Course Guide

IBM Cognos Analytics: Author Reports Fundamentals

Course code B6058 ERC 3.0
April 2016

NOTICES

This information was developed for products and services offered in the USA.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service. IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
United States of America

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

TRADEMARKS

IBM, the IBM logo, ibm.com and Cognos are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.

Adobe, and the Adobe logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

© Copyright International Business Machines Corporation 2016.

This document may not be reproduced in whole or in part without the prior written permission of IBM.

US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
Contents

Preface................................................................................................................. P-1

Contents............................................................................................................. P-3
Course overview................................................................................................. P-9
Verify services used in the course environment................................................ P-10
Optional configuration for the IBM Cognos Analytics 11.0
  course environment ........................................................................................ P-12
Document conventions..................................................................................... P-15
Exercises.......................................................................................................... P-16
Additional training resources ............................................................................ P-17
IBM product help.............................................................................................. P-18

Unit 1 Introduction to IBM Cognos Analytics - Reporting ............................. 1-1

Unit objectives .................................................................................................... 1-3
What is IBM Cognos Analytics - Reporting? ....................................................... 1-4
Explore the environment ..................................................................................... 1-6
Examine the side panel ...................................................................................... 1-7
Explore authoring templates ............................................................................... 1-8
Generate the report ............................................................................................ 1-9
Change the properties of an object ................................................................... 1-10
Demonstration 1: Create a simple report .......................................................... 1-14
Dimensionally-modeled and dimensional data sources .................................... 1-21
Demonstration 2: Create a report from a dimensionally-modeled
  relational data source ..................................................................................... 1-22
Examine personal data sources and data modules .......................................... 1-27
Demonstration 3: Create a report from a personal data source ....................... 1-28
Unit summary ................................................................................................... 1-31
Exercise 1: Create a revenue report ................................................................. 1-32

Unit 2 Create list reports ................................................................................. 2-1

Unit objectives .................................................................................................... 2-3
Examine list reports ............................................................................................ 2-4
Group data ......................................................................................................... 2-5
Format list columns ............................................................................................ 2-6
Include list headers and footers .......................................................................... 2-7
Demonstration 1: Enhance a list report ............................................................... 2-8
Understand fact/measure data ......................................................................... 2-17
Understand aggregate data .............................................................................. 2-18
Understand difference in aggregation ............................................................... 2-19
Demonstration 2: Explore data aggregation ..................................................... 2-20
Use shared dimensions to create multi-fact queries ......................................... 2-26
Add data-driven baselines and markers to charts ............................................... 5-8
Demonstration 1: Create and format a chart report ............................................ 5-9
Compare values and highlight proportions using gauge charts and pie charts ............................................... 5-22
Demonstration 2: Create a gauge report and a pie chart report ........................ 5-23
Display items on separate axes........................................................................ 5-29
Demonstration 3: Show the same data graphically and numerically ............... 5-30
Customize charts .............................................................................................. 5-36
What is RAVE? ................................................................................................. 5-37
What is a visualization? .................................................................................... 5-38
Demonstration 4: Display visualizations ........................................................... 5-39
Unit summary ................................................................................................... 5-44
Exercise 1: Create a dashboard report ............................................................. 5-45

Unit 6  Focus reports using prompts ................................................................. 6-1

<table>
<thead>
<tr>
<th>Unit objectives</th>
<th>6-3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Examine parameters and prompts</td>
<td>6-4</td>
</tr>
<tr>
<td>Create a parameter item on the report</td>
<td>6-5</td>
</tr>
<tr>
<td>Build a prompt page</td>
<td>6-6</td>
</tr>
<tr>
<td>Add a prompt item to a report</td>
<td>6-7</td>
</tr>
<tr>
<td>Demonstration 1: Create a prompt by adding a parameter</td>
<td>6-8</td>
</tr>
<tr>
<td>Identify prompt type</td>
<td>6-12</td>
</tr>
<tr>
<td>Demonstration 2: Add a value prompt to a report</td>
<td>6-13</td>
</tr>
<tr>
<td>Add pages to a report</td>
<td>6-19</td>
</tr>
<tr>
<td>Demonstration 3: Add a Select &amp; search prompt to a report</td>
<td>6-20</td>
</tr>
<tr>
<td>Create a cascading prompt</td>
<td>6-24</td>
</tr>
<tr>
<td>Demonstration 4: Create a cascading prompt</td>
<td>6-25</td>
</tr>
<tr>
<td>Unit summary</td>
<td>6-32</td>
</tr>
<tr>
<td>Exercise 1: Focus a report using value prompts</td>
<td>6-33</td>
</tr>
</tbody>
</table>

Unit 7  Extend reports using calculations ...................................................... 7-1

<table>
<thead>
<tr>
<th>Unit objectives</th>
<th>7-3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Derive additional information from the data source</td>
<td>7-4</td>
</tr>
<tr>
<td>Add run-time information to your report</td>
<td>7-5</td>
</tr>
<tr>
<td>Add Date/Time functions to your report</td>
<td>7-7</td>
</tr>
<tr>
<td>Add string functions to your report</td>
<td>7-9</td>
</tr>
<tr>
<td>Demonstration 1: Add calculations to a report</td>
<td>7-10</td>
</tr>
<tr>
<td>Display prompt selections in report titles</td>
<td>7-16</td>
</tr>
<tr>
<td>Demonstration 2: Display prompt selections in the report title</td>
<td>7-17</td>
</tr>
<tr>
<td>Unit summary</td>
<td>7-23</td>
</tr>
<tr>
<td>Exercise 1: Sales percent by sales representative and country</td>
<td>7-24</td>
</tr>
</tbody>
</table>
Unit 8  Use additional report building techniques ........................................ 8-1
Unit objectives .................................................................................................. 8-3
Enhance report design .................................................................................... 8-4
Add objects ....................................................................................................... 8-6
Organize objects using tables ......................................................................... 8-7
Break a report into sections ........................................................................... 8-8
Convert a list to a crosstab .......................................................................... 8-9
Reuse objects within the same report .......................................................... 8-10
Demonstration 1: Reuse objects within the same report ............................... 8-11
Share layout components among separate reports .................................... 8-18
Demonstration 2: Reuse layout components in a different report ................. 8-20
Handle reports with no data .......................................................................... 8-27
Demonstration 3: Explore options for reports that contain do data ............... 8-28
Unit summary .................................................................................................. 8-35
Exercise 1: Analyze product quantities sold by month .................................. 8-36

Unit 9  Customize reports with conditional formatting ................................. 9-1
Unit objectives .................................................................................................. 9-3
Change displays based on conditions .............................................................. 9-4
3 steps for conditional formatting ................................................................. 9-5
Step 1. Create a variable .................................................................................. 9-6
Step 2. Assign the variable to a report object ............................................... 9-8
Step 3. Apply formatting to object based on condition value ....................... 9-10
Demonstration 1: Create a multilingual report (optional) ....................... 9-11
Demonstration 2: Highlight exceptional data ................................................. 9-18
Conditionally render objects in reports ......................................................... 9-22
Demonstration 3: Create a report with a conditionally rendered column ...... 9-23
Conditionally format one crosstab measure based on another ................. 9-27
Demonstration 4: Conditionally format one crosstab measure based on another ................................................................. 9-28
Unit summary .................................................................................................. 9-33
Exercise 1: Distinguish yearly data ................................................................. 9-34
Unit 10  Drill-through definitions

Unit objectives ........................................................................................................... 10-3
Let users navigate to related data in IBM Cognos Analytics .................................. 10-4
Set up drill-through access from a report ................................................................. 10-5
Package-based drill through .................................................................................... 10-6
Specify the values passed to target parameters ..................................................... 10-8
Steps to set up a package-based drill-through definition ........................................ 10-9
Limit the items that users can drill through from .................................................. 10-10
Measure-based scope ............................................................................................... 10-11
Drill Through Assistant ........................................................................................... 10-12
Demonstration 1: Set up drill-through access for a package .................................... 10-13
Dynamic drill-through .............................................................................................. 10-21
Dynamic drill-through - matching criteria ............................................................... 10-23
Demonstration 2: Configure dynamic drill-through and set measure scope .......... 10-24
Unit summary ............................................................................................................. 10-30
Exercise 1: Configure dynamic drill-through ......................................................... 10-31

Unit 11  Enhance report layout

Unit objectives ............................................................................................................. 11-3
View the structure of the report .................................................................................. 11-4
Force page breaks in reports ..................................................................................... 11-5
Demonstration 1: Create a report structured on data items ...................................... 11-6
Horizontal pagination ................................................................................................. 11-13
Add horizontal page numbers .................................................................................... 11-14
Demonstration 2: Format a report for horizontal viewing .......................................... 11-15
Modify structures ....................................................................................................... 11-20
Demonstration 3: Create a condensed list report ..................................................... 11-21
Change PDF page orientation to suit report objects ................................................ 11-25
Set PDF security options .......................................................................................... 11-26
Demonstration 4: Change a PDF page from portrait to landscape orientation .......... 11-27
Format objects across a report .................................................................................. 11-33
Demonstration 5: Format objects across a report (optional) ...................................... 11-34
Unit summary ............................................................................................................. 11-38
Exercise 1: Analyze retailer contacts by country ..................................................... 11-39
Appendix A  Introduction to IBM Cognos Analytics................................. A-1

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit objectives</td>
<td>A-3</td>
</tr>
<tr>
<td>What is IBM Cognos Analytics?</td>
<td>A-4</td>
</tr>
<tr>
<td>Redefined Business Intelligence</td>
<td>A-5</td>
</tr>
<tr>
<td>Self-service</td>
<td>A-6</td>
</tr>
<tr>
<td>Navigate to content in IBM Cognos Analytics</td>
<td>A-7</td>
</tr>
<tr>
<td>Interact with the user interface</td>
<td>A-8</td>
</tr>
<tr>
<td>Model data with IBM Cognos Analytics</td>
<td>A-9</td>
</tr>
<tr>
<td>IBM Cognos Analytics components</td>
<td>A-10</td>
</tr>
<tr>
<td>Create reports with IBM Cognos Analytics - Reporting</td>
<td>A-12</td>
</tr>
<tr>
<td>Perform self-service analysis with Dashboards</td>
<td>A-13</td>
</tr>
<tr>
<td>IBM Cognos Analytics architecture (high level)</td>
<td>A-14</td>
</tr>
<tr>
<td>IBM Cognos Analytics security</td>
<td>A-15</td>
</tr>
<tr>
<td>IBM Cognos Analytics groups and roles</td>
<td>A-16</td>
</tr>
<tr>
<td>Package/Data source relationship</td>
<td>A-17</td>
</tr>
<tr>
<td>Create Data Modules</td>
<td>A-18</td>
</tr>
<tr>
<td>Upload files</td>
<td>A-19</td>
</tr>
<tr>
<td>Demonstration 1: Explore IBM Cognos Analytics</td>
<td>A-20</td>
</tr>
<tr>
<td>Extend IBM Cognos Analytics</td>
<td>A-43</td>
</tr>
<tr>
<td>Unit summary</td>
<td>A-44</td>
</tr>
</tbody>
</table>
Course overview

Preface overview

IBM Cognos Analytics: Author Reports Fundamentals (v11.0) provides report authors an opportunity to learn report building techniques using relational data models. Techniques to enhance, customize, and manage professional reports will be explored. Demonstrations and exercises will illustrate and reinforce key concepts during this learning opportunity.

Intended audience

Report Authors

Topics covered

Topics covered in this course include:

- Explore IBM Cognos Analytics report authoring, different report object types (list, crosstab, chart, visualization, etc.)
- Create and format reports using grouping, headers, footers, and other formatting options
- Focus reports by filtering data and using prompts
- Add value to your reports using calculations and additional report building techniques
- Enhance reports with advanced formatting and exceptional data highlighting

Course prerequisites

Participants should have:

- Knowledge of business requirements
- Experience using the Windows operating system
- IBM Cognos Analytics for Consumers (v11.0) WBT or equivalent knowledge
Verify services used in the course environment

The environment provided in this course requires the following services to be started before you begin performing demonstrations and exercises:

- Apache Directory Server
- DB2 -DB2COPY 1 - DB2
- DB2DAS - DB2DAS00
- IBM Cognos
- World Wide Web Publishing Service
- IBM Cognos

To review the services, on the Taskbar of your environment, click the Services icon, and ensure that the above services are running. If you have closed your image and launched it again, it is a best practice to review the status of the services before continuing with your demonstrations and exercises.

If the Apache Directory Server or DB2 -DB2COPY 1 - DB2 service have stopped, you will need to stop the IBM Cognos service, start the stopped services, and then start the IBM Cognos service once the previously stopped services have started successfully. You can start and stop a specific service by double-clicking the service to open the Properties dialog box, and then clicking the Stop or Start buttons.

Note that it may take 15 minutes or more for the IBM Cognos service to start. You may also see the following message, which is normal:

![Services dialog box]

Windows could not start the IBM Cognos service on Local Computer.
Error 1053: The service did not respond to the start or control request in a timely fashion.

Click OK, then continue to monitor and check that the Service is started.

Periodically check to see if the IBM Cognos service has started, by clicking Refresh.
Once the service has started, you will see a status of Started:

![Services window](image)

Once you have confirmed that the Service has started, close the Services window.
Optional configuration for the IBM Cognos Analytics 11.0 course environment

The training environment provided has been configured and tested to work with the demonstrations and exercises in this course.

For instructors teaching in an ILT or ILO environment, or for students in a Self-paced Virtual Classroom (SPVC) environment, there is an optional configuration that can be performed. This configuration provides access to a wide range of templates that can be used to create reports in the new IBM Cognos Analytics - Reporting environment. If you proceed with the optional configuration presented here, in the course environment, this change impacts the IBM Cognos Analytics - Reporting user interface and the initial steps used to create reports as scripted in the demonstration and exercise steps. If you want to use the optional configuration, you are advised to first complete the course as it is originally scripted, and then you may explore this optional configuration.

When creating a report in the current environment (as per the course set up tasks), the user interface and dialog box for choosing a report type (List, Crosstab, Chart, etc.) appears as follows:

![Dialog box for choosing report type](image)

The dialog box is a legacy component from the IBM Cognos BI 10.2.2 version of the product. It is available, by default, when performing the "Custom" installation of IBM Cognos Analytics 11.0. The "Custom" installation was chosen for the setup of the course environment.

Note: Steps in demonstrations and exercises in this course are currently scripted to use this dialog box for creating reports.
The optional configuration includes the deployment of the Templates.zip deployment archive. This can be performed using the following steps in the course environment:

1. In **Internet Explorer**, log on to **IBM Cognos Analytics** as **admin\Education1**.
2. On the side panel (left pane), click **Manage**, and then click **Administration console**.
3. Click the **Configuration** tab, and then click **Content Administration**.
4. On the toolbar, click **New Import**.
5. On the **Select a deployment archive** page, select the **Templates** archive, and then click **Next**.
6. On the **Specify name and description** page, click **Next**.
7. On the **Public folders, directory and library content** page, select the **Templates** check box, and then click **Next**.
8. On the **Specify the general options** page, click **Next**.
9. On the **Review the summary** page, click **Next**.
10. On the **Select an action** page, ensure **Save and run once** is selected, and then click **Finish**.
11. On the **Run with options** page, ensure **Now** is selected, click **Run**, and then click **OK**.
12. Close the **IBM Cognos Administration** tab.

Once this configuration has been performed, use the following steps to create a report.

1. Follow the steps from the demonstration or exercise to log on as a user with the ability to create reports.
2. From the side panel (left pane), click **New**.
3. Click **Report**.
   The user interface now appears as follows:

![User Interface](image)

You are presented with a list of templates to choose from.

4. Select a template. Note: there are various layouts to choose from. In most instances, you will be successful with current steps using the "1 column" option (1st column, 2nd row). If issues are encountered, the "Blank" option is also available.

5. In the report layout on the right, click +, and then choose the report type (List, Crosstab, Chart), or object (Table, Text item, Block)

6. Continue creating the report as scripted in the demonstration or exercise. Note: Some screen captures may appear different than what is presented in the demonstration or exercise, if you choose to incorporate the Templates.zip deployment.
Document conventions

Conventions used in this guide follow Microsoft Windows application standards, where applicable. As well, the following conventions are observed:

- **Bold**: Bold style is used in demonstration and exercise step-by-step solutions to indicate a user interface element that is actively selected or text that must be typed by the participant.

- **Italic**: Used to reference book titles.

- **CAPITALIZATION**: All file names, table names, column names, and folder names appear in this guide exactly as they appear in the application. To keep capitalization consistent with this guide, type text exactly as shown.
Exercises

Exercise format
Exercises are designed to allow you to work according to your own pace. Content contained in an exercise is not fully scripted out to provide an additional challenge. Refer back to demonstrations if you need assistance with a particular task. The exercises are structured as follows:

The business question section
This section presents a business-type question followed by a series of tasks. These tasks provide additional information to help guide you through the exercise. Within each task, there may be numbered questions relating to the task. Complete the tasks by using the skills you learned in the unit. If you need more assistance, you can refer to the Task and Results section for more detailed instruction.

The task and results section
This section provides a task based set of instructions that presents the question as a series of numbered tasks to be accomplished. The information in the tasks expands on the business case, providing more details on how to accomplish a task. Screen captures are also provided at the end of some tasks and at the end of the exercise to show the expected results.
Additional training resources

- Visit IBM Analytics Product Training and Certification on the IBM website for details on:
  - Instructor-led training in a classroom or online
  - Self-paced training that fits your needs and schedule
  - Comprehensive curricula and training paths that help you identify the courses that are right for you
  - IBM Analytics Certification program
  - Other resources that will enhance your success with IBM Analytics Software
- For the URL relevant to your training requirements outlined above, bookmark:
  - Information Management portfolio: http://www-01.ibm.com/software/data/education/
### IBM product help

<table>
<thead>
<tr>
<th>Help type</th>
<th>When to use</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task-oriented</td>
<td>You are working in the product and you need specific task-oriented help.</td>
<td><em>IBM Product</em> - Help link</td>
</tr>
<tr>
<td>Books for Printing (.pdf)</td>
<td>You want to use search engines to find information. You can then print out selected pages, a section, or the whole book. Use Step-by-Step online books (.pdf) if you want to know how to complete a task but prefer to read about it in a book. The Step-by-Step online books contain the same information as the online help, but the method of presentation is different.</td>
<td><em>Start/Programs/IBM Product/Documentation</em></td>
</tr>
<tr>
<td>IBM on the Web</td>
<td>You want to access any of the following:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• IBM - Training and Certification</td>
<td><em><a href="http://www-01.ibm.com/software/analytics/training-and-certification/">http://www-01.ibm.com/software/analytics/training-and-certification/</a></em></td>
</tr>
<tr>
<td></td>
<td>• Online support</td>
<td><em><a href="http://www-947.ibm.com/support/entry/portal/Overview/Software">http://www-947.ibm.com/support/entry/portal/Overview/Software</a></em></td>
</tr>
<tr>
<td></td>
<td>• IBM Web site</td>
<td><em><a href="http://www.ibm.com">http://www.ibm.com</a></em></td>
</tr>
</tbody>
</table>
Unit objectives

- Examine IBM Cognos Analytics - Reporting and its interface
- Explore different report types
- Create reports in preview or design mode
- Create a simple, sorted, and formatted report
- Examine dimensionally modelled and dimensional data sources
- Explore how data items are added queries
- Examine personal data sources and data modules
What is IBM Cognos Analytics - Reporting? (1 of 2)

- Reporting is a Web-based report authoring tool.
- Reporting lets you create business intelligence reports that analyze corporate data according to specific information needs.
- Reporting lets you format, present, and distribute your corporate data using many different methods.
Reporting, in IBM Cognos Analytics, can be performed based on the following:

- **Create New**: Author a report with a choice from preset template options.
- **Open Existing**: Open a copy of an existing report, make changes and then save under a new name.
- **New from Template**: Navigate to and open a saved template. This option shows saved templates and all reports. You will be able to choose any saved report that is appropriate for your needs and Reporting will then automatically create a template based on the chosen report.
Explore the environment

Some notable actions and features:

- You can build reports by adding objects and data items from the Content pane.
- The Source tab provides query items that can be added to report data containers directly from a published package.
- You can make changes to the structure of the package by using Framework Manager.
- The Data Items tab allows you to insert data items into your report that already exist in your query.
- The Toolbox tab allows you to insert report objects into your report.
- Modify objects and query items using the Properties pane.
- Use the Navigate tab to move through the Page structure of a report, including using the Page explorer, Query explorer, Condition explorer, and Active report controls and variables.
Examine the side panel

**Data** - use this to add query items to a report template.

**Toolbox** - use this to modify the report by adding unique options to the report template that aid in creating a professional look and feel to the report and additional options to a unique view of the data.

**Navigate** - use this to navigate through the different pages of the report including the Query and Conditional Explorer.
Explore authoring templates

- IBM Cognos Analytics - Reporting contains several report templates to structure your reports.

Creating a new report gives the report author a chance to pick a layout presentation for the report screen. Combined with a Theme, the author can quickly pick a layout a set of colors that are attractive.

Once the report layout has been selected, for each section in the layout, there is the Add button, which when clicked provides the option to create:

- List reports which are useful for presenting tabular list information.
- Crosstab reports which are useful for comparative analysis.
- Chart reports which are useful for graphically showing comparisons, relationships, and trends.
- Text items which are useful for labeling report objects in a meaningful way.
- Blocks which are useful for extending the layout.
- Table which are useful for formatting data in a repeated fashion.
Generate the report

- You can view the results of the designed report by running the report in the web browser.

While working in Page design mode, you will only see metadata, such as column or row labels. You will not see actual data values in the report. While working in Page preview mode, you will see sample data as you create the report. To see the final report results, you must run the report. The results appear in a separate web browser tab.

You can navigate through the report using the Page Up, Page Down, Top, and Bottom links at the bottom of the report.

You can return to Reporting to alter your report by clicking on the reporting tab containing your work area, or by closing the tab.

You can distribute reports by email, through the Web, or you can save them on your desktop. You would first render the report in the format that is most suitable for your needs: HTML, PDF, Excel, Excel Data, Run Delimited Text (CSV).
Change the properties of an object

- The Properties pane lets you view and change the properties of an item or object in your work area.

When you click an item or object in the report, the properties for that item appear in the Properties pane.

You can verify the object type selected by the name displayed at the top of the Properties pane. It is a best practice to verify the object type selected before making any modifications to it.

You can select an ancestor (or parent object) of the object previously selected in your work area by clicking the Select Ancestor button.

In the slide example, the Properties pane shows the properties and settings for Revenue, which is a column in a list data object. The data displayed will be summarized by total.

There are different ways to change a property setting. If there are only two options for a certain property, double-click the setting to toggle to the other option. If there are multiple options, you can click the setting and then click the ellipsis and choose the desired setting from the dialog box that appears, or choose a selection from a drop-down list.
Demonstration and exercise start point information

This section is a reference so that you will know how to interpret the start point information in this course.

Before you begin the steps of a demonstration or exercise, you will see start point information to help you set your environment for the tasks that you will perform in that demonstration or exercise. The start point format appears similar to the following:

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content > Samples > Models > GO data warehouse (query)
Report Type: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

This information provides you with a unique starting point for that demonstration or exercise. It tells how you will access IBM Cognos Analytics through a browser, it provides the user ID and password to use, the package to use, the report type to start with, and within the data source, the folder and namespace for items to be used in your report as you build it. Use this to set your environment before beginning the first task of the demonstration or exercise.

For example, if your demonstration provided the start point information above, you would do the following steps to create the list report using the Sales and Marketing (query)/Sales (query) namespace:

1. From the Start menu, click All Programs > Internet Explorer to launch a browser session.
2. In the Address box, type the portal address http://vclassbase:9300/bi and then press Enter.
   The Log On to IBM Cognos Analytics window appears in a browser tab, with a Log on dialog box prompting for a User ID and Password. You will log on with the credentials listed in the start point information.
3. In the User ID box type brettonf, in the Password box type Education1, and then click Sign in.
   The IBM Cognos Analytics software page displays, the features of the application that your user has permission to use are displayed.
4. Click New and then click Report.
   You are prompted to select a package to author your reports with and also to select a reporting object, such as a List or a Crosstab.
5. Click the ellipsis to the right of the Package text box.
6. Click **Samples > Models > GO data warehouse (query)**, and then click **Open**. The New dialog box displays the package you will work with, and the report objects that you will begin to author your report. You can select one object from the list, and then click OK, or you can double-click an object to select it and launch the new report. The start point information in this example requires a list report template.

7. Click **List** and then click **OK**.

Most demonstrations and exercises will require a new report. If your demonstration or exercise requires you to use an existing report, you will be provided with a report name and location in the start point information. Because this start point example lists a Report type, this is your cue that a new report is required.

8. Click **Data** in the side panel on the left.

This opens two tabs: Source and Data items. The Source tab includes a package explorer for using the data in the package that you have selected, while the Data items tab contains a list of data items that are in use in your report.

9. Expand the **Sales and Marketing (query)** folder.

The expanded folder displays the namespaces that are available to you in this package. The starting point information in this example will work with the Sales (query) namespace.
10. Expand the **Sales (query)** namespace.

    The results appear as follows:

    ![Diagram of Sales (query) namespace]

    From here, you would work with the metadata of query subjects, query items, and facts within this selected namespace unless otherwise mentioned.

    Follow the start point information carefully, as there will be different logins, packages, report types, and namespaces used for each demonstration and exercise in this course.

    At the end of each unit, in the last demonstration or exercise, you will be instructed to log off. This is a best practice, to free up resources, and another best practice is to close all browser windows, especially if you are finished working with IBM Cognos Analytics for the day.
Demonstration 1: Create a simple report

<table>
<thead>
<tr>
<th>Country</th>
<th>City</th>
<th>Last name</th>
<th>First name</th>
<th>Position name</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switzerland</td>
<td>Genève</td>
<td>Bruno</td>
<td>Fausta</td>
<td>Level 3 Sales Rep</td>
<td>$79,955,838.92</td>
</tr>
<tr>
<td>Switzerland</td>
<td>Genève</td>
<td>Giordano</td>
<td>Florenza</td>
<td>Level 3 Sales Rep</td>
<td>$72,784,594.30</td>
</tr>
<tr>
<td>Switzerland</td>
<td>Genève</td>
<td>Chambers</td>
<td>Warren</td>
<td>Level 3 Sales Rep</td>
<td>$82,843,459.76</td>
</tr>
<tr>
<td>Finland</td>
<td>Kuopio</td>
<td>Lindholm</td>
<td>Helena</td>
<td>Level 3 Sales Rep</td>
<td>$59,769,153.93</td>
</tr>
<tr>
<td>Korea</td>
<td>Seoul</td>
<td>Kim</td>
<td>Chang-ho</td>
<td>Level 3 Sales Rep</td>
<td>$59,422,592.32</td>
</tr>
<tr>
<td>United States</td>
<td>Los Angeles</td>
<td>Laurel</td>
<td>Charles</td>
<td>Level 3 Sales Rep</td>
<td>$59,406,874.73</td>
</tr>
<tr>
<td>Switzerland</td>
<td>Genève</td>
<td>Bichot</td>
<td>Lotta</td>
<td>Level 3 Sales Rep</td>
<td>$54,436,904.60</td>
</tr>
<tr>
<td>Netherlands</td>
<td>Amsterdam</td>
<td>Jansen-Velasquez</td>
<td>Beinda</td>
<td>Level 3 Sales Rep</td>
<td>$52,822,234.19</td>
</tr>
</tbody>
</table>
Demonstration 1: 
Create a simple report

Purpose:
Sales executives would like you to create a report that lists all of the sales representatives and the revenue they have generated to date. The report should include their name, position, city, and country. Sort the report by revenue, in descending order, and display revenue in American dollars.

Task 1. Open IBM Cognos Analytics and choose a report template.
1. Launch Internet Explorer, in the address field type http://vclassbase:9300/bi, and then press Enter.
   In the training environment, this may be the default browser landing page, so if you are presented with a Sign In dialog box, you may proceed.
2. On the IBM Cognos Analytics page, in the User ID box, type brettonf, in the Password box, type Education1, and then click Sign in.
   The Welcome to IBM Cognos Analytics page appears.
3. From the side panel on the left, click New, and then click Report.
4. From the New dialog, select the Blank template, and then click OK.
   You will check the state of an option available in Reporting, to ensure that the automatic group and summary behavior for lists is off. In your own work environment, you may have this on or off, depending on the needs of your organization. For this course, you will have it off, so that summary rows, for example, do not automatically appear in lists.
5. In the application toolbar at the top, click the More ellipses, and then click Options.
6. Click the Report tab, and ensure Automatic group and summary behavior for lists is deselected, and then click OK.

Task 2. Add a data source and a list.
1. From the side panel, click Data, and notice two tabs appear: Source and Data items.
2. Under the Source tab, click Add report data, which is a circle icon with a plus sign in it.
3. Navigate to: Team content\Samples\Models\GO data warehouse (query). The results appear as follows:

![Open]

4. Click Open.

5. On the Application bar, click Page views, and then click Page preview.

6. Click Add in the center of the screen. This will bring up a set of data container choices.

7. Click the List template, and then click OK to accept the default query name.

**Task 3. Add data to the list.**

1. On the side panel, Source tab, expand the Sales and Marketing (query) folder, expand the Sales (query) namespace, and then expand the Employee by region query subject.

2. Double-click the Country query item to add it to the list report object. The list report object now has one column.

3. Double-click City to add it to the list report object. City is automatically added to the end of the list.

4. Right-click Last name, and then click Properties. The Properties dialog box appears, with details about the item.

5. Click Close.

6. Click First name, and then Ctrl-click Last name, Employee level and Position name.
7. Right-click **Position name**, and then click **Insert**.
The items are added to the list in the order in which they are selected.

A section of the results appear as follows:

```
<table>
<thead>
<tr>
<th>Country</th>
<th>City</th>
<th>First name</th>
<th>Last name</th>
<th>Employee level</th>
<th>Position name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switzerland</td>
<td>Genève</td>
<td>Aaghe</td>
<td>Heiman</td>
<td>5</td>
<td>Information Technology Manager</td>
</tr>
<tr>
<td>Switzerland</td>
<td>Genève</td>
<td>Aaltje</td>
<td>Hansen</td>
<td>6</td>
<td>Level 1 Sales Representative</td>
</tr>
<tr>
<td>Brazil</td>
<td>São Paulo</td>
<td>Abel</td>
<td>Antunes</td>
<td>4</td>
<td>Product Manager</td>
</tr>
<tr>
<td>Switzerland</td>
<td>Genève</td>
<td>Abram</td>
<td>Ruiz</td>
<td>6</td>
<td>Level 2 Sales Representative</td>
</tr>
<tr>
<td>Italy</td>
<td>Milano</td>
<td>Ada</td>
<td>Morales</td>
<td>5</td>
<td>Warehouse Worker</td>
</tr>
<tr>
<td>Italy</td>
<td>Milano</td>
<td>Adara</td>
<td>Cruz</td>
<td>2</td>
<td>Accountant 2</td>
</tr>
</tbody>
</table>
```

8. From the **Source** tab, expand the **Sales fact** query subject, and then click and drag **Revenue** to add it to the end of the list (you should see a flashing black bar inside of a white bar, indicating the correct drop zone).

If you place the query item outside of the list report object you will receive a message indicating that you have created a singleton. You instead want the new query item to be added to the end of the List, so if this is the case, undo your last action, then redo step 8.

You would like to see Last name appearing before First name.

9. In the work area, click the data for **Last name** (not the header) to select the list column body, and then drag it to the left of the **First name** list column body.

A flashing black bar appears when the item is over a drop zone.

Note: Make sure that the list column body is selected by clicking any one of the cells in the column, not the column header. To check to see what element of the report you have selected, check the title bar of the Properties pane.

Now that you have built the report you can view the data items in the query.

**Task 4. View the data items in the query.**

1. On the side panel, click **Navigate** ☿, then on the content pane, click **Query explorer** ⬠, and then click **Query1**.

The data items you added to the list appear in the Data Items pane for the query. The names of the data items correspond to the column titles in the report layout.

2. In the **Data Items** pane, click **Position name**.

You want to view information about the data the Position name data item retrieves from the data source.
3. From the **Application** bar, click **Show properties**.

4. In the **Properties** pane, double-click the **Expression** property.

   In the Data item expression dialog box, you can see that this data item retrieves data from the Position name query item, in the Employee by region query subject, in the Sales (query) namespace.

5. Click **OK**, and then in the **Data Items** pane, click **Last name**.

6. In the **Properties** pane, double-click the **Expression** property.

   The Data item expression dialog box appears. You can see that this data item retrieves data from the Last name query item, in the Employee by region query subject, in the Sales (query) namespace.

7. Click **OK** to close the dialog box.

8. In the content pane, click **Page explorer**, and then click **Page1** to return to the work area.

**Task 5. Remove a column from the report.**

It has been decided that Employee level in the list report object is not needed in the report. You will remove it from the list.

1. In the list report object, click a data cell for **Employee level** (not the header).

2. From the now visible container toolbar, click **More**, and then click **Cut**.

   The column is removed from the list report.

3. On the content toolbar, click **Query explorer**, and then click **Query1**.

   The Employee level data item still appears in the Data Items pane. Although you removed the Employee level data item from the report layout in Page Explorer, the data item has not been removed from the query. However, keeping the data item in the query can be useful for other tasks such as creating a calculation.

   Other examples of where you would keep a data item in the query, but remove it from the report layout are: creating an expression based on the query item, or, using this item when sorting or formatting data in the list.

4. On the content toolbar, click **Page explorer**, and then click **Page1** to return to the work area.

5. On the **Application** bar, click **Undo**.

6. With the **Employee level** data column still selected, on the container toolbar, click **More**, and then click **Delete**.
7. On the content toolbar, click **Query explorer**, and then click **Query1**. The Employee level data item has been removed from the report layout and the query and no longer appears in the Data Items pane.

**Task 6. Format and sort the data, and then run the report.**

1. On the content toolbar, click **Page explorer**, and then click **Page1**.
2. In the list report object, click a data cell for **Revenue**, in the list column body (not the column title).
   The Revenue cells are highlighted to show that they are selected. The Properties pane shows the properties for this column.
3. On the container toolbar, click **Sort** and then click **Descending**. Our sales reps will now be ranked starting with our top performers.
4. With the **Revenue** column still selected, in the **Properties** pane, under the **DATA** category, click **Data format**, and then click the ellipsis **...**. The Data Format dialog box appears.
5. In the **Format type** list, select **Currency**.
6. Under **Properties**, click **Currency**, click the down arrow button in the column to the right of **Currency**, and then select **$ (USD) - United States of America, dollar** from the list.

Revenue will now be displayed in American dollars. By default, it will use a comma as a Thousands separator, and two decimal places.

Note: Changing the currency will not perform a currency conversion (for example, it will not convert one currency into the value of another). It will simply show the value with a different currency symbol, thousands separator, decimal place, and so on. If you want to see data displayed in a particular currency, the data must be stored in the data source in that currency.
7. Click **OK** to close the Data format dialog box.
8. Click **Page views** and then click **Page design**.

The results appear as follows:

<table>
<thead>
<tr>
<th>Country</th>
<th>City</th>
<th>Last name</th>
<th>First name</th>
<th>Position name</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Country&gt;</td>
<td>&lt;City&gt;</td>
<td>&lt;Last name&gt;</td>
<td>&lt;First name&gt;</td>
<td>&lt;Position name&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Country&gt;</td>
<td>&lt;City&gt;</td>
<td>&lt;Last name&gt;</td>
<td>&lt;First name&gt;</td>
<td>&lt;Position name&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Country&gt;</td>
<td>&lt;City&gt;</td>
<td>&lt;Last name&gt;</td>
<td>&lt;First name&gt;</td>
<td>&lt;Position name&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
</tbody>
</table>

Page design mode is an alternate way to edit report objects such as the List.
9. On the main toolbar, click **Run options**  
and then click **Run HTML**.
A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Country</th>
<th>City</th>
<th>Last name</th>
<th>First name</th>
<th>Position name</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switzerland</td>
<td>Genève</td>
<td>Bruno</td>
<td>Fausta</td>
<td>Level 3 Sales Representative</td>
<td>$79,955,838.92</td>
</tr>
<tr>
<td>Switzerland</td>
<td>Genève</td>
<td>Giordano</td>
<td>Fiorenza</td>
<td>Level 3 Sales Representative</td>
<td>$72,784,594.30</td>
</tr>
<tr>
<td>Switzerland</td>
<td>Genève</td>
<td>Chambers</td>
<td>Warren</td>
<td>Level 3 Sales Representative</td>
<td>$62,843,459.76</td>
</tr>
<tr>
<td>Finland</td>
<td>Kuopio</td>
<td>Lindholm</td>
<td>Helena</td>
<td>Level 3 Sales Representative</td>
<td>$59,799,153.93</td>
</tr>
<tr>
<td>Korea</td>
<td>Seoul</td>
<td>Kim</td>
<td>Chang-ho</td>
<td>Level 3 Sales Representative</td>
<td>$59,422,582.32</td>
</tr>
<tr>
<td>United States</td>
<td>Los Angeles</td>
<td>Laurel</td>
<td>Charles</td>
<td>Level 3 Sales Representative</td>
<td>$59,406,874.73</td>
</tr>
<tr>
<td>Switzerland</td>
<td>Genève</td>
<td>Bichot</td>
<td>Lotta</td>
<td>Level 3 Sales Representative</td>
<td>$54,436,904.60</td>
</tr>
<tr>
<td>Netherlands</td>
<td>Amsterdam</td>
<td>Jansen-Velasquez</td>
<td>Belinda</td>
<td>Level 3 Sales Representative</td>
<td>$52,822,234.19</td>
</tr>
</tbody>
</table>

You can see that revenue is sorted in descending order.

10. At the bottom of the page, click **Page down**  
to navigate to each page of the report.

11. Close the rendered report web page (tab) to return to the work area.

12. Leave **IBM Cognos Analytics** open for the next demonstration.

**Results:**
You created a list report and added the necessary items from the model as required by the sales executives. You sorted the data in descending order and formatted the revenue in American dollars.
Dimensionally-modeled and dimensional data sources

- In IBM Cognos Analytics, reports using dimensionally-modeled relational data sources and dimensional data sources enable you to drill down to a detailed level.

Dimensionally-modeled relational metadata is data taken from a relational source and modeled as a star schema. As well, hierarchies are applied to allow for drill behavior.

Dimensionally-modeled relational data extends dimensional capabilities (such as drill-down) to relational sources.

With dimensional analysis, your corporate data is organized in the way you think about your business so that you spend more time on value added analysis, rather than on data retrieval.

Only dimensional models allow drill up and drill down behavior in analyses and reports.

Note: The purpose of this course is to explore how Reporting can use relational data sources to create reports. The next demonstration provides an opportunity to create a report using a dimensional data source. The IBM Cognos Analytics: Author Reports with Multidimensional Data course explores, in greater detail, how IBM Cognos Analytics - Reporting can be used to analyze DMR or OLAP data sources.
Demonstration 2

Create a report from a dimensionally-modeled relational data source

<table>
<thead>
<tr>
<th>Year</th>
<th>Country</th>
<th>Store</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>2011</td>
<td>Canada</td>
<td>Star Dome</td>
<td>621</td>
</tr>
<tr>
<td>Q1 2011</td>
<td>Canada</td>
<td>Star Dome</td>
<td>531</td>
</tr>
<tr>
<td>Q2 2011</td>
<td>Canada</td>
<td>Star Dome</td>
<td>586</td>
</tr>
<tr>
<td>Q3 2011</td>
<td>Canada</td>
<td>Star Dome</td>
<td>665</td>
</tr>
</tbody>
</table>

Demonstration 2: Create a report from a dimensionally-modeled relational data source
Demonstration 2:
Create a report from a dimensionally-modeled relational data source

Purpose:
You want to explore a dimensionally-modeled relational data source and create a report that enables you to drill down to a lower level of detail.

Task 1. Examine a dimensionally modelled relational data source.

1. From the side panel, click **New**, and then click **Report**.
2. Select **Blank**, and then click **OK**.
3. Using **Data > Source > Add report data**, navigate to **Team content > Samples > Models > GO data warehouse (analysis)**.
4. Click **Open**.
The Source tab on the left displays the folders available in the package. Notice the folder symbols.
5. Click **Add** in the center of the work area, click **List**, and then click **OK** to accept the default options presented.
6. Expand **Sales and Marketing (analysis)**.
   You see the namespaces in the Sales and Marketing (analysis) folder. Notice the namespace symbols.
7. Expand the **Sales** namespace.
   A section of the results appear as follows:

   ![Data Tree Example]

   The available measures and dimensions are displayed in the data tree.
   Notice the measures query subject and the dimensions.
**Task 2.  Continue examining the data source.**

1. **Expand the Sales fact measures query subject.**
   Review the measures available in the Sales fact measures query.

2. **Expand the Retailers dimension, and then expand the Retailers hierarchy.**
   Review the Members folder and its five levels.

3. **Expand the Region level.**
   The Members folder and the Region code query item display in the data tree.

4. **Expand the Members folder (under Region) to see the five sales regions.**
   The results appear as follows:

   ![Data Tree](image)

**Task 3.  Add items to the list report object.**

You want to create a report that shows the quantity of Star Dome tents sold in Canada in 2011. Because this is dimensionally-modeled relational data, you can drill down to a greater level of detail than in a relational model.

1. **Expand the Time dimension, Time hierarchy, Year level, and Members.**

2. **Drag 2011 to the list report object, in the work area.**
   Notice how you can add specific members to a report, instead of having all years added and filtering for only the years you want (as in relational data sources).

3. **Under the Retailers dimension, Retailers hierarchy, Region level, Members folder, expand the Americas member, and then drag the Canada member to the end of the list report object, dropping when the flashing black bar appears.**
4. Expand the **Products** dimension, **Products** hierarchy, **Product line** level, **Members** folder, **Camping Equipment** member, **Tents** member, and then drag **Star Dome** to the list report object.

5. Expand **Sales fact** measures (if necessary), and then drag the **Quantity** measure to the list report object.

The results appear as follows:

![Table](image_url)

**Task 4. Allow drill-up and drill-down on the report.**

1. From the side panel, click **Navigate**.

2. Under **Find**, click **Report**, and then on the **Application** bar, click **Show properties**.

3. In the **DATA** section, change the **Drill-up and drill-down** property to **Yes**.

4. Click **Page explorer**, and then click **Page 1**.

5. Select the entire List data container by clicking the container selection handle in the upper left corner of the List.

6. If necessary, open the **Properties** panel.

7. Click **Select Ancestor** (next to the List panel header title) in the upper left of the Properties panel, and then click **Report**.

8. In the **RUNNING & VALIDATING** section, change the property for **Run with full interactivity** to **No**.

9. On the **Application** bar, click **Run options**, and then click **Run HTML**.

You see that 2,403 Star Dome tents were sold in Canada in 2011.
10. Notice that 2011 appears as a hyperlink; click **2011** to drill-down for more detail. The results appear as follows, showing the quarterly time periods of 2011.

You can drill-down on any underscored data values.

11. Close the rendered report page (Reporting) tab to return to the work area.

12. Leave **IBM Cognos Analytics** open.

---

**Results:**
You have explored a dimensionally-modeled relational data source in IBM Cognos Analytics - Reporting. You created a report that demonstrated how you can drill down to a lower level of detail in the data source.
Examine personal data sources and data modules

IBM Cognos Analytics has the ability to allow you to import personal data sources like a CSV or XLS file. The personal data source can then be converted to a Data Module, after which it may be used as a data source for a report like any other.

The process begins by using the Upload files feature available from the Welcome screen. Next, the uploaded personal data source must be contained as a data module. This is done by selecting New > Data module. Data can be dragged from the Selected sources pane to the Data module pane to define the data module. Once it has been defined, the data module can be renamed and saved, making it available as a data sources for report creation.
Demonstration 3

Create a report from a personal data source

<table>
<thead>
<tr>
<th>Retailer country</th>
<th>Order method type</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Japan</td>
<td>Sales visit</td>
<td>106,513</td>
</tr>
<tr>
<td>Germany</td>
<td>Telephone</td>
<td>61,322</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>Mail</td>
<td>5,068</td>
</tr>
<tr>
<td>Denmark</td>
<td>Web</td>
<td>507,390</td>
</tr>
<tr>
<td>Singapore</td>
<td>Sales visit</td>
<td>176,874</td>
</tr>
<tr>
<td>Korea</td>
<td>Sales visit</td>
<td>343,578</td>
</tr>
<tr>
<td>Spain</td>
<td>Mail</td>
<td>31,371</td>
</tr>
</tbody>
</table>
Demonstration 3: Create a report from a personal data source

Purpose:
The purpose of this demonstration is to show you how to take a simple Microsoft Excel file and use its data as the source for a report.

Task 1. Upload a Microsoft Excel file.
1. In the Application bar, use the dropdown menu in the middle (displaying New*), and then click Welcome.
2. From the Welcome to IBM Cognos Analytics page, click Upload files from the side bar.
3. In the Choose File to Upload dialog box, browse to: C:\Training\B6058\Instructor Files.
4. Click the SampleFile_GOSales.xlsx file to select it, and then click Open.
The system will take a few moments (up to 2 minutes) to load the file. When complete, a preview of the data is displayed, such as Retailer country, Order method type, Retailer type, etc.
Note: If the upload is not complete after several minutes, restart the browser and repeat steps 1 to 4.
5. Click OK at the top of the page, once the load is complete.

Task 2. Create the data module.
1. Using the actions from the Step 1 of the previous task, open the Welcome page.
2. From the Welcome page, click New > Data module.
3. From the Sources list, click Uploaded files.
4. Select the checkbox for SampleFile_GOSales.xlsx (with the My content designation).
5. Click the Start button at the bottom left of the page.
6. Under Selected sources, expand SampleFile_GOSales.xlsx.
7. Expand Sample File Go Sales Xlsx, and then drag it to the bee icon under the Data module area.
8. Click the Actions to the right of New data module.
9. Click Rename, and then click the X to clear the textbox that appears.
10. In the textbox, type **Sales Data Module**.

11. Click **Save**，and then click **My content**.
12. In the **Save as** textbox, type **Sales Data Module**, and then click **Save**.

13. From the dropdown menu in the Application bar at the top, click **Remove** to the right of **Sales Data Module**.

**Task 3. Create a report from the data module.**

1. Return to the **Welcome** page, and then click **New > Report**.
2. Click the ellipsis to the right of the **Package** textbox, and then click **My content**.
3. Click **Sales Data Module**, and then click **Open**.
4. Click **List**, and then click **OK**.
5. On the side bar, click **Data**.
6. Expand **Sales Data Module > Sample File Go Sales Xlsx**.
7. Drag **Retailer country** to the list.
8. Drag **Order method type** to the right of **Retailer country**.
9. Drag **Quantity** to the right of **Order method type**.
10. At the top, click **Run options**, and then click **Run HTML**.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Retailer country</th>
<th>Order method type</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Japan</td>
<td>Sales visit</td>
<td>106,513</td>
</tr>
<tr>
<td>Germany</td>
<td>Telephone</td>
<td>61,322</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>Mail</td>
<td>5,068</td>
</tr>
<tr>
<td>Denmark</td>
<td>Web</td>
<td>507,390</td>
</tr>
<tr>
<td>Singapore</td>
<td>Sales visit</td>
<td>176,874</td>
</tr>
<tr>
<td>Korea</td>
<td>Sales visit</td>
<td>343,578</td>
</tr>
<tr>
<td>Spain</td>
<td>Mail</td>
<td>31,371</td>
</tr>
</tbody>
</table>

11. Close the rendered report page tab to return to the reporting work area.
12. Leave **IBM Cognos Analytics** open.

**Results:**

You created a report from a Microsoft Excel file. By uploading the file, and creating a data module from it, you were then able to take that result and create a standard list report.
Unit summary

• Examine IBM Cognos Analytics - Reporting and its interface
• Explore different report types
• Create reports in preview or design mode
• Create a simple, sorted, and formatted report
• Examine dimensionally modelled and dimensional data sources
• Explore how data items are added queries
• Examine personal data sources and data modules
Exercise 1: Create a revenue report

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Product</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Tents</td>
<td>Star Lite</td>
<td>168,191,550.48</td>
</tr>
<tr>
<td>Personal Accessories</td>
<td>Eyewear</td>
<td>Zone</td>
<td>157,369,344.95</td>
</tr>
<tr>
<td>Camping Equipment</td>
<td>Tents</td>
<td>Star Gazer 2</td>
<td>147,783,128.88</td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>Woods</td>
<td>Hailstorm Titanium Woods Set</td>
<td>117,598,685.56</td>
</tr>
<tr>
<td>Personal Accessories</td>
<td>Watches</td>
<td>TX</td>
<td>112,878,735.7</td>
</tr>
<tr>
<td>Personal Accessories</td>
<td>Eyewear</td>
<td>Inferno</td>
<td>104,705,055.75</td>
</tr>
<tr>
<td>Camping Equipment</td>
<td>Packs</td>
<td>Canyon Mule Journey Backpack</td>
<td>99,216,132.92</td>
</tr>
</tbody>
</table>
Exercise 1: Create a revenue report

Sales executives, from the Great Outdoors Samples Company, want you to create a report showing revenue for each product within each product type for each product line. The report must list the revenue from the greatest to the least.

- Create a list report using the GO data warehouse (query) package.
- Navigate to Sales and Marketing (query)/ Sales (query).
- Add the following query items to a new list report object in the order provided:
  - Products: Product line, Product type, and Product
  - Sales fact: Revenue
  - Sort Revenue in descending order.

For more information about where to work and the exercise results, refer to the Tasks and results section that follows. If you need more information to complete a task, refer to earlier demonstrations for detailed steps.
Exercise 1:
Tasks and results

Server: http://vclassbase:9300/bi
User/Password: brettonf/Education1

Task 1. Create a list report.

- **Toolbar:** Open a new List template using the Team content\Samples\Models\GO data warehouse (query) package.
- **Data/Source tab:** Navigate to Sales and Marketing (query) / Sales (query) / Products.
  - Add Product line, Product type, and Product to the List object.
- **Source tab:** Navigate to Sales and Marketing (query) / Sales (query) / Sales fact.
  - Add Revenue to the List object.

The results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Product</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product type&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product type&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
</tbody>
</table>

Task 2. Format and test the List object.

- **List object:** Click the <Revenue> list column body.
- **Toolbar:** Sort Revenue in descending order.
  - Run the report in HTML to test the new report.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Product</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Tents</td>
<td>Star Lite</td>
<td>168,191,550.48</td>
</tr>
<tr>
<td>Personal Accessories</td>
<td>Eyewear</td>
<td>Zone</td>
<td>157,369,344.95</td>
</tr>
</tbody>
</table>

- Close all web browser windows and tabs, and if prompted, select Leave this page.

Note: If your environment appears to not react to your clicks, there may be another browser tab in the background with a message awaiting your response. Try clicking the other open browser tab(s), and select Leave this page, when you see this prompt.

You created a report showing revenue for each product within each product type for each product line, and the list is sorted on revenue in descending order.
Create list reports

IBM Cognos Analytics (v11.0)
Unit objectives

- Group, format, and sort list reports
- Describe options for aggregating data
- Create a multi-fact query
- Create a report with repeated data
Examine list reports

- You can use list reports to:
  - present tabular information
  - show detailed information from your database

<table>
<thead>
<tr>
<th>Country</th>
<th>Employee name</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switzerland</td>
<td>Adriaantje Haanraads</td>
<td>$27,600,413.97</td>
</tr>
<tr>
<td>Spain</td>
<td>Agatha Reyes</td>
<td>$24,097,530.30</td>
</tr>
<tr>
<td>Japan</td>
<td>Aimi Tanaka</td>
<td>$16,468,860.28</td>
</tr>
</tbody>
</table>
Group data

- Group your data and choose how often to display item names by changing the group span properties.

### Group on Country and City

<table>
<thead>
<tr>
<th>Canada</th>
<th>Calgary</th>
<th>Tammy Sherwood</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Vittorio Rizzo</td>
</tr>
<tr>
<td></td>
<td>Toronto</td>
<td>Brendon Pike</td>
</tr>
</tbody>
</table>

### Group on Country and City with Group Span by City

<table>
<thead>
<tr>
<th>Canada</th>
<th>Calgary</th>
<th>Tammy Sherwood</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Vittorio Rizzo</td>
</tr>
<tr>
<td></td>
<td>Toronto</td>
<td>Brendon Pike</td>
</tr>
</tbody>
</table>

Group data

You can group on one or more columns depending on how you want to see your data.

The list report should preferably follow a 1:n cardinality from left to right in order to properly display the grouping.

Spanning one group of items by a second group can be helpful if the second group contains many items.

You can level span grouped items only by other grouped items on the report.

To group related information together, select a column and click Group/Ungroup on the toolbar. For example, when country and city are both grouped, you can choose to show the country name each time the country changes (span Country by Country), each time the city changes (span Country by City), or every time there is a new record (no level spanning).

A grouped item will appear at the top of a new page regardless of level spanning. For example, when Country is spanned by City, the Country name will appear at the top of the next page, even for records in the same City.

Grouping a column in a list generates an "order by" clause in the generated SQL, so your data is returned grouped and automatically sorted ascending.
Format list columns

- You can emphasize certain data to make your reports easier to read and understand.

<table>
<thead>
<tr>
<th>Order number</th>
<th>Retailer name</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>100003</td>
<td>Universo Acampando</td>
<td>2004</td>
</tr>
<tr>
<td>100009</td>
<td>Sporting Goods Direct</td>
<td>2004</td>
</tr>
</tbody>
</table>

Before

<table>
<thead>
<tr>
<th>Order number</th>
<th>Retailer name</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>100003</td>
<td>Universo Acampando</td>
<td>2004</td>
</tr>
<tr>
<td>100009</td>
<td>Sporting Goods Direct</td>
<td>2004</td>
</tr>
</tbody>
</table>

After

You can format list report columns at different levels depending on your requirements:

- lowest level: format the cells on a list column
- higher level: format both cells and the title in a list column
- highest level: format both the cells and titles in all columns in the list

In the slide example, if you wanted to sort the Product line column by Gross profit instead of by Product line, you would delete the Product line sort item from under the Product line Sort List folder, and would then drag the Gross profit query item to the Product line Sort List folder. Items in a report that are grouped appear under the Groups folder.

- You can modify the item used to sort a grouped item, add or remove a sort item, and determine the sort order. Click an object that can be sorted, on the toolbar, click Sort, and then click Advanced Sorting.
- Select the List object, and then in the Properties pane, double-click the Grouping & Sorting property.

The item used to sort specific grouped items in a report or to sort ungrouped items in a report does not need to be on the report page but does need to be in the query.
Include list headers and footers

- You can add headers and footers to a list report to provide additional information about the contents of the report.

<table>
<thead>
<tr>
<th>Country</th>
<th>City</th>
<th>Employee name</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austria</td>
<td>Wien</td>
<td>Jutta Shulz</td>
<td>29,274,108.98</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sabine Grüner</td>
<td>32,895,343.27</td>
</tr>
</tbody>
</table>

List headers and footers can be placed:
- at the top or bottom of a list on each page
- at the top of the first page or bottom of the last page
- before or after a group of details

Choose where to place headers and footers based on your requirements.
Demonstration 1

Enhance a list report

<table>
<thead>
<tr>
<th>Product type Sales and Revenue by Product</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Retailer type</strong></td>
</tr>
<tr>
<td>-------------------</td>
</tr>
<tr>
<td><strong>Attention: Sales Managers</strong></td>
</tr>
<tr>
<td>Outdoor Protection</td>
</tr>
<tr>
<td>First Aid</td>
</tr>
<tr>
<td>Aloe Relief</td>
</tr>
<tr>
<td>Department Store</td>
</tr>
<tr>
<td>Direct Marketing</td>
</tr>
<tr>
<td>Sports Store</td>
</tr>
<tr>
<td>Outdoors Shop</td>
</tr>
<tr>
<td>Warehouse Store</td>
</tr>
<tr>
<td>Golf Shop</td>
</tr>
<tr>
<td>Equipment Rental Store</td>
</tr>
<tr>
<td><strong>Aloe Relief</strong></td>
</tr>
<tr>
<td><strong>Total</strong></td>
</tr>
</tbody>
</table>

**Revenue by Retailer type**
Demonstration 1: Enhance a list report

**Purpose:**
Executives want you to create and format a report to highlight and sort the product lines based on the revenue that they generated. They also want you to highlight the retailer type and sort revenue descending by quantity sold.

- **Portal:** http://vclassbase:9300/bi
- **User/Password:** brettonf/Education1
- **Package:** Team content\Samples\Models\GO data warehouse (query)
- **Report Type:** List
- **Folder:** Sales and Marketing (query)
- **Namespace:** Sales (query)

**Note:** If you are unsure as to how to begin using the starting point information above, please refer to Unit 1, in the section titled Demonstration and exercise start point information.

**Task 1. Create the list.**

1. Add the following query items to a new list template object:
   - **Products:** Product line, Product type, Product
   - **Retailer type:** Retailer type
   - **Sales fact:** Quantity, Revenue

2. From the Application bar, click Run options, and then click Run HTML. A new web browser tab opens with the rendered report.
3. Click the Bottom navigation button found on the lower left of the page to view the final rows of the report, noting that there is no summary data.
   Due to the complexity of the final report, you will not include any summary row in your final report. This will make it easier for the consumer to review the data.
4. Close the rendered report tab and return to the report authoring work area tab.
Task 2. Group and span columns, and then add a report title.

1. In the list data container, click the <Product line> list column body, Ctrl-click the <Product type> and <Product> list column bodies, and then from the List toolbar, click Group / Ungroup.

Results appear as follows:

2. Click the <Product type> list column body.
3. On the application bar, click Show properties to open the Properties pane, and then in the DATA section, click Group span.
4. From the list, click Product, and then click Show properties to close the Properties pane.
5. On the side panel, click Toolbox, and then drag a Block to the left of the list container, dropping it when there is a flashing black vertical bar.
6. Drag a Text item inside the Block object that you just placed in the work area.
7. In the Text dialog box, type Product type Sales and Revenue by Product, and then click OK.
8. Click the title text, and from the Font list, click Arial Black.
9. Change the font size to 16pt, and then click Underline.

A section of the result appears as follows:
10. On the Application bar, click Run options, and then click Run HTML. A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Product</th>
<th>Retailer type</th>
<th>Quantity</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Cooking Gear</td>
<td>TrailChef Canteen</td>
<td>Department Store</td>
<td>211,339</td>
<td>2,426,658.9</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Direct Marketing</td>
<td>38,688</td>
<td>468,360.18</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Equipment Rental Store</td>
<td>6,641</td>
<td>72,910.87</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Outdoors Shop</td>
<td>222,831</td>
<td>2,682,916.23</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Sports Store</td>
<td>362,970</td>
<td>4,170,027.41</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Warehouse Store</td>
<td>123,254</td>
<td>1,512,645.05</td>
</tr>
<tr>
<td>Cooking Gear</td>
<td>TrailChef Cook Set</td>
<td>Department Store</td>
<td>229,456</td>
<td>11,509,856.38</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Direct Marketing</td>
<td>72</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Equipment Rental Store</td>
<td>15,597</td>
<td>824,622.11</td>
</tr>
</tbody>
</table>

Product type is spanned by Product. Every time Product changes the Product type is repeated.

11. Close the rendered report tab.

**Task 3. Add a list page header, an overall header, and a group header.**

You want to add a list page header for the report and an overall header to add additional information to the report.

1. Click the list data container to select it, and then from the container toolbar, click **Headers & footers**

2. Click **List headers & footers**.

3. Select **List page header** and **Overall header**, and then click **OK**.

4. In the list data container, double-click **List page header**.

5. In the **Text** box, replace the default text with **Revenue by Retailer type**, and then click **OK**.

6. With the list page header still selected, on the toolbar click **More**, point to **Style**, and then click **Font**.

7. In the **Font** dialog box, change **Family** to **Arial Black**, change **Size** to **12 pt.**, and then click **OK**.

8. In the list data container, double-click **Overall**.

9. In the **Text** box, replace the default text with **Attention: Sales Managers**, and then click **OK**.
10. In the list data container, click the <Product line> list column body.
11. On the toolbar, click Headers & footers, and then click Create header.
    When a header is created from a column, the header stays within the list object. You cannot create a header from a spanned column. Also, the list column titles can be moved to the start of the details of the report by selecting the container object and changing the GENERAL/Column titles property to "At start of details".
12. With the <Product line> list column body still selected, press the Delete key to remove the redundant column.
    A section of the results appear as follows:

    ![Diagram of list report with headers and footers created and column titles moved]

13. Run the report in HTML.
    A section of the results appear as follows:

    ![HTML rendering of list report with headers and footers]


**Task 4. Format and sort a column.**

1. In the list data container, click <Revenue>.
2. On the List toolbar, click Sort, and then click Descending.
    When a column is sorted the Sort icon appears in the list column title cell.
3. With the <Revenue> list column body still selected, on the List toolbar, click More, point to Style, and then click Data format.
4. In the Data format dialog box, under Format type, select Currency.
5. Under Properties, click Currency.
6. From the list, select $ (USD) United States of America, dollar, and then click OK.

**Task 5. Format the list column body.**

1. Click the <Retailer type> list column body.
2. On the List toolbar, click More.
3. Point to Style, click Font, change Family to Arial, and then change Style to Italic.
4. Click Foreground Color, click Purple, click OK, and then click OK again.

The font properties are applied to the body cells in the Retailer type column.

A section of the results appear as follows:

![Product type Sales and Revenue by Product](image)

**Task 6. Format a column.**

1. With <Retailer type> list column body still selected, from the Application bar, click Show properties.
2. On the Properties pane title bar, click Select Ancestor, and then click List column.
3. In the Properties pane, under FONT & TEXT, double-click the Font property; change the properties to Arial, 12pt, Bold, and then change the Foreground Color to Green.
4. Click **OK** to close the **Foreground Color** dialog box, and then click **OK** to close the **Font** dialog box.

A section of the results appear as follows:

```plaintext
The color property is applied only to the column title because the list column body formatting overrides the list column formatting. However, because you have not set the size or weight for the list column body, the value in the cells now appears in 12pt bold font.
```

**Task 7. Sort the Product line column by the Revenue generated.**

1. Click the `<Revenue>` list column body, on the list toolbar click **Summarize** and then click **Total**.
2. In the upper left corner of the **Product type** header cell click the **Container Selector** to select the entire list.
   
   You may need to click Esc to clear the list toolbar so that you can see the list column headers.
3. On the **Application** bar, click **Show properties**, if it isn't already displayed.
4. In the **Properties** pane, under **DATA**, double-click **Grouping & sorting**.
   
   Because Product line, Product type, and Product are grouped, these items appear under the Groups folder.
5. In the Groups pane, expand Product line, and then from the Data items pane, drag Revenue onto the Product line / Sort list folder.

The results appear as follows:

![Diagram of grouping and sorting]

The Product line column will now be sorted in ascending order based on the revenue generated by each product line. The product line that generated the least revenue will appear at the beginning of the report.
6. Click OK, and then, on the Application bar, run the report in HTML. A section of the results appear as follows:

Since Outdoor Protection generated the least revenue, it appears at the beginning of the report.

7. Close the rendered report tab.

8. On the Application bar, close the Properties pane.

9. Return to the Welcome screen.

Results:
You have created a list report that grouped Product line, Product type, and Product name. You highlighted retailer type; and you have sorted revenue in descending order according to the quantity sold.
Understand fact/measure data

- You can aggregate fact data to show trends or summaries.

<table>
<thead>
<tr>
<th>Employee name</th>
<th>Product line</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agatha Reyes</td>
<td>Camping Equipment</td>
<td>9,596,483.77</td>
</tr>
<tr>
<td></td>
<td>Golf Equipment</td>
<td>1,966,340.45</td>
</tr>
<tr>
<td></td>
<td>Mountaineering Equipment</td>
<td>5,546,852.83</td>
</tr>
<tr>
<td></td>
<td>Outdoor Protection</td>
<td>991,736.35</td>
</tr>
<tr>
<td></td>
<td>Personal Accessories</td>
<td>5,996,116.9</td>
</tr>
<tr>
<td><strong>Agatha Reyes</strong></td>
<td><strong>Total</strong></td>
<td><strong>24,097,530.3</strong></td>
</tr>
</tbody>
</table>

Show minimum, maximum, average, total, count, or calculated data.

The Rollup Aggregate Function specifies the type of aggregation to apply to summarize values. These values appear at the higher levels of list and crosstabs. The default setting is Automatic. The setting of Automatic indicates that the aggregation applied is based on the data type of the query item. Therefore, an integer data type with rollup aggregation set to automatic provides total aggregation. The report on the slide illustrates rollup aggregation set to Total.

The Aggregate function specifies the type of aggregation to apply to individual values which appear as detail rows in lists or crosstabs.

These property values and many others can be set for all authors by the modelers in Framework Manager to centralize administration.
Understand aggregate data

- You can show your data as summarized aggregated data or as detailed non-aggregated data.

By default, the data will be grouped and summarized, at its lowest level of detail, because of the Auto Group and Summarize property that is applied to the query. This aggregation is applied at the initial query.

The rollup aggregated function summarizes grouped data and is applied after data is retrieved.

The list on the left shows a list report with all of the default aggregation settings and no grouping applied; Aggregate Function is set to Total, by default, in the model package, Rollup Aggregation is set to Automatic since there is no grouping.

**Aggregate function**: aggregates items at the lowest level of detail and is set by the data modeler for the package. This aggregation is applied only when the query’s Auto Group and Summary is set to Yes.

**Rollup aggregation**: is applied, by the report author, to grouped items and provides a higher level aggregation, as seen by the center list report.

The list on the right shows results with the query’s Auto Group and Summary set to No.
Understand difference in aggregation

- You can use data items for your query from the Source tab or the Data Items tab.

Data items selected from the source tab will be calculated and summarized prior to aggregation.

Data items selected from the Data Items tab will be calculated and summarized after aggregation.

Fact data items should be selected from the Data Items tab if they are to be used multiple times in a report or calculation, since they would not be re-aggregated based upon the entire query. This prevents any double counting of the fact data item and provides predictable results.
## Demonstration 2

Explore data aggregation

<table>
<thead>
<tr>
<th>Product line</th>
<th>Order method type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>E-mail</td>
<td>75,899.954.63</td>
</tr>
<tr>
<td></td>
<td>Fax</td>
<td>23,054.398.46</td>
</tr>
<tr>
<td></td>
<td>Mail</td>
<td>21,348.844.09</td>
</tr>
<tr>
<td></td>
<td>Sales visit</td>
<td>100,611.961.87</td>
</tr>
<tr>
<td></td>
<td>Special</td>
<td>12,308.969.44</td>
</tr>
<tr>
<td></td>
<td>Telephones</td>
<td>153,864.892.13</td>
</tr>
<tr>
<td></td>
<td>Web</td>
<td>1,133,838.883.39</td>
</tr>
</tbody>
</table>

**Camping Equipment - Average**
227,805,257.719571

<table>
<thead>
<tr>
<th>Product line</th>
<th>Order method type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Golf Equipment</td>
<td>E-mail</td>
<td>47,903.933.16</td>
</tr>
<tr>
<td></td>
<td>Fax</td>
<td>15,241.303.27</td>
</tr>
<tr>
<td></td>
<td>Mail</td>
<td>12,693.287.48</td>
</tr>
<tr>
<td></td>
<td>Sales visit</td>
<td>39,240.918.73</td>
</tr>
<tr>
<td></td>
<td>Special</td>
<td>4,904.762.97</td>
</tr>
<tr>
<td></td>
<td>Telephones</td>
<td>70,730.112.56</td>
</tr>
<tr>
<td></td>
<td>Web</td>
<td>527,607.049.63</td>
</tr>
</tbody>
</table>

**Golf Equipment - Average**
103,773,062.55574

*Demonstration 2: Explore data aggregation*
Demonstration 2: Explore data aggregation

Purpose:
You have been asked by management to create a report that compares how different order methods are performing for each product line. This report should display the revenue that individual order methods generate for each product line and the average revenue all order methods generate for each product line. You will create this report and examine the underlying query model at various stages.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create a basic report and examine the query model.

2. Add the following query items to a new List template, using the GO data warehouse (query) package:
   - Products: Product line
   - Order method: Order method type
   - Sales fact: Revenue

3. On the side panel, click Navigate, click the Query explorer tab, and then under Queries, click Query1.
   Note the three data items in the Data Items pane. Each data item corresponds to an item in a column in the list report.
4. In the **Data Items** pane, click **Revenue**, and then click **Show properties** from the **Application** bar.

In the Properties pane notice that the Detail aggregation property is set to Total. When the query groups and summarizes data at the lowest level of detail, the query will summarize data by calculating the total revenue generated at the lowest level of detail. In our report, the lowest level of detail is Revenue generated by each Order method type.

You have not yet added any aggregate revenue values for grouped data items in report layout. Therefore, the Summary aggregation, Detail property for Revenue is set to Automatic.

5. Run the report in **HTML**.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Order method type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>E-mail</td>
<td>75,899,094.63</td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>E-mail</td>
<td>47,933,933.16</td>
</tr>
<tr>
<td>Mountaineering Equipment</td>
<td>E-mail</td>
<td>7,476,451.96</td>
</tr>
<tr>
<td>Outdoor Protection</td>
<td>E-mail</td>
<td>5,882,477.87</td>
</tr>
<tr>
<td>Personal Accessories</td>
<td>E-mail</td>
<td>42,651,086.54</td>
</tr>
<tr>
<td>Camping Equipment</td>
<td>Fax</td>
<td>23,054,398.48</td>
</tr>
</tbody>
</table>

6. You can examine the revenue generated by each product line using each order method.

7. Close the rendered report tab.

**Task 2. View individual records rather than data grouped and summarized at the lowest level of detail.**

You would like to review the amount of revenue generated by each order made using a particular Order method type for each product line. To achieve this result, you will set the Auto Group & Summarize property for this query to No.

1. Verify that **Query1** is selected.
2. From the **Properties** pane, under **DATA**, click the **Auto group & summarize** property, and then change it to **No**.
3. Run the report in HTML.
A section of the results appears as shown below:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Order method type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Golf Equipment</td>
<td>Telephone</td>
<td>10,469.76</td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>Telephone</td>
<td>41,958.76</td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>Telephone</td>
<td>35,949.86</td>
</tr>
</tbody>
</table>

The report no longer displays a single row for the total revenue generated by all sales of each product line using a specific order method type. Instead, it displays individual rows containing the revenue generated by each individual sale that used a specific order method type for each product line.

For example, a row in the report displays data for a golf equipment sale made by telephone. This sale generated $41,958.76 in revenue.

If you wanted to display these individual records in your final report, you would group and sort this data to make it easier to read. However, you decide you would prefer to have this data grouped and summarized at the lowest level of detail.

4. Close the rendered report tab.
5. From the Properties pane for Query1, click the Auto group & summarize property, and then change it back to Yes.

Task 3. **Group query items, add aggregate data, and observe the results in the query.**

As requested, you will now group this data by product line and add aggregate data to display the average revenue generated by all order method types for each product line.

1. On the Content pane, click Page explorer, and then click Page1.
2. In the list, click the <Revenue> list column body, and then under DATA in the Properties pane, double-click Data format.
3. Change the Format type to Currency.
4. Change the Currency property to USD - United States of America, dollar, and then click OK.
5. In the list, click the `<Product line>` list column body, and then on the container toolbar, click **Group / Ungroup**.

The product line column is grouped and you can now include aggregate data at a higher level of detail. You want to see the average revenue generated by all order method types for each product line, and for all product lines.

6. In the list, click the `<Revenue>` list column body.

7. On the container toolbar, click **Summarize**, and then click **Average**.

The results appear as follows:

![Show results](image)

You will examine how the aggregation you specified has changed the Rollup Aggregate Function for the Revenue data item in this query.

8. On the **Content** pane, click the **Query explorer** tab.

9. Click **Query1**, and then in the **Data Items** pane, click **Average(Revenue)**.

In the Properties pane, notice that the Summary property for `Average (Revenue)` is now set to `Average`. This is because you have specified that revenue for grouped items in the report be aggregated to display the average revenue generated.
10. Run the report in **HTML**.

   A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Order method type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>E-mail</td>
<td>$75,899,094.63</td>
</tr>
<tr>
<td></td>
<td>Fax</td>
<td>$23,054,398.48</td>
</tr>
<tr>
<td></td>
<td>Mail</td>
<td>$21,348,644.09</td>
</tr>
<tr>
<td></td>
<td>Sales visit</td>
<td>$168,611,961.87</td>
</tr>
<tr>
<td></td>
<td>Special</td>
<td>$12,388,989.44</td>
</tr>
<tr>
<td></td>
<td>Telephone</td>
<td>$153,894,892.13</td>
</tr>
<tr>
<td></td>
<td>Web</td>
<td>$1,133,838,683.39</td>
</tr>
</tbody>
</table>

   In this report, data is grouped by product line. Below each product line row is an aggregate row displaying the average revenue generated by all order method types for that product line.

   You can see that for all product lines, revenue generated by the Web method far exceeded those of other order methods.

11. Close the rendered report tab.

**Task 4. View tabular data.**

   1. Click **Query1**.
   2. Right-click the **Query Explorer** content pane, and then click **View Tabular Data**.
   3. Click **OK** to the warning message.

   Notice that although you grouped the Product line data item in the report layout, in the tabular data retrieved for the query, product line data is still ungrouped. This option retrieves the data without any grouping or formatting.

   4. Close the rendered report tab.
   5. Return back to the **Welcome** page.

**Results:**

You created a list report displaying revenue generated by each order method for each product line and the average revenue all order methods generate for each product line. You also specified that the query should display individual data records instead of grouped and summarized data, and you then compared the results.
Use shared dimensions to create multi-fact queries

• When authoring reports with multiple facts across the business, it is necessary to use at least one shared dimension item to ensure correlated and predictable results.

A shared dimension is created by the data modeler to provide consistent results throughout the company's different business units. When business units report with these shared query items, they communicate more efficiently as a whole by providing the same base of information.

Shared dimensions are also known as conformed dimensions.

Results of multiple-fact queries can vary if the level of granularity differs or you use a non-conformed dimension. For example, in the GO Data Warehouse (query) package, the granularity for time differs between Sales target and Revenue. Sales targets are recorded monthly, whereas, the Revenue is recorded on a daily basis. This is not an issue when reporting and will not cause confusing results if you report at a common level of granularity, such as in this case, the month level. If you report at the day level, inventory levels will simply display repeating values, the month total for every day of the month in the report. These values will not be double-counted.
Demonstration 3

Create a multi-fact query in a list report

<table>
<thead>
<tr>
<th>Year</th>
<th>Revenue</th>
<th>Sales target</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010</td>
<td>914,352,803.72</td>
<td>812,885,300</td>
</tr>
<tr>
<td>2011</td>
<td>1,159,195,590.16</td>
<td>1,036,923,300</td>
</tr>
<tr>
<td>2012</td>
<td>1,495,891,100.9</td>
<td>1,332,553,100</td>
</tr>
<tr>
<td>2013</td>
<td>1,117,336,274.07</td>
<td>1,023,006,840</td>
</tr>
</tbody>
</table>

Demonstration 3: Create a multi-fact query in a list report
Demonstration 3:
Create a multi-fact query in a list report

Purpose:
You have been asked to create a report showing sales revenue and target revenue for each year. You will need to use conformed query items in the report to ensure the results are accurate and consistent with expected results.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content/Samples/Models/GO data warehouse (query)
Report Type: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)/Sales target (query)

Task 1. Add two facts from different query subjects to a list report.

1. Add the following query items to a new list data container, using the GO data warehouse (query) package:
   - Sales (query)/Sales fact: Revenue
   - Sales target (query)/Sales target fact: Sales target

2. On the Application bar, run the report in HTML.
   The results appears as follows:

   ![Revenue and Sales target table]

   These are the two distinct aggregated totals for Revenue and Sales target. These values were returned as a result of two separate Select statements.

3. Close the rendered report tab.
Task 2. Add context to the list.

You will include a query item to give context and meaning to the performance indicators that are already in the list. You will add the year in which the orders closed as a point in time to compare revenue to sales target.

1. From the Sales (query) namespace, add the following query item to the beginning of the report:
   - Time (close day): Year (close date)

2. On the Application bar, run the report in HTML.

   The results appear as follows:

<table>
<thead>
<tr>
<th>Year (close date)</th>
<th>Revenue</th>
<th>Sales target</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010</td>
<td>907,292,137.51</td>
<td>4,205,368,540</td>
</tr>
<tr>
<td>2011</td>
<td>1,144,204,628.01</td>
<td>4,205,368,540</td>
</tr>
<tr>
<td>2012</td>
<td>1,497,596,605.86</td>
<td>4,205,368,540</td>
</tr>
<tr>
<td>2013</td>
<td>1,137,682,397.47</td>
<td>4,205,368,540</td>
</tr>
</tbody>
</table>

   The Revenue values change with each year, but the Sales target values do not. This is because the Time (close day) is not a conformed dimension. This dimension is not shared by both the Revenue and Sales target facts. The Sales target fact has no relationship to Time (close day).

3. Close the rendered report tab.

Task 3. Add a query item from a shared dimension to the list report.

You will add a shared dimension to the report. This dimension will have a relationship to both Revenue and Sales target.

1. Under Sales target (query), point to Time.

   The Sales target (query) namespace contains a query object called Time. Notice there is no query object called Time (close date), which confirms what you already saw from running the report: Time (close date) is not shared across the facts.

2. Under Sales (query), point to Time.

   Time exists in both the Sales target (query) and the Sales (query) namespaces; therefore, it is a shared dimension.

3. Under Sales (query), expand Time, and then drag Year to the beginning of the list.
4. On the **Application** bar, run the report in **HTML**.
   A section of the results appear as follows:

   ![Table](image)

   The Sales target numbers now change from year to year. In 2010, there was 7,060,666.21 worth of orders that were placed in that year, but did not close until 2011. The orders that were placed in 2010 and closed in that same year totaled 907,292,137.51. Because Sales target has no relationship to the non-conformed dimension, Year (close date), it just repeats the value it knows for 2010. This is an example of the inaccurate results that can occur when using non-conformed query items with multi-fact reports. Therefore, you should use conformed query items.

5. Close the rendered report tab.

**Task 4. Delete a query item from the list report.**

You want to delete the Year (close date) query item and only have the Year query item, from a conformed dimension, in the list.

1. In the list, click the **Year (close date)** list column body, on the list toolbar, click **More**, and then click **Delete**.

2. Run the report in **HTML**.
   The results appear as shown below:

   ![Table](image)

   The Revenue and Sales target numbers now change from year to year. The report runs as expected.
3. Close the rendered report tab.
4. Leave the reporting tab open for the next demonstration.

Results:
You created a report showing sales revenue and target revenue for each year. You used a conformed dimension in the report to ensure the results were accurate and consistent with expected results.
Add repeated information to reports

- You can use either repeaters or repeater tables to present repeated information.

**Repeater table**

<table>
<thead>
<tr>
<th>Mailing List</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address line 1</td>
</tr>
<tr>
<td>Address line 2</td>
</tr>
<tr>
<td>Address line 3</td>
</tr>
</tbody>
</table>

Add repeated information to reports

Use repeaters to duplicate individual item(s) across a single row without a particular structure.

Use repeater tables to repeat items in a table structure, such as mailing label information.
Demonstration 4

Create a mailing list report

<table>
<thead>
<tr>
<th>Australia</th>
<th>Austria</th>
<th>Belgium</th>
</tr>
</thead>
<tbody>
<tr>
<td>2315 Queen's Ave Level 2 Melbourne VIC 2088 Australia</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Jedlaser Straße 7 Wien A-1210 Austria</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Interleuvenlaan 2 Heverlee B-3001 Belgium</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Brazil</th>
<th>Canada</th>
<th>Canada</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avenida Paulista, 333 CJ 231 2o. Andar São Paulo SP 01403-090 Brazil</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>789 Yonge Street Toronto Ontario M2M 4K8 Canada</td>
<td></td>
</tr>
<tr>
<td></td>
<td>7800, 756 - 6th Avenue, S.W. Calgary Alberta T2P 3Z0 Canada</td>
<td></td>
</tr>
</tbody>
</table>

Create list reports

© Copyright IBM Corporation 2016

Demonstration 4: Create a mailing list report
Demonstration 4: Create a mailing list report

Purpose:
You will create a mailing list for all of your sales offices. The addresses must be listed alphabetically by county with the country name appearing at the top. For easy readability, each page must contain no more than three addresses across and four down.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content/Samples/Models/GO data warehouse (query)
Report Type: Repeater Table
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create a repeater table.

1. Open a new Repeater table template, using the GO data warehouse (query) package.
2. From the Toolbox, expand LAYOUT.
3. Drag a Table data container to the Repeater table drop zone, at the top of the work area.
4. In the **Insert table** dialog box, change the number of columns to 1, the number of rows to 7, and then click **OK**.

The results appear as follows: The work area contains a two-column, three-row repeater table containing six tables, each having one column and seven rows.

A section of the results appears as follows:

![Task 2. Add items to the tables.](image)

**Task 2. Add items to the tables.**

1. Click **Data**.
2. From the **Source** tab, navigate to **Sales and Marketing (query)/Sales (query)**.
3. Expand **Employee by region**, and then drag **Country** into the first cell of the first 1x7 table.
4. Drag **Address 1**, **Address 2**, **City**, **Province or State**, **Postal zone**, and again **Country** into the remaining table cells.

A section of the results appear as follows:

```
<table>
<thead>
<tr>
<th>&lt;Country&gt;</th>
<th>&lt;Country&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Address 1&gt;</td>
<td>&lt;Address 1&gt;</td>
</tr>
<tr>
<td>&lt;Address 2&gt;</td>
<td>&lt;Address 2&gt;</td>
</tr>
<tr>
<td>&lt;City&gt;</td>
<td>&lt;City&gt;</td>
</tr>
<tr>
<td>&lt;Province or State&gt;</td>
<td>&lt;Province or State&gt;</td>
</tr>
<tr>
<td>&lt;Postal zone&gt;</td>
<td>&lt;Postal zone&gt;</td>
</tr>
<tr>
<td>&lt;Country&gt;</td>
<td>&lt;Country&gt;</td>
</tr>
</tbody>
</table>
```

When you add multiple instances of the same data item (as in this case when you added the same Country item twice) the second and subsequent items will be numbered to show that it is a duplicate entry. An alternative would have been to drag Country from the Data Items tab.

**Task 3. List countries in alphabetical order and apply a style to the headers.**

1. Click the `<Country>` item at the top of one of the tables, ensuring you select only the item and not the entire cell.
   All of the Country items at the top of each table are selected.
2. On the List toolbar, click **Sort**, and then click **Ascending**.
   A Sort Ascending icon appears beside the Country item in the first table.
3. With the `<Country>` item still selected, from the toolbar expand the **Size** list, and then select **12 pt**.
4. From the toolbar, click **Bold**.
   The `<Country>` items appear in bold, black text.
Task 4. Change the frequency and positioning of the tables.

1. Click the **Container Selector** in the top left-hand corner of the **Repeater Table**, to select the entire container.

2. In the **Properties** pane, under **GENERAL**, change the **Across** property to a value of 3, change **Down** to 4, and then press **Enter**.

3. In the **Properties** pane, under **POSITIONING** (you may need to scroll down), double-click **Table properties**, select **Fixed size**, and then click **OK**.

4. Click the **Container Selector** in the top left corner of the first table, to select all of the tables (you should only see the tables selected, not the entire list).

5. In the **Properties** pane, under the **BOX** section, double-click **Margin**.

6. In the **Right margin** and **Top margin** text boxes, type 10, and then click **OK**. This adds the appropriate space for the printed labels.

7. On the **Application** toolbar, run the report in **PDF**. PDF would be the appropriate run output for mailing labels.

A section of the results appear as follows:

```
<table>
<thead>
<tr>
<th>Australia</th>
<th>Austria</th>
<th>Belgium</th>
</tr>
</thead>
<tbody>
<tr>
<td>2315 Queen's Ave</td>
<td>Jedlese Straße 7</td>
<td>Interluevenaun 2</td>
</tr>
<tr>
<td>Level 2</td>
<td>Wien</td>
<td>Heverlee</td>
</tr>
<tr>
<td>Melbourne</td>
<td>A-1210</td>
<td>B-3001</td>
</tr>
<tr>
<td>VIC</td>
<td>Austria</td>
<td>Belgium</td>
</tr>
<tr>
<td>2088</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Australia</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Brazil</th>
<th>Canada</th>
<th>Canada</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avenida Paulista, 333</td>
<td>789 Yonge Street</td>
<td>7800, 756 - 6th Avenue, S.W.</td>
</tr>
<tr>
<td>C2 231 20, Andar</td>
<td>Toronto</td>
<td>Calgary</td>
</tr>
<tr>
<td>São Paulo</td>
<td>Ontario</td>
<td>Alberta</td>
</tr>
<tr>
<td>SP</td>
<td>M2M 4K8</td>
<td>T2P 320</td>
</tr>
<tr>
<td>01403-090</td>
<td>Canada</td>
<td>Canada</td>
</tr>
<tr>
<td>Brazil</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

8. Close the rendered report tab.

9. Leave report authoring tab open for the exercise.

**Results:**
You created a mailing list and added the country name at the top of each address as a header and displayed the addresses alphabetically by country. The addresses were displayed, with no more than three addresses across and four down each page.
### Unit summary

- Group, format, and sort list reports
- Describe options for aggregating data
- Create a multi-fact query
- Create a report with repeated data
Exercise 1
Create and format a list report

Gross Profit by Retailer Type and Region

<table>
<thead>
<tr>
<th>Retailer type</th>
<th>Region</th>
<th>Gross profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department Store</td>
<td>Americas</td>
<td>111,543,822.41</td>
</tr>
<tr>
<td></td>
<td>Asia Pacific</td>
<td>98,425,260.8</td>
</tr>
<tr>
<td></td>
<td>Central Europe</td>
<td>77,587,318.45</td>
</tr>
<tr>
<td></td>
<td>Northern Europe</td>
<td>39,559,096.87</td>
</tr>
<tr>
<td></td>
<td>Southern Europe</td>
<td>36,177,713.46</td>
</tr>
<tr>
<td>Department Store - Total</td>
<td></td>
<td>363,293,214.09</td>
</tr>
</tbody>
</table>

Create list reports

Exercise 1: Create and format a list report
Exercise 1: Create and format a list report

You have been asked to create a list report where users can review the gross profit generated by retailer types for each region.

To accomplish this:

- Create a list report using the GO data warehouse (query) package.
- Add the following items:
  - Retailer type: Retailer type
  - Retailers: Region
  - Sales fact: Gross profit
  - Group Retailer type.
  - Sort Gross profit as descending.
  - Aggregate Gross profit by Total.

For more information about where to work and the exercise results, refer to the Tasks and Results section that follows. If you need more information to complete a task, refer to earlier demonstrations for detailed steps.
Exercise 1: Tasks and results

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content/Samples/Models/GO data warehouse (query)
Report Template: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create, group, and sort a list.

- Side panel: Open a new list report template using the GO data warehouse (query) package.
- Data tab: Navigate to Sales and Marketing (query)/Sales (query).
  - From the Retailer type query subject, add the Retailer type query item, to the list report object.
  - From the Retailers query subject, add the Region query item, to the list report object.
  - From the Sales fact query subject, add the Gross profit query item, to the list report object.
- List Toolbar: Group the <Retailer type> list column body.
- Sort the <Gross profit> list column body in descending order.

The results appear as follows:

![Image of list report columns]

Task 2. Format and summarize the list report.

- Change the report title to Gross Profit by Retailer Type and Region.
- Toolbar: Left-justify the header block (not the text item).
- Change the report title font to Arial Black.
- Summarize the <Gross profit> list column body, by Total.
Run the report in **HTML**.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Retailer type</th>
<th>Region</th>
<th>Gross profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department Store</td>
<td>Americas</td>
<td>111,543,822.41</td>
</tr>
<tr>
<td></td>
<td>Asia Pacific</td>
<td>98,425,260.8</td>
</tr>
<tr>
<td></td>
<td>Central Europe</td>
<td>77,587,318.45</td>
</tr>
<tr>
<td></td>
<td>Northern Europe</td>
<td>39,559,098.97</td>
</tr>
<tr>
<td></td>
<td>Southern Europe</td>
<td>36,177,713.46</td>
</tr>
</tbody>
</table>

**Department Store - Total** 363,293,214.09

You have created a list report where users can review the gross profit generated by retailer types for each region.

- Close the rendered report tab.
- Sign out of **IBM Cognos Analytics**, if prompted click **OK** to continue without saving and then close the browser.
Unit objectives

• Create filters to narrow the focus of reports
• Examine detail and summary filters
• Determine when to apply filters on aggregate data
Create filters

- To narrow the focus of your report, you can create a filter expression in three different ways:

  - Custom based on data item
  - Combined
  - Advanced
Filter your data with advanced detail filters

- Create a detail filter to narrow your focus and report on specific data.

**Filter to show only sales revenue greater than $100,000**

<table>
<thead>
<tr>
<th>Expression Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Revenue]&gt;100000</td>
</tr>
</tbody>
</table>

**Filter to show only data from January to June for the year 2012**

<table>
<thead>
<tr>
<th>Expression Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Sales (query)].[Time].[Date] between 2012-01-01 and 2012-06-30</td>
</tr>
</tbody>
</table>

When you create a filter, you define conditions around query items to report on a specific subset of data.

A detail filter will be applied to all rows in the report.

For detail filters, filter any item in the package using the Source tab, or filter items in the report using the Data Items tab or Queries tab. Use the Functions tab to create filter calculations. Use the Parameters tab to use existing filters.
Demonstration 1: Apply filters to a report
Demonstration 1: 
Apply filters to a report

Purpose:
The Vice President of Sales has requested a report that shows sales performance in each country for 2012. He wants to see the performance for representatives in Southern Europe so he can present an award to the top seller when he visits next month.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the list.
1. Add the following query items to a new list template:
   - Employee by region: Country, City, First name, Last name, Position name
   - Sales fact: Revenue

<table>
<thead>
<tr>
<th>Country</th>
<th>City</th>
<th>First name</th>
<th>Last name</th>
<th>Position name</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Country&gt;</td>
<td>&lt;City&gt;</td>
<td>&lt;First name&gt;</td>
<td>&lt;Last name&gt;</td>
<td>&lt;Position name&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Country&gt;</td>
<td>&lt;City&gt;</td>
<td>&lt;First name&gt;</td>
<td>&lt;Last name&gt;</td>
<td>&lt;Position name&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Country&gt;</td>
<td>&lt;City&gt;</td>
<td>&lt;First name&gt;</td>
<td>&lt;Last name&gt;</td>
<td>&lt;Position name&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
</tbody>
</table>

2. Ctrl-click <Country> and <City>, and then on the list toolbar, click Group / Ungroup.
3. Click <Country>, on the list toolbar click Headers & footers, and then click Create header.
4. With <Country> still selected, press the Delete key to delete the redundant <Country> list column body.
5. Click the <Revenue> list column body, on the list toolbar click Summarize, and then click Total.
6. Click the <Revenue> list column body, on the list toolbar click Sort, and then click Descending.
7. Run the report in **HTML**.
   A section of the results appear as follows:

   ![HTML Report](image)

8. Close the rendered report tab.

**Task 2. Add a filter to show sales from 2012.**

1. Select the list data container by clicking 📊 in the upper left corner of the list.

2. On the list toolbar, click **Filters** 🔄, and then click **Edit Filters**.
   The Filters - Query 1 dialog box appears. There are two tabs: one for creating filters at the detail level, and one for creating filters at the summary level.

3. With the **Detail Filters** tab selected, click **Add** 📊, click **Advanced**, and then click **OK**.

4. Under **Available Components**, from the **Source** tab, expand **Sales and Marketing (query)**, expand **Sales (query)**, and then expand **Time**.
5. Create and validate the following expression. (Using the Hint outlined below, you can create the expression differently):

\[ \text{[Sales (query)].[Time].[Year]=2012} \]

**Hint:**
- Drag Year from the Time query subject, into the Expression Definition pane. There are different ways of creating filters to achieve the same result:
- create the expression \[ \text{[Sales (query)].[Time].[Date]between 2012-01-01 and 2012-12-31} \]
- create filters by adding operators and conditions to query items using SQL syntax
- Click Validate to check the syntax of the expression.

6. Click **OK** to close the Detail filter expression dialog box, and then click **OK** to close the Filters - Query1 dialog box.

7. Run the report in **HTML**.

8. At the bottom of the page, click **Bottom** to navigate to the end of the report. A section of the results appear as follows:

<table>
<thead>
<tr>
<th>City</th>
<th>Sales Rep</th>
<th>Sales Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seattle</td>
<td>George Harrows</td>
<td>17,924,373.12</td>
</tr>
<tr>
<td>Bart</td>
<td>Scott Lastman</td>
<td>14,538,997.37</td>
</tr>
<tr>
<td>Audrey</td>
<td>Lastman</td>
<td>13,535,227.17</td>
</tr>
<tr>
<td>Melanie</td>
<td>White</td>
<td>6,906,978.7</td>
</tr>
</tbody>
</table>

**Seattle - Total**: 52,905,576.36
**United States - Total**: 164,986,189.21
**Overall - Total**: 1,495,891,100.9

Only 2012 sales are included in the report. On the last page of the report, the Overall - Total revenue is $1,495,891,100.90 for 2012.

9. Close the rendered report tab.
Task 3. Filter data to show only Southern European countries.

The Southern European countries consist of Austria, Italy, and Spain.

1. Select the list data container, on the list toolbar, click Filters, and then click Edit Filters.

   The Filters - Query 1 dialog box appears showing the detail filter you just created. You will create another detail filter.

2. Click Add, ensure that Country is selected under Custom based on data item, and then click OK.

3. In the Values section, ensure that Specific values is selected from the list. Text pattern matching is also available and includes:
   - Starts with
   - Ends with
   - Contains
   - Matches SQL pattern

   Advanced search options are also available.

4. From the Values list, click Austria, and then Ctrl-click Italy.

5. Click the arrow to add the items to the Selected values pane.

6. Click Page down, click Spain, and then add it into the Selected values pane.

7. Click OK to close the Filter condition dialog box, and then click OK to close the Filters dialog box.

8. Run the report in HTML.

   A section of the results appear as follows:

<table>
<thead>
<tr>
<th>City</th>
<th>First name</th>
<th>Last name</th>
<th>Position name</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wien</td>
<td>Sabine</td>
<td>Grüner</td>
<td>Level 3 Sales Representative</td>
<td>12,193,198.67</td>
</tr>
<tr>
<td>Jutta</td>
<td>Shultz</td>
<td></td>
<td>Level 2 Sales Representative</td>
<td>9,938,792.37</td>
</tr>
<tr>
<td>Thomas</td>
<td>Schirmer</td>
<td></td>
<td>Level 1 Sales Representative</td>
<td>6,216,976.62</td>
</tr>
<tr>
<td>Austria-Total</td>
<td></td>
<td></td>
<td></td>
<td>28,348,967.66</td>
</tr>
<tr>
<td>Wien</td>
<td></td>
<td></td>
<td></td>
<td>28,348,967.66</td>
</tr>
<tr>
<td>Austria-Total</td>
<td></td>
<td></td>
<td></td>
<td>28,348,967.66</td>
</tr>
</tbody>
</table>

   In 2012, Italy generated the most revenue of Southern European countries, and Sabine Grüner from Austria earned the top sales rep award.
9. Close the rendered report tab.
10. Leave the report authoring tab open for the next demonstration.

Results:
You created a report with filters to show the revenue generated by the top sales representatives for 2012 in Southern Europe.
### Determine when to apply a filter with aggregation

**Before Auto-aggregation**

<table>
<thead>
<tr>
<th>Product Type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Navigation</td>
<td>121,958.34</td>
</tr>
<tr>
<td>Navigation</td>
<td>104,207.4</td>
</tr>
<tr>
<td>Knives</td>
<td>100,045.74</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>1,378,713.67</strong></td>
</tr>
<tr>
<td><strong>Overall</strong></td>
<td><strong>496,713,003.2</strong></td>
</tr>
</tbody>
</table>

Individual data values for Navigation product type where revenue is greater than $100,000.

**After Auto-aggregation**

<table>
<thead>
<tr>
<th>Product Type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personal Accessories</td>
<td>305,646.3</td>
</tr>
<tr>
<td>Navigation</td>
<td>1,073,067.37</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>1,378,713.67</strong></td>
</tr>
<tr>
<td><strong>Overall</strong></td>
<td><strong>496,713,003.2</strong></td>
</tr>
</tbody>
</table>

Summarized data values for Navigation product type where revenue is greater than $100,000.

---

Aggregated data can show totals, averages, or other formats of summarized data.
Demonstration 2

Apply a detail filter on fact data in a report

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Cooking Gear</td>
<td>1,863,445.82</td>
</tr>
<tr>
<td></td>
<td>Packs</td>
<td>52,076,711.17</td>
</tr>
<tr>
<td></td>
<td>Sleeping Bags</td>
<td>21,034,472.39</td>
</tr>
<tr>
<td></td>
<td>Tents</td>
<td>282,028,081.98</td>
</tr>
<tr>
<td><strong>Camping Equipment - Total</strong></td>
<td>357,002,711.36</td>
<td></td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>Irons</td>
<td>41,032,759.96</td>
</tr>
<tr>
<td></td>
<td>Putters</td>
<td>1,184,967.25</td>
</tr>
<tr>
<td></td>
<td>Woods</td>
<td>87,453,875.01</td>
</tr>
<tr>
<td><strong>Golf Equipment - Total</strong></td>
<td>129,874,602.22</td>
<td></td>
</tr>
</tbody>
</table>

*Demonstration 2: Apply a detail filter on fact data in a report*
Demonstration 2: Apply a detail filter on fact data in a report

Purpose:
You want to make a report displaying the total revenue produced by top performing products. To create this report, you will add several filters and examine how they affect the query.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the list.
1. Add the following query items to a new list template:
   - Products: Product line, Product type
   - Sales fact: Revenue

2. Click the <Product line> list column body, and then on the toolbar, click Group / Ungroup.
3. Click <Revenue> list column body, on the toolbar click Summarize, and then click Total.
4. Run the report in **HTML**.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Cooking Gear</td>
<td>272,835,984.18</td>
</tr>
<tr>
<td></td>
<td>Lanterns</td>
<td>126,925,660.64</td>
</tr>
<tr>
<td></td>
<td>Packs</td>
<td>351,880,402.84</td>
</tr>
<tr>
<td></td>
<td>Sleeping Bags</td>
<td>309,172,888.35</td>
</tr>
<tr>
<td></td>
<td>Tents</td>
<td>528,221,728.02</td>
</tr>
</tbody>
</table>

The Product line data is grouped and an aggregate row displays the total revenue generated by all product types in each product line. Notice that Cooking Gear for the Camping Equipment product line generated $272,835,984.18 in revenue. You will compare this number with the revenue number generated later in Task 3.

5. Close the rendered report tab.

6. On the Side panel, click Navigate, click the **Query explorer** tab, and then click **Query1**.

7. In the Data Items pane, click Revenue, and then click Show properties from the Application bar.

   In the Properties pane, notice that the Detail aggregation property for Revenue is set to Total. This is because in the layout you added an aggregate row displaying total revenue for grouped items in the report.

8. On the content pane, click Queries, and then in the work area, click **Query 1**.

   In the Properties pane, you notice that the Auto group & summarize property for the query is set to Yes. You want to view each individual data record, so you will change this property to No.

9. In the Properties pane, click the Auto group & summarize property, and then select No from the list.

10. Run the report in **HTML**.

    Note: Do not click the Bottom navigation button as this report returns a large amount of data and it will take a considerable amount of time to render the last
The order you see displayed in the results may vary, as there has been no sorting applied.

A section of the results appears similar to the following:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Packs</td>
<td>41,543.16</td>
</tr>
<tr>
<td>Cooking Gear</td>
<td>Cooking Gear</td>
<td>10,278.44</td>
</tr>
<tr>
<td>Cooking Gear</td>
<td>Cooking Gear</td>
<td>30,838.38</td>
</tr>
<tr>
<td>Cooking Gear</td>
<td>Cooking Gear</td>
<td>40,776.32</td>
</tr>
<tr>
<td>Tents</td>
<td></td>
<td>61,075.08</td>
</tr>
<tr>
<td>Sleeping Bags</td>
<td></td>
<td>49,704.2</td>
</tr>
<tr>
<td>Sleeping Bags</td>
<td></td>
<td>22,737.78</td>
</tr>
</tbody>
</table>

The report displays separate rows for revenue generated by individual sales of each product type.

11. Close the rendered report tab.

**Task 2. Apply a detail filter before auto aggregation and examine the effects.**

You want this report to include only data from individual orders of each product type that generated more than $100,000 in revenue. You will create a detail filter and apply it before auto aggregation.

1. Click the **Page explorer** tab, and then click **Page1**.
2. Select the list data container.
3. On the list toolbar, click **Filters**, click **Edit Filters**, and then ensure the **Detail Filters** tab is selected.
4. Click **Add**, from the **Custom based on data item** list select **Revenue**, and then click **OK**.
5. Ensure that the **Operator** is >, and then in the **Value** text box, type **100000** (100 thousand).
6. Click **OK**, and then in the **Application** area, click **Before auto aggregation**.
7. Click **OK** to close the dialog box.
8. Run the report in **HTML**.
A section of the results appear similar to the following:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Sleeping Bags</td>
<td>115,144.26</td>
</tr>
<tr>
<td>Tents</td>
<td></td>
<td>114,969.48</td>
</tr>
<tr>
<td>Tents</td>
<td></td>
<td>111,038.25</td>
</tr>
<tr>
<td>Tents</td>
<td></td>
<td>114,829.8</td>
</tr>
</tbody>
</table>

The report now displays only data for individual sales of product types that generated more than $100,000 in revenue.

9. In the report, click **Bottom**.
The total revenue generated by product type orders of over $100,000 is $496,713,003.20.

10. Close the rendered report tab.
11. Click the **Query explorer** tab, and then click **Query 1**.
The filter that you created appears in the Detail Filters pane.
12. In the **Detail Filters** pane, click **Revenue > 100000**.
In the Properties pane, the properties specified for the filter display as follows:
- **Definition**: displays the expression you created for this filter
- **Usage**: is set to Required
- **Application**: is set to Before Auto Aggregation

**Task 3. Set the query to group and summarize data.**
You want to see only one row for sales of each product type, so you will set the Auto Group & Summarize property for the query back to Yes.

1. On the **Query explorer** tab, click **Queries**, and then in the work area, click **Query 1**.
2. In the **Properties** pane, under **DATA**, change the **Auto group & summarize** property to **Yes**.
3. Run the report in **HTML**.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Cooking Gear</td>
<td>1,863,445.82</td>
</tr>
<tr>
<td>Packs</td>
<td></td>
<td>52,076,711.17</td>
</tr>
<tr>
<td>Sleeping Bags</td>
<td></td>
<td>21,034,472.39</td>
</tr>
<tr>
<td>Tents</td>
<td></td>
<td>282,028,081.98</td>
</tr>
<tr>
<td><strong>Camping Equipment - Total</strong></td>
<td></td>
<td><strong>357,002,711.36</strong></td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>Irons</td>
<td>41,032,759.96</td>
</tr>
</tbody>
</table>

There is only one row for each product type because the query will group and summarize the data at the lowest level of detail.

The revenue generated by Cooking Gear is $1,863,445.82. When you ran this report without the filter in Task 1, the revenue generated by Cooking Gear was $272,835,984.18. The value is different because it no longer includes individual orders that generated less than one hundred thousand dollars in revenue.

The total revenue generated by all product lines is $496,713,003.20, which is the same as when you ran the report in Task 2 with the Auto group & summarize property for the query set to No.

Since you specified that the filter was to be applied before the query will group and summarize retrieved data, the filter will exclude the same data regardless of whether the query retrieves data that is summarized or not summarized.

4. Close the rendered report tab.

**Task 4. Apply a detail filter after auto aggregation.**

You want the report to display only product types for which the total revenue for all sales is greater than ten million dollars. To achieve this, you will create a detail filter and apply it after auto aggregation.

1. Click the **Page Explorer** tab, and then click **Page1**.
2. Select the list data container, on the list toolbar click **Filters**, and then click **Edit Filters**.
3. Click **Add**, select **Revenue** from the **Custom based on data item** list, and then click **OK**.
4. Ensure that the **Operator** is >, and then in the **Value** text box, type 10000000 (10 million).
5. Click **OK**, and then ensure that under the **Application** section, **After auto aggregation** has been selected.
6. Click **OK** to close the **Filters** dialog box.
Task 5. Observe the effects of the filters.

1. Run the report in HTML.
   The results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Packs</td>
<td>52,076,711.17</td>
</tr>
<tr>
<td></td>
<td>Sleeping Bags</td>
<td>21,034,472.39</td>
</tr>
<tr>
<td></td>
<td>Tents</td>
<td>282,028,081.98</td>
</tr>
<tr>
<td><strong>Camping Equipment - Total</strong></td>
<td></td>
<td><strong>355,139,265.54</strong></td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>Irons</td>
<td>41,032,759.96</td>
</tr>
<tr>
<td></td>
<td>Woods</td>
<td>87,453,875.01</td>
</tr>
<tr>
<td><strong>Golf Equipment - Total</strong></td>
<td></td>
<td><strong>128,486,634.97</strong></td>
</tr>
<tr>
<td><strong>Overall - Total</strong></td>
<td></td>
<td><strong>483,625,900.51</strong></td>
</tr>
</tbody>
</table>

   Only the five product types that generated total revenue greater than ten million display in the report.

2. Close the rendered report tab.
   You have decided to include product types in the report even if the aggregated revenue generated by all sales of the product type is less than ten million dollars. However, in case you may want to use this filter in the future, you will disable this filter instead of deleting it.

3. Select the list data container.

4. On the toolbar, click Filters, and then click Edit Filters.

5. Click Revenue > 10000000, and then in the Usage area, click Disabled.

6. Click OK to close the dialog box.
7. Run the report in **HTML**.

   A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Cooking Gear</td>
<td>1,863,445.82</td>
</tr>
<tr>
<td>Packs</td>
<td></td>
<td>52,076,711.17</td>
</tr>
<tr>
<td>Sleeping Bags</td>
<td></td>
<td>21,034,472.39</td>
</tr>
<tr>
<td>Tents</td>
<td></td>
<td>282,026,081.98</td>
</tr>
<tr>
<td><strong>Camping Equipment - Total</strong></td>
<td></td>
<td><strong>357,802,711.36</strong></td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>Irons</td>
<td>41,032,759.96</td>
</tr>
</tbody>
</table>

   All product types that generated over $100,000 in revenue (in at least one order), again appear in the report - which indicates that the second filter you added has been disabled.

8. Close the rendered report tab.

9. On the **Content** pane, click **Query explorer**, and then click **Query1**.

   Notice that the Revenue > 10000000 filter still appears in the query, though it is grayed out and unavailable to the query.

10. In the **Detail Filters** pane, click **Revenue > 10000000**.

    In the Properties pane notice that, as specified, the Usage property for the filter is set to **Disabled**.

11. Leave the report authoring tab open for the next demonstration.

**Results:**

You created a report that displayed the total revenue produced by top performing products. You applied detail filters to the report so that only products producing a certain amount of revenue were displayed. You disabled a filter and viewed the effects.
Filter your data with summary filters

- Create a summary filter to filter your grouped data on summary values.

The summary filter focuses on Product lines that generated total revenues greater than $1,000,000,000

To add a filter that will apply to groups in the report, click the Summary Filters tab in the Filters dialog box.

When you use a summary filter, you can specify the group on which you want to filter.

When you combine detail and summary filters, be aware that the detail filter will affect the summarized numbers that you are filtering on. Be sure to check that the results are as expected.
Demonstration 3

Apply a summary filter to a report

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Cooking Gear</td>
<td>272,835</td>
</tr>
<tr>
<td></td>
<td>Lanterns</td>
<td>126,925</td>
</tr>
<tr>
<td></td>
<td>Packs</td>
<td>351,806</td>
</tr>
<tr>
<td></td>
<td>Sleeping Bag</td>
<td>309,172</td>
</tr>
<tr>
<td></td>
<td>Tents</td>
<td>528,221</td>
</tr>
<tr>
<td>Overall - Total</td>
<td></td>
<td>3,474,709</td>
</tr>
<tr>
<td>Personal Accessories</td>
<td>Binoculars</td>
<td>130,834</td>
</tr>
<tr>
<td></td>
<td>Eyewear</td>
<td>867,125</td>
</tr>
<tr>
<td></td>
<td>Knives</td>
<td>153,420</td>
</tr>
<tr>
<td></td>
<td>Navigation</td>
<td>237,400</td>
</tr>
<tr>
<td></td>
<td>Watcheas</td>
<td>526,802</td>
</tr>
<tr>
<td>Overall - Total</td>
<td></td>
<td>3,474,709</td>
</tr>
</tbody>
</table>

Demonstration 3: Apply a summary filter to a report
Demonstration 3:  
Apply a summary filter to a report

Purpose:  
You have been asked to create a report that focuses on product lines that have generated revenues greater than $1 billion. You will use a summary filter to focus on this data.

Portal: http://vclassbase:9300/bi  
User/Password: brettonf/Education1  
Package: Team content\Samples\Models\GO data warehouse (query)  
Report Type: List  
Folder: Sales and Marketing (query)  
Namespace: Sales (query)

Task 1. Create the list and apply a summary filter.

1. Add the following query items to a new list template using the GO data warehouse (query) package:
   - Products: Product line, Product type
   - Sales fact: Revenue

2. Click <Product line> list column body, and then on the list toolbar, click Group / Ungroup.
3. Click <Revenue> list column body, on the list toolbar, click Summarize, and then click Total.
4. Select the list data container, on the list toolbar, click Filters, and then click Edit Filters.
5. Click the Summary Filters tab, click Add, click Advanced, and then click OK.
6. Create and validate the following expression: Total(Revenue)>1000000000
   Hint:
   - drag Total(Revenue) from the Data items tab
   - 1,000,000,000 (1 billion)
7. Click OK.
8. Next to the **Scope**, field click the **ellipsis**.

9. Select **Product line**, click **OK** to close the **Scope** dialog box, and then click **OK** to close the **Filters** dialog box.

10. Run the report in **HTML**.

The results appear as follows:

![Table of product lines and their revenues]

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Cooking Gear</td>
<td>272,835,984.18</td>
</tr>
<tr>
<td></td>
<td>Lanterns</td>
<td>126,925,660.64</td>
</tr>
<tr>
<td></td>
<td>Packs</td>
<td>351,080,402.64</td>
</tr>
<tr>
<td></td>
<td>Sleeping Bags</td>
<td>309,172,888.35</td>
</tr>
<tr>
<td></td>
<td>Tents</td>
<td>526,221,728.02</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Camping Equipment - Total</strong></th>
<th>1,589,036,664.03</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personal Accessories</td>
<td></td>
</tr>
<tr>
<td>Binoculars</td>
<td>130,834,653.2</td>
</tr>
<tr>
<td>Eyewear</td>
<td>887,125,198.48</td>
</tr>
<tr>
<td>Knives</td>
<td>153,420,439.59</td>
</tr>
<tr>
<td>Navigation</td>
<td>207,490,641.92</td>
</tr>
<tr>
<td>Watches</td>
<td>526,802,374.59</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Personal Accessories - Total</strong></th>
<th>1,885,673,307.78</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Overall - Total</strong></td>
<td>3,474,709,971.81</td>
</tr>
</tbody>
</table>

Only two product lines generated total revenues greater than $1,000,000,000:
Camping Equipment and Personal Accessories.

**Task 2. Navigate to the query explorer.**

1. Close the rendered report tab.
2. On the side bar, click **Navigate**.
3. On the **Content** pane, click **Query explorer**, and then click **Query 1**. The summary filter you added appears in the Summary Filters pane.
4. On the **Application** bar, click **Show Properties**.
5. In the **Summary Filters** pane, click **[Total(Revenue)] > 1000000000**.
   In the Properties pane, the Scope property for this filter is set to Product line.
6. Leave the report authoring tab open for the following Exercise.

**Results:**
You have created a report that used a summary filter to focus on product lines that generated total revenues greater than $1 billion.
Apply pre-defined source filters

- Save time and effort by applying filters published with your source package rather than creating your own.

Pre-defined filters have been included in the report package to assist in report authoring.
Unit summary

- Create filters to narrow the focus of reports
- Examine detail and summary filters
- Determine when to apply filters on aggregate data
Exercise 1

Create a report focused on top performing product types and product lines

<table>
<thead>
<tr>
<th>Product Type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cooking Gear</td>
<td>272,835,804.16</td>
</tr>
<tr>
<td>Lanterns</td>
<td>126,925,661.64</td>
</tr>
<tr>
<td>Packs</td>
<td>351,880,402.04</td>
</tr>
<tr>
<td>Sleeping Bags</td>
<td>309,172,888.55</td>
</tr>
<tr>
<td>Tents</td>
<td>520,221,726.07</td>
</tr>
<tr>
<td><strong>Camping Equipment- Total</strong></td>
<td><strong>1,889,830,864.25</strong></td>
</tr>
<tr>
<td>Irons</td>
<td>254,814,337.39</td>
</tr>
<tr>
<td>Putters</td>
<td>105,184,271.37</td>
</tr>
<tr>
<td>Woods</td>
<td>313,698,414.65</td>
</tr>
<tr>
<td><strong>Golf Equipment- Total</strong></td>
<td><strong>1,674,287,824.61</strong></td>
</tr>
<tr>
<td>Binooculars</td>
<td>156,854,653.2</td>
</tr>
<tr>
<td>Eyewear</td>
<td>867,325,195.90</td>
</tr>
<tr>
<td>Knives</td>
<td>153,420,439.59</td>
</tr>
<tr>
<td>Navigation</td>
<td>207,490,641.62</td>
</tr>
<tr>
<td>Wristwatches</td>
<td>529,882,974.59</td>
</tr>
<tr>
<td><strong>Personal Accessories- Total</strong></td>
<td><strong>1,886,873,307.78</strong></td>
</tr>
<tr>
<td><strong>Overall- Total</strong></td>
<td><strong>4,149,898,095.82</strong></td>
</tr>
</tbody>
</table>
Exercise 1: Create a report focused on top performing product types and product lines

You have been asked to create a report that displays revenue by product line and product type. The report must show the product types that generated revenue greater than $100 million and product lines that generated revenue greater than $400 million.

To accomplish this:

- Add the following query items to a new list template using the GO data warehouse (query) package/Sales and Marketing (query)/Sales (query).
  - Products: Product line, Product type
  - Sales fact: Revenue
- Group by Product line, Total on Revenue
- Add a detail filter (After auto-aggregation) for revenue greater than $100 million.
- Add a summary filter on Product line that generated total revenue greater than $420 million.

For more information about where to work and the exercise results, refer to the Tasks and Results section that follows. If you need more information to complete a task, refer to earlier demonstrations for detailed steps.
Exercise 1: Tasks and Results

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content/Samples/Models/GO data warehouse (query)
Report Template: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create a new list report and observe the results.

- **Toolbar:** Open a new List template using the GO data warehouse (query) package.

- **Source tab:** From the Products query subject add Product line and Product type to the list report object.
  - From the Sales fact query subject, add Revenue to the list report object.

  The results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product type&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product type&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product type&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
</tbody>
</table>

- **List Toolbar:** Group <Product line>.
- **List Toolbar:** Summarize/Total <Revenue>.

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product type&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Product line&gt; - Total</td>
<td>&lt;Total(Revenue)&gt;</td>
<td></td>
</tr>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product type&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Product line&gt; - Total</td>
<td>&lt;Total(Revenue)&gt;</td>
<td></td>
</tr>
<tr>
<td>Overall - Total</td>
<td>&lt;Total(Revenue)&gt;</td>
<td></td>
</tr>
</tbody>
</table>
• Run the report in **HTML** and then observe the results for Product type and Revenue totals.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Lanterns</td>
<td>126,925,660.64</td>
</tr>
<tr>
<td></td>
<td>Packs</td>
<td>351,680,402.84</td>
</tr>
<tr>
<td></td>
<td>Sleeping Bags</td>
<td>309,172,888.35</td>
</tr>
<tr>
<td></td>
<td>Tents</td>
<td>528,221,728.02</td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>Golf Accessories</td>
<td>51,514,343.88</td>
</tr>
<tr>
<td></td>
<td>Irons</td>
<td>254,814,337.99</td>
</tr>
<tr>
<td></td>
<td>Putters</td>
<td>106,184,271.37</td>
</tr>
<tr>
<td></td>
<td>Woods</td>
<td>313,898,414.65</td>
</tr>
<tr>
<td>Mountaineering Equipment</td>
<td>Climbing Accessories</td>
<td>81,096,582.48</td>
</tr>
<tr>
<td></td>
<td>Rope</td>
<td>114,426,644.73</td>
</tr>
<tr>
<td></td>
<td>Safety</td>
<td>83,236,883.98</td>
</tr>
<tr>
<td><strong>Camping Equipment - Total</strong></td>
<td></td>
<td><strong>1,589,036,864.03</strong></td>
</tr>
<tr>
<td><strong>Golf Equipment - Total</strong></td>
<td></td>
<td><strong>726,411,367.89</strong></td>
</tr>
</tbody>
</table>

• Close the rendered report tab.

**Task 2. Apply a detail filter on Revenue.**

• **List Toolbar:** Create a detail filter that shows revenue greater than 100,000,000 after auto aggregation.

The results appear as follows:
Run the report in **HTML** and then compare to the previous run. Observe that most of the Product line total revenues have changed and that Product types that generated less than $100 million are not included in these totals. Outdoor Protection is no longer included in the report because all the Product types that belong to it generated less than $100 million.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Cooking Gear</td>
<td>272,835,984.18</td>
</tr>
<tr>
<td></td>
<td>Lanterns</td>
<td>126,925,660.64</td>
</tr>
<tr>
<td></td>
<td>Packs</td>
<td>351,880,402.84</td>
</tr>
<tr>
<td></td>
<td>Sleeping Bags</td>
<td>309,172,888.35</td>
</tr>
<tr>
<td></td>
<td>Tents</td>
<td>528,221,728.02</td>
</tr>
<tr>
<td><strong>Camping Equipment - Total</strong></td>
<td></td>
<td><strong>1,589,036,664.03</strong></td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>Irons</td>
<td>254,814,337.99</td>
</tr>
</tbody>
</table>

Close the rendered report tab.

**Task 3. Apply a summary filter on Total(Revenue).**

- **List Toolbar**: Create a summary filter that shows generated revenue for each product line greater than $420,000,000.

The results appear as follows (notice that the Scope is set to Product line):

- Run the report in **HTML**.
Now with both the detailed and summary filters applied, you should see that only three product lines remain in the results queries. This includes only product lines that had an original summary over 420,000,000 and excluding any product types with revenue less than 100,000,000.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Cooking Gear</td>
<td>272,835,984.18</td>
</tr>
<tr>
<td></td>
<td>Lanterns</td>
<td>126,925,660.64</td>
</tr>
<tr>
<td></td>
<td>Packs</td>
<td>351,880,402.84</td>
</tr>
<tr>
<td></td>
<td>Sleeping Bags</td>
<td>309,172,888.35</td>
</tr>
<tr>
<td></td>
<td>Tents</td>
<td>528,221,728.02</td>
</tr>
<tr>
<td>Camping Equipment</td>
<td></td>
<td><strong>1,589,036,664.03</strong></td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>Irons</td>
<td>254,814,337.99</td>
</tr>
<tr>
<td></td>
<td>Putters</td>
<td>106,184,271.37</td>
</tr>
<tr>
<td></td>
<td>Woods</td>
<td>313,898,414.65</td>
</tr>
<tr>
<td>Golf Equipment</td>
<td></td>
<td><strong>674,897,024.01</strong></td>
</tr>
<tr>
<td>Personal Accessories</td>
<td>Binoculars</td>
<td>130,834,653.2</td>
</tr>
<tr>
<td></td>
<td>Eyewear</td>
<td>867,125,198.48</td>
</tr>
<tr>
<td></td>
<td>Knives</td>
<td>153,420,439.59</td>
</tr>
<tr>
<td></td>
<td>Navigation</td>
<td>207,490,641.92</td>
</tr>
<tr>
<td></td>
<td>Watches</td>
<td>526,802,374.59</td>
</tr>
<tr>
<td>Personal Accessories</td>
<td></td>
<td><strong>1,886,673,307.78</strong></td>
</tr>
<tr>
<td>Overall - Total</td>
<td></td>
<td><strong>4,149,606,995.82</strong></td>
</tr>
</tbody>
</table>

- Close rendered report tab.
- Close all browser tabs and if prompted, select **Leave this Page**.
Unit 4  Create crosstab reports

Create crosstab reports

IBM Cognos Analytics (v11.0)
Unit objectives

- Format and sort crosstab reports
- Create complex crosstabs using drag and drop functionality
- Create crosstabs using unrelated data items
Create a crosstab report

- Add query items to rows and columns, and measures to the body of the crosstab.

<table>
<thead>
<tr>
<th>Query Items</th>
<th>Measures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Silver</td>
<td>$500,382,422.83</td>
</tr>
<tr>
<td>Gold</td>
<td>$352,910,329.97</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Revenue</th>
<th>2007</th>
<th>2006</th>
</tr>
</thead>
<tbody>
<tr>
<td>Golf Equipment</td>
<td>$174,740,819.29</td>
<td>$230,110,270.55</td>
</tr>
<tr>
<td>Camping Equipment</td>
<td>$352,910,329.97</td>
<td>$500,382,422.83</td>
</tr>
</tbody>
</table>

Create a crosstab report

A crosstab is a tabular display of data with data items appearing on rows and columns, and is useful for analyzing and comparing summary data. Crosstab edge cells have four drop zones: one on each side, one at the top of the cell, and one at the bottom of the cell. Use the crosstab drop zones to add items as parents, peers, or children of other items in the crosstab. Using crosstab drop zones, you can quickly create crosstabs using drag-and-drop functionality.
Add measures to crosstab reports

- You can add measures to either the row or column edges of a crosstab report.
- You can add a default measure that is used in cells where the measure is not defined on the row or column edge.

<table>
<thead>
<tr>
<th>Default measure</th>
<th>Defined measure for a crosstab node</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revenue</td>
<td>&lt;#Quarter#&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;#Quantity#&gt;</td>
</tr>
<tr>
<td>Product line</td>
<td>&lt;#1234#&gt;</td>
</tr>
<tr>
<td>Product line</td>
<td>&lt;#1234#&gt;</td>
</tr>
</tbody>
</table>

Add measures to crosstab reports

Any data item that can be aggregated can be added to the body of the crosstab as the measure. The measure defines the data in the report, such as revenue, quantity, or profit margin.

The crosstab fact cells contain the measure values. Default measure is a property of the crosstab object. If the measures of the crosstab cannot be determined by what is being rendered on the edges, then the default measure will be rendered.

In crosstabs, you can now show values as a percentage of a summary instead of the actual values. For example, you can show the revenue that was generated by each product line as a percentage of the total revenue.
Data sources for crosstabs

- Relational models have a basic metadata structure that looks like tables and columns in a database.
- Dimensionally Modeled Relational (DMR) models are built from relational data sources, but are modeled with a dimensional structure (like OLAP) consisting of measures and dimensions.
- Because crosstabs use rows and columns to define the basic structure and determine cell values, they are better suited to dimensional reporting.

Best practices to keep in mind when using crosstab report objects:

- crosstabs are, by design, a dimensional reporting object
- insert the query items you wish to view in the rows and columns to focus the report rather than using filters
- filters in a crosstab may cause unpredictable results and should be used only when necessary
- crosstabs can be used in relational data reporting, but take care to maintain predictable results
## Demonstration 1

Create a simple crosstab report

<table>
<thead>
<tr>
<th>Revenue</th>
<th>Camping Equipment</th>
<th>Golf Equipment</th>
<th>Outdoor Protection</th>
<th>Personal Accessories</th>
<th>Mountaineering Equipment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Telephone</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010</td>
<td>50,487,595.83</td>
<td>44,244,120.93</td>
<td>6,141,159.76</td>
<td>45,940,522.79</td>
<td></td>
</tr>
<tr>
<td>2011</td>
<td>47,562,256.31</td>
<td>27,340,352.57</td>
<td>3,293,287.7</td>
<td>18,428,095.15</td>
<td>10,826,292.36</td>
</tr>
<tr>
<td>2012</td>
<td>17,715,451.4</td>
<td>6,411,333.64</td>
<td>507,455.63</td>
<td>5,970,547.46</td>
<td>6,586,129.67</td>
</tr>
<tr>
<td>2013</td>
<td>8,149,587.54</td>
<td>734,405.51</td>
<td>76,371.43</td>
<td>3,173,298.96</td>
<td>5,689,410.37</td>
</tr>
<tr>
<td>Web</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010</td>
<td>125,829,518.92</td>
<td>45,580,401.41</td>
<td>13,735,716.85</td>
<td>264,622,836.47</td>
<td></td>
</tr>
<tr>
<td>2011</td>
<td>270,403,415.68</td>
<td>116,938,646.30</td>
<td>16,476,270.8</td>
<td>411,577,877.16</td>
<td>85,655,485.48</td>
</tr>
<tr>
<td>2012</td>
<td>406,355,675.75</td>
<td>203,905,656.61</td>
<td>8,579,078.91</td>
<td>563,050,077.83</td>
<td>132,736,443.67</td>
</tr>
<tr>
<td>2013</td>
<td>311,102,071.84</td>
<td>157,608,957.23</td>
<td>4,156,746.33</td>
<td>427,367,391.96</td>
<td>117,012,256.92</td>
</tr>
</tbody>
</table>

**Demonstration 1: Create a simple crosstab report**
Demonstration 1: 
Create a simple crosstab report

Purpose:
You want to create and format a report to show revenue generated by order method for each year. You want to see yearly trends in sales for each order method.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: Crosstab
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create a crosstab.

1. Open a new Crosstab template using the GO data warehouse (query) package.
2. From the Data/Source tab, use the click and drag method to add the following query items to the new crosstab data container object:
   Rows:
   • Products: Product line
   Columns:
   • Order method: Order method type
   Measures:
   • Sales fact: Revenue

<table>
<thead>
<tr>
<th>Revenue</th>
<th>&lt;#Order method type#&gt;</th>
<th>&lt;#Order method type#&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;#Product line#&gt;</td>
<td>&lt;#1234#&gt;</td>
<td>&lt;#1234#&gt;</td>
</tr>
<tr>
<td>&lt;#Product line#&gt;</td>
<td>&lt;#1234#&gt;</td>
<td>&lt;#1234#&gt;</td>
</tr>
</tbody>
</table>
3. Run the report in HTML.
   The results appear as follows:
   
   Your report shows the revenue generated for each product line by each order
   method. You want to add relevancy to the revenue items by adding years to the
   report to compare revenue generated in each year.

4. Close the rendered report tab.

Task 2. Add Year to the crosstab report and sort on Year.

1. Expand the Time query subject, and then drag Year to the Columns, nested
   under <#Order method type#> as a child (or nested) cell.

2. Click the <#Year#> column title.

3. From the list toolbar, click Sort, and then click Ascending.

4. Run the report in HTML.
   Your report is very wide. When consumers are viewing the report, they will
   always have to scroll horizontally. You can swap the rows and columns to make
   it easier for consumers to read the report.

5. Close the rendered report tab.

6. Select the entire crosstab.

7. On the toolbar, click Swap Rows and Columns.

8. Run the report in **HTML**.

   A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Revenue</th>
<th>Camping Equipment</th>
<th>Golf Equipment</th>
<th>Outdoor Protection</th>
<th>Personal Accessories</th>
<th>Mountaineering Equipment</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-mail</td>
<td>2010: 39,124,634.73</td>
<td>29,322,537.92</td>
<td>4,139,915.35</td>
<td>22,019,708.21</td>
<td>2,517,063.13</td>
</tr>
<tr>
<td></td>
<td>2011: 21,297,066.31</td>
<td>8,851,223.61</td>
<td>1,406,531.47</td>
<td>10,263,053.91</td>
<td>6,129,791.95</td>
</tr>
<tr>
<td></td>
<td>2012: 10,612,304.02</td>
<td>5,401,733.78</td>
<td>289,343.01</td>
<td>5,568,561.15</td>
<td>1,829,100.61</td>
</tr>
<tr>
<td></td>
<td>2013: 4,871,150.57</td>
<td>3,450,428.85</td>
<td>60,688.04</td>
<td>4,009,763.27</td>
<td>3,130,288.22</td>
</tr>
<tr>
<td>Fax</td>
<td>2010: 9,634,783.39</td>
<td>6,255,930.08</td>
<td>1,435,512.2</td>
<td>11,313,264.67</td>
<td>6,129,791.95</td>
</tr>
<tr>
<td></td>
<td>2011: 6,229,274.72</td>
<td>3,539,563.59</td>
<td>385,329.2</td>
<td>3,613,228.75</td>
<td>3,536,047.25</td>
</tr>
<tr>
<td></td>
<td>2012: 5,226,451.57</td>
<td>2,408,222.14</td>
<td>123,028.48</td>
<td>2,149,810.49</td>
<td>3,536,047.25</td>
</tr>
<tr>
<td></td>
<td>2013: 1,964,909.25</td>
<td>3,037,587.46</td>
<td>22,614.84</td>
<td>886,679.75</td>
<td>2,180,530.88</td>
</tr>
</tbody>
</table>

9. Click **Page down** to view the rest of the report.

   Your report shows that Web sales have been increasing while Telephone sales have been decreasing. (Be aware that the 2013 values are based on only 7 months of data, not 12 months of data like the others)

<table>
<thead>
<tr>
<th>Revenue</th>
<th>Camping Equipment</th>
<th>Golf Equipment</th>
<th>Outdoor Protection</th>
<th>Personal Accessories</th>
<th>Mountaineering Equipment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Telephone</td>
<td>2010: 80,467,596.88</td>
<td>44,244,120.93</td>
<td>8,141,169.76</td>
<td>45,940,652.79</td>
<td>10,626,292.36</td>
</tr>
<tr>
<td></td>
<td>2011: 47,562,256.31</td>
<td>27,349,352.57</td>
<td>3,203,287.7</td>
<td>18,428,095.15</td>
<td>10,626,292.36</td>
</tr>
<tr>
<td></td>
<td>2012: 17,715,451.4</td>
<td>6,411,233.64</td>
<td>507,485.63</td>
<td>5,979,547.46</td>
<td>6,586,124.57</td>
</tr>
<tr>
<td></td>
<td>2013: 8,149,507.54</td>
<td>734,495.51</td>
<td>76,371.43</td>
<td>3,173,296.86</td>
<td>5,699,410.37</td>
</tr>
<tr>
<td>Web</td>
<td>2010: 125,829,519.92</td>
<td>49,583,491.41</td>
<td>13,735,716.85</td>
<td>284,622,826.47</td>
<td>65,855,489.46</td>
</tr>
<tr>
<td></td>
<td>2012: 426,353,676.75</td>
<td>203,385,896.61</td>
<td>568,078.91</td>
<td>13,273,443.67</td>
<td>132,736,443.67</td>
</tr>
<tr>
<td></td>
<td>2013: 3,112,371.64</td>
<td>157,698,057.23</td>
<td>4,166,745.33</td>
<td>427,363,318.9</td>
<td>117,010,256.92</td>
</tr>
</tbody>
</table>

10. Close the rendered report tab.

11. Leave the report authoring tab open for the next demonstration.

**Results:**

You created and formatted a report to show revenue generated by order method for each year. The report displayed yearly trends in telephone sales for each order method.
Create complex crosstab reports

- Crosstab drop zones let you create a wide variety of crosstab layouts to meet your business requirements.

Add Region as a peer of Product line

To add a second item as a peer below an existing item, drop the new item below the bottom instance of the item on the row edge. To add a second item as a peer above the existing item, drop the new item above either instance of the item on the row edge.

To add a second item as a peer to the right of the existing item, drop the new item to the right of the far right instance of the item on the column edge. To add a second item as a peer to the left of the existing item, drop the new item to the left of either instance of the item on the column edge.
Create crosstab nodes and crosstab node members

- When you add items to crosstabs, you create crosstab nodes and crosstab node members.

This crosstab node contains two crosstab node members: Region and City.

The row and column edges of a crosstab are composed of sets of crosstab nodes. A crosstab node contains one crosstab node member, as well as any crosstab node members nested under it.

Each crosstab node member refers to a data item.

Crosstab nodes and crosstab node members let you easily create and modify complex crosstabs.
Create complex crosstab reports

<table>
<thead>
<tr>
<th>Model Category</th>
<th>2010</th>
<th>2011</th>
<th>2012</th>
<th>2013</th>
<th>E-mail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>323,908,338.06</td>
<td>402,757,573.17</td>
<td>500,382,422.83</td>
<td>352,610,329.97</td>
<td>75,689,094.65</td>
</tr>
<tr>
<td>Quantity</td>
<td>5,825,053</td>
<td>8,003,376</td>
<td>8,339,156</td>
<td>6,103,916</td>
<td>1,413,504</td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>135,553,050.88</td>
<td>106,095,427.07</td>
<td>230,110,276.55</td>
<td>174,740,019.29</td>
<td>47,933,933.16</td>
</tr>
<tr>
<td>Quantity</td>
<td>1,002,862</td>
<td>1,277,793</td>
<td>1,536,172</td>
<td>1,681,164</td>
<td>333,000</td>
</tr>
<tr>
<td>Outdoor Protection</td>
<td>36,105,521.07</td>
<td>25,000,574.08</td>
<td>10,349,175.84</td>
<td>4,471,025.26</td>
<td>582,427.87</td>
</tr>
<tr>
<td>Quantity</td>
<td>5,914,356</td>
<td>8,111,056</td>
<td>1,599,585</td>
<td>809,446</td>
<td>905,156</td>
</tr>
<tr>
<td>Personal Accessories</td>
<td>391,647,033.61</td>
<td>456,323,359.93</td>
<td>904,909,400.42</td>
<td>443,834,449.85</td>
<td>652,619,098.64</td>
</tr>
<tr>
<td>Quantity</td>
<td>7,572,330</td>
<td>8,587,357</td>
<td>10,706,015</td>
<td>8,061,994</td>
<td>781,005</td>
</tr>
<tr>
<td>Mountaineering Equipment</td>
<td>1,077,099,659.9</td>
<td>161,039,823.26</td>
<td>141,520,549.7</td>
<td>7,478,491.96</td>
<td>199,214</td>
</tr>
<tr>
<td>Quantity</td>
<td>2,644,713</td>
<td>3,700,202</td>
<td>3,555,110</td>
<td>699,456</td>
<td>199,214</td>
</tr>
<tr>
<td>Canada</td>
<td>41,468,082.87</td>
<td>49,366,410.29</td>
<td>67,341,094.59</td>
<td>53,511,041.08</td>
<td>40,596,757.97</td>
</tr>
<tr>
<td>France</td>
<td>50,542,272.09</td>
<td>45,745,704.76</td>
<td>53,967,276.85</td>
<td>43,928,068.57</td>
<td>305,039.79</td>
</tr>
<tr>
<td>Germany</td>
<td>41,462,245.81</td>
<td>43,631,063.88</td>
<td>56,037,217.04</td>
<td>41,352,286.51</td>
<td>37,914,586.54</td>
</tr>
<tr>
<td>Italy</td>
<td>22,237,050.92</td>
<td>31,113,860.15</td>
<td>45,822,153.53</td>
<td>33,788,696.68</td>
<td>22,058,934.4</td>
</tr>
</tbody>
</table>

Demonstration 2: Create complex crosstab reports
Demonstration 2:
Create complex crosstab reports

Purpose:
Management needs you to create a crosstab report for users to analyze the revenue generated and the quantity sold for different order methods. You will add data to examine the revenue generated by different order methods in the countries where your products are sold. You will also add order year data to the report and explore the flexibility of layout options using the crosstab drop zones.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: Crosstab
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create a crosstab report.

1. Open a new Crosstab template using the GO data warehouse (query) package.
2. From the Data/Source tab, add the following query items to the new crosstab report object:
   Rows:
   - Products: Product line
   Columns:
   - Order method: Order method type
Task 2. Nest on a crosstab edge.

You want to examine the revenue generated and quantity sold by each order method for each product line. To do this, you will nest both of these measures in the rows of the report.

1. From the Source tab, expand Sales fact, and then drag Revenue to the Rows area as a child of `<#Product line#>`.

Revenue is nested in the Product line rows of the crosstab.

The results appear as follows:

You also want to nest Quantity in the Product line rows.
2. From the **Source** tab, from **Sales fact**, drag **Quantity** to the **Rows** area as a peer of **<#Revenue#>**.

Both Revenue and Quantity are now nested in the Product line rows of the crosstab.

The results appear as follows:

3. Run the report in **HTML**.

A section of the results appear as follows:

You can analyze the revenue generated and the quantity sold by each order method for each product line.

4. Close the rendered report tab.
Task 3. **Add items as peers on a crosstab edge.**

You are also interested in how revenue generated by different order methods varies from country to country.

1. From the **Source** tab, expand **Employee by region**, and then drag **Country** to the **Rows** area, as a peer of `<#Product line#>`.

Both Product line and Country now appear on the row edge of the crosstab. The results appear as follows:

Country has no measure associated with it, since Revenue and Quantity are children of Product line only.
2. From the **Data Items** tab, drag **Revenue** to the **Rows** area as a child of `<#Country#>`.

The results appear as follows:

Revenue is nested within the Country rows of the crosstab.

The results appear as follows:

3. **Run the report in HTML.**

A section of the results appear as follows:

You can examine the revenue generated by each order method in different countries as well as the revenue generated and the quantity sold by each order method for each product line.
4. Close the rendered report tab.
   You now want to examine data for years and order method types. To do this, you will add Year to the column edge of the crosstab.

5. From the **Source** tab, expand **Time**, and then drag **Year** to the left of **<#Order method type#>** in the **Columns** area of the crosstab.
   The results appear as follows:

   ![Image](image1.png)

   Both Year and Order method types appear on the column edge of the crosstab.
   The results appear as follows:

   ![Image](image2.png)

6. Click the **<#Year#>** column header.
7. On the list toolbar, click **Sort**, and then click **Ascending**.
8. Run the report in **HTML**.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th></th>
<th>2010</th>
<th>2011</th>
<th>2012</th>
<th>2013</th>
<th>E-mail</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Camping Equipment</strong></td>
<td>332,906,333.06</td>
<td>402,757,573.17</td>
<td>500,382,422.33</td>
<td>352,910,329.97</td>
<td>75,899,094.63</td>
</tr>
<tr>
<td>Revenue</td>
<td>5,895,053</td>
<td>6,903,764</td>
<td>8,399,156</td>
<td>6,103,176</td>
<td>1,413,084</td>
</tr>
<tr>
<td><strong>Golf Equipment</strong></td>
<td>153,553,059.90</td>
<td>186,006,427.07</td>
<td>230,110,270.55</td>
<td>174,740,198.29</td>
<td>47,933,933.16</td>
</tr>
<tr>
<td>Revenue</td>
<td>1,092,982</td>
<td>1,297,793</td>
<td>1,536,772</td>
<td>1,186,154</td>
<td>333,300</td>
</tr>
<tr>
<td><strong>Outdoor Protection</strong></td>
<td>36,165,521.07</td>
<td>25,008,574.08</td>
<td>10,349,175.34</td>
<td>4,471,025.26</td>
<td>5,882,477.87</td>
</tr>
<tr>
<td>Revenue</td>
<td>5,614,356</td>
<td>4,111,058</td>
<td>1,599,535</td>
<td>689,446</td>
<td>905,156</td>
</tr>
<tr>
<td><strong>Personal Accessories</strong></td>
<td>391,647,093.61</td>
<td>456,323,355.9</td>
<td>594,009,408.42</td>
<td>443,693,449.85</td>
<td>42,651,086.54</td>
</tr>
<tr>
<td>Revenue</td>
<td>7,572,339</td>
<td>8,567,357</td>
<td>10,760,015</td>
<td>8,061,994</td>
<td>791,905</td>
</tr>
<tr>
<td><strong>Mountaineering Equipment</strong></td>
<td>707,099,559.94</td>
<td>161,039,823.26</td>
<td>141,520,649.7</td>
<td>7,476,451.96</td>
<td></td>
</tr>
<tr>
<td>Revenue</td>
<td>2,644,713</td>
<td>3,700,262</td>
<td>3,555,116</td>
<td>199,214</td>
<td></td>
</tr>
<tr>
<td><strong>Canada</strong></td>
<td>41,468,882.87</td>
<td>48,366,410.09</td>
<td>67,341,094.59</td>
<td>53,511,041.09</td>
<td>40,596,757.97</td>
</tr>
<tr>
<td>Revenue</td>
<td>50,546,272.09</td>
<td>45,745,704.79</td>
<td>53,867,275.35</td>
<td>43,928,068.57</td>
<td>365,839.79</td>
</tr>
<tr>
<td><strong>France</strong></td>
<td>41,462,248.81</td>
<td>43,631,063.98</td>
<td>55,037,217.94</td>
<td>41,352,298.31</td>
<td>37,914,006.54</td>
</tr>
<tr>
<td><strong>Germany</strong></td>
<td>22,227,855.92</td>
<td>31,113,888.15</td>
<td>45,622,153.53</td>
<td>33,788,864.66</td>
<td>22,650,934.4</td>
</tr>
<tr>
<td><strong>Italy</strong></td>
<td>22,227,855.92</td>
<td>31,113,888.15</td>
<td>45,622,153.53</td>
<td>33,788,864.66</td>
<td>22,650,934.4</td>
</tr>
</tbody>
</table>

You can examine revenue generated and quantity sold for your product lines as well as by different order methods. You can also examine the revenue generated in different countries by different order methods. For example, you can see that no Mountaineering Equipment was sold in 2010.

9. Close the rendered report tab.

10. Leave the report authoring tab open for the next demonstration.

**Results:**
You created a report that displayed revenue generated and quantity sold by your product lines in different years and by different order methods. The report also displayed the revenue generated in different countries in different years and by different order methods. You explored the flexibility of layout options using the crosstab drop zones.
Format crosstab reports

- You can specify formatting for cells displaying data for a specific row or column edge item, such as Product line or Region.

<table>
<thead>
<tr>
<th>Gross Profit</th>
<th>2011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personal Accessories</td>
<td>186,535,159.07</td>
</tr>
<tr>
<td>Asia Pacific</td>
<td>118,203,277.67</td>
</tr>
</tbody>
</table>

- Formatting applied to Crosstab Fact Cells
- No formatting applied

Bold, Blue
Bold, Italic, and Green
Add unrelated items to crosstabs edges

- You can create discontinuous crosstabs that have unrelated data in the row and column edges.

Creating discontinuous crosstabs lets you present a wide variety of information in one report and customize the way it is displayed.

If you want items on the edges of your crosstab to be discontinuous (contain different nested items), you can turn on the Create crosstab node option. This can be found under the Structure menu item.

If you want items on the edges of your crosstab to be related (contain the same nested items), you can turn off the Create crosstab node option.
## Demonstration 3

Sort and format a crosstab report

<table>
<thead>
<tr>
<th></th>
<th>Revenue 2010</th>
<th>Revenue 2011</th>
<th>Revenue 2012</th>
<th>Revenue 2013</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personal Accessories</td>
<td>29,346,444.08</td>
<td>30,310,573.76</td>
<td>39,974,429.94</td>
<td>31,303,208.42</td>
<td>130,834,653.2</td>
</tr>
<tr>
<td>Eyewear</td>
<td>154,310,479.02</td>
<td>200,648,005.39</td>
<td>292,226,165.14</td>
<td>221,939,948.93</td>
<td>687,125,198.46</td>
</tr>
<tr>
<td>Knitwear</td>
<td>36,374,834.09</td>
<td>33,144,183.28</td>
<td>47,784,444.36</td>
<td>36,177,477.89</td>
<td>163,420,483.59</td>
</tr>
<tr>
<td>Navigation</td>
<td>51,508,510.99</td>
<td>43,724,569.8</td>
<td>62,330,073.61</td>
<td>49,837,487.52</td>
<td>297,490,641.02</td>
</tr>
<tr>
<td>Watches</td>
<td>126,717,325.43</td>
<td>140,475,423.7</td>
<td>161,774,586.37</td>
<td>104,436,327.09</td>
<td>526,607,374.59</td>
</tr>
<tr>
<td>Total(Product type)</td>
<td>395,648,018.04</td>
<td>456,323,355.9</td>
<td>594,009,408.42</td>
<td>443,603,449.85</td>
<td>2,081,673,697.8</td>
</tr>
</tbody>
</table>

Central Europe        | 428,821,198.74 | 559,235,928.65 | 675,574,387.12 | 499,983,272.05 | 2,143,194,784.55 |
Americas               | 102,330,458.3  | 359,215,347.86 | 312,537,662.91 | 233,865,783.74 | 977,067,480.8 |
Asia Pacific           | 166,746,977.65 | 212,250,513.92 | 275,691,859.59 | 204,564,826.67 | 859,254,378.14 |
Northern Europe        | 70,230,147.41  | 90,215,646.65  | 117,140,087.64 | 91,945,289.28  | 369,535,160.96 |
Southern Europe        | 56,324,025.02  | 78,279,853.09  | 115,438,805.33 | 87,857,102.35  | 337,399,674.59 |

© Copyright IBM Corporation 2016

Demonstration 3: Sort and format a crosstab report
Demonstration 3: Sort and format a crosstab report

Purpose:
Sales Managers want you to create a crosstab report with data in which users can easily understand the sort order and can distinguish between data based on appearance. The report should show revenue for each year of operation for each Product type within each Product line. In the same crosstab, you want to display Revenue for each Branch Region.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: Crosstab
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create and sort a crosstab.
1. Open a new Crosstab template, using the GO data warehouse (query) package.
2. From the Source tab, add the following query items to the new crosstab report object:
   - **Rows:**
     - Products: Product line
     - Employee by region: Branch region as a peer of <#Product line#>.
   - **Columns:**
     - Time: Year
   - **Measures:**
     - Sales fact: Revenue
   The results appear as follows:

   ![Crosstab Table]

3. Click <#Year#>. 
4. On the list toolbar, click **Sort**, and then click **Ascending**.
5. Click **Ascending**.
6. On the toolbar, click **Sort**, and then click **Ascending**.

**Task 2. Format the crosstab and perform advanced sorting.**

1. Click **<#Product line#>**, and then on the toolbar click **More**.
2. Click **Select Member Fact Cells**.
3. On the list toolbar, click **More**.
4. Point to **Style**, click **Font**, click **Foreground Color**, and then click **Blue**.
5. Click **OK** to close the **Foreground Color** dialog box, and then click **OK** again to close the **Font** dialog box.
6. Click **<#Branch region#>**, and then on the main toolbar, click **Show properties**.
7. In the **Properties** pane, under **DATA**, double-click the **Sorting** property.
8. In the **Sorting** dialog box, from the **Data items** pane, drag **Revenue** to the **Sort List** pane.
9. Double-click the **Revenue** item that you just added, to change the sort order from ascending to descending (arrow pointing down).

The results appear as follows:

![Sorting - Branch region](image)

10. Click **OK**.
11. Click **<#Year#>**.
12. On the list toolbar, click **Summarize**, and then click **Total**.
Task 3. Add aggregate data to the crosstab.

1. From the Source tab, under Products, drag Product type to the Rows as a child of <#Product line#>. The results appear as follows:

![Crosstab example](image)

2. Click <#Product type#>.
3. On the toolbar, click Summarize, and then click Total.
4. In the list report, under <#Product type#>, click Total.
5. In the Properties pane, under TEXT SOURCE, click the Source type property, and then from the list, select Data item value.
6. Run the report in HTML.
7. Click Page down to view the rest of the report. A section of the results appear as follows:

![Crosstab result](image)

8. Close the rendered report tab.
Task 4. Examine crosstab nodes and crosstab node members.

1. Click the `<#Product line#>` row, and then drag it below the `<#Branch region#>` row.
The results appear as follows:

2. On the toolbar, click **Undo**.
3. Click the `<#Branch region#>` row and drag it above the `<#Product line#>` row.
The results appear as follows:

4. Close the **Properties** pane.
5. Leave the report authoring tab open for the exercise.

Results:
You have created a crosstab report with data in which users can easily understand the sort order and can distinguish between data based on appearance. The report now shows revenue for each year of operation for each Product type within each Product line. In the same crosstab, you have displayed Revenue for each Branch Region.
Unit summary

- Format and sort crosstab reports
- Create complex crosstabs using drag and drop functionality
- Create crosstabs using unrelated data items
Exercise 1

Present unrelated items in a crosstab using a discontinuous crosstab

<table>
<thead>
<tr>
<th></th>
<th>Americas</th>
<th>Asia Pacific</th>
<th>Central Europe</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Revenue</td>
<td>Quantity</td>
<td>Revenue</td>
</tr>
<tr>
<td>Camping Equipment</td>
<td>481,448.781</td>
<td>8,101,682</td>
<td>421,839.331</td>
</tr>
<tr>
<td>Golf Equipment</td>
<td>217,262,995</td>
<td>1,544,411</td>
<td>193,677,673.68</td>
</tr>
<tr>
<td>Mountaineering Equipment</td>
<td>123,127,397</td>
<td>2,949,533</td>
<td>197,505,775.01</td>
</tr>
<tr>
<td>Outdoor Protection</td>
<td>23,002,647</td>
<td>3,619,457</td>
<td>19,716,018.32</td>
</tr>
<tr>
<td>Personal Accessories</td>
<td>132,246,598</td>
<td>2,739,299</td>
<td>118,715,219.51</td>
</tr>
<tr>
<td>2013</td>
<td>Q1</td>
<td>59,688,309.86</td>
<td>1,752,555</td>
</tr>
<tr>
<td>Q2</td>
<td>60,828,777.21</td>
<td>1,698,239</td>
<td>86,438,349.68</td>
</tr>
<tr>
<td>Q3</td>
<td>55,168,598.87</td>
<td>559,659</td>
<td>50,802,577.53</td>
</tr>
<tr>
<td>2012</td>
<td>Q1</td>
<td>72,914,470.22</td>
<td>1,200,166</td>
</tr>
<tr>
<td>Q2</td>
<td>82,514,841.44</td>
<td>1,553,153</td>
<td>75,405,563.84</td>
</tr>
<tr>
<td>Q3</td>
<td>79,028,210.68</td>
<td>1,219,021</td>
<td>71,904,554.10</td>
</tr>
<tr>
<td>Q4</td>
<td>76,675,470.87</td>
<td>1,127,524</td>
<td>66,439,262.34</td>
</tr>
</tbody>
</table>
Exercise 1: 
Present unrelated items in a crosstab using a discontinuous crosstab

The sales managers have asked you to create a report showing revenue and quantity for each product line by year and quarter. The report should show revenue and quantity data for each sales region and they should be formatted in different colors to be more easily distinguished, blue for revenue and red for quantity. Since the report will have rows with unrelated data, you will be creating a discontinuous crosstab report.

To accomplish this:

- Open a new crosstab template using the GO data warehouse (query) package.
- Add the following query items to the new crosstab report object:
  - **Rows:**
    - Products: Product line,
    - Time: Year (below Product line as a peer),
    - Time: Quarter (nested to the right of Year as a child)
  - **Columns:** Employee by region: Branch region
    - Sales fact: Revenue and Quantity (nested under Branch region as children)
  - Sort `<#Branch region#>` as ascending.
  - Sort `<#Product line#>` as ascending.
  - Sort `<#Year#>` as descending.
- Format `<#Revenue#>` Member Fact Cells with a Blue foreground color.
- Format `<#Quantity#>` Member Fact Cells with a Red foreground color.

For more information about where to work and the exercise results, refer to the Tasks and Results section that follows. If you need more information to complete a task, refer to earlier demonstrations for detailed steps.
Exercise 1: Tasks and Results

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content/Samples/Models/GO data warehouse (query)
Report Template: Crosstab
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create a discontinuous crosstab.

- Toolbar: Open a new Crosstab template using the GO data warehouse (query) package.

- Source tab:
  - Add Product line to the rows of the crosstab report object.
  - Add Branch region to the columns of the crosstab report object.
  - Add Year to the rows as a peer of <#Product line#>.
  - Add Quarter to the rows as a child of <#Year#>.
  - Add Revenue as a nested column under <#Branch region#>.
  - Add Quantity to the columns as a peer of <#Revenue#>.

- Toolbar:
  - Sort the <#Branch region#> column ascending.
  - Sort the <#Product line#> row ascending.
  - Sort the <#Year#> row descending.

Results appear as follows:

![Crosstab Report](image-url)
Task 2. Apply formatting to the crosstab fact cells.

- **Toolbar:**
  - Set the Foreground Color for the `<#Revenue#>` Member Fact Cells to Blue.
  - Set the Foreground Color for the `<#Quantity#>` Member Fact Cells to Red.
  - Run the report in HTML.

The results appear as follows:

![Table showing revenue and quantity data for each sales region](image)

- Close the rendered report tab.
- Sign out of IBM Cognos Analytics, if prompted click OK to continue without saving.
- Close all browser windows.

You have created a report that shows revenue and quantity for each product line by year and quarter. The report shows revenue and quantity data for each sales region. You have created create a discontinuous crosstab report that shows rows of unrelated data. You have formatted the measure columns in different colors to be more easily distinguished.
Unit 5    Present data graphically

Present data graphically

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2016
Course materials may not be reproduced in whole or in part without the written permission of IBM.
Unit objectives

- Create charts containing peer and nested columns
- Present data using different chart type options
- Add context to charts
- Create and reuse custom chart palettes
- Introduce visualizations
- Present key data in a single dashboard report
Create a chart report

The chart user interface lets you format and customize different objectives in charts. Data can be displayed graphically to effectively show comparisons, relationships, and trends using one or more of the available chart types.

The IBM Cognos Analytics v11 release includes two charting engines. Users can switch between the legacy chart engine and the default chart engine. The legacy chart engine is no longer the default. Default chart authoring allows you to create several types of standard charts. The chart authoring feature offers rich, visually-appealing charts with additional options to enhance them.

The features and benefits of each chart type are displayed at the bottom of the Insert Chart dialog box when you click a chart type. Many chart aspects can be customized including the title, the axes, 2D and 3D properties, and adding baselines.

Visualizations are also available in this released version of IBM Cognos Analytics.
Different chart options

IBM Cognos Analytics includes a default chart technology. You can continue to use and work with the legacy charts or upgrade your legacy charts to the default charts.

The new default chart technology provides a greater and more updated list of chart types and options for presenting your data in a meaningful way.

Visualizations are also available, but must be added by the administrator to the visualization library.
Create charts containing peer and nested items

You can use chart drop zones to add items as parents, peers or children of other items in the chart, allowing you to quickly create and customize charts to meet your business needs.

Since multiple items are often added to the vertical axis of a chart, the Series area of chart types contains an additional drop zone that can be used to create peer unions between items.

Even though there is no additional drop zone shown for the horizontal axis, you can create peer unions between items on the horizontal axis.
Create and reuse custom chart palettes

Patterns are especially useful when users print charts in black and white.

You can create a chart palette that contains only patterns, or you can create a palette that contains a combination of patterns, colors, and gradients.

To reuse a custom palette, copy the palette to the clipboard and then paste the palette into a different chart report.

If you create a custom chart palette, to save time, you can copy the palette from one chart, and then paste it into different charts.

When using patterns in charts, the chart displays best when you include borders for chart elements such as the bars or pie slices.

There is a ready-made Patterns palette that report authors can select when defining chart palettes.

You can change the foreground and background colors for patterns in the palette. For example, you could change the foreground color of a pattern to white and the background of the pattern to black.
Add data-driven baselines and markers to charts

To help consumers analyze data, you can add data-driven baselines to charts. Baselines help report consumers to quickly identify target or threshold values in charts.
Demonstration 1: Create and format a chart report
Demonstration 1: Create and format a chart report

Purpose:
You will create a combination chart displaying yearly revenue generated by different regions, product lines. You want users to easily distinguish between regional data and yearly data. Because this report will be printed in black and white, you will create a custom palette for the chart and then reuse it for the second series chart. You will add baselines for this chart to display the mean, and plus or minus one standard deviation.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Report Type: Blank
Package: Team content\Samples\Models\GO data warehouse (query)
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the combination chart.
1. Open a new Blank reporting template, using the GO data warehouse (query) package.
2. Click a blank part of the work area to bring it into focus.
3. On the Application bar, click Show properties, click Select Ancestor, and then click Report.
5. Click the Advanced tab, ensure that Use legacy chart authoring is cleared, and then click OK.
6. On the Application bar, click Page views , and then click Page preview.
7. In the work area, click Add , and then click Chart.
8. From the left pane, click Combination, then from the right pane, click Stacked Bar and Stacked Area , and then click OK.
9. Click Page views , and then click Page design.
10. From the **Source** tab, add the following query items to the new chart report object:

**Default measure (y-axis) drop zone:**
- Sales fact: **Revenue**

**Categories (x-axis) drop zone:**
- Products: **Product line**

**Series (primary axis) drop zone (drop in the upper area of this drop zone):**
- Retailers: **Region**

### Task 2. Combine area and bar charts in a single presentation.

You want the Region to appear as an area clustered chart rather than an area stacked chart.

1. In the work area, in the **Series (primary axis)** area of the chart object, click the Chart icon next to `<#Region#>`.

2. In the Properties pane, under **GENERAL**, click the **Series type** property and then select **Clustered** from the list.

You also want to add a clustered bar chart to display the revenue generated for each product line by year.

3. From the **Source** tab, expand **Time**, and then drag **Year** to the empty **Series (primary axis) drop zone** beneath `<#Region#>`.

4. Click the Chart icon for the `<#Year#> series.
5. In the **Properties** pane, under **GENERAL**, click **Series type**, and then select **Clustered** from the list.

The results appear as follows:

![Graphical representation of clustered series type](image1.png)

6. Run the report in **HTML**.

The results appear similar to as follows:

![HTML rendered report](image2.png)

Both the region and the year data appear on the chart, however it is difficult to see the values of the different regions. The chart can be customized further in order to provide a better view of the data.

7. Close the rendered report tab.
**Task 3. Modify by sorting and by changing bar shape.**

1. In the **Series** area, click the `<#Region#>` text.
2. On the list toolbar, click **Sort**, and then click **Ascending**.
3. In the **Series** area, click the `<#Region#>` chart icon, and then in the **Properties** pane, under **BOX**, click **Borders**, and then select **Show** from the list.
4. In the **Series** area, click the `<#Year#>` text.
5. On the list toolbar, click **Sort**, and then click **Ascending**.
6. In the **Series** area, click the `<#Year#>` chart icon, then in the **Properties** pane, under **GENERAL**, click **Bar shape**, and then select **Cylinder**.
7. Click the chart background, to select it.
8. In the **Properties** pane, under **GENERAL**, click **Depth**, and then select 75.
9. Run the report in **HTML**.

The results appear as follows:

![Graph showing bar and area charts for region and year data](image)

The year data appears as a bar chart and the region data appears as an area chart. This allows the yearly revenue generated by each product line to be compared with the revenue generated by each region.

10. Close the rendered report tab.

**Task 4. Format an axis title.**

1. In the chart area, expand **Axis titles**.
2. Click **(Default category axis title)**.
3. From the **Properties** pane, under **GENERAL**, change **Default Title** to **No**.
4. Double-click **Double-click to edit text**, to open the **Text** dialog box.
5. In the **Text** dialog box, type **Product Lines - Total Revenue:**, press the space bar, and then click **OK**.
6. Repeat steps 2 to 5 to add the title **Revenue by Year and Region** to the **Primary Axis Title**.
7. Click **Toolbox**, and then expand **TEXTUAL**.
8. Drag a **Query calculation** to the end of the text in the horizontal axis title drop zone.

Note: Insert a query calculation into your report to add a new row or column with values that are based on a calculation.
Insert a layout calculation to add run-time information, such as current date, current time, and user name.

9. In the **Name** box, replace the text with **Total Revenue for Product Lines**, and then create and validate the following expression:

   ```
   total([Revenue])
   ```

   Hint: drag Revenue from the Data Items tab.
10. Click **OK** to close the dialog box.
11. Run the report in **HTML**.
The results appear as follows:

![Graph displaying product line revenue](image)

The total product line revenue displays under the horizontal axis.


**Task 5. Create a custom palette (optional).**

Tasks 5-7 are optional; however, all tasks must be completed - otherwise, they should not be done at all.

This chart will be printed in black and white, therefore, you will create a custom palette that uses the Gray Scale palette and patterns.

1. Click the `<#Region#>` series chart icon, in the **Properties** pane, under **COLOR & BACKGROUND**, double-click the **Palette** property.

2. In the **Chart Palette Presets** list, select **Gray Scale**.

You want to add some patterns to the palette so that there are enough palette entries for all the items in your chart.

3. Under the left pane, click **New**.

4. In the **Fill type** list, select **Pattern**, change the **Default color** and **Foreground color** properties to the Basic color **Black**, and then change the **Background color** to **White**.

A new pattern entry (horizontal line) is added to the palette.

You will now add four additional entries.

5. Click **New**.
6. In the Fill type list, select Pattern, in the Pattern pane, click the second option (vertical lines), change the Default color and Foreground color to Black, and then change the Background color to White.

   The results appear as follows:

   ![Palette Image](image.png)

7. With the new vertical line pattern still selected, under the left pane, click Move Down to move the new pattern below the second gray scale entry.

8. Repeat steps 5 to 7, to add these three additional patterns to the palette:
9. For the last pattern created, click **Move Up** until the new pattern (hash marks), that you just added, appears at the top of the list of palette entries. The results appear as follows (note you will not see one of the patterns, due to scrolling…but it is there…):

10. Click **OK** to close the **Palette** dialog box.
11. Run the report in **HTML**.
The results appear as follows:

![Chart](image)

12. Close the rendered report tab.

**Task 6. Use transparent colors.**

We want to be able to clearly see both the regions and the years without the two different types of palettes obscuring the chart. To do this, we will change the Year data (vertical cylinders) to use transparent colors.

1. Click the background of the chart, and then click the `<#Year#>` series chart icon.
2. In the **Properties** pane, double-click the **Palette** value.
3. With the top color selected, change the **Fill type** to **Color**, and then change the **Transparency (%)** to 30.
4. Click the second color (orange), and then change the **Fill type** to **Color**.
5. Change the **Transparency (%)** to 30.
6. Click the third color (dark green), and then change the **Fill type** to **Color**.
7. Change the **Transparency (%)** to 30.
8. Click the fourth color (light green), and then change the **Fill type** to **Color**.
9. Change the **Transparency (%)** to 30, and then click **OK** in the **Palette** dialog.
10. Run the report in **HTML**.

The results appear as follows:

![Graph showing revenue by year and region](image)

11. Close the rendered report tab.

**Task 7. Add baselines to the chart.**

1. Click the chart background, and then in the **Properties** pane, under **CHART ANNOTATIONS**, double-click the **Numeric baselines** property. You will add a baseline to display the mean revenue based on year.
   1. In the **Baselines** dialog box, click **New**, and then from the list, select **Mean**.
   2. Ensure that the following properties are set as follows:
      - Based on: **Year**, **Year**
      - * Number of standard deviations: 0
      - Baseline Label: **Mean**
      - Line Styles: Weight: 2 px, Color: **Blue**

        * Note: Do not use the Delete key when modifying the Number of standard deviations input field as it might delete the newly created baseline. Instead, select the contents of the input field and then enter 0 or use the Backspace key to clear the default number and then enter 0.

2. Click **OK** to close the **Line styles** dialog box.

You will add a baseline to display a +1 standard deviation from the mean revenue based on Year.
4. In the **Baselines** dialog box, click **New**, and then from the list, select **Mean**.
5. Ensure that the following properties are set as follows:
   - Based on: **Year, Year**
   - Number of standard deviations: **1**
   - Baseline Label: **Mean + 1 STD Deviation**
   - Line Styles: Weight: **2 px**, Color: **Green**
6. Click **OK** to close the **Line styles** dialog box.
   You will add a baseline to display a -1 standard deviation from the mean revenue based on Year.
7. In the **Baselines** dialog box, click **New**, and then from the list, select **Mean**.
8. Ensure that the following properties are set as follows:
   - Based on: **Year, Year**
   - Number of standard deviations: **-1**
   - Baseline Label: **Mean - 1 STD Deviation**
   - Line Styles: Weight: **2 px**, Color: **Red**
9. Click **OK** to close the **Line styles** dialog box.
   The results appear as follows:

```
Baselines
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td></td>
</tr>
<tr>
<td>Mean + 1 STD Deviation</td>
<td></td>
</tr>
<tr>
<td>Mean - 1 STD Deviation</td>
<td></td>
</tr>
</tbody>
</table>
```
10. Click **OK** to close the **Baselines** dialog box.
11. Run the report in **HTML**.
   The results appear as follows:

   ![Chart Image]

   The chart uses a custom palette and displays the baselines you specified.

12. Close the rendered report tab.

**Results:**
You created a report using a combination chart to display revenue generated in different regions as an area chart, and a bar chart displaying revenue generated for different years. You added data to the horizontal axis title displaying the total revenue generated by all product lines and created a custom palette for the region area chart. You then reused this palette for the year bar chart, and then added data-driven baselines to this chart.
Compare values and highlight proportions using gauge charts and pie charts

Gauge charts are useful for comparing values between a small number of variables.
A gauge chart plots a data series against a measure using a dial or gauge for the measure, and needles or indicators for the series members.
Pie charts highlight data proportionally against a measure, allowing for quick identification of major performers.
Demonstration 2

Create a gauge report and a pie chart report

![Gauge report and pie chart illustrations](image-url)
Demonstration 2: Create a gauge report and a pie chart report

Purpose:
You want to create a chart for users to quickly compare how different product lines are selling. You would also like to see this data represented proportionally. A gauge chart is a good way to show comparisons between multiple variables, while a pie chart will show the data proportionally.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: Gauge Chart with Beveled Border
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create a gauge chart.
1. Open a new Chart template using the GO data warehouse (query) package.
2. From the left pane click Gauge, and then click OK to accept the default.
3. Click the background of the chart to select it, and then from the Source tab, add the following query items to the new list report object:
   - Default measure (y-axis) drop zone:
     - Sales fact: Revenue
   - Categories (gauges) drop zone:
     - Time: Year
   - Gauge Axes drop zone:
     - Products: Product line
4. Double-click **Double-click to edit text**, at the top of the chart.
5. In the **Text** dialog box, type **Revenue by Product Line**, and then click **OK**.
6. On the **Application** bar, click **Show properties**.
7. Click the title block (not the text) to select it.
8. Under **POSITIONING**, double-click **Size & overflow**, in the **Width** box, type **500**, and then click **OK**.
9. Click the chart background (only the Guage chart should show selected), and then in the **Properties** pane, under **POSITIONING** (note: you may have to scroll down to find the property), double-click the **Size & overflow** property.
10. In the **Width** and **Height** boxes, type **500**, and then click **OK**.

**Task 2. Modify the axis labels and gauge properties.**

1. Click the **Axis Label**, and then under **FONT & TEXT** of the **Properties** pane, double-click **Font**.
2. Click **Foreground Color**, click **Black**, and then click **OK**.
3. Click **Bold**, and then click **OK**.
4. In the **Properties** pane, under the **DATA** section, click the **Data format** ellipsis.
5. Change the **Format type** to **Number**.
6. Change the **Scale** to **-6**.
7. In the **Pattern** box (scroll down), enter **$###,### M**.
8. Click **OK**.

**Task 3. Modify the gauge properties.**

1. Click the chart background.
2. In the **Properties** pane, under **GENERAL**, double-click **Gauge border**.
3. Click **Color**, click **Navy**, and then click **OK** to close the **Color** dialog box.
4. Click **OK** to close the **Gauge border** dialog box.
5. Under **COLOR & BACKGROUND** (scroll down), double-click **Dial face fill**.
6. From **Fill type**, select **Linear Gradient**, under **Colors**, click the first color option, and then click **Color**.
7. Click the **Color swatch** tab, click **#CCCCCC** (10th row, 11th column), and then click **OK**.
8. Under **Colors**, click the second color option, and then click **Color**.
9. Click the **Color swatch** tab, click **#CCFFFF** (10th row, last column), and then click **OK**.
10. In the **Position (%)** box, change the value to **50**.

The results appear as follows:

![Fill Effects dialog box](image)

11. Click **OK** to close the Fill Effects dialog box.

**Task 4. Modify the arc colors.**

1. Click the icon under **Gauge Axes**, for `<#Product line#>`.

2. In the **Properties** pane, under **COLOR & BACKGROUND**, double-click **Gauge axis colors**.

   To indicate product lines that are selling poorly, the low end of the arc will appear in red.

3. In the **Gauge axis Colors** dialog box, click the top color, and then click **Color**.

4. From the **Basic colors** tab, click **Red**, and then click **OK**.

5. In the **Gauge axis Colors** dialog box, click the middle color, and then click **Color**.

6. From the **Basic colors** tab, click **Yellow**, and then click **OK**.

7. Ensure the center color (yellow) in the list is selected, and then change the position percentage to **50%**.
8. In the **Gauge axis Colors** dialog box, click the bottom color, and then click **Color**.
9. From the **Basic colors** tab, click **Green**, and then click **OK**.
10. Click **OK** to close the **Gauge axis Colors** dialog box.
11. Run the report in **HTML**.
   A section of the results are as follows:
   ![Revenue by Product Line](image)

12. Close the rendered report tab.

**Task 5. Create a pie chart.**
1. Open a new **Chart** template using the **GO data warehouse (query)** package.
2. On the left pane, click **Pie, Donut**, click **Pie with 3-D Effects and Rounded Bevel**, and then click **OK**.
3. Click the background of the chart to select it.
4. Add the following query items to the pie chart:
   - **Default measure**: drop zone:
   - **Sales fact**: **Revenue**
   - **Series (pie slices)**: drop zone:
     - **Products**: **Product line**
   Reminder: It only takes two values to define a chart.

**Task 6. Set the properties of the chart.**
1. Click the chart background.
2. On the **Application** bar, click **Show properties**.
3. In the **Properties** pane, under **GENERAL**, double-click the **Exploded slices** property, click **New**, and then change the **Slice number** to **2**.
4. Click **OK** to close the **Exploded slice** dialog box.
5. Click **OK** to close the **Exploded slices** dialog box.
6. In the **Properties** pane, under **BOX**, change **Borders** to **Show**.
7. In the Properties pane, under COLOR & BACKGROUND, double-click Palette.
8. From Chart palette presets, click Dynamic, and then OK.

**Task 7. Create chart title.**
1. Double-click the chart title box labeled Double-click to edit text.
2. In the text dialog box, type Revenue by Product Line, and then click OK.
3. With the text title still selected, change the Font to Arial Black, 16pt, and Underlined.
4. Select the header block, and then left justify it so it is no longer centered.
5. Run the report in HTML.
6. Move the cursor over the expanded green slice to view the tooltip.
The results are as follows:

![Revenue by Product Line Chart](image)

7. Close the rendered report tab.

**Results:**
You have created a gauge chart for users to quickly compare how different product lines are selling. You have also created a pie chart to show the data proportionally.
Display items on separate axes

You can improve the clarity of charts by displaying values for different items on separate axes.

Using separate axes is useful when the value ranges for different items displayed in the chart are significantly different.
Demonstration 3

Show the same data graphically and numerically
Demonstration 3: Show the same data graphically and numerically

Purpose:
You want to create a report that shows revenue and quantity by Product line and Region. You want the report to focus on Camping Equipment, Mountaineering Equipment, and Personal Accessories sales for the three European sales regions. You will build a crosstab report and add a combination chart that reports on the same information. You will add a microchart to the crosstab for a quick overview of specified regions and product lines.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team Content\Samples\Models\GO data warehouse (query)
Report Type: Crosstab
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Add query items to a new crosstab.

1. Open a new Crosstab template using the GO data warehouse (query) package.
2. Add the following query items:
   - Rows:
     - Order method: Order method type
   - Columns:
     - Time Year
     - Sales Fact: Revenue, Quantity (nested under Year)
Task 2. Create a combination chart.

1. From the Toolbox tab, drag a Chart data container to the left of the crosstab. The Insert chart dialog opens.
2. Select Fill with data from.
3. In the left pane, click Combination, and then click OK to accept the default Clustered Bar and Clustered Line chart.
   A new chart is added and populated based on the crosstab.
4. Click the chart background to provide focus.
5. Run the report in HTML.

You can see in the combination chart that the top revenue generating order method type is the Web. The crosstab provides the exact numbers and provides detail to what is seen graphically in the chart.

6. Close the rendered report tab.
   You want the chart to use the same query as the crosstab and to show the Revenue and the Quantity on separate y axes.

Task 3. Show two measures on different y axes.

1. Click the chart background, and then click Show properties.
2. In the Properties pane under DATA, click Query2, and then change it to Query1 from the list.
3. From the Properties pane, under GENERAL, double-click Combinations.
4. Click the Secondary Axis checkbox to select it, and then click Edit.
5. Click **Line**, click **OK** to close the **Combination Element** dialog box, and then **OK** to close the **Combinations** dialog box.

6. From the **Series (primary axis)**, drag `<#Order method type#>` from the **Line** chart type to the **Series (secondary axis)**, and then drag `<#Quantity#>` under `<#Order method type#>`.

7. Click the now empty **Line** chart type icon, under **Series (primary axis)**, and then press **Delete**.

   The entire section may disappear. If it does, you may need to click the chart background again, to see the results of your Delete action just performed.

   The results appear as follows:

   ![Chart](image)

8. Run the report in **HTML**.

   A section of the results appear as follows:

   ![Report](image)

   This chart may be too complicated for your consumers to read clearly. In Task 4 you will add filters to report only on E-mail, Sales visits, and Telephone. Web will not be included since it is the clear winner for Revenue and Quantity.

9. Close the rendered report tab.
Task 4. Add filters to focus the data.
1. Click the combination chart to select it.
2. On the chart toolbar, click Filters, and then click Edit Filters.
3. Click Add, and then click OK.
4. From the Values box, move the following to the Selected values box: E-mail, Sales visit, Special, and Telephone, and then click OK.
5. In the Filters dialog box, under Application, ensure that Before auto aggregation is selected, and then click OK to close the Filters dialog box.
6. Run the report in HTML.
The result appears as shown below:

![Graph showing data]

7. Close the rendered report tab.

Task 5. Add a microchart to the crosstab to preview data in a chart.
1. In the crosstab, click <#Order method type#>, on the crosstab toolbar click More, and then click Insert chart for Row Data.
2. In the left pane, verify that Microchart is selected, and then click OK to accept the default Line microchart.
3. In the Categories (x-axis) section of the microchart, drag <#Revenue> to the Default Measure (y-axis).
4. In the **Categories (x-axis)** section of the microchart, click **Quantity**, and then press **Delete**.

The entire section may disappear. If it does, you may need to click the chart background again, to see the results of your Delete action just performed.

The results appear as follows:

![Microchart](image)

5. Run the report in **HTML**.

The results appear as follows:

![HTML Report](image)

6. Close the rendered report tab.

**Results:**

You created a combination chart with two measures on different Y axes and then added a crosstab to see product line sales revenue and quantity by region. You focused on Camping Equipment, Mountaineering Equipment, and Personal Accessories sales for the three European sales regions. You added a microchart to the crosstab for a quick overview of product line revenue for all regions specified.
Customize charts

Custom elements such as color schemes, rescaling of axes numbers, renaming axes, and displaying details can enhance reports.

Fills and Background customization can greatly enhance the visual appeal of charts. Tool tips are available by default, and provide additional information while adding a level of interaction to the chart.

Notes can hide whatever is under them, so it is important to properly position them in the chart so as to not block important information.
What is RAVE?

The Rapidly Adaptive Visualization Engine (RAVE) is being used to enable advanced visualization technology in many different IBM projects and products today.

It is not a traditional charting engine with pre-defined chart types (such as column, and pie charts). Rather, it is a general purpose visualization engine that can produce both traditional and new charts and visualizations.

- RAVE does not describe charts by type (barchart, linechart, histogram, and so on) but by mapping. For example:
  - bar chart - basic 2D coordinates, categorical x numeric displayed with intervals dropped from locations
  - line chart - basic 2D coordinates, any x numeric displayed with lines connecting locations

RAVE supports statistical operations (such as sum, count), and styling (such as color). The grammar-based approach provides flexibility: new charts, or chart attributes, can be added without requiring a new product binary. The declarative language for visualizations (charts, interactivity, events, etc.), is a cross-IBM standard.
What is a visualization?

- A visualization is:
  - intuitive
  - immediate
  - language-independent

Visualization exploits the human visual system to provide an intuitive, immediate and language-independent way to view and show your data. It is an essential tool for understanding information. Visualization can play a key role by making the individual analytic components understandable and by tying them together into a comprehensible "big picture".

Visualizations provide an empowering technology that delivers context to raw data and maximize the perspective of the data.
Demonstration 4: Display visualizations
Demonstration 4: Display visualizations

Purpose:
You have been asked to create a report that compares multiple key performance indicators for all product lines. Users need to be able to quickly identify product line performance. You will use a visualization that was made available in the portal Library to accomplish this task.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: Blank
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Select a visualization.
1. Open a new Blank template using the GO data warehouse (query) package.
2. From the Toolbox tab, expand DATA CONTAINER, and then drag a Visualization object to the right pane.

The Visualization Gallery dialog displays the visualizations that are available:

You could directly click the visualization that you want, from those available in the center panel, but if you have many items to choose from, you will want to filter on a specific type to make it easier to find.
3. In the **Refine by** pane, select the **Bar** checkbox.
   The number displayed for each type, indicates how many visualizations tagged with that type are available in the gallery. Notice how several bar diagrams are available in the center pane.

4. In the **Refine by** pane, select the **Column** checkbox.
   There are now bar diagrams and column diagrams available, and all other visualization types have been filtered from the view.

5. In the **Refine by** pane, click **Clear all**.
   All filters have been removed, and all visualizations are displayed in the center pane.

6. In the **Refine by** pane, select the **Bubble** checkbox, and then in the center pane, select **Packed Bubble**.
   Notice the description of the visualization in the right pane.

7. Click **OK** to open the visualization.
   You could have also clicked directly on Packed bubble in the center pane, instead of filtering it first.

**Task 2. Populate the packed bubble visualization and run the report.**

1. Click **OK** to accept the default values for the **Object and query names** dialog box.
   The page displays a preview of the Packed bubble chart visualization:
2. Open the **Properties** pane.
3. Click the visualization background, and then notice the available properties in the **Properties** pane.

Other properties can be made available by using the Visualization Customizer tool. The tool allows you to create properties that are needed in reporting to further customize the report.

A section of the Properties pane appears as follows:

4. From the **Source** tab, add the following query items to the visualization report object:
   - **Values**: Size Value
     - Sales fact: *Quantity*
   - **Categories**: Series
     - Sales fact: *Planned revenue*, *Revenue*, and *Gross profit*
   - **Categories**: Bubbles (Categories)
   - Products: *Product line*

A section of the results appears as follows:
5. Click the visualization background.
6. In the Properties pane, in the COMMON section, set Show Values to No.
7. Run the report as HTML.
8. Once the report is generated, hover the cursor over one of the bubbles. The results will appear similar to the following:

![Packed Bubble Chart](image)

In this visualization, you can very quickly identify that Personal Accessories came close to its planned revenue goal and had the greatest gross profit. However, you would need an additional detail report to accurately identify the revenue achieved and to compare planned revenue with actual revenue.

9. Close the rendered report tab.

Results:
You used IBM Cognos Analytics - Reporting to create a packed bubble chart report, based on an imported visualization that was made available in the Library. You also successfully added data to the visualization, and ran the report to display the results.
Unit summary

- Create charts containing peer and nested columns
- Present data using different chart type options
- Add context to charts
- Create and reuse custom chart palettes
- Introduce visualizations
- Present key data in a single dashboard report
Exercise 1: Create a dashboard report

Present data graphically

Exercise 1: Create a dashboard report
Exercise 1: Create a dashboard report

You are Frank Bretton, a report author, and have been asked to create an interactive report that lets users examine a variety of important sales data in one view. To do this, you will create a dashboard report that contains a gauge chart that compares the gross profit of each product line by region, a combination chart that shows revenue earned by each product line by retailer type and region on separate axis, and finally a crosstab report that shows the gross margin of each product line by year and region.

To accomplish this:

- Using the GO data warehouse (query) package, the Sales and Marketing (query) folder, and the Sales (query) namespace, add a gauge chart, a combination chart, a crosstab, and a table to a blank template.
- Create a gauge chart (Gauge Chart with Bevelled Border) with Gross profit, Product line, and Region, and then format the gauge chart to enhance its visual appeal.
- Create a combination chart (Clustered Bar and Clustered Line) with Revenue, Product line, and Retailer type with Region as a peer. Format and enhance the visual appeal of the chart.
- Create a crosstab with Gross margin, Product line, and Year with Region as a peer.

For more information about where to work and the exercise results, refer to the Tasks and Results section that follows. If you need more information to complete a task, refer to earlier demos for detailed steps.
Exercise 1: Tasks and Results

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content/Samples/Models/GO data warehouse (query)
Report Template: Blank
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Add charts and a crosstab to a blank template.

- **Toolbar**: Open a new Blank template using the GO data warehouse (query) package.

- **Work area**: Add a Table container that is 2 columns by 2 rows to the work area.

- **Table**: Ctrl-click the 2 bottom cells, from the Table toolbar, select Merge Cells.

- **Work area**: Add a Chart data container to the top left table cell.

- **Insert chart** dialog box: Change the Name field to Gauge Query.
  - From the left pane, click Gauge.
  - Click OK to accept the default gauge chart (Gauge Chart with Bevelled Border).

- **Work area**: Add a Chart data container to the top right table cell.

- **Insert chart** dialog box: Change the Name field to Combination Query.
  - From the left pane, click Combination.
  - Click OK to accept the default combination chart (Clustered Bar and Clustered Line).

- **Work area**: Add a Crosstab data container to the bottom table cell.

- **Object and query name** dialog box: Change the query name to Crosstab Query.
  - Click OK.
Task 2. Add data to the reporting objects.

- **Data/Source** tab: Navigate to Sales and Marketing (query)/Sales (query), and then add the following query items to the gauge chart:
  - **Default measure (y-axis)** drop zone:
    - Sales fact: **Gross profit**
  - **Categories (gauges)** drop zone:
    - Retailers: **Region**.
  - **Gauge Axes** drop zone:
    - Products: **Product line**

- **Data/Source** tab: Navigate to Sales and Marketing (query)/Sales (query), and then add the following query items to the combination chart:
  - **Default measure (y-axis)** drop zone:
    - Sales fact: **Revenue**
  - **Category (x-axis)** drop zone:
    - Products: **Product line**
  - **Series (primary axis)** drop zone:
    - Retailers: **Retailer type**

- **Data/Source** tab: Navigate to Sales and Marketing (query)/Sales (query), and then add the following query items to the crosstab:
  - **Rows area**:
    - Products: **Product line**
  - **Columns area**:
    - Time: **Year**
    - Retailers: **Region** to the right (peer) of **Year**
  - **Measures area**:
    - **Gross margin**
  - **Work area**: Click the combination chart background.
  - **Properties pane**: Under **GENERAL**, double-click **Combinations**.
Combinations dialog box: Under Combinations, click Clustered line, and then click Delete.
- Under Numeric axes, select the Secondary Axis checkbox.
- Click Edit.
- Combination Element dialog box: Under Combination Type, click Line, and then click OK.
- Combinations dialog box: Click OK.
- Data/Source tab: Under Retailers query subject, drag Region to the Series (secondary axis) drop zone of the combination chart.
- In the crosstab, click <#Year#>, and then sort Descending.

The results appear as follows:

Task 3. Enhance the visual appeal of the charts.
- Work area: Click the gauge chart background, and then Ctrl-click the combination chart background to select them.
- Properties pane: Under COLOR & BACKGROUND, double-click Background effects.
- Background effects: Click the Border checkbox to select it.
  - Click Black from the Color list.
  - In the Corner radius box, type 10.
  - Click the Fill checkbox to select it.
  - Click Color, and then click the Color swatch tab.
  - Click #CCFFFF (10th row, last column), and then click OK.
• In the **Position** box, type 45.
• Click **New**.
• Click **Color**, and then click the **Color swatch** tab.
• Click #CCCCCC (10th row, 11th column), and then click **OK**.
• In the **Position** box, type 100.
• In the **Angle** box, type 90.
• Click the **Drop Shadow** checkbox to select it, and then click **OK**.

**Toolbar**: Run the report in HTML.

**Rendered report** tab: Verify the results, and then close the tab.

A section of the results appear as follows:

![Diagram of a combination chart showing regions and product lines with corresponding data]

**Task 4. Format the combination chart.**

• **Work area**: In the combination chart, click **Series (primary axis)** bar.
• **Toolbar**: Click **Chart Palette Presets**.
• **Palette** dialog box: Click the **Contemporary style palette**.
• **Work area**: In the combination chart, click **Series (secondary axis)** line.
• **Toolbar**: Click **Chart Palette Presets**
• **Palette** dialog box: Click the **Contemporary style palette**.
• **Work area**: Click the combination chart background.
• Properties pane: Under Positioning, double-click Size & Overflow.
• Size & Overflow dialog box: In the Width box type 550, in the Height box, type 300, and then click OK.
• Properties pane: Under Chart Titles, click Title, and then in the list, click Show.
• Combination chart: Expand Axis titles, and then click Default category axis title.
• Properties pane: under General, click the Default title property, and then in the list, click No.
• Work area: Double-click the combination chart title text.
• Text dialog box: Type Product Lines: Revenue by Retailer Type and Region, and then click OK.
• Properties pane: Click Select Ancestor, click Chart title.
• Toolbar: Click Arial, 12 pt., Bold.

**Task 5. Format the gauge chart.**

• Work area: Click the background of the gauge chart.
• Properties pane: Under Positioning, double-click the Size & Overflow property, in the Width box type 550, and in the Height box type 300 and then click OK.
• Work area: In the gauge chart, expand Axis titles, and then click Default matrix columns axis title.
• Properties pane, under General, click Default title, and then in the list click No.
• Work area: In the gauge chart, click Gauge Axes for <#Product line#>.
• Properties pane: Under General, double-click Axis Angles.
• Axes Angles dialog box: In the Start angle box, type 320.
  • In the End Angle box, type 220.
  • In the Axis direction list, click Counterclockwise, and then click OK.
• Properties pane: Under Color & Background, double-click Gauge axis colors.
• Gauge axis colors: Click the center color (yellow).
  • In the Position box, type 50, and then click OK.
• Work area: Click the gauge chart background.
• Properties pane: Under Chart Titles, click the Title property, and then click Show.
• **Work** area: In the gauge chart, double-click the chart title text.

• **Text** dialog box: Type *Gross Profit for Product Lines by Region*, and then click OK.

• **Toolbar**: Click Arial, 12 pt., Bold.

**Task 6. Format the crosstab.**

• **Crosstab table cell toolbar**: Click Left and then click Top.

• **Work** area: Add a Block above the crosstab and then add a **Text item** to the Block.

• **Title**: In the **Text item** field, type *Gross margin for Product lines by Year and Region*.

• **Toolbar**: Click Arial, 12 pt., Bold.

• Run the report in **HTML**.

The results appear as follows:

![Image of charts and tables showing gross profit and revenue by region and year](image)

• Close the rendered report tab.

• Sign out from **IBM Cognos Analytics** and if prompted, click OK to continue without saving.

• Close all browser windows.
Unit 6  Focus reports using prompts

Focus reports using prompts

IBM Cognos Analytics (v11.0)
Unit objectives

• Identify various prompt types
• Use parameters and prompts to focus data
• Search for prompt types
• Navigate between pages
Examine parameters and prompts

• Prompts ask the user to provide the value for the parameter that will filter the report on specific data values.

There are three ways to prompt for report specifications:
• create a parameter for an item on the report
• add a prompt page to the report containing one or more prompt items
• add a prompt item to a report

Parameters are placeholders that require a value to determine what data to report on. Prompts function as dynamic filters.

Parameters are based on parameterized filters. The filter consists of a query item and operator. The operator you choose will determine some of the default properties of the prompt. For example, if you choose the = operator the user will only be able to select a single option from the prompt (Multi-Select - No). If you choose the ‘in’ operator, the user will be able to select multiple options from the prompt.

A prompt is nothing more than a dynamic (parameterized) filter.
Create a parameter item on the report

• Use a parameterized filter to create a prompt.

Reporting can automatically generate prompted reports based on parameters you create. When you run the report, Reporting can generate a prompt page for each parameter not associated to an existing prompt page depending on whether the prompt run option is selected or not.

If you create a parameter for an item on a report, when you run the report you will be prompted to specify a value for that item. The report displays the information according to the value given in the prompt. The prompt is created automatically and must be answered in order to view the report.
You can create a prompt page to control how prompts appear in the report. A prompt page can be built by dragging prompt tools from the tool box.

You can also create a blank prompt page by adding a new page to the Prompt Pages section in Page Explorer. Once on the new prompt page, you can drag prompt items onto the work area.

A date item will automatically generate a Calendar prompt, a number item generates a Text Box prompt, and a value item will generate a Value prompt.
Add a prompt item to a report

• When a prompt item is dragged onto a report a prompt wizard walks the report author through the prompt building process.

A prompt item can be added directly to a report. When added, a prompt wizard dialog box appears and steps you through building the prompt. Prompt customization is done at this time.

The wizard will add a prompt control and a parameterized filter to the report:

1. Create a parameter.
2. Add a filter to the data container with the parameter.
3. Create a query for the prompt.
4. Add the query and the parameter to the prompt.

If you add a prompt directly onto a report page, you will either need to set the prompt to automatically submit the selection, or add a Finish prompt button to the report so that the report will regenerate using the new criteria.
Demonstration 1

Create a prompt by adding a parameter

<table>
<thead>
<tr>
<th>Order number</th>
<th>Date</th>
<th>Product</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>104734</td>
<td>Jan 8, 2013</td>
<td>Blue Steel Max Putter</td>
<td>34,320</td>
</tr>
<tr>
<td>104734</td>
<td>Jan 8, 2013</td>
<td>Course Pro Gloves</td>
<td>5,974.5</td>
</tr>
<tr>
<td>104734</td>
<td>Jan 8, 2013</td>
<td>Lady Hailstorm Titanium Irons</td>
<td>73,477.59</td>
</tr>
<tr>
<td>104735</td>
<td>Jan 8, 2013</td>
<td>Course Pro Putter</td>
<td>38,178.52</td>
</tr>
<tr>
<td>104735</td>
<td>Jan 8, 2013</td>
<td>Firefly Multi-light</td>
<td>7,670.36</td>
</tr>
<tr>
<td>104735</td>
<td>Jan 8, 2013</td>
<td>Hailstorm Steel Irons</td>
<td>22,773.4</td>
</tr>
<tr>
<td>104735</td>
<td>Jan 8, 2013</td>
<td>Hailstorm Steel Woods Set</td>
<td>52,234.8</td>
</tr>
<tr>
<td>104735</td>
<td>Jan 8, 2013</td>
<td>Lady Hailstorm Steel Irons</td>
<td>43,525.46</td>
</tr>
</tbody>
</table>
Demonstration 1:
Create a prompt by adding a parameter

Purpose:
You have been asked to provide a report showing product sales by date to
determine the revenue generated by each individual order. Because the report
contains detailed information, you want to be able to filter the report to show
only orders made after a specified date. You will create a parameter to prompt
a user for a date and the report will return all dates greater than the one
specified.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the list.
1. Open a new List template using the GO data warehouse (query) package.
2. From the Source tab, add the following query items to the new list report object:
   • Sales order: Order number
   • Time: Date
   • Products: Product
   • Sales fact: Revenue
   The results appear as follows:

<table>
<thead>
<tr>
<th>Order number</th>
<th>Date</th>
<th>Product</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Order number&gt;</td>
<td>&lt;Date&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Order number&gt;</td>
<td>&lt;Date&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Order number&gt;</td>
<td>&lt;Date&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
</tbody>
</table>

3. Click the <Date> list column body, on the list toolbar click Sort, and then click
   Ascending.
4. Run the report in **HTML**.
A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Order number</th>
<th>Date</th>
<th>Product</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>100001</td>
<td>Jan 12, 2010</td>
<td>Flicker Lantern</td>
<td>8,624.64</td>
</tr>
<tr>
<td>100001</td>
<td>Jan 12, 2010</td>
<td>Polar Ice</td>
<td>9,411.6</td>
</tr>
<tr>
<td>100002</td>
<td>Jan 12, 2010</td>
<td>Bear Edge</td>
<td>6,690.8</td>
</tr>
<tr>
<td>100002</td>
<td>Jan 12, 2010</td>
<td>Edge Extreme</td>
<td>18,032.22</td>
</tr>
</tbody>
</table>

The earliest date is Jan 12, 2010.

5. Click **Bottom** to see the last page of the report.
The last date is Jul 20, 2013.

6. Close the rendered report tab.

**Task 2. Add a date parameter and run the report.**

1. Click the list to regain focus.
2. On the list toolbar, clicks **Filters**, **Edit Filters**, and then click **Add**.
3. Click **Advanced**, and then click **OK**.
4. Create and validate the following expression:
   
   
   \[ \text{[Date]} > {?Date} \]
   
   Hint:
   
   • drag Date from the Data Items tab
   • validate for 2013-Jan-1
   
   The report will only retrieve data where the order date is greater than the date specified by the user.

5. Click **OK** to close the **Detail filter expression** dialog box, and then click **OK** to close the **Filters** dialog box.

6. Run the report in **HTML**.
You are prompted to select a date and time.

7. Select **2013-Jan-1**, accept the default time, and then click **OK**.
A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Order number</th>
<th>Date</th>
<th>Product</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>104734</td>
<td>Jan 8, 2013</td>
<td>Blue Steel Max Putter</td>
<td>34,320</td>
</tr>
<tr>
<td>104734</td>
<td>Jan 8, 2013</td>
<td>Course Pro Gloves</td>
<td>5,974.5</td>
</tr>
<tr>
<td>104734</td>
<td>Jan 8, 2013</td>
<td>Lady Hailstorm Titanium Irons</td>
<td>73,477.59</td>
</tr>
<tr>
<td>104735</td>
<td>Jan 8, 2013</td>
<td>Course Pro Putter</td>
<td>38,178.52</td>
</tr>
</tbody>
</table>
8. Click **Bottom** to see the last page of the report.
The report displays results from Jan 8, 2013 to Jul 20, 2013.
9. Close the rendered report tab.
10. Leave the report authoring tab open for the next demonstration.

**Results:**
You created a parameter to prompt a user for a date, and when the list report ran, it returned information based on the response to the prompt.
### Identify prompt type

- Choose the appropriate prompt type and style for your reporting requirements.

<table>
<thead>
<tr>
<th>Text box prompt</th>
<th>Time prompt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value prompt</td>
<td>Interval prompt</td>
</tr>
<tr>
<td>Select &amp; search prompt</td>
<td>Tree prompt</td>
</tr>
<tr>
<td>Date &amp; time prompt</td>
<td>Generated prompt</td>
</tr>
<tr>
<td>Date prompt</td>
<td>Prompt button</td>
</tr>
</tbody>
</table>

If you add prompt items to a report or prompt page, you can choose from the different types of prompts available in the Toolbox tab according to your needs. If you select items on a report and then create a prompt page, Reporting will choose an appropriate prompt type for you.

Similarly, a generated prompt acts as a placeholder in the work area, but when the report is run, Reporting selects the appropriate prompt type for that report.

If there are a large number of choices available (such as sales rep name), then Select & Search is a good option. This saves time in scrolling to look for the desired option. If the exact name or spelling of an item is unknown, then avoid using the Text Box prompt as the value must be typed in exactly as it appears in the report.

Interval prompts are valuable for reporting on very specific time frames as they let you choose lowest to highest time intervals in days, hours, and minutes.
Demonstration 2

Add a value prompt to a report
Demonstration 2: Add a value prompt to a report

Purpose:
You will create a report to help reduce production costs. Because you have many products, you will add a prompt so that users can view product data within a specified product line.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the list.
1. Open a new List template using the GO data warehouse (query) package.
2. From the Source tab, add the following query items to the new list report object:
   - Products: Product line, Product type, Product
   - Sales fact: Unit cost
   - Gross margin (calculated fact under Sales (query))
3. In the report, click <Product line>, Ctrl-click <Product type>, and then click Group / Ungroup on the List toolbar.
   The results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Product</th>
<th>Unit cost</th>
<th>Gross margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product line</td>
<td>Product type</td>
<td>Product</td>
<td>Unit cost</td>
<td>Gross margin</td>
</tr>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product type&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Unit cost&gt;</td>
<td>&lt;Gross margin&gt;</td>
</tr>
<tr>
<td>&lt;Product type&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Unit cost&gt;</td>
<td>&lt;Gross margin&gt;</td>
<td></td>
</tr>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product type&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Unit cost&gt;</td>
<td>&lt;Gross margin&gt;</td>
</tr>
</tbody>
</table>

4. Delete the Double-click to edit text from the block at the top of the report.
5. From the Toolbox tab, drag a Table object into the Block object, clear the Maximize width checkbox, and then click OK.
6. Click in the Block object, and then from the toolbar, left justify the table.
Task 2. Add a product line prompt to the report page.

1. From the Toolbox tab, expand PROMPTING, and then drag a Value prompt into the left table cell.

You want the prompt to filter on product line code to make the query more efficient. However, the prompt will display product line names, not codes, to make selections easier.

2. In the Prompt Wizard - Value prompt dialog box, change the parameter name to ProductLineCode, and then click Next.

3. On the Create Filter page, ensure that Create a parameterized filter is selected, and then beside Package item, click the ellipsis.

4. Expand the Sales and Marketing (query) folder, Sales (query) namespace, Products, and then Codes folder.

5. Click Product line code, and then click OK.

Product line code is used because it is an indexed field. Querying on an indexed field is much faster and more efficient.

6. Select Make the filter optional.

A section of the results appear as follows:

7. Click Next, and then beside Values to display, click the ellipsis.

8. Expand the Sales and Marketing (query) folder, expand the Sales (query) namespace, and then expand Products.

9. Click Product line, click OK, and then click Finish.

The results appear as follows:
Task 3. Add a prompt button and set the properties for the value prompt.

1. From the Toolbox tab, in the PROMPTING section, drag a Prompt button into the right table cell.
2. In the Properties pane, under GENERAL, click Type, and then from the list, select Finish.
3. From the side panel, click Navigate, and then click the Query explorer.
4. Click Query2, and then under Data Items, click Product line code1.
5. In the Properties pane, under DATA ITEM, double-click Expression.
6. Change the Expression Definition to [Sales (query)].[Products].[Product line].
7. Validate the expression, and then click OK.
8. Run the report in HTML.
   The report opens in the browser, displaying data for all product lines. The report can run because the parameterized filter is defined as optional.
9. From the Product line prompt list, select Golf Equipment, and then click Finish.
   Only Golf Equipment product line information is displayed.
10. Close the rendered report tab.
11. Click Page explorer, and then click Page1 to return to the report design.
12. Click the Finish prompt button, and then press Delete.
13. Click the Value Prompt, to select it.
14. In the Properties pane, under GENERAL, click Auto-submit, and then change the property to Yes.

Task 4. Customize the prompt.

You want to customize the prompt header to provide instructions on how to use the prompt.

1. With the value prompt selected, in the Properties pane, under PROMPT TEXT, click Header Text, and then click the ellipsis.
2. Select Specified text, and then click the ellipsis to the right of the text box.

To control the usage of your prompt between required or optional, make the change directly on your filter through your filters Usage property instead of changing this setting on your prompt. The filters Usage property overrides the prompts Required property.
3. Type the following in the **Default text** box: **Select the desired Product line**.
The results appear as follows:

![Localized text dialog box]

You can add information here as well for localization.

4. Click **OK** to close the **Localized Text** dialog box, and then **OK** to close the **Header Text** dialog box.

   You want to have the value prompt separate from the list. You will add a space between the prompt and the list by increasing the top margin of the list.

5. Click the list **Container selector** to select the entire list.

6. From the **Properties** pane, under **BOX**, double-click the **Margin** property.

7. Type **20** in the Top margin cell, and then click **OK**.
Task 5. Run the report.

1. Run the report in HTML.
   The report opens in the browser displaying data for all product lines. The report can run because the parameterized filter is defined as optional.

   ![Select the desired Product line](image)

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Product</th>
<th>Unit cost</th>
<th>Gross margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Golf Equipment</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mountaineering Equipment</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Outdoor Protection</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Personal Accessories</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

   ![Golf Equipment](image)

   With the Auto-submit property set to Yes, you can use the Product line list to select which Product line data you want to display without having to click an additional button to submit your selection.

2. In the list, select Golf Equipment.
   A section of the results appear as follows:

   ![Golf Equipment](image)

   Results:
   You created a report to show cost and gross margin for each product. You added a prompt so that users can view product data within a specified product line.
Add pages to a report

Enhance your report by adding multiple report and prompt pages.

By accessing Page Explorer from the Explorer bar, you can navigate between report pages and prompt pages. You can also add or delete report pages and prompt pages by clicking the Report Pages link or the Prompt Pages link.
Demonstration 3

Add a Select & search prompt to a report

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Product</th>
<th>Unit cost</th>
<th>Gross margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Lanterns</td>
<td>EverGlow Butane</td>
<td>40.63</td>
<td>40.02%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EverGlow Kerosene</td>
<td>20.00</td>
<td>41.05%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly 2</td>
<td>16.30</td>
<td>40.99%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly 4</td>
<td>17.04</td>
<td>44.64%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly Extreme</td>
<td>29.10</td>
<td>51.59%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly Lite</td>
<td>6.75</td>
<td>62.84%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly Maestro</td>
<td>7.50</td>
<td>52.14%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly Multi-light</td>
<td>17.70</td>
<td>37.67%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly Charger</td>
<td>22.38</td>
<td>55.47%</td>
</tr>
<tr>
<td>Mountaineering Equipment</td>
<td>Climbing Accessories</td>
<td>Firefly Climbing Lamp</td>
<td>21.57</td>
<td>38.33%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly Rechargeable Battery</td>
<td>3.15</td>
<td>54.09%</td>
</tr>
</tbody>
</table>
Demonstration 3: Add a Select & search prompt to a report

**Purpose:**
You want to change your current report to allow users to select multiple products to show in the report. To do this you must delete the current value prompt and replace it with the Select & search prompt.

Note: Before starting this demonstration, be sure to complete Demonstration 2. The report ("Unit 6-Prompt") from Demonstration 2 is used for Demonstration 3.

User/Password: brettonf/Education1
Package: Team \Samples\Models\GO data warehouse (query)
Report Type: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

**Task 1. Run the report.**

1. With the report from the previous demonstration still open (My Content\Unit 6-Prompt), run the report in HTML.
2. In the Product line list, select Camping Equipment.
   A section of the results appear as follows:

   ![Camping Equipment Table](image)

   The report includes information on only one product line. You will now modify the prompt to let users search and select one or more product names.
3. Close the rendered report tab.
**Task 2. Add a Select & search prompt on a prompt page.**

1. In the work area, click the **Value** prompt, and then delete it.
2. Click the list data container, from the list toolbar, click **Filters**, and then click **Edit Filters**.
   Notice that the prompt filter remained even when the Value prompt was deleted.
3. Select the filter, delete it, and then click **OK**.
4. From the side panel, click **Navigate**, if necessary, and then click **Prompt pages**.
5. From the **Toolbox**, drag a **Page** object to the **Prompt pages** pane.
6. Double-click **Prompt page1**, expand **PROMPTING**, and then drag a **Select & search prompt** object onto the prompt page.
7. In the **Choose Parameter** page, change the parameter name to **Productnames**, and then click **Next**.
8. In the **Create Filter** page, ensure that **Create a parameterized filter** is selected, and then beside **Package item**, click the **ellipsis**.
9. Expand the **Sales and Marketing (query)** folder, **Sales (query) namespace**, **Products** folder, click **Product**, and then click **OK**.
10. Select **Make the filter optional**.
11. In the **Operator** list, select **in**, click **Next**, and then click **Finish**.
   The report appears with the Select & search prompt on the prompt page.
   You use the 'in' operator to allow for multiple selections. If you used the '=' operator, the prompt would allow for only a single selection.
12. Left justify the **Double-click to edit text** in the header block.

**Task 3. Run the report.**

1. Run the report in **HTML**.
2. Click **Finish** to accept the default of all product lines.
   You can navigate to view product data on other pages. The report ran because you made the prompt optional. The name of the product you want to search for contains the keyword "Firefly".
3. Click **Run** to run the report again.
4. In the **Keywords** text box, type **Firefly** (no other text should appear in the text box), and then expand the options drop down list at the bottom.
   The list of search options allows you to refine your search.
5. Select **Contains any of these keywords**, and then click **Search**.
   All product lines with "Firefly" in the name appear in the Results box.
6. Click **Select all**, and then click **Finish** to see the **Firefly** results.

7. Click **Run Report** to run the report again.

8. In the **Keywords** text box, type **Firefly Butane Kerosene**.

9. Select **Contains any of these keywords**, and then click **Search**.

10. Click **Select all**, and then click **Finish**.

The results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Product</th>
<th>Unit cost</th>
<th>Gross margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camping Equipment</td>
<td>Lanterns</td>
<td>EverGlow Butane</td>
<td>40.63</td>
<td>49.092%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EverGlow Kerosene</td>
<td>20.00</td>
<td>41.057%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly 2</td>
<td>16.38</td>
<td>48.909%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly 4</td>
<td>17.84</td>
<td>44.545%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly Extreme</td>
<td>29.10</td>
<td>51.590%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly Lite</td>
<td>6.75</td>
<td>62.846%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly Mapreader</td>
<td>7.50</td>
<td>62.146%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly Multi-light</td>
<td>17.78</td>
<td>37.673%</td>
</tr>
<tr>
<td>Mountaineering Equipment</td>
<td>Climbing Accessories</td>
<td>Firefly Charger</td>
<td>22.36</td>
<td>55.479%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly Climbing Lamp</td>
<td>21.57</td>
<td>38.336%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Firefly Rechargeable Battery</td>
<td>3.15</td>
<td>54.869%</td>
</tr>
</tbody>
</table>

The report runs and is filtered to display only the data associated with the products that you selected based on your search.

11. Close the rendered report tab.

12. Leave the report authoring tab open for the next demonstration.

**Results:**

You removed the existing value prompt and filter and updated the existing report with a Select & search prompt. This allowed users to search for and select from, a list of product names based on keyword options.
Create a cascading prompt

• Use values from a previous prompt to filter the values in the current prompt or picklist.

In the slide example, the selection that the user makes for Product line determines what is populated in the Product type prompt. The selection made for Product type determines what is populated in the Product name prompt.

When you create a series of prompts that have a hierarchical relationship, you can define them as cascading, so that a prompt selection is determined by the choice of the user in the previous prompt.
Demonstration 4: Create a cascading prompt
Demonstration 4: Create a cascading prompt

Purpose:
Executives need a report that lets them analyze product returns. They want a report that enables them to focus on specific product lines and product types within those product lines for all order methods. This report will be delivered to the shareholders during their monthly meeting, so the executives would like a cover page to add a more official look.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: List
Folder: Sales and Marketing (query)
Namespace: Sales (query), Returned items (query)

Task 1. Create a list report with title, and then create a prompt page with a cascading prompt.
1. Open a new List template, using the GO data warehouse (query) package.
2. From the Data/Source tab, add the following query items to the list report object:
   - Sales (query) > Products: Product line and Product type
   - Sales (query) > Order method: Order method type
   - Returned items (query) > Returned items fact: Return quantity

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product type</th>
<th>Order method type</th>
<th>Return quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product type&gt;</td>
<td>&lt;Order method type&gt;</td>
<td>&lt;Return quantity&gt;</td>
</tr>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product type&gt;</td>
<td>&lt;Order method type&gt;</td>
<td>&lt;Return quantity&gt;</td>
</tr>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product type&gt;</td>
<td>&lt;Order method type&gt;</td>
<td>&lt;Return quantity&gt;</td>
</tr>
</tbody>
</table>

Returned quantity works in this query because the Returned Items (query) query subject has the same query items as the Sales (query) query subject. Returned quantity is a conformed value.
3. In the Text field at the top of the report, change the title to: Product type by Product line for all Order methods.
4. With the title text still selected, change the font to Arial Black, 14pt.
5. Left justify the text field.
6. Ctrl-click <Product line> and <Product type>, and then on the list toolbar, click Group / Ungroup.
7. Click <Return quantity>, on the toolbar click Summarize, and then click Total.
8. In the work area, click <Product line>, and then Ctrl-click <Product type>, and <Order method type>.
9. On the Navigate tab, right-click Report  and then click Build Prompt Page .

Task 2. Set behavior patterns for prompts.

1. In the Prompt page work area, click the Product type value prompt.
2. From the Application bar, click Show properties.
3. In the Properties pane, under GENERAL, double-click Cascade source, from the list, select Product line, and then click OK.
4. In the Properties pane, under GENERAL, ensure that Multi-select is set to Yes, and that Auto-submit is set to No.
   The Product types available to choose from will depend on the Product line selected when the prompt submits.
5. In the work area, click the Product line value prompt.
6. In the Properties pane, under GENERAL, in the Multi-select list, select No.
7. Under GENERAL, in the Auto-submit list, select Yes.
   The user can only select one product line, and the selection will be submitted automatically.
8. In the work area, click the Order method type value prompt.
9. In the Properties pane, under GENERAL, in the Multi-select list, click No.
   The user can select multiple product types, but the selection will not submit automatically. Once all of the selections for the prompts are complete, the user must submit the request by clicking Finish.
Task 3. Create a static value to select all order method type values.

1. With the Order method type value prompt still selected, under DATA, double-click Static choices, and then click Add.
2. Type ALL for both the Use and Display values, click OK to close the Edit dialog box, and then click OK to close the Static choices dialog box.

The value entered for static choices is case sensitive and should be entered the exact same way in your filter expression.

You will specify what values to return when ALL is selected in the Order method prompt.

3. From the Page explorer pane, click Page1.
4. Click anywhere in the list, from the list toolbar, click Filters, and then click Edit Filters.
5. Click the Order method type filter, and then click Edit.
6. Replace and validate the existing expression with the following:
   
   if (?Order method type?='ALL') then ([Order method type]=[Order method type]) else([Sales (query)].[Order method].[Order method type] = ?Order method type?)

   Hint: Drag Order method type from the Data Items tab.

7. Choose any options for the prompts, and then click OK to close the validation box.
8. Click OK to close the Detail filter expression dialog box, and then click OK to close the Filters dialog box.

Task 4. Create a cover page.

2. From the Toolbox tab, drag a Page to the Report pages section - placing it above Page1.
3. In the Properties pane, under MISCELLANEOUS, in the Name box, modify it to a name of CoverPage, and then press Enter.
4. Double-click CoverPage to open it.
5. From the Toolbox tab, drag a Table onto the work area with 2 columns and 1 row, and then click OK.
6. Click the left table cell, and then Ctrl-click the right table cell.
7. On the table toolbar, click Center.
8. Click anywhere on the page below the table, and then on the toolbar, click Middle.
Task 5. Create title and image for cover page.

1. Drag a Text item into the left table cell.
2. In the Text dialog box, type GO Data Warehouse - Revenue Generated, and then click OK.
3. Click the text item, and then change the font to Arial Black, 16 pt.
4. From Toolbox, expand LAYOUT, and then drag an Image object into the right table cell.
5. In the toolbar, click More, and then click Edit Image URL, cancelling any message that appears.
6. In the Image URL box, ensure the path is set to http://vclassbase:88/images/ (modify the URL, if necessary), and then click the Browse button.
7. Locate and select cover2.jpg, and then click OK to close the Image Browser dialog box.
8. Click OK to close the Image URL dialog box.
9. In the Properties pane, under POSITIONING, double-click the Size & overflow property.
10. Set the Width to 150 pixels, the Height to 75 pixels, and then click OK.

Task 6. Run the report and view details for specific products.

1. Run the report in HTML.

A section of the results appear as follows:

```
Product line
- Camping Equipment
- Golf Equipment
- Mountaineering Equipment
- Outdoor Protection
- Personal Accessories
```

The Prompt page appears prompting for a Product line. The star icon indicates that this selection is mandatory.
2. In the Product line prompt, click **Camping Equipment**.
The results for the **Product type** prompt update and appear as follows:

```
Product type
* Cooking Gear
* Lanterns
* Packs
* Sleeping Bags
* Tents

Select all  Deselect all
```

3. Under **Product type**, click **Lanterns**, and then Ctrl-click **Tents**.
The results for the Order method type prompt appear as follows:

```
Order method type
* ALL
* E-mail
* Fax
* Mail
* Sales visit
* Special
* Telephone
* Web
```

4. In the **Order method type** prompt, click **ALL**, and then click **Finish**.
The report cover page appears.
5. Click **Bottom**.

The results appear as follows:

![Product type by Product line for all order methods](image)

6. Close the rendered report tab.
7. Leave the report authoring tab open for the following exercise.

**Results:**
You created a report that lets you analyze product returns. The report enabled users to focus on specific product lines and product types within those product lines. In particular, you focused on tent and lantern returns for all order methods. You gave the report a cover page for a more professional look.
# Unit summary

- Identify various prompt types
- Use parameters and prompts to focus data
- Search for prompt types
- Navigate between pages
Exercise 1: Focus a report using value prompts
Exercise 1: Focus a report using value prompts

Company executives have asked you to create a report that shows revenue data by product line where they can choose the region(s) and the year that they want the report to include. They would like the prompts to show up on a separate prompt page.

To accomplish this:

- Open a Combination chart (Clustered Bar and Clustered Line) template with the GO data warehouse (query) package.
- Add the following query items:
  - Default measure (y-axis):
    - Sales fact: Revenue
  - Categories (x-axis):
    - Products: Product line
  - Series (primary axis) - Bar Chart Type:
    - Employee by region: Branch Region
  - Series (primary axis) - Line Chart Type:
    - Time: Year
- Create a prompt page with two value prompts:
  - Branch region
  - Year
- Create a cover page.
- Add a table with 2 columns and one row
- Add a title and company logo to the cover page.

For more information about where to work and the exercise results, refer to the Tasks and Results section that follows. If you need more information to complete a task, refer to earlier demonstrations for detailed steps.
Exercise 1: Tasks and Results

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content/Samples/Models/GO data warehouse (query)
Report Template: Chart/Combination
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create a chart, then create a prompt page and add the Region prompt.

- **Toolbar**: Open a new Combination chart (Clustered Bar and Clustered Line) template using the GO data warehouse (query) package.
- **Data pane**: Navigate to Sales and Marketing (query)/Sales, and add:
  - Sales fact: Revenue to the Default measure (y-axis).
  - Products: Product line to the Categories (x-axis).
  - Employee by region: Branch region to the Series (primary axis) - Bar Chart Type.
  - Time: Year to the Series (primary axis) - Line Chart Type.

The results appear as follows:

- **Navigate pane**: Click Prompt pages.
- **Toolbox pane**: Drag a Page to the Prompt pages pane.
- **Prompt Pages work area**: Open Prompt page1.
• **PROMPTING**: Add a *Value prompt* object to the prompt page.
• **Prompt Wizard - Value prompt** dialog box: Create a multi-select parameter named *RegionPrompt*, based on *Branch region* (in Employee by region).
• **Toolbox** pane: Add a *Text Item* to the left of the value prompt.
Task 2. Add a Year prompt.

- **Toolbox tab**: Drag a Block below the RegionPrompt prompt.
  - Add a Value Prompt object into the block.
- **Prompt Wizard - Value Prompt** dialog box: Create a single-select parameter named YearPrompt, based on Year.
  - Click Next, click Next again, and then click Finish.
- **Toolbox tab**: Add a Block to the left of the YearPrompt prompt.
- Add a Text Item into the block, type Choose Year: and then click OK.
- **Toolbar**: Set text to 14 pt.
- **Properties pane**: In the YearPrompt properties, under GENERAL, set the Select UI to Radio button group.

The results appear as follows:
Task 3. Create a cover page (Optional).
As an additional challenge, create a cover page to give your report a finished look.

- **Navigate pane:** Click Report pages.
- **Toolbox pane:** Add a Page to the Report Pages work area, above Page 1.
- **Properties page:** Change the Name property to CoverPage.
- **Report Pages work area:** Open CoverPage.
- **Toolbox pane:** Add a 2 column by 1 row Table to the work area.
- **Toolbar:** Set the page to Middle.

Task 4. Add a text item and image to the cover page (optional).
This task will add a title and image to the cover page created in Task 3.

- **Toolbox pane:** Add a Text Item into the left table cell.
- **Text box:** Type GO Data Warehouse - Revenue Generated.
- **Toolbar:** Set text to Arial Black and 16 pt.
- **Toolbox pane:** Expand LAYOUT and then drag an Image into the right table cell
- **Toolbar:** Click More, click Edit Image URL
- **Image URL box:** Enter http://vclassbase:88/images/, and then browse to and select cover1.jpg

The results appear as follows:
Task 5. Run the report.

- **Toolbar**: Run the report in **HTML**.
- **RegionPrompt**: Select **Americas** and **Asia Pacific**.
- **YearPrompt**: Select **2011**.
- **Cover Page**: Go to the next page.

The results appear as follows:

You have created a report that shows revenue data by product line where users can choose the region(s) and the year that they want the report to include. You have put the prompts on a separate prompt page and created a cover page.

- Close the rendered report tab.
- Sign out of **IBM Cognos Analytics**, if prompted click **OK** to continue without saving.
- Close all browser windows.
Unit 7  Extend reports using calculations

Extend reports using calculations

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2016
Course materials may not be reproduced in whole or in part without the written permission of IBM.
Unit objectives

• Create calculations based on data in the data source
• Add run-time information to the reports
• Create expressions using functions
Derive additional information from the data source

Create a layout calculation to add information to your report.
Create calculated columns based on existing items in the model using query calculations.
Calculations can be added to a list, crosstab, or chart report, as well as to the body, headers, and footers.
Add run-time information to your report

Layout calculations can include run-time information such as current date, current time, and user name.

Including calculated columns can help provide further insight into your data. You can create a query or layout calculation by inserting a calculation into your report and then creating the expression in the Expression Editor.

Create a calculated column to make a report more meaningful by deriving additional information from the data source. For example, you create an invoice report, and you want to see the total sale amount for each product ordered. Create a calculated column that multiplies the product price by the quantity ordered.

If an expression is used in multiple reports, or used by different report authors, ask your modeler to create the expression as a standalone object in the model and include it in the relevant package.
To build the expression use the:

- Source tab to find all query items available from the package.
- Data Items tab to find the query items currently found in your report.
- Functions tab to find operators, summaries, constants, and constructs to create the expression you want to display your customized data.
- Parameters tab to find query items used for prompts and parameters within the report.
Add Date/Time functions to your report

Use date and time functions in calculations and filters to query on specific dates and times in your report.

Date/Time functions can be used to build dates, modify existing dates or to filter the report for specific dates.

A useful date/time function is the extract() function which returns an integer representing the value of datepart (year, month, day, hour, minute, second) in your datetime expression.

Date/Time functions can be found under the Business Date/Time Functions folder, Vendor Specific Functions folder or the Common Functions folder. If you are going to use vendor specific functions, ensure that they are specific to the database that is currently being queried.
Not all data sources support functions the same way. The data modeler can set a quality of service indicator (icon appearing beside some functions) on functions to give a visual clue about the behavior of the functions. Report authors can use the quality of service indicators to determine which functions to use in a report. The quality of service indicators are:

- **(X) not available** - This function is not available for any data sources in the package.
- **(!!) limited availability** - The function is not available for some data sources in the package.
- **(!) poor performance** - The function is available for all data sources in the package but may have poor performance in some data sources.
- **(no symbol) unconstrained** - The function is available for all data sources.
Add string functions to your report

Use string functions in calculations and filters in your report to manipulate text data.

Some examples of string functions include:

- `substring()` function to return part of a string
- `trim()` function removes specific characters from the beginning or end of a specific text data item
- `upper()` function changes the text returned to be in uppercase
- `lower()` function changes the text returned to be in lowercase

String functions can be found under the Common Functions folder, or Vendor Specific Functions folder. For the above slide, if a user entered ‘Golf Equipment ‘(with a space at the end) into the database, the report author would get unexpected results if they queried the database and was doing a comparison against ‘Golf Equipment’ (without a space at the end). This is a real world example where the `trim()` function should be used to remove trailing spaces before doing a comparison.
Demonstration 1: Add calculations to a report

### 2010 First Quarter Sales Figures and Overall Calculated Percent of Goal
Report run date: Mar 4, 2015

<table>
<thead>
<tr>
<th>Date</th>
<th>Product line</th>
<th>Revenue</th>
<th>Planned revenue</th>
<th>Percent of Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 12, 2010</td>
<td>Camping Equipment</td>
<td>20,217,372.96</td>
<td>21,714,739.59</td>
<td>93%</td>
</tr>
<tr>
<td>Jan 12, 2010</td>
<td>Golf Equipment</td>
<td>8,141,599.89</td>
<td>9,615,364.17</td>
<td>85%</td>
</tr>
<tr>
<td>Jan 12, 2010</td>
<td>Outdoor Protection</td>
<td>2,263,380.47</td>
<td>2,393,932.12</td>
<td>95%</td>
</tr>
<tr>
<td>Jan 12, 2010</td>
<td>Personal Accessories</td>
<td>7,414,443.91</td>
<td>7,797,959.04</td>
<td>95%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Camping Equipment</td>
<td>5,000,710.65</td>
<td>5,350,515.31</td>
<td>93%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Golf Equipment</td>
<td>2,536,524.65</td>
<td>2,723,637.61</td>
<td>93%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Outdoor Protection</td>
<td>474,025.75</td>
<td>496,360.65</td>
<td>95%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Personal Accessories</td>
<td>3,477,197.59</td>
<td>3,588,395.95</td>
<td>97%</td>
</tr>
</tbody>
</table>
Demonstration 1: Add calculations to a report

Purpose:
You have been asked to create a report that will return revenue and planned revenue for product lines for January 2010. In addition to looking at actual revenue versus planned revenue, users want to see a percentage for how much of the planned revenue was met. The report should also display the date that the report is run.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content \Samples\Models\GO data warehouse (query)
Report Type: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the list and include a calculated column for percent of goal.
1. Open a new List template using the GO data warehouse (query) package.
2. From the Source tab, add the following query items to the new list report object:
   - Time: Date
   - Products: Product line
   - Sales fact: Revenue, Planned revenue

<table>
<thead>
<tr>
<th>Date</th>
<th>Product line</th>
<th>Revenue</th>
<th>Planned revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Date&gt;</td>
<td>&lt;Product line&gt;</td>
<td>&lt;Revenue&gt;</td>
<td>&lt;Planned revenue&gt;</td>
</tr>
<tr>
<td>&lt;Date&gt;</td>
<td>&lt;Product line&gt;</td>
<td>&lt;Revenue&gt;</td>
<td>&lt;Planned revenue&gt;</td>
</tr>
<tr>
<td>&lt;Date&gt;</td>
<td>&lt;Product line&gt;</td>
<td>&lt;Revenue&gt;</td>
<td>&lt;Planned revenue&gt;</td>
</tr>
</tbody>
</table>
3. From the Toolbox tab, expand TEXTUAL, and then drag a Query calculation to make it the last column.
   The Data item expression dialog box appears.

4. In the Name field, type Percent of Goal, and then create and validate the following expression:
   
   \[\frac{\text{Revenue}}{\text{Planned revenue}}\]

   Hint: Drag Revenue and Planned Revenue form the Data Items tab.

5. Click OK.
   This column will show the percentage of revenue achieved for each product line on each day.

**Task 2. Add a detail filter to filter dates.**

1. On the list toolbar, click Filters, Edit Filters, and then click Add.
2. Click Advanced, and then OK.
3. Create and validate the following expression:
   
   \[\text{[Date] between } \text{_first_of_month(2010-01-01)} \text{ and } \text{_last_of_month (2010-03-31)}\]

   Hint: Drag Date from the Data Items tab.
   Data items selected from the Source tab will be calculated and summarized after aggregation, whereas data items selected from the Data Items tab will be calculated and summarized after aggregation.
   This filter will return dates between January 1 2010 and March 31 2010 (First quarter of 2010). The _first_of_month() function returns the first day of the month in the date expression, while the _last_of_month() function returns the last day of the month in the date expression. You can use any date for the expression as long as it’s in the proper format and it contains data. Make sure that there is a space between each function.

4. Click OK to close the expression dialog box, and then click OK to close the Filters dialog box.

**Task 3. Format the data in the list.**

1. In the work area, click the <Date> list column body.
2. On the list toolbar, click Sort, and then click Ascending.
3. Click the <Percent of Goal> list column body.
4. In the Properties pane, under DATA, double-click Data format.
5. In the Format type list, select Percent, for Percentage symbol select %, and then click OK.
6. With **Percent of Goal** still selected, on the list toolbar click **Summarize**, and then click **Calculated**.

Calculated is applied if the data item expression:

- contains a summary function
- is an if then else or case expression that contains a reference to at least a modeled measure in its condition
- contains a reference to a model calculation or to a measure that has the Regular Aggregate property set to a value other than Unsupported
- contains a reference to at least one data item that has the Rollup Aggregate Function property set to a value other than None

**Task 4. Add run-time information to the report.**

You want to display the run date of the report under the title of the report.

1. Delete the **Double-click to edit text**.
2. From the Toolbox tab, drag a Table, with 1 column and 2 rows, into the block.
3. From the Toolbox tab, drag a Text item into the top cell.
4. In the Text field, type **2010-First Quarter Sales Figures and Overall Calculated Percent of Goal**, and then click OK.
5. From the list toolbar, set the font size to **14pt**.
6. From the Toolbox tab, drag a Text item into the bottom table cell, type **Report run date:**, press the spacebar, and then click OK.
7. From the Toolbox tab, drag a Layout calculation to the right of the text item.
8. Create and validate the following expression:
   
   **AsOfDate()**

   Hint: Drag AsOfDate from the functions Tab/Report Functions folder.
9. Click **OK**.
10. Run the report in HTML.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Date</th>
<th>Product line</th>
<th>Revenue</th>
<th>Planned revenue</th>
<th>Percent of Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 12, 2010</td>
<td>Camping Equipment</td>
<td>20,217,372.98</td>
<td>21,714,739.59</td>
<td>93%</td>
</tr>
<tr>
<td>Jan 12, 2010</td>
<td>Golf Equipment</td>
<td>9,141,599.89</td>
<td>9,815,894.17</td>
<td>93%</td>
</tr>
<tr>
<td>Jan 12, 2010</td>
<td>Outdoor Protection</td>
<td>2,263,380.47</td>
<td>2,393,032.12</td>
<td>95%</td>
</tr>
<tr>
<td>Jan 12, 2010</td>
<td>Personal Accessories</td>
<td>7,414,443.06</td>
<td>7,797,859.04</td>
<td>95%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Camping Equipment</td>
<td>5,000,710.6</td>
<td>5,350,515.31</td>
<td>95%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Golf Equipment</td>
<td>2,536,524.65</td>
<td>2,723,837.61</td>
<td>95%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Outdoor Protection</td>
<td>474,025.75</td>
<td>496,960.85</td>
<td>95%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Personal Accessories</td>
<td>3,477,197.59</td>
<td>3,586,395.95</td>
<td>97%</td>
</tr>
</tbody>
</table>

11. Click **Bottom** to see that the report includes all of the months of the first quarter, and the overall calculated percent of goal.

12. Close the rendered report tab.

**Task 5. Overwrite query expression.**

1. Double-click the `<Percent of Goal>` list column body to open the expression dialog box.

2. Overwrite the current expression using query items from the **Source** tab as follows: `[Sales (query)]].[Sales fact].[Revenue]/[Sales (query)].[Sales fact].[Planned revenue]`

3. Validate, and then click **OK**.

4. Run the report in HTML.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Date</th>
<th>Product line</th>
<th>Revenue</th>
<th>Planned revenue</th>
<th>Percent of Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 12, 2010</td>
<td>Camping Equipment</td>
<td>20,217,372.98</td>
<td>21,714,739.59</td>
<td>66.858%</td>
</tr>
<tr>
<td>Jan 12, 2010</td>
<td>Golf Equipment</td>
<td>9,141,599.89</td>
<td>9,815,894.17</td>
<td>24.356%</td>
</tr>
<tr>
<td>Jan 12, 2010</td>
<td>Outdoor Protection</td>
<td>2,263,380.47</td>
<td>2,393,032.12</td>
<td>25.745%</td>
</tr>
<tr>
<td>Jan 12, 2010</td>
<td>Personal Accessories</td>
<td>7,414,443.06</td>
<td>7,797,859.04</td>
<td>80.525%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Camping Equipment</td>
<td>5,000,710.6</td>
<td>5,350,515.31</td>
<td>20.130%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Golf Equipment</td>
<td>2,536,524.65</td>
<td>2,723,837.61</td>
<td>7.577%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Outdoor Protection</td>
<td>474,025.75</td>
<td>496,960.85</td>
<td>6.954%</td>
</tr>
</tbody>
</table>

The Percent of Goal calculation does not match because the timing of the aggregation is different.

5. Close the rendered report tab.
6. Click `<Percent of Goal>` list column body.
7. In the **Properties** pane, under the **DATA ITEM** section, change the **Detail aggregation** function to **Calculated**.
8. Run the report in **HTML**.

Now the expression returns the same results as the previous expression.

```
<table>
<thead>
<tr>
<th>Date</th>
<th>Product line</th>
<th>Revenue</th>
<th>Planned revenue</th>
<th>Percent of Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 12, 2010</td>
<td>Camping Equipment</td>
<td>20,217,372.90</td>
<td>21,714,739.59</td>
<td>93%</td>
</tr>
<tr>
<td>Jan 12, 2010</td>
<td>Golf Equipment</td>
<td>9,141,599.89</td>
<td>9,815,894.17</td>
<td>93%</td>
</tr>
<tr>
<td>Jan 12, 2010</td>
<td>Outdoor Protection</td>
<td>2,263,380.47</td>
<td>2,393,032.12</td>
<td>95%</td>
</tr>
<tr>
<td>Jan 12, 2010</td>
<td>Personal Accessories</td>
<td>7,414,443.06</td>
<td>7,797,859.04</td>
<td>95%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Camping Equipment</td>
<td>5,000,710.6</td>
<td>5,350,515.31</td>
<td>93%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Golf Equipment</td>
<td>2,536,524.85</td>
<td>2,723,837.61</td>
<td>93%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Outdoor Protection</td>
<td>474,025.75</td>
<td>496,960.85</td>
<td>95%</td>
</tr>
<tr>
<td>Jan 13, 2010</td>
<td>Personal Accessories</td>
<td>3,477,197.59</td>
<td>3,586,395.95</td>
<td>97%</td>
</tr>
</tbody>
</table>
```

9. Close the rendered report tab.
10. Leave the report authoring tab open for the next demonstration.

**Results:**
You created a report to show revenue and planned revenue and the percentage of planned revenue that was achieved for product lines for the first quarter of 2010. You also included the date when the report was run.
Display prompt selections in report titles

- You can display information in the report title that describes the prompt option a user selects.

```
Sales region

Report title: if the user selects to view only data for the Asia Pacific sales region
Quantity Sold in Asia Pacific

Report title: if the user does not select a prompt option and therefore views data for all sales regions
Quantity Sold in All Sales Regions
```

To display the selected prompt option in the report title, add a layout calculation to the report title that returns a different value depending on the prompt option a user selects.
Demonstration 2

Display prompt selections in the report title

<table>
<thead>
<tr>
<th>Quantity Sold in Asia Pacific</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantity</td>
</tr>
<tr>
<td>PERSONAL ACCESSORIES</td>
</tr>
<tr>
<td>Binoculars</td>
</tr>
<tr>
<td>Eyewear</td>
</tr>
<tr>
<td>Knives</td>
</tr>
<tr>
<td>Navigation</td>
</tr>
<tr>
<td>Watches</td>
</tr>
<tr>
<td>PERSONAL ACCESSORIES</td>
</tr>
<tr>
<td>MOUNTAINEERING EQUIPMENT</td>
</tr>
<tr>
<td>Climbing Accessories</td>
</tr>
<tr>
<td>Rope</td>
</tr>
<tr>
<td>Safety</td>
</tr>
<tr>
<td>Tools</td>
</tr>
</tbody>
</table>

Demonstration 2: Display prompt selections in the report title
Demonstration 2: Display prompt selections in the report title

Purpose:
You have been asked for a report that displays the quantity of products sold for each order year. You also need to display all product lines in uppercase. The report should contain an optional prompt that lets users view data by sales region. Add a report title that indicates which sales region users select in the prompt. It should also indicate if they do not select a region as well. You will use a layout calculation to display the report title.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: Crosstab
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the crosstab and edit the expression to return product line values in uppercase.

1. Open a new Crosstab template using the GO data warehouse (query).
2. From the Source tab, add the following query items to the new crosstab report object:
   - Rows area:
     - Products: Product line, Product type (nested to the right of Product line)
   - Columns area:
     - Time: Year
   - Measures area:
     - Sales fact: Quantity
3. Click the <#Product type#> crosstab node member, from the toolbar click Summarize, and then click Total.
4. On the crosstab, click Total.
5. In the Properties pane, under TEXT SOURCE, change Source type to Data item value.
6. In the Properties pane, under TEXT SOURCE, change Data item value to Product line.
7. In the crosstab, double-click <#Product line#>.
8. Update and validate the existing expression as follows:
   upper([Sales (query)].[Products].[Product line])
9. Click OK.

**Task 2. Add an optional parameter.**

You will now add an optional filter containing a parameter that lets users specify the sales region for which they want to view data.

1. With the entire crosstab selected, on the crosstab toolbar, click Filters, and then Edit Filters.
2. Ensure the Detail Filters tab is selected, click Add, click Advanced, and then click OK.
3. From the Available Components pane, expand Sales and Marketing (query), Sales (query), and then expand Employee by region.
4. Create and validate the following expression (validate using Americas):
   [Sales (query)].[Employee by region].[Branch region]=?Region?
5. Click OK to close the validation dialog box, and then OK to close the Detail filter expression dialog box.
6. In the Filters dialog box, with the filter you just added selected, click Optional, and then click OK.
   You will run this report to test the prompt.
7. Run the report in HTML.
8. At the prompt, select **Asia Pacific**, and then click **OK**. A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Quantity</th>
<th>2010</th>
<th>2011</th>
<th>2012</th>
<th>2013</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERSONAL ACCESSORIES</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Binoculars</td>
<td>43,340</td>
<td>45,626</td>
<td>62,144</td>
<td>49,788</td>
</tr>
<tr>
<td>Eyewear</td>
<td>22,252</td>
<td>50,760</td>
<td>79,760</td>
<td>69,607</td>
</tr>
<tr>
<td>Knives</td>
<td>396,185</td>
<td>275,620</td>
<td>388,853</td>
<td>307,093</td>
</tr>
<tr>
<td>Navigation</td>
<td>117,074</td>
<td>84,358</td>
<td>107,223</td>
<td>113,107</td>
</tr>
<tr>
<td>Watches</td>
<td>33,936</td>
<td>46,015</td>
<td>60,211</td>
<td>44,995</td>
</tr>
<tr>
<td><strong>PERSONAL ACCESSORIES</strong></td>
<td>612,787</td>
<td>502,379</td>
<td>697,991</td>
<td>584,590</td>
</tr>
<tr>
<td>MOUNTAINEERING EQUIPMENT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Climbing Accessories</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rope</td>
<td>410,155</td>
<td>526,482</td>
<td>573,585</td>
<td></td>
</tr>
<tr>
<td>Safety</td>
<td>30,530</td>
<td>45,981</td>
<td>38,024</td>
<td></td>
</tr>
<tr>
<td>Tools</td>
<td>89,114</td>
<td>104,518</td>
<td>87,855</td>
<td></td>
</tr>
<tr>
<td><strong>MOUNTAINEERING EQUIPMENT</strong></td>
<td>713,054</td>
<td>922,000</td>
<td>936,245</td>
<td></td>
</tr>
</tbody>
</table>

The report displays data only for the Asia Pacific region. Notice that the Total line caption now reflects the product line that it summarizes and that all Product line titles are uppercase.

9. Close the rendered report tab.

**Task 3. Display the parameter value in the report title.**

To give this report some context, you want the region selected to appear in the report title. If no region is selected, you want the report title to indicate that the data displayed represents quantity sold in all regions.

1. On the report page, double-click the report title text, type **Quantity Sold in**, press the spacebar, and then click **OK**.
2. Left justify the report title within the block. You will create a layout calculation to display the prompt option selected in the report title.
3. From the **Toolbox**, expand **TEXTUAL**, and then drag a **Layout calculation** object to the end of the report title.

You will create an expression that specifies that if a parameter value is selected, the layout calculation should show the display value for the selected parameter value. Otherwise, the layout calculation should show **All Regions**.
4. Create and validate the following expression:
   \[ \text{if(ParamDisplayValue('Region')<> ' ') then (ParamDisplayValue('Region')) else 'All Regions'} \]
   
   Hint: ParamDisplayValue('Region') is found on the parameter tab. The empty quotes represent no display value. This will be the case when the prompt is optional and the user does not select anything.

5. Click OK to close the dialog box.
   You will format the layout calculation text to look like the report title text.

6. Select the Quantity Sold in text, and make sure it is not underlined.

**Task 4. Test the prompt.**

1. Run the report in HTML.
2. On the prompt page, ensure that Branch region is selected, and then click OK to run the report.
   A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Quantity Sold in All Regions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantity</td>
</tr>
<tr>
<td>-------------------------------</td>
</tr>
<tr>
<td>Personal Accessories</td>
</tr>
<tr>
<td>Binoculars</td>
</tr>
<tr>
<td>Eyewear</td>
</tr>
<tr>
<td>Knives</td>
</tr>
<tr>
<td>Navigation</td>
</tr>
<tr>
<td>Watches</td>
</tr>
<tr>
<td>Mountaineering Equipment</td>
</tr>
<tr>
<td>Climbing Accessories</td>
</tr>
<tr>
<td>Rope</td>
</tr>
<tr>
<td>Safety</td>
</tr>
<tr>
<td>Tools</td>
</tr>
</tbody>
</table>

   The report title explains that this report contains data about quantity sold in all regions.

3. Close the rendered report tab, and then run the report in HTML.
4. On the prompt page, select **Asia Pacific**, and then click **OK**. A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Quantity Sold in Asia Pacific</th>
<th>2010</th>
<th>2011</th>
<th>2012</th>
<th>2013</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PERSONAL ACCESSORIES</strong></td>
<td>Binoculars</td>
<td>43,340</td>
<td>45,626</td>
<td>62,144</td>
</tr>
<tr>
<td></td>
<td>Eyewear</td>
<td>22,252</td>
<td>60,760</td>
<td>79,760</td>
</tr>
<tr>
<td></td>
<td>Knives</td>
<td>396,185</td>
<td>275,620</td>
<td>388,653</td>
</tr>
<tr>
<td></td>
<td>Navigation</td>
<td>117,074</td>
<td>84,358</td>
<td>107,223</td>
</tr>
<tr>
<td></td>
<td>Watches</td>
<td>33,936</td>
<td>46,015</td>
<td>60,211</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td>612,787</td>
<td>502,379</td>
<td>697,991</td>
</tr>
<tr>
<td><strong>MOUNTAINEERING EQUIPMENT</strong></td>
<td>Climbing Accessories</td>
<td>410,155</td>
<td>526,482</td>
<td>573,585</td>
</tr>
<tr>
<td></td>
<td>Rope</td>
<td>30,630</td>
<td>45,981</td>
<td>38,024</td>
</tr>
<tr>
<td></td>
<td>Safety</td>
<td>85,114</td>
<td>104,518</td>
<td>67,655</td>
</tr>
<tr>
<td></td>
<td>Tools</td>
<td>187,255</td>
<td>245,019</td>
<td>236,781</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td>713,054</td>
<td>922,000</td>
<td>936,245</td>
</tr>
</tbody>
</table>

The report title explains that this report contains data about quantity sold in Asia Pacific.

5. Close the rendered report tab.

6. Leave the report authoring tab open for the following exercise.

**Results:**
You created a report that displays the quantity sold for products by order year. You also displayed all product lines in uppercase. Users have the option to select a region for which to view data. To add context to the report, the user's prompt selection appears in the report title, by using a layout calculation.
Unit summary

- Create calculations based on data in the data source
- Add run-time information to the reports
- Create expressions using functions
Exercise 1

Sales percent by sales representative and country

Exercise 1: Sales percent by sales representative and country
Exercise 1:
Sales percent by sales representative and country

Sales management would like to improve overall product line sales. To do this they need to start with a report that shows which product lines each salesperson (sales representative, or sales rep) tends to sell the most of. Sales management would like to be able to also filter by specified year and country or countries.

To accomplish this:
- Open a new list template using the GO data warehouse (query) package.
- Add the following query items to the list report object:
  - Employee by region: Country
  - Employee by region: Employee name
  - Time: Year
  - Products: Product Line
  - Sales fact: Revenue
- Add a calculated column called EmpRevPercent.
- Group Country, Year and Employee Name.
- Create a header using Country.
- Summarize the Revenue by Total and format the data as $(USD).
- Add an EmpRevPercent summary row and format as a percent.
- Format summary row to display the data item value.
- Create a Year parameter.
- Add a prompt to allow users to focus on one or more countries.
- Add a report title.
- Run the report and then focus on information for Canada and the United States, and for the year 2012.

For more information about where to work and the exercise results, refer to the Tasks and Results section that follows. If you need more information to complete a task, refer to earlier demonstrations for detailed steps.
Exercise 1: Tasks and Results

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Template: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the list and add a calculated column.

- **Side panel**: Open a new List template using the GO data warehouse (query) package.
- **Toolbar**: Justify the report header text to the Left.
- **Source tab**: Add the following query items to the list data container:
  - Employee by Region: Country.
  - Time: Year.
  - Employee by Region: Employee name.
  - Products: Product line.
  - Sales fact: Revenue.
- **Toolbox tab**: Create and validate the EmpRevPercent query calculation, and add it to the end of the list:
  
  \[
  \text{[Revenue]} / \text{Total([Revenue] for [Country])}.
  \]

  Note: You can retrieve the Total function from the Function tab, Summaries folder. You can also drag Revenue and Country from the Data Items tab.

  The results appear as follows:

<table>
<thead>
<tr>
<th>Country</th>
<th>Year</th>
<th>Employee name</th>
<th>Product line</th>
<th>Revenue</th>
<th>EmpRevPercent</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Country&gt;</td>
<td>&lt;Year&gt;</td>
<td>&lt;Employee name&gt;</td>
<td>&lt;Product line&gt;</td>
<td>&lt;Revenue&gt;</td>
<td>&lt;EmpRevPercent&gt;</td>
</tr>
<tr>
<td>&lt;Country&gt;</td>
<td>&lt;Year&gt;</td>
<td>&lt;Employee name&gt;</td>
<td>&lt;Product line&gt;</td>
<td>&lt;Revenue&gt;</td>
<td>&lt;EmpRevPercent&gt;</td>
</tr>
<tr>
<td>&lt;Country&gt;</td>
<td>&lt;Year&gt;</td>
<td>&lt;Employee name&gt;</td>
<td>&lt;Product line&gt;</td>
<td>&lt;Revenue&gt;</td>
<td>&lt;EmpRevPercent&gt;</td>
</tr>
</tbody>
</table>
Task 2. Group and summarize the report.

- Toolbar: Group <Country>, <Year> and <Employee name>.
  - Create a header using the <Country> column, and delete the redundant list column body.
  - Summarize <Revenue> by Total.

Task 3. Format the data.

- Toolbar: Format all of the <Total(Revenue)> summary cells for $(USD) currency.
  - Summarize the <EmpRevPercent> by Calculated.
  - Format all of the <EmpRevPercent> cells and <Calculated(EmpRevPercent)> summary cells for Percent (with Percentage Symbol %).

Task 4. Add a parameter and a prompt.

- Toolbar: Create and validate the following advanced filter expression: [Year] =?Year?. (Validate using 2012)
- Toolbox tab: To the left of the list data container, create a multi-select Value prompt, named Countries, based on the Country query item.
  - Add a Prompt button to the right of the value prompt.
- Properties pane: Change the Prompt button type to Finish.

A section of the results appear as follows:
Task 5. Add a report title and run the report.

- **Text pane:** (type the following report title) *Sales Percent by Sales Rep and Country.*
- **Toolbar:** Run the report in **HTML**.
- **Year Parameter:** Type **2012**.
- **Value Prompt:** Select **Canada** and **United States**.

The results appear as follows:

![Sales Percent by Sales Rep and Country](image)

You have created a report that shows which product lines each sales person tends to sell the most of. The report is focused on a specified year and country or countries.

- **Close the rendered report tab.**
- **Sign out of IBM Cognos Analytics.**
- **Close all browser windows.**
Additional information: Some common functions

- **cast (expression, datatype_specification)** Converts "expression" to a specified data type. Some data types allow for a length and precision to be specified. Make sure that the target is of the appropriate type and size.
- **char_length (string_expression)** Returns the number of logical characters in "string_expression". The number of logical characters can be distinct from the number of bytes in some East Asian locales.
- **current_date** Returns a date value representing the current date of the computer that the database software runs on.
- **current_time** Returns a time with time zone value, representing the current time of the computer that runs the database software if the database supports this function. Otherwise, it represents the current time of the computer that runs IBM® Cognos® Analytics software.
- **floor (numeric_expression)** Returns the largest integer that is less than or equal to "numeric_expression".
- **localtime** Returns a time value, representing the current time of the computer that runs the database software.
- **mod (integer_expression1, integer_expression2)** Returns the remainder (modulus) of "integer_expression1.
- **position (string_expression1, string_expression2)** Returns the integer value representing the starting position.
- **substring (string_expression, integer_expression1 [, integer_expression2])** Returns the substring of "string_expression" that starts at position
- **trim ( [ [ trailing|leading|both ] [ match_character_expression ] , ] string_expression)** Returns "string_expression" trimmed of leading and trailing blanks.
- **upper (string_expression)** Returns "string_expression" with all lowercase characters converted to uppercase.
- **percentage** Returns the percent of the total value for selected data items.
- **total** Returns the total value of selected data items.
- **between** Determines if a value falls in a given range.
Unit 8  Use additional report building techniques

Use additional report building techniques

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2016
Course materials may not be reproduced in whole or in part without the written permission of IBM.
Unit objectives

• Enhance report design with report objects
• Reuse objects within the same report
• Share layout components among separate reports
• Discuss report templates
• Choose options to handle reports with no available data
Enhance report design (1 of 2)

- When creating a report, keep in mind that reports have:
  - horizontal bands
  - vertical bands
  - data frame objects (lists, crosstabs, charts, etc.)

When information runs in horizontal bands, use a block to hold the objects.

Reports may also have headers and footers. Determine what objects to use when building a report based on the kind of information you want to display, and how you want it to appear. If information applies to the entire report and you want it to appear on every page, place it in the header or footer.

When information runs vertically, such as text beside an image, use a table to organize the objects.
Enhance report design (2 of 2)

- To ensure professional results remember to:
  - use padding, margins, and blocks to create white space
  - set properties on the highest level item
  - avoid fixed size objects

Properties applied to an object will also be applied to any child items; therefore, it is best to set styling properties at the highest level.

Avoid fixed size objects because they are rigid and may not work with your overall design.

An empty block does not add space between objects. The block must contain an object, or you must specify the padding of the block to use the block for spacing.

Property Inheritance is the passing of parent attributes to child items. Use the Select Ancestor button on the Properties pane title bar to help determine the level at which to apply settings.

Applying properties at the highest level saves time and effort. For example, if you set the font type for a list object then all items in the list or added to the list will inherit the same font.

If objects have borders, use margins to make the objects look spaced apart.
Add objects

- Add, format, and organize objects to enhance the appearance of reports.

You can format items and objects to change their size, shape, location, and behavior according to your needs.

You can use text items to communicate relevant information about the report to its users.

You can add a background image to a data frame object like a list or crosstab, a cell in a table, or to the entire page. It is important to be aware that a background image can obscure the data in the report to some degree.
Organize objects using tables

- Add a table to a page to hold and organize objects such as titles, list, images, and charts.

You can use tables to assist with the spatial layout of report types and layout objects. You need tables to control where objects are placed. Unlike some graphics software, you cannot place objects anywhere on the work area.
Break a report into sections

Create sections in a report to show grouped information in separate report objects. This makes information easier to locate, and lets you view data for one group of items at a time.

Creating sections is similar to grouping on a query item. The difference is that section headers and footers appear outside the list, crosstab, or chart.

Create separate lists, crosstabs, or charts for specific query items by creating a section header.

When you run the report, separate sections appear for each value.

To remove section headers or footers, click the header or footer, and then from the Structure menu, click List Headers & Footers, clear the appropriate checkboxes, then the item will disappear from the report.
Convert a list to a crosstab

• Condense a report and view data from a different perspective by converting a list to a crosstab.

When you convert a list to a crosstab, the list columns you select become columns and nested columns in the crosstab, and the unselected columns become rows and nested rows.

If you have one measure, it becomes the cells of the crosstab. If you have more than one measure, then the measures will appear as columns.
Reuse objects within the same report

- You can change the contents of a reused object by overriding the child components and replacing them with other objects.

If you reuse an object that contains other objects, you can replace the child objects with a different object to customize your report.

To change the contents of a reused object, you must override the child object using the Properties pane.
Demonstration 1

Reuse objects within the same report

Product Line Sales by Year

Please contact Sales Manager for more details
Demonstration 1: Reuse objects within the same report

Purpose:
You have been asked to add some descriptive information to a sectioned report. The report must include a title on each page describing the contents of the report, and information about whom to contact if users have any questions.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: Blank
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Add a list to a blank page.

1. Open a new Blank report template using the GO data warehouse (query) package.

2. On the work area, click Add, and then click List.

3. Click OK to accept the Object and query name defaults.

4. From the Data/Source tab, add the following query items to the new list report object:
   - Employee by region: Country
   - Products: Product line
   - Time: Year
   - Sales fact: Revenue
5. Click the `<Revenue>` list column body, click **Summarize**, and then click **Default summary**.

![Revenue table](image)

**Task 2. Section data and convert to a crosstab.**

1. Click the `<Country>` list column body, and then on the list toolbar, click **Section / unsection**.

There is now a separate list displayed for each country.

2. Run the report in **HTML**.

The report is sectioned at the Country level; however, it is difficult to interpret.

![Country data](image)

3. Close the rendered report tab.

4. Click the `<Year>` list column body, and then on the list toolbar, click **Pivot List to Crosstab**.
5. Run the report in HTML.
   A section of the results appear as follows:

   Because you selected Year before you converted the list into a crosstab, it now
   appears as columns. Product line appears on rows, and Revenue, because it
   can be aggregated, appears as measures on the report. You can now interpret
   the data more quickly.

6. Close the rendered report tab.

**Task 3. Add a header and footer, and add objects to the header.**

1. Click to the right of the crosstab data container.
2. From the list toolbar, click **Headers & footers**, click **Page header & footer**,
   select the **Header and Footer** checkboxes, and then click **OK**.
3. From the **Toolbox**, drag a **Block** object to the **Page header**.
4. From the Toolbox, drag a **Text Item** onto the **Block** object in the **Page header**.
5. In the **Text field**, type **Product Line Sales by Year**, and then click **OK**.
Task 4. **Apply style to the header block and text.**

You will format the objects that you added to the header.

1. Click the **Page header** block, on the toolbar, click the arrow next to **Background Color**, click the **Basic colors** tab, and then click **Teal**.
2. In the **Page header**, click the text item to select it, and then on the toolbar, change the font to **Arial Black, 16 pt**, with a **Foreground Color** of **White**. The result appears as follows:

The report contains a header with the title that you specified. It has been formatted according to the properties you have set.

You now want to reuse the objects that you created and formatted to avoid repeating steps in building the footer.

**Task 5. Specify unique object names.**

1. Click the header block, in the **Properties** pane, under **MISCELLANEOUS**, in the **Name** property, type **Title Block**, and then press **Enter**.
2. Click the header text, in the **Properties** pane, under **MISCELLANEOUS**, in the **Name** property, type **Title Text**, and then press **Enter**.

If you try to assign a name that is not unique, Reporting displays a warning message informing you that the name must be unique.

If you select an element of the report, such as a column in a list, and want to deselect it, press **Esc** on your keyboard.
Task 6. Reuse the header block and change the text in the footer.

1. From the Toolbox, expand the ADVANCED section, drag a Layout component reference object into the footer.
   To reuse an object in the footer, you need to specify the object to be referenced. You can choose from the two objects to which you have previously assigned names, as well as the list containing the crosstab. In this case, you will select the block object because it also contains the text item object.

2. Under Available components to reference, click Title Block, and then click OK.
   The footer now contains the same object and formatting as the header.

3. Click the text in the footer.
   In the Properties pane you can only select the layout component reference object and not the block or text item objects individually. This is because it is referencing the block object in the header. Remember, the block object in the header also contains the text item object.

   You want to change the text in the footer to contain contact information.

4. In the Properties pane, click Overrides, and then click the ellipsis.

5. In the Overrides dialog box, select the Title Text checkbox, and then click OK.
   The layout component reference object in the footer no longer contains text. Only the referenced block object remains.

6. Drag a Text item object into the component override area of the footer block, type Please contact Sales Manager for more details, and then click OK.

7. Click the text item object in the footer, and then change the font to 12 pt, Bold, and Foreground color of White.
8. Run the report in **HTML**, and then click **Bottom** to view the footer. A section of the results appear as follows:

![Report Table](image)

**Please contact Sales Manager for more details**

This is a simplified example of reusing report objects. This technique might be best for reusing an object with numerous format properties applied.

You can also reuse objects between different reports and will be presented later in this unit.

9. Close the rendered report tab.
10. Leave the report authoring tab open for the next demonstration.

**Results:**
You enhanced the Product Line Sales by Year report by adding a header and footer. To build the footer and to minimize your work, you reused objects from the header.
Share layout components among separate reports

In Reporting, you can reuse layout components in different reports. You can choose to update shared layout objects manually or automatically. Be sure to name each layout component you want to reuse in other reports. Create a report containing all the objects you want to reuse in different reports, and then use it as an object library.

Instead of creating new layout components (such as page headers) for each report, you can create an object in one report and then reuse it in different reports. Reusing layout components saves you time and lets you apply standard company formatting to multiple reports.

By default, reused objects are automatically updated each time the report is run. This means that when you open or run a report containing a reused object, if the object has been changed in the source report, this change will automatically be applied in your report.

If you want a reused object to be updated manually instead of automatically, in the report where the object is reused, select the object, and then in the Properties pane, change the Embed property from Reference to Copy.

When you reuse a layout object in a different report, you can override child objects within this object (such as a text item in a page header object) if the child objects have been named in the source report.
Shared objects are stored in the layout component cache. The cache contains the definitions of the shared objects. When you open a report that contains layout component reference objects, the report(s) containing the shared layout objects is opened and the definitions are copied into the Reporting cache. Object names cannot contain white space and must begin with a letter. When you override child objects, you can replace the child object with any other object, not just an object of the same type. For example, if the child object is a text item, you can replace it with an image.
Demonstration 2

Reuse layout components in a different report

<table>
<thead>
<tr>
<th>Order method type</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Order method type&gt;</td>
<td>&lt;Quantity&gt;</td>
</tr>
<tr>
<td>&lt;Order method type&gt;</td>
<td>&lt;Quantity&gt;</td>
</tr>
<tr>
<td>&lt;Order method type&gt;</td>
<td>&lt;Quantity&gt;</td>
</tr>
</tbody>
</table>

Demonstration 2: Reuse layout components in a different report
Demonstration 2: Reuse layout components in a different report

Purpose:
To save time when creating new reports, you will create one report containing a standard page header that can be used in many. Next, you will create one report that will reuse this page header.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: Blank
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create a report with a page header that can be reused in other reports.
1. Open a new Blank template using the GO data warehouse (query) package.
2. On the work area, click Add, and then click Table
3. Create a table with 3 columns and 1 row.
   Because you want to reuse this table as a page header in other reports, you will name the table object.
4. Click the Container selector in the upper left cell, to select the entire table.
5. In the Properties pane, under MISCELLANEOUS, in the Name property, type StandardPageHeader, and then press Enter.
   You want to add your company logo to the left side of the page header.
6. From the Toolbox, expand LAYOUT, and then drag an Image object to the left cell of the table.
7. Click the Image object that you just added, and then in the Properties pane, under URL SOURCE, double-click the URL property.
8. In the Image URL dialog box, click Browse, navigate to http://vclassbase:88/images, and then click go_logo_small.jpg.
9. Click OK to close the Image Browser dialog box, and then click OK to close the Image URL dialog box.
   You want to add a text item in the middle of the page header that can be used to add a report title.
10. From the Toolbox tab, drag a Text Item to the center cell of the table, and then click OK to close the Text dialog box without adding any text. You will not specify the text to be used yet, because this will be different for each report. You will name this text object so that it can be overridden when the page header is reused in other reports.

11. Click the Text Item you just added, in the Properties pane, under MISCELLANEOUS, for the Name property type ReportTitle, and then press Enter.

**Task 2. Add additional details to the page header, and save the report.**

You want to add date and time information to the report header.

1. From the Toolbox tab, drag a Table object to the right cell of the table.

2. Set Number of columns to 1, Number of rows to 2, and then click OK.

3. From the Toolbox, from the TEXTUAL section, drag a Layout calculation object to the top cell of the table you added in the previous step.

4. In the Report expression dialog box, click the Functions tab, expand Report Functions, and then drag AsOfDate to the Expression Definition pane.

5. Validate the expression, and then click OK to close the Report expression dialog box.

You want to add a time stamp, to appear in the bottom-right corner of the page header.

6. From the Toolbox, drag a second Layout calculation object to the bottom cell of the table you added previously.

7. Click the Functions tab, expand Report Functions, and then drag AsOfTime to the Expression Definition pane.

8. Validate the expression, and then click OK to close the Report expression dialog box.

9. From the File menu, click Save.

10. Navigate to My content, in the Save as box type Layout Library, and then click Save.

**Task 3. Create a second report that reuses the standard page header.**

1. Open a new Blank template, using the GO data warehouse (query) package.

2. Add a List report to the work area, accepting the defaults.

3. From the Toolbox, drag a Block to the left of the list.
4. From Toolbox/ADVANCED, drag a Layout component reference object to the Block.
5. In the Component reference dialog, select Another report, click the ellipsis, and then navigate to My Folders.
6. Click Layout Library, and then click Open.
7. Under Available components to reference, click StandardPageHeader, and then click OK.

The page header from the Layout Library report appears. You want to customize the report.

8. In the header, click the Layout Component Reference object (this is the image), and then in the Properties pane, under GENERAL, double-click the Overrides property.

The Overrides dialog box appears. Because you gave the report title text object a distinct name, you can now override its contents in the shared page header.

9. Select the ReportTitle checkbox, and then click OK.
10. From the Toolbox, drag a Text Item to the component override area of the center cell of the header.
11. In the Text dialog box, type Quantity by Order Method, and then click OK.
12. Click the Quantity by Order Method title you just added, and then on the toolbar, change the font to Arial, and the size to 22 pt.

**Task 4. Add data to the list report and format the report.**

1. Select the list, to put it in focus.
2. From Data/Source, expand the Sales and Marketing (query) folder, and then expand the Sales (query) namespace.
3. Add the following query items to the list:
   • Order method: Order method type
   • Sales fact: Quantity

4. Save the report to the My content folder, as Quantity by Order Method.
5. Run the report in HTML.
   The results appear as follows:

   ![Quantity by Order Method](image)

   The header you created in the Layout Library report displays the title that you added to this report.

6. Close the rendered report tab.

**Task 5. Modify the shared page header and observe the results.**

1. In the Page header of the list report, click the Layout component reference object.
   In the Properties pane, the Embed property is set to Reference. This means any changes made to the shared page header in the Layout Library source report will be automatically applied in this report. You will now modify the shared page header in the source report.

2. On the Application bar, click the down arrow next to the report title.
3. Click the Layout Library report to open it.
4. In the Page header, click `<%AsOfTime()%>`, and then press Delete.
5. On the **Application** bar, click **Save**.

6. Click the down arrow next to the report title.

7. Click the **Quantity by Order Method** report to open it, and then run the report in **HTML**.

The result appears as follows:

![Quantity by Order Method](image)

The change that you made to the page header in the source report has automatically been applied to this shared page header, as seen in the top right corner.

8. Close the rendered report tab.

**Task 6. Manually update changes to the shared page header.**

You decide you do not want changes to the page header in this report to be applied automatically when the header changes in the source (Layout Library) report.

1. In the **Quantity by Order Method** report, in the page header, click the **Layout component reference** object.

2. In the **Properties** pane, under **GENERAL**, change the **Embed** property to **Copy**.

3. Save the report.

4. Open the **Layout Library** report, from **My content**.

5. In the **Page header**, drag the **Image** object from the left cell, to the center cell, so that it appears to the right of the title.

6. In the table, in the center cell, click and drag the **Text** object into the left cell.

7. Save the report.
8. Open the **Quantity by Order Method** report from **My content**. Although you switched the order of the image and text objects in the source report, this change is not reflected in the page header in this report. To make the page header in the Quantity by Order Method report consistent with the standard page header you created in the Layout Library report, you will now manually update the shared page header.

9. In the report, in the **Page header**, click the **Layout component reference** object, from the toolbar click the ellipsis, and then click **Update Component Copy**.

10. Save the report.

11. Open the **Quantity by Order Method** report from **My content**.

12. From the **Application** bar, click the **Edit** button to enable editing, if not already enabled.

13. In the report, in the **Page header**, click the **Layout component reference** object, from the toolbar click the ellipsis, and then click **Update Component Copy**.

A section of the results appears as follows:

![Quantity by Order Method](image)

The page header is updated with the changes made in the Layout Library.

14. Save the report.

15. Leave report authoring tab open for the next demonstration.

**Results:**
You created and reused a standard page header and then compared automatically and manually updating the reused page header when it changed in the source report.
Handle reports with no data

- When a query returns no data, you can provide alternate content or remove the data frame from the report.

Each data frame has a property called No Data Contents. When this is set to Yes, a new frame appears that you can populate with a text message, or an alternate data frame, and so on.

Each data frame also has a Render Page when Empty property. When this property is set to No, the page does not render.
Demonstration 3

Explore options for reports that contain no data

<table>
<thead>
<tr>
<th>List:</th>
<th>Crosstab:</th>
</tr>
</thead>
<tbody>
<tr>
<td>List contains no data!</td>
<td>Crosstab contains no data!</td>
</tr>
</tbody>
</table>

Demonstration 3: Explore options for reports that contain no data
Demonstration 3: 
Explore options for reports that contain no data

Purpose:
You want to create a report with three pages showing different methods of handling no data being returned. The first page will show default data handling, the second page will not display when the list is empty, and the third page will generate a custom message to replace the empty container.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: List and Crosstab
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create a list and a crosstab.
1. Open a new List template, using GO data warehouse (query).
2. From the Data/Source tab, add the following query items to the new list data container:
   - Products: Product line
   - Time: Year
   - Sales fact: Revenue

3. From the Toolbox, add a new crosstab report to the right of the list.
4. From the Query Name list, click Query1, and then click OK.
5. From **Data/Data Items**, add the following query items to the new crosstab:
   - **Rows** area:
     - Time: *Year*
   - **Column area**:
     - Products: *Product line*
   - **Measure area**:
     - Sales fact: *Revenue*

6. From the **Toolbox**, drag a **Table** to the page below the crosstab.
7. Clear the **Maximize width** checkbox, and then click **OK**.
8. Click inside the left table cell (but not the Add icon), and then in the **Properties** pane, under **BOX**, double-click **Padding**.
9. In the **Right padding** box, type 10, and then click **OK**.

**Task 2. Modify layout.**

1. Click the list **Container Selector**, and then drag the list into the left table cell.
2. Click the crosstab **Container Selector**, and then drag the crosstab into the right table cell.
3. From the **Toolbox** tab, drag a **Text Item** to the left of the list - but within the left table cell.
4. Type List:, and then click **OK**.
5. From the **Toolbox** tab, drag a **Text Item** to the left of the crosstab, within the right table cell.
6. Type Crosstab: then click **OK**.
7. Click to the right of the crosstab, and then on the toolbar, click **Top**.
8. Double-click the text in the header, type **Page 1 - Default Behavior**, and then click **OK**.
9. Left justify the text.
   The results appear as follows:

   ![Diagram of Page 1 - Default Behavior]

**Task 3. Add filters to your list and crosstab.**

Both reporting objects are linked to Query1, so only one set of filters will be needed;

1. Click anywhere in the list, click **Filters**, and then click **Edit Filters**.
2. Click **Add**, click **Advanced**, and then click **OK**.
3. Create and validate the following expression:
   
   \[\text{[Year]} = \text{?pyear}\]  
   Validate using 2011.
4. Click **OK** to close the **Detail filter expression** dialog box.
5. Create and validate another detail filter expression as follows:
   
   \[\text{[Product line]} = \text{?ppline}\]  
   Validate using Camping Equipment.
6. Click **OK** to close the **Detail filter expression** dialog box.
7. Click **OK** to close the **Filters** dialog box.

**Task 4. Create additional pages.**

1. On the **Navigate** tab, click **Page explorer**, and then click **Report pages**.
2. Right-click **Page1**, and then click **Copy**.
3. Right-click in the **Report pages** pane, and then click **Paste** to create **Page2**.
4. Right-click in the **Report pages** pane, and then click **Paste** to create **Page3**.

**Task 5. Configure a page that does not display when the list is empty.**

You do not want Page2 to render when the list is empty.

1. Double-click **Page2**, and then double-click the text in the **Page header** to edit the text.
2. Type **Page 2 - Do Not Render Page if No Data is Returned in the List**, and then click **OK**.
3. Click the list Container Selector to select the entire list.
4. From the Property pane, under GENERAL, set the Render page when empty property to No.

**Task 6. Configure a page with a custom No Data Handler that replaces an empty container with a message.**

You want to display a custom message when the list or crosstab is empty.

1. On the Page explorer tab, click Page 3.
2. Double-click the text in the Page header, and then update the text to: Page 3 - Show Custom Message When No Data is Returned.
3. Click OK.
4. Click the list Container Selector to select the entire list.
5. From the Property pane, under CONDITIONAL, double-click the No data contents property.
6. Select Content specified in the No data tab, and then click OK.

The No Data Contents property specifies whether to show the No Data Contents tab for the selected query frame. When set to Yes, you can specify on this tab what to show when there is no data. When set to No, the tab is hidden and the query frame reverts to the default behavior.

Your list should now appear as follows, with a new No Contents Data tab:

![No Data Available](image)

7. Double-click the object showing No Data Available, update the text to List contains no data!, and then click OK.

The results appear as follows:

![List contains no data!](image)

8. Click the crosstab Container Selector to select the entire crosstab.
9. From the Property pane, under CONDITIONAL, double-click the No data contents property.
10. Select Content specified in the No data tab, and then click OK.
11. Double-click the object showing No Data Available, update the text to Crosstab contains no data!, and then click OK.
Task 7. Add a prompt page.
1. On the Page explorer tab, click Prompt pages.
2. From the Toolbox, drag Page to the Prompt pages pane, and then double-click Prompt page1.
3. Insert a Table of 2 rows by 2 columns into the work area.
4. Insert a Text Item in the top left cell, type Select a Product Line:, press the space bar, and then click OK.
5. Insert a Text Item in the bottom left cell, type Select a Year:, press the space bar, and then click OK.
6. Expand PROMPTING, and then insert a Value prompt in the top right cell.
7. Select Use existing parameter, select ppline from the list, click Next, and then click Finish.
8. Insert a Value prompt in the bottom left cell.
9. Select Use existing parameter, select pyear from the list, click Next, and then click Finish.
10. Click the top left cell
11. In the Properties window, under POSITIONING, double-click Size & overflow.
12. Type 150 for the Width, and then click OK.

Task 8. Run report displaying data, and with no data to display.
1. Run the report in HTML.
2. When prompted, next to Select a Product Line:, select Camping Equipment, next to Select a Year:, select 2010, and then click Finish.
The results for page 1 appear as follows:

Since all of the queries in this report are filtered by the same parameters, all lists and crosstabs on the three report pages should look the same when data is returned. The page numbers refer to the pages in Page Explorer and not in the HTML view.
3. Click Page Down to see the Page 2 - Do Not Render Page if No Data is Returned in the List page.
4. Click **Page Down** to see the Page 3 - Show Custom Message When No Data is Returned page. 
Notice that all three pages appear with a list and crosstab.

5. Click the **Run** button to run the report again.

6. When prompted, select **Mountaineering Equipment**, select **2010**, and then click **Finish**.

The results for page 1 appear as follows:

```none
Page 1 - Default Behavior

List:
No Data Available  
Crosstab:
No Data Available
```

Notice how the individual pages are affected in the report since there is no data for 2010 for the product line Mountaineering Equipment.

The first page shows default behavior for the list and crosstab when there is no data returned. The list only shows the column titles, where the crosstab is not even rendered.

7. Click **Page Down**.

Notice that the Page 2 - Do Not Render Page if No Data is Returned in the List page did not display at all. This list contains no data and the list property Render page when empty is set to No, so the page did not render. You are taken directly to the Page 3 - Show Custom Message When No Data is Returned page. Both the list and crosstab are showing the custom message you created when no data is returned.

The results appear as follows:

```none
Page 3 - Show Custom Message When No Data is Returned

List:
List contains no data!  
Crosstab:
Crosstab contains no data!
```

8. Close the rendered report tab.

9. Leave the report authoring tab open for the following exercise.

**Results:**
You created a report with three pages showing different methods of handling no data being returned. The first page showed default data handling, the second page did not display when the list was empty, and the third page generated a custom message to replace the empty container.
Unit summary

- Enhance report design with report objects
- Reuse objects within the same report
- Share layout components among separate reports
- Discuss report templates
- Choose options to handle reports with no available data
Exercise 1: Analyze product quantities sold by month
Exercise 1:
Analyze product quantities sold by month

The Production Department has asked you to prepare a report that shows the quantity of products sold in each month of 2012 for all product lines, to help estimate production requirements for next year. The report must be broken into separate sections for each product line so that products from each line can be analyzed separately. The report name and logo must appear at the top and bottom of each page of the report.

To accomplish this:

- Open a new List template, based on the GO data warehouse(query) package, without saving the previous report.
- Add the following query items to the list report object:
  - Products: Product line
  - Products: Product
  - Time: Month
  - Sales fact: Quantity
- Section data and convert to a crosstab.
- Filter data so that only 2012 data is displayed and sort month data in ascending order.
- Edit title and add an image to the header block with the Company name and logo (cover2.jpg).
- Add a layout component reference (CompanyBlock) to the footer block.

For more information about where to work and the exercise results, refer to the Tasks and Results section that follows. If you need more information to complete a task, refer to earlier demonstrations for detailed steps.
Exercise 1: Tasks and results

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Template: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the list.

- **Toolbar**: Open a new List template, using the GO data warehouse (Query).
- **Source** tab: Add the following, from Sales (query):
  - Products: Product line to the new list report object.
  - Products: Product to the new list report object.
  - Time: Month to the new list report object.
  - Sales fact: Quantity to the new list report object.

The results appear as follows:

<table>
<thead>
<tr>
<th>Product line</th>
<th>Product</th>
<th>Month</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Month&gt;</td>
<td>&lt;Quantity&gt;</td>
</tr>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Month&gt;</td>
<td>&lt;Quantity&gt;</td>
</tr>
<tr>
<td>&lt;Product line&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Month&gt;</td>
<td>&lt;Quantity&gt;</td>
</tr>
</tbody>
</table>

- **Toolbar**: Section on <Product line>.
- Pivot the list to a crosstab using <Month>.

The results appear as follows:

![Crosstab Example]
Task 2. Filter and sort month data.

- **Toolbar**: Create and validate the following advanced filter expression for the crosstab:
  \[
  \text{[Sales (query)].[Time].[Year]}=\text{2012}
  \]
- Sort the `<#Month#>` column header ascending.

Task 3. Edit the title and add an image to the block.

- **Properties pane**: Name the header block `CompanyBlock`.
- **CompanyBlock**: Change the Text item to the following title: *The Sample Outdoors Company*.
  - Left align the text within the block.
  - Remove the underlining from the title.
- **Toolbox tab**: Add the `cover2.jpg` image to the right of the title.

Task 4. Add a reference to the block and its components.

- **Work area**: Delete the table in the footer block.
- ADVANCED: Add a Layout component reference for `CompanyBlock` into the page footer.
• **Toolbar**: Run the report in **HTML**.

• **Rendered Report** tab: scroll down.

The results appear as follows:

```
<table>
<thead>
<tr>
<th>Product line: Camping Equipment</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Quantity</strong></td>
</tr>
<tr>
<td>April</td>
</tr>
<tr>
<td>------</td>
</tr>
<tr>
<td>Canyon Mule Carryall</td>
</tr>
<tr>
<td>Canyon Mule Climber Backpack</td>
</tr>
<tr>
<td>Canyon Mule Cooler</td>
</tr>
<tr>
<td>Canyon Mule Extreme Backpack</td>
</tr>
<tr>
<td>Canyon Mule Journey Backpack</td>
</tr>
<tr>
<td>Canyon Mule Weekender Backpack</td>
</tr>
<tr>
<td>EverGlow Butane</td>
</tr>
<tr>
<td>EverGlow Double</td>
</tr>
<tr>
<td>EverGlow kalezone</td>
</tr>
<tr>
<td>EverGlow Lamp</td>
</tr>
<tr>
<td>EverGlow Single</td>
</tr>
<tr>
<td>Firefly 2</td>
</tr>
<tr>
<td>Firefly 4</td>
</tr>
<tr>
<td>Firefly Extreme</td>
</tr>
<tr>
<td>Firefly Lite</td>
</tr>
<tr>
<td>Firefly Waverider</td>
</tr>
<tr>
<td>Firefly Multi-light</td>
</tr>
<tr>
<td>Flicker Lantern</td>
</tr>
<tr>
<td>Hibernator</td>
</tr>
<tr>
<td>Hibernator Camp Out</td>
</tr>
</tbody>
</table>
```

• Close the rendered report tab.

• Sign out of the **IBM Cognos Analytics**.

• Close all browser windows.
Unit 9  Customize reports with conditional formatting

Customize reports with conditional formatting

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2016
Course materials may not be reproduced in whole or in part without the written permission of IBM.
Unit objectives

- Create multilingual reports
- Highlight exceptional data
- Show and hide data
- Conditionally render objects in reports
- Conditionally format one crosstab measure based on another
Change displays based on conditions

Display conditional text depending on language

Display conditional styles depending on data

Change displays based on conditions
3 steps for conditional formatting

1. Create a variable.
   - Define the condition and create values.
2. Assign the variable to an object in the report.
   - Properties pane, under Conditional, assign variable to object.
3. Apply formatting to object based on condition value.
   - Select specific value condition and apply formatting to object.
Step 1. Create a variable (1 of 2)

• Create variables and values to decide what element of the report will determine the change and the possible outcomes.

This step can be performed in Condition explorer.

Boolean variables are used if there are only two possible outcomes, where the values will be Yes or No.

String variables are used if there is more than one outcome, based on string values you will specify.

Language variables are used when the values are different languages.

The variable determines what will change in the report. For example, the report will vary depending on revenue, product line or the language in which the report is run.

The values define the possible scenarios or outcomes for the variable. For example, revenue is either above $150,000 ('yes') or not ('no'), product line is 'Camping Equipment', 'Golf Equipment', or the language may be 'Chinese' or 'Dutch', and so on.
Step 1. Create a variable (2 of 2)

• Define the condition and create values.

**Boolean**

<table>
<thead>
<tr>
<th>Expression Definition:</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Gross profit] &gt; 150000</td>
</tr>
</tbody>
</table>

**String**

<table>
<thead>
<tr>
<th>Expression Definition:</th>
</tr>
</thead>
<tbody>
<tr>
<td>if ([Revenue] &gt; 1000000) then ('high') else if ([Revenue] &lt; 25000) then ('low') else if ([Revenue] between 300000 and 600000) then ('medium')</td>
</tr>
</tbody>
</table>

**Report Language**

- **Languages**
  - [ ] Finnish (Finland)
  - [x] French
  - [ ] French (Belgium)

If you create a Boolean or string variable, you must define the condition.

If you create a language variable, you do not need to define the condition. You must choose the languages you want to support.

In the slide example for the string variable, revenue will be deemed 'high' if it is more than $1,000,000 or 'low' if it is less than $25,000.

The string variable's condition does not need to test all possible cases or the language variable hold all the possible languages because of the 'other' value. For example, in the slide string Expression Definition above, revenue between $25,000 and $300,000 is 'other'.

© Copyright IBM Corp. 2003, 2016
Course materials may not be reproduced in whole or in part without the prior written permission of IBM.
Step 2. Assign the variable to a report object (1 of 2)

- Select the condition style or variable that is appropriate to the object that you want to change based on the variable you want to assign to it.

Conditional Styles: Add conditional styles to highlight data in your report, based on set ranges.

Style Variable: Specifies a variable based on which the object can be conditionally styled.

Text Source Variable: Specifies a variable based on which the text can be chosen.

Render Variable: Specifies a variable based on which the object can be conditionally rendered.
Step 2. Assign the variable to a report object (2 of 2)

- Once you have created a variable, assign the variable to the object that you wish to conditionally format.

Select your object in the report layout that you want to add conditional formatting to. This step is performed in Conditional explorer.

After you have created your variable, you must define how the report will appear for each value. To do this, select the text or part of the report that will vary, and then apply the variable to it using the Properties pane.

When you apply the variable, the values for which you can perform conditional report authoring appear. By default, Reporting selects all the values assuming you will format all of them. If you wish to create conditional formatting for only some values, you can deselect the others.

If you apply a language variable, an additional value called Other appears by default. When you create a string or report language variable, and create or choose variables for it, an additional value called Other appears by default.
Step 3. Apply formatting to object based on condition value

- Select the condition value, and apply formatting to the object.

Once the report element has been designated as conditional, set the display for that value by modifying the report to appear the way you want it to look if that condition is satisfied.

This step is performed in Conditional explorer.

This step does not apply when working with a Render Variable.

When you select a value in the Condition explorer, the Explorer bar will turn green. This is to notify you that conditional formatting is turned on, and to remind you that all changes you make to the report only apply to the variable you selected.

After you have set the display for each value, turn the conditional formatting off by double-clicking the Explorer bar, or by selecting No Variable from the Condition explorer.
Demonstration 1

Create a multilingual report (optional)

Rapport sur les produits

<table>
<thead>
<tr>
<th>Lignes de produits</th>
<th>Types de produit</th>
<th>Revenus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessoires personnels</td>
<td>Couteaux</td>
<td>153 420 439,59</td>
</tr>
<tr>
<td></td>
<td>Jumelles</td>
<td>136 834 653,2</td>
</tr>
<tr>
<td></td>
<td>Lunettes</td>
<td>867 125 198,48</td>
</tr>
<tr>
<td>Matériel d'orientation</td>
<td></td>
<td>207 496 641,92</td>
</tr>
<tr>
<td>Montres</td>
<td></td>
<td>526 302 374,59</td>
</tr>
<tr>
<td>Accessoires personnels - Total</td>
<td></td>
<td>1 885 673 307,78</td>
</tr>
</tbody>
</table>

Demonstration 1: Create a multilingual report (optional)
Demonstration 1:  
Create a multilingual report (optional)

Purpose:
Your regional sales managers want to examine the revenue for all of your product types to promote the most profitable ones. Because this report will be distributed to offices in Germany, France, and the United States, you must run the report in different languages.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the list.
1. Open a new List template without saving any previous report.
2. From Source, add the following query items to the new list data container:
   - Products: Product line, Product type
   - Sales fact: Revenue

   ![Product line, Product type, Revenue]

3. Click the <Product line> list column body, and then click Group / Ungroup.
4. Click the `<Revenue>` list column body, click Summarize, and then click Total.
5. Run the report in HTML to examine the report.
6. Close the rendered report tab.

**Task 2. Create a language variable and choose the languages.**

You will apply conditional formatting to the header text so that a report title will appear in the language in which the report is run.

1. On the Navigate tab, click Condition explorer. The Condition explorer pane shows that there are currently no variables for this report.
2. Click Variables.
3. From the Toolbox, drag Report Language Variable to the Variables pane.
4. Scroll through the list to locate and select the checkboxes beside all of the English, French, and German languages, and then click OK.
5. In the Values pane, click English, Shift-click English (Zimbabwe), and then click Group Values.
6. Repeat Step 5 to group all French languages together.
7. Repeat Step 5 to group all German languages together.
   The results appear as follows:

   There are seventeen English languages, six French, and six German. You want to select all of these languages so that you can group them together. That way you only have to format the report for three grouped values, rather than for each individual language.

   The report now has one variable with three grouped values, one for each language in which the report will be run. Because you created a language variable, the expression is created for you.

8. In the Properties pane, under MISCELLANEOUS, in the Name box, replace Report Language1 with Language, and then press Enter.

9. On the Navigate tab, click Page explorer, and then click Page1.

**Task 3. Define the title as conditional text.**

1. Double-click the Text field at the top, in the Text dialog type Product Report Title, and then click OK.

2. Ensure that the report header text is left justified (use the containing block).

3. Click the Text field at the top to select it.

4. In the Properties pane, under CONDITIONAL, double-click Text source variable.
   The Text source variable dialog box appears.
5. From the **Variable** list, select **Language**.
The Values pane shows the three languages you chose, plus an option called Other. The three languages are selected by default so that you can use the Condition explorer to perform conditional authoring for any of these three languages.

6. Click **OK**.

**Task 4. Set the display for each value.**

1. On the **Navigate** tab, click **Condition explorer**, and then click **English**.
   The Explorer bar turns green to remind you that any changes you make to the report will apply to the value you selected. The previous title also disappears because you must specify the text for this value.

2. Change the title text to **Product Report**.

3. Within the **Condition explorer** pane, click **French**.

4. Change the title text to **Rapport sur les produits**.

5. Within the **Condition explorer** pane, click **German**.

6. Change the title text to **Produktbericht**.

7. Turn off conditional formatting by clicking **(No variable)** in the **Condition explorer** pane.

In order to run a report in different languages, the data source must be multilingual. Your browser must also be able to support multilingual characters, or else the characters will appear as boxes.

Now you can run the report in various languages.
Task 5. Run the report in various languages.

1. Run the report in HTML.
   Your report appears in English as this is our current default language. The report title appears as you created it for the English value. You will now run the report in French to see the results.
2. Close the rendered report tab.
3. Click Run options, and then click Show run options.
   You want to choose a language other than our current default.
4. Under Language, scroll down to and select French (France), and then click OK.
5. Run the report in HTML.
   A section of the results appear as follows:

   ![](image)

   The report appears in French, including the title you created.
   IBM Cognos Analytics cannot translate the data returned by the query. This must be done as part of data modeling and must be included in the published package.
6. Close the rendered report tab.
7. Repeat Steps 3 to 5 to run the report in **German (Austria)** and in **English (Zimbabwe)**.

A section of the results appears as follows:

![Product Report](image1)

8. Close the rendered report tab.
9. Click **Run options**, and then click **Show run options**.
10. Under **Language**, scroll up, select **(Default)**, and then click **OK**.
11. Leave the report authoring tab open for the next demonstration.

**Results:**
Regional Sales managers can examine the revenue for all of your product types to promote the most profitable products. This report can be distributed to offices in German, French, and English speaking countries in the appropriate languages.
### Demonstration 2

Highlight exceptional data

<table>
<thead>
<tr>
<th>Revenue</th>
<th>Americas</th>
<th>2010</th>
<th>2011</th>
<th>2012</th>
<th>2013</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personal Accessories</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Binoculars</td>
<td>8,640,023.69</td>
<td>9,204,463.69</td>
<td>11,564,263.97</td>
<td>9,698,328.44</td>
<td></td>
</tr>
<tr>
<td>Eyewear</td>
<td>51,694,482.68</td>
<td>64,664,792.82</td>
<td>80,687,164.04</td>
<td>88,629,626.99</td>
<td></td>
</tr>
<tr>
<td>Knives</td>
<td>11,328,089.94</td>
<td>12,661,267.75</td>
<td>15,647,263.69</td>
<td>10,513,594.42</td>
<td></td>
</tr>
<tr>
<td>Navigation</td>
<td>15,667,527.25</td>
<td>13,543,263.78</td>
<td>15,757,628.59</td>
<td>14,352,706.56</td>
<td></td>
</tr>
<tr>
<td>Watches</td>
<td>40,969,813.22</td>
<td>44,372,904.39</td>
<td>49,213,306.66</td>
<td>32,816,631.11</td>
<td></td>
</tr>
<tr>
<td>Mountaineering</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Equipment</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Climbing Accessories</td>
<td>0,938,753.87</td>
<td>0,838,514.65</td>
<td>3,578,473.95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rope</td>
<td>8,065,819.48</td>
<td>13,962,404.12</td>
<td>11,968,118.86</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Safety</td>
<td>0,001,059.96</td>
<td>10,137,737.95</td>
<td>7,140,543.96</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tents</td>
<td>10,170,549.49</td>
<td>16,631,906.21</td>
<td>12,306,958.95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Camping Equipment</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cooking Gear</td>
<td>18,885,808.93</td>
<td>20,262,604.67</td>
<td>22,792,669.65</td>
<td>16,933,277.29</td>
<td></td>
</tr>
<tr>
<td>Lanterns</td>
<td>3,935,045.78</td>
<td>8,516,888.22</td>
<td>12,274,847.87</td>
<td>8,136,631.46</td>
<td></td>
</tr>
<tr>
<td>Pots</td>
<td>20,796,915.16</td>
<td>25,622,161.46</td>
<td>30,664,705.37</td>
<td>24,919,022.74</td>
<td></td>
</tr>
<tr>
<td>Sleeping Bags</td>
<td>19,652,776.61</td>
<td>20,629,479.54</td>
<td>20,724,366.29</td>
<td>19,755,659.05</td>
<td></td>
</tr>
<tr>
<td>Tents</td>
<td>15,466,623.45</td>
<td>43,218,516.95</td>
<td>43,699,626.23</td>
<td>35,031,372.65</td>
<td></td>
</tr>
<tr>
<td>Outdoor Protection</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>First Aid</td>
<td>12,123,619.60</td>
<td>15,841,665.15</td>
<td>53,855,858.46</td>
<td>125,402,515</td>
<td></td>
</tr>
<tr>
<td>Insect Repellent</td>
<td>5,872,455.60</td>
<td>3,410,199.19</td>
<td>5,864,821.12</td>
<td>634,776.13</td>
<td></td>
</tr>
<tr>
<td>Sunscreen</td>
<td>3,224,068.68</td>
<td>2,817,622.68</td>
<td>990,919.08</td>
<td>440,176.4</td>
<td></td>
</tr>
</tbody>
</table>

**Demonstration 2: Highlight exceptional data**
Demonstration 2: Highlight exceptional data

Purpose:
A manager wants to quickly identify revenue greater than $20,000,000 and less than $5,000,000 to identify high and low revenue-generating product types in all sales regions. You need to create a report that displays revenue data in different colors depending on revenue values.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: Crosstab
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the crosstab.
1. Open a new Crosstab template using the GO data warehouse (query) package.
2. From the Source tab, add the following query items to the new crosstab data container:
   - Rows area:
     - Products: Product line, Product type (nested as a child)
   - Columns area:
     - Retailers: Region
     - Time: Year (nested under Region)
   - Measure area:
     - Sales fact: Revenue

   ![Crosstab table](image)

You will create a variable to define revenue as 'high' or 'low' if the amount is above or below specified amounts.
3. On the **Navigate** tab, click the **Condition explorer** tab.

4. In the **Condition explorer** pane, click **Variables**, and then from the **Toolbox** tab, drag a **String Variable** to the **Variables** pane.  
Because you are creating a string variable, you must specify the condition on which revenue will change, and then create values for the possible outcomes.

5. Create and validate the following expression:
   \[
   \text{if ([Query1].[Revenue]>20000000) then ('high') else if ([Query1].[Revenue]<5000000) then ('low')}
   \]
   Hint: You can double-click Revenue in the Available Components pane to add it to the expression as you write it.

6. Click **OK**.

7. Under the **Values** box, click **Add**.

8. In the **Add** dialog box, type **high**, and then click **OK**.

9. Repeat steps 7 and 8 to add a second value called **low**.  
The values created in steps 7 - 8 must be spelled exactly as they are spelled in the expression definition for the variable.

10. In the **Properties** pane, under **MISCELLANEOUS**, in the **Name** box, modify the text to **Revenue_high_low**, and then press **Enter**.

11. Navigate back to **Page1**.
Now that you have created a variable and specified its values, you must format the revenue cells for each value.

**Task 2. Define the measures as conditional and set the display for each value.**

1. Click any of the **Revenue** (**<#1234#>**) cells in the crosstab, in the **Properties** pane, click **Select Ancestor**, and then click **Crosstab fact cells**.

2. Under **CONDITIONAL**, double-click **Style variable**. 
The Style variable dialog box appears.

3. From the **Variable** list, select **Revenue_high_low**, and then click **OK**.  
The measures cells are now conditionally formatted using the variable you just created. You must now set the display for each value.

4. Click the **Condition explorer** tab, and then click **high**. 
The Explorer bar turns green.

5. With the **Revenue** cells still selected, on the crosstab toolbar, click **Style**, and then click **Edit** next to the **Font** option.

6. Set **Foreground Color** to **Green**, set **Weight** to **Bold**, click **OK** to close the **Font** dialog box, and then click **OK** again to close the **Style** dialog box.
7. Repeat steps 4 - 6 to change the **Foreground** color for the **low** value to **Red**.
8. Click **(No variable)** to turn the conditional formatting off.
9. Run the report in **HTML**.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Revenue</th>
<th>Americas</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2010</td>
<td>2011</td>
<td>2012</td>
<td>2013</td>
</tr>
<tr>
<td>Personal Accessories</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Binoculars</td>
<td>8,846,023.06</td>
<td>9,264,403.99</td>
<td>11,564,233.97</td>
<td>9,696,326.44</td>
</tr>
<tr>
<td>Eyewear</td>
<td>51,204,488.85</td>
<td>64,364,795.63</td>
<td>89,687,954.94</td>
<td>69,628,625.09</td>
</tr>
<tr>
<td>Knives</td>
<td>11,328,889.94</td>
<td>10,861,897.75</td>
<td>15,047,283.69</td>
<td>19,513,994.42</td>
</tr>
<tr>
<td>Navigation</td>
<td>15,667,887.05</td>
<td>13,943,283.78</td>
<td>19,757,826.39</td>
<td>14,382,780.56</td>
</tr>
<tr>
<td>Watches</td>
<td><strong>40,506,913.37</strong></td>
<td><strong>44,372,804.39</strong></td>
<td><strong>51,213,366.66</strong></td>
<td><strong>32,018,631.1</strong></td>
</tr>
<tr>
<td>Mountaineering Equipment</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Climbing Accessories</td>
<td>6,928,733.67</td>
<td>8,238,351.85</td>
<td>8,576,743.95</td>
<td></td>
</tr>
<tr>
<td>Rope</td>
<td>8,065,819.48</td>
<td>13,952,484.12</td>
<td>11,568,116.88</td>
<td></td>
</tr>
<tr>
<td>Safety</td>
<td>6,881,859.98</td>
<td>10,137,737.5</td>
<td>9,410,543</td>
<td></td>
</tr>
<tr>
<td>Tools</td>
<td>10,178,540.49</td>
<td>16,831,986.21</td>
<td>12,356,858.95</td>
<td></td>
</tr>
<tr>
<td>Camping Equipment</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cooking Gear</td>
<td>18,686,808.03</td>
<td><strong>20,266,504.57</strong></td>
<td><strong>22,792,656.65</strong></td>
<td><strong>16,033,277.29</strong></td>
</tr>
<tr>
<td>Lanterns</td>
<td>9,550,941.73</td>
<td>9,516,850.22</td>
<td>12,274,647.67</td>
<td>8,136,831.8</td>
</tr>
<tr>
<td>Packs</td>
<td><strong>20,705,015.1</strong></td>
<td>25,822,181.45</td>
<td>35,194,708.35</td>
<td>24,619,832.74</td>
</tr>
<tr>
<td>Sleeping Bags</td>
<td>19,652,376.61</td>
<td><strong>23,529,473.94</strong></td>
<td>29,772,358.26</td>
<td>19,755,038.84</td>
</tr>
<tr>
<td>Tents</td>
<td><strong>35,466,023.41</strong></td>
<td><strong>43,248,510.95</strong></td>
<td>49,990,030.23</td>
<td><strong>35,031,372.05</strong></td>
</tr>
<tr>
<td>Outdoor Protection</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>First Aid</td>
<td><strong>2,127,816.09</strong></td>
<td><strong>641,985.19</strong></td>
<td>536,893.86</td>
<td>258,402.95</td>
</tr>
<tr>
<td>Insect Repellents</td>
<td>5,872,450.5</td>
<td><strong>3,410,919.19</strong></td>
<td><strong>1,644,331.12</strong></td>
<td><strong>634,775.13</strong></td>
</tr>
<tr>
<td>Sunscreen</td>
<td><strong>3,324,046.67</strong></td>
<td><strong>2,937,932.68</strong></td>
<td>980,915.8</td>
<td>440,176.4</td>
</tr>
</tbody>
</table>

You can see that some Camping Equipment product types generated high revenue over a four-year period in Central Europe, whereas Outdoor Protection generated low revenue. Notice that when the revenue condition is not satisfied (when it is neither high nor low) revenue appears in black.

10. Close the rendered report tab.
11. Leave the report authoring tab open for the next demonstration.

**Results:**

You created a report that compares product line revenue for all sales regions to quickly identify by color the product type revenues greater than $20,000,000 and less than $5,000,000.
Conditionally render objects in reports

- Using conditional rendering, you can determine whether certain objects will be included in a report when the report is run.

If objects are not rendered, they do not take up space in the report when it is run.

Conditional rendering is useful when your report contains sensitive data or data that may be relevant for some consumers but not for others.

When conditional rendering is applied to a column in a list report, the conditional rendering applies to all portions of the column including the title, the body cells, and header and footer cells.

In the slide example, the product description column is rendered because the report was run in HTML format. An expression was created on the product description column to only render if the report output is HTML.
Demonstration 3: Create a report with a conditionally rendered column

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>06/12/2009</td>
<td>Achieved a goal related to the project.</td>
</tr>
<tr>
<td>06/13/2009</td>
<td>Completed a task ahead of schedule.</td>
</tr>
<tr>
<td>06/14/2009</td>
<td>Improved our team's communication skills.</td>
</tr>
<tr>
<td>06/15/2009</td>
<td>Participated in a training session.</td>
</tr>
<tr>
<td>06/16/2009</td>
<td>Attended a meeting with key stakeholders.</td>
</tr>
</tbody>
</table>

© Copyright IBM Corporation 2016
Demonstration 3:
Create a report with a conditionally rendered column

Purpose:
Some users want a report to include descriptions of each product, while others are familiar with the products and do not want these descriptions in the report. You will create a report that can be run with or without a column displaying product descriptions based on the format in which you run the report.

Portal:  http://vclassbase:9300/bi
User/Password:  brettonf/Education1
Package:  Team content\Samples\Models\GO data warehouse (query)
Report Type:  List
Folder:  Sales and Marketing (query)
Namespace:  Sales (query)

Task 1. Create the list.
1. Open a new List template without saving the previous report.
2. From the Source tab, add the following query items to the new list data container:
   - Time: Date
   - Sales order: Order number
   - Products: Product, Product Description
   - Sales fact: Revenue

<table>
<thead>
<tr>
<th>Date</th>
<th>Order number</th>
<th>Product</th>
<th>Product description</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Date&gt;</td>
<td>&lt;Order number&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Product description&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Date&gt;</td>
<td>&lt;Order number&gt;</td>
<td>&lt;Product&gt;</td>
<td>&lt;Product description&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
</tbody>
</table>

3. Click <Date>, and then on the toolbar, click Section / unsection.
4. Click <Order number>, and then on the toolbar, click Group / Ungroup.
5. Click <Revenue>, on the toolbar click Summarize, and then click Total.
6. On the toolbar, click Filters, and then click Edit Filters.
7. Click Add, click Advanced, and then click OK.
8. Create and validate the following expression:
\[ \text{Sales (query).Time.Month key}=201001 \]

9. Click OK to close the Detail filter expression dialog box, and then click OK to close the Filters dialog box.
The report runs more efficiently with this filter.

10. Change the title text at the top of the report to Order Details, and then left justify the text field within the block.

**Task 2. Add a Boolean variable.**

1. Click <Product description>, in the Properties title bar, click Select Ancestor, and then click List column at the bottom of the list.

2. In the Properties pane, under CONDITIONAL, double-click the Render variable property.

3. From the Variable list, select New boolean variable.

4. In the New variable dialog box, type ShowDescrip, and then click OK.

5. Create and validate the following expression:
   \( \text{ReportOutput ()='HTML'} \)
   Hint: Find ReportOutput () from the Functions tab, Report Functions folder.

6. Click OK until all dialog boxes are closed.

**Task 3. Run the report in HTML, and then in PDF.**

1. Run the report in HTML.
   A section of the results appear as follows:

   ![Order Details](image)

   The report contains a column displaying a description of each product.
2. Click the down arrow to the right of the **Run report** button, and then run the report in **PDF**.

A section of the results appear as follows:

![Order Details](image)

When the report is rendered in PDF format, Product description is not rendered.

3. Close the rendered report tab.
4. Leave report authoring tab open for the next demonstration.

**Results:**
**You created a report you can run with or without a column displaying product descriptions based on the format in which you run the report.**
Conditionally format one crosstab measure based on another

You can conditionally format one crosstab measure based on another crosstab measure using the Conditional Styles dialog box.
Demonstration 4

Conditionally format one crosstab measure based on another

<table>
<thead>
<tr>
<th></th>
<th>2010</th>
<th>2011</th>
<th>2012</th>
<th>2013</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Gross profit</td>
<td>Revenue</td>
<td>Gross profit</td>
<td>Revenue</td>
</tr>
<tr>
<td>Camping Equipment</td>
<td>117,156,311.6</td>
<td>332,886,338.96</td>
<td>140,869,111.12</td>
<td>402,757,573.17</td>
</tr>
<tr>
<td>Gulf Equipment</td>
<td>70,765,385.29</td>
<td>153,553,850.58</td>
<td>70,616,344.54</td>
<td>100,006,427.07</td>
</tr>
<tr>
<td>Outdoor Protection</td>
<td>21,549,287.72</td>
<td>36,105,521.97</td>
<td>46,961,634.83</td>
<td>25,000,076.05</td>
</tr>
<tr>
<td>Personal Accessories</td>
<td>150,040,408.96</td>
<td>391,647,093.61</td>
<td>203,710,120.40</td>
<td>496,322,300.89</td>
</tr>
<tr>
<td>Maintenance Equipment</td>
<td>342,723,307.2</td>
<td>197,999,059.84</td>
<td>424,235,927.4</td>
<td>161,038,523.20</td>
</tr>
<tr>
<td></td>
<td>152,570,742.95</td>
<td>428,385,422.17</td>
<td>238,587,774.25</td>
<td>628,382,472.87</td>
</tr>
<tr>
<td></td>
<td>238,587,774.25</td>
<td>628,382,472.87</td>
<td>132,639,885.65</td>
<td>382,910,328.57</td>
</tr>
</tbody>
</table>

**Demonstration 4: Conditionally format one crosstab measure based on another**
Demonstration 4:  
Conditionally format one crosstab measure based on another

Purpose:  
Consumers would like to see conditional formatting for revenue values based on Gross profit values in a crosstab. To achieve this, you will take advantage of the IBM Cognos ability to conditionally format one crosstab value based on another.

Portal:  http://vclassbase:9300/bi  
User/Password:  brettonf/Education1  
Package:  Team content\Samples\Models\GO data warehouse (query)  
Report Type:  Crosstab  
Folder:  Sales and Marketing (query)  
Namespace:  Sales (query)

Task 1. Create the crosstab report.

1. Open a new Crosstab template using the GO data warehouse (query) package.
2. From the Source tab, add the following query items to the new crosstab report object:
   - Rows area:
     - Products: Product line
   - Columns area:
     - Time: Year
     - Sales fact: Gross Profit, Revenue (Nested under Year)

3. Run the report in HTML.
4. Examine the results.
5. Close the rendered report tab.
**Task 2. Conditionally format one crosstab measure based on another.**

1. Click the **Revenue** fact cells (<#1234#>), on the crosstab toolbar, click **More**, point to **Style**, and then click **Conditional styles**.

2. Click **New Conditional Style** , and then click **New Conditional Style** from the list.

3. In the **Base it on the following data item** list, click **Gross profit**, and then click **OK**.

4. In the **Name** box, type **Gross Profit Performance**.

5. Click **New Value** , type **180000000** (180,000,000), and then click **OK**.

6. Repeat step 5 to add values for **130000000** (130,000,000), **70000000** (70,000,000), and **20000000** (20,000,000).

7. In the **Style** column, in the top drop down list, select **Excellent**, and then for the remaining drop down lists select **Very good**, **Average**, **Below average**, and **Poor** respectively.

8. To the right of **Poor**, click **Edit Style**.

9. Click **Edit** to the right of **Font**, change **Weight** to **Bold**, and then click **OK**.
10. Click **OK** to close the **Style** dialog box. The results appear as follows:

![Conditional style - numeric range](image)

11. Click **OK** to close the **Conditional style - numeric range** dialog box, and then click **OK** again to close the **Conditional styles** dialog box.

**Task 3. Run the report and use existing conditional styles.**

1. Run the report in **HTML**.

   The results appear as follows:

   ![Revenue values](image)

   Revenue values are conditionally formatted based on Gross profit values.

2. Close the rendered report tab.

   You will now apply the same Gross Profit Performance style to the Gross profit measure.
3. Click the **Gross profit** fact cells, from the toolbar click the ellipsis, then point to **Style**, and then click **Conditional styles**.

4. Click **New Conditional Style**, click **Use Existing Conditional Style**, and then click **Gross Profit Performance**.

5. Click **OK** to close the **Select existing conditional styles** dialog box, and then click **OK** to close the **Conditional styles** dialog box.

6. Run the report in **HTML**.

   The results appear as follows:

   Now the same conditional style is applied to both measures.

7. Close the rendered report tab.

8. Leave the report authoring tab open for the following exercise.

   **Results:**

   By taking advantage of the IBM Cognos ability to conditionally format one crosstab value based on another, you were able to create a crosstab in which conditional formatting for revenue values were based on the Gross profit values. You then applied the same conditional formatting to the Gross profit values to create a uniform look for the crosstab.
Unit summary

• Create multilingual reports
• Highlight exceptional data
• Show and hide data
• Conditionally render objects in reports
• Conditionally format one crosstab measure based on another
### Exercise 1: Distinguish yearly data

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Americas</th>
<th>Asia Pacific</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department Store</td>
<td>1,727,876</td>
<td>1,487,381</td>
</tr>
<tr>
<td>Direct Marketing</td>
<td>148,802</td>
<td>141,203</td>
</tr>
<tr>
<td>Equipment Rental Store</td>
<td>81,440</td>
<td>101,775</td>
</tr>
<tr>
<td>Eyewear Store</td>
<td>290,533</td>
<td>246,970</td>
</tr>
<tr>
<td>Golf Shop</td>
<td>271,114</td>
<td>312,422</td>
</tr>
<tr>
<td>Outdoors Shop</td>
<td>1,000,571</td>
<td>2,010,483</td>
</tr>
<tr>
<td>Sports Store</td>
<td>1,794,199</td>
<td>1,680,846</td>
</tr>
<tr>
<td>Warehouse Store</td>
<td>695,084</td>
<td>406,603</td>
</tr>
</tbody>
</table>
Exercise 1: Distinguish yearly data

You have been asked to create a report that shows the volume of sales in each region by retailer type. To make the yearly data easier to distinguish, you will format the report so each year column will have a different background color.

To accomplish this you will:

- Open a new crosstab template, using the GO data warehouse (query) package, without saving the previous report.
- Add the following data items to a new crosstab using the GO data warehouse (query)\Sales (query):
  - Retailers: Region
  - Time: Year (Nested under Region)
  - Retailer type: Retailer type
  - Sales fact: Quantity
- Create a year string variable with values for each year.
- Assign the measures in the crosstab a conditional style using the year variable.
- Set the display for each year to display a yellow background for 2010, a green background for 2011, a blue background for 2012, and a red background for 2013.

For more information about where to work and the exercise results, refer to the Tasks and Results section that follows. If you need more information to complete a task, refer to earlier demonstrations for detailed steps.
Exercise 1: Tasks and results

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content/Samples/Models/GO data warehouse (query)
Report Template: Crosstab
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the crosstab report and add an Order year variable.

- Toolbar: Open a new Crosstab template, using the GO data warehouse (query) package.
- Source tab: Add the following query items to the crosstab:
  - Retailer type: Retailer type to the rows area of the crosstab report object.
  - Retailers: Region to the columns area of the crosstab report object.
  - Time: Year as a nested column under Region.
  - Sales fact: Quantity to the measures area of the crosstab report object.

The results appear as follows:

- Condition explorer: Create and validate the following String Variable expression: [Query1].[Year]
  - Name the variable Year.
- Toolbar: Go back to Page 1.
Task 2. Define the measures as conditional and set the display for each value.

- **Properties pane:** Link the Crosstab fact cells, Style variable, to the Year variable.
- **Variable list,** click Year, and then click **OK.**
- **Condition Explorer:** Click the 2010 variable.
- **Toolbar:** For the Crosstab Fact Cells, set the background color for each conditional value, using the Color Swatch colors, as follows:

<table>
<thead>
<tr>
<th>Year</th>
<th>Color</th>
<th>Color Code</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010</td>
<td>Light yellow</td>
<td>#FFFF99</td>
<td>Bottom row, 3rd from right</td>
</tr>
<tr>
<td>2011</td>
<td>Light green</td>
<td>#33FF99</td>
<td>4th row, 3rd from right</td>
</tr>
<tr>
<td>2012</td>
<td>Light blue</td>
<td>#66FFFF</td>
<td>6th row, last column</td>
</tr>
<tr>
<td>2013</td>
<td>Light red</td>
<td>#FF6666</td>
<td>2nd last row, 4th from right</td>
</tr>
</tbody>
</table>

- **Explorer bar:** Turn the conditional formatting off.
- **Toolbar:** Run the report in HTML.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Americas</th>
<th>Asia Pacific</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department Store</td>
<td>1,727,876</td>
<td>1,487,381</td>
</tr>
<tr>
<td>Direct Marketing</td>
<td>148,802</td>
<td>141,290</td>
</tr>
<tr>
<td>Equipment Rental Store</td>
<td>81,480</td>
<td>101,775</td>
</tr>
<tr>
<td>Eyewear Store</td>
<td>206,533</td>
<td>240,070</td>
</tr>
<tr>
<td>Golf Shop</td>
<td>271,114</td>
<td>313,422</td>
</tr>
<tr>
<td>Outdoors Shop</td>
<td>1,600,571</td>
<td>2,510,484</td>
</tr>
<tr>
<td>Sports Store</td>
<td>1,794,109</td>
<td>1,890,046</td>
</tr>
<tr>
<td>Warehouse Store</td>
<td>609,004</td>
<td>400,000</td>
</tr>
</tbody>
</table>

You have created a report that shows the volume of sales in each region by retailer type. To make the yearly data easier to distinguish, you have formatted the report so each year column will have a different background color.

- Close the rendered report tab.
- Sign out of IBM Cognos Analytics without saving any reports.
- Close all browser windows.
Unit 10  Drill-through definitions

Drill-through definitions

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2016
Course materials may not be reproduced in whole or in part without the written permission of IBM.
## Unit objectives

- Discuss parameter-driven drill through
- Discuss dynamic drill through
- Set up package-based drill-through definitions
- Set scope
- Use the Drill Through Assistant
Let users navigate to related data in IBM Cognos Analytics

Drill-through access lets users navigate between reports to view related data to help them answer business questions.

In IBM Cognos Analytics, report authors can set up drill-through access to and from reports, queries, advanced queries, and analyses using dimensional and relational data sources.

It is also possible to set up drill-through access to IBM Cognos Analytics targets from third party sources and to third party targets from Cognos sources. Setting up drill-through access to and from third-party sources and targets can be accomplished using URL requests or by using the Software Development Kit.
Set up drill-through access from a report

You can set up drill-through access from reports that are created from relational data sources or dimensional data sources.

You can let users navigate to target reports, queries, and analyses created from both relational and dimensional data sources.

You must create the target report before you can set up drill-through access.
Package-based drill through

To let users navigate to a specific target report from reports, analyses, and queries created using a package, you can create a drill-through definition for the package.

The source reports do not need to be created when you create the drill-through definition. This lets you set up drill-through access to the target report, and then later, report authors can create as many source reports as required.

Each package drill-through definition can have only one target. You can create multiple drill-through definitions for a package.

In the slide example, a drill-through definition has been created for the GO Data Warehouse (analysis) package. Users can drill through to the target report from a variety of sources created using the same package.

A model in Framework Manager is a business presentation of the structure of the data from one or more data sources (such as IBM Cognos PowerCubes and relational databases). A model describes the metadata objects, structure, and grouping, as well as relationships and security.
You can set permissions properties for target reports to determine which users will be able to open them when they attempt to drill through. You can also set permissions properties for drill-through definitions to determine which users have access to these drill-through definitions.

In IBM Cognos Analytics (v11.0), targets can be reports or packages.
Specify the values passed to target parameters

When you set up drill-through access, you must map the values that the source report will pass to the target parameters.

If you do not specify which values to pass to target parameters, then when users drill through they will be prompted to select values for any required target parameters.

When dealing with dimensional sources, you can also select which property of the member you would like to pass to the target report (for example: Member Unique Name (MUN), Member Caption, or Business Key). It is important to know which values are conformed between the source report and the target report data sources.
Steps to set up a package-based drill-through definition

Before you can set up drill-through access for a package, you must have a target report created. Next, create a drill-through definition that lets users navigate to the target report from reports created using the package. To let users drill through from a report using a package drill-through definition, you must enable this drill behavior in the source report. You can create the target report in Cognos Analysis or any studio.

When you set up drill through for a package, the following steps are optional:

- Create the target report (may be done by a different report author)
- Add a parameter to the target report.
- Limit the data items from which users can drill through.
- Map the parameter values passed to the target report.

Steps to set up a package-based drill-through definition

1. Create the target report.
2. Create the drill-through definition.
3. Drill through to the target report.

Create the target report
Add a parameter to the target report
Save the target report
Hide the target report

Select the package
Limit the data items from which users can drill through
Select the target report
Map the parameter values passed to the target report if needed, or specify dynamic filtering of target

Create a report, using the source package
Drill through to view related data in the target report
The target report appears and displays filtered data

* Optional
Limit the items that users can drill through from

- Package drill-through definitions control where users can start drill through in source reports.
- To do this, set a data item in the source package as the scope of the drill-through definition.

<table>
<thead>
<tr>
<th>Product type values</th>
<th>Quantity</th>
<th>2004</th>
<th>2005</th>
<th>2006</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cooking Gear</td>
<td>2,905,120</td>
<td>3,501,329</td>
<td>4,060,635</td>
<td></td>
</tr>
<tr>
<td>Eyewear</td>
<td>4,066,410</td>
<td>5,180,407</td>
<td>6,354,258</td>
<td></td>
</tr>
<tr>
<td>First Aid</td>
<td>450,978</td>
<td>186,317</td>
<td>133,692</td>
<td></td>
</tr>
<tr>
<td>Golf Accessories</td>
<td>613,311</td>
<td>791,935</td>
<td>963,013</td>
<td></td>
</tr>
<tr>
<td>Insect Repellents</td>
<td>2,864,588</td>
<td>1,806,770</td>
<td>808,715</td>
<td></td>
</tr>
</tbody>
</table>

With scope set to Product type, users can drill through to the target report from any of the fact cells in this report.

If a target report contains one parameter, it makes sense to limit the scope of the drill-through definition to the item that must be passed to this parameter. This ensures that users will not be prompted to select a parameter value when they drill through.

Once you have set the scope of a drill-through definition to a particular data item, users can drill through from a cell in source reports only if its context includes this item.

If you do not set the scope of a drill-through definition for a package, users can drill through from any cell in any report created using the package.

It is useful to set the scope of drill-through access to limit the number of possible target reports users see when they drill through. If you have created many drill-through definitions for a package and you do not set the scope, users may be presented with an overwhelming number of possible target reports when they drill through.

When you create a drill-through definition for a relational package, set the scope to a specific fact/measure or query item, such as the Revenue fact from the Sales fact query subject or the Product type query item from the Products query subject.

When you create drill-through definitions for OLAP or dimensionally modeled relational (DMR) packages, you can set the scope to a dimension, a level in the dimension, or a measure.
Measure-based scope

- Set scope based on a measure in a drill-through definition.
- The source report must use the selected measure in order to drill through to target report.

When defining a drill-through definition, the user has a choice to set the scope on the target report. If you base scope on a measure, then the target report that is specified in the drill-through definition will only appear on the Related Links list of a source report if the selected measure is in the source report.
Drill Through Assistant

- The Drill Through Assistant lets you see the values that are passed from the source report to the target report.

IBM Cognos includes a debugging functionality, called the Drill Through Assistant, which you can use to troubleshoot your drill-through definitions created in the IBM Cognos portal. It can also help you understand how the drill-through functionality works, especially across different types of data sources.

The Drill Through Assistant is especially useful for Report Authors and Report Administrators. By default, no user, group, or role can use the Drill Through Assistant until the capability is granted.

For more information about the Drill Through Assistant see the Administration and Security guide.
Demonstration 1: Set up drill-through access for a package
Demonstration 1:  
Set up drill-through access for a package

Purpose:  
You have been asked to create a drill-through definition to let users navigate to a detailed product line sales report that is created from reports using the GO Data Warehouse (query) package. To let users focus on specific areas of interest, the target report will only display data for the product line from which users drill through. Finally, you will enable the Drill Through Assistant and view the values that are passed.

Portal:  
http://vclassbase:9300/bi

User/Password:  
hirschb/Education1 (Branka Hirsch is a Report Administrator)

Task 1. Create the target report.

In this task you will open and run an existing report. Be sure you are logged in as Branka Hirsch (see above) before starting this particular demonstration.

1. From the Welcome screen, click Team content and then navigate to Legacy Samples > Samples_Drillthrough > Models > Go Data Warehouse (query) > Report Studio Report Samples.
2. Click the Total Revenue by Country entry.

This is a report based on a relational model that provides information on region, country, retailer name, and revenue for product lines.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Revenues</th>
<th>Golf Equipment</th>
<th>Outdoor Recreation</th>
<th>Personal Accessories</th>
<th>Camping Equipment</th>
<th>Mountaineering Equipment</th>
<th>Total (Product line)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asia Pacific</td>
<td>Australia</td>
<td>2,106,790.0</td>
<td>1,953.3</td>
<td>1,037,833.78</td>
<td>5,091,375.56</td>
<td></td>
</tr>
<tr>
<td>Beech Beds Pty Ltd.</td>
<td></td>
<td>4,107,255.17</td>
<td>293,119.01</td>
<td>5,230,905.74</td>
<td>15,708,235.03</td>
<td>25,549,404.97</td>
</tr>
<tr>
<td>Black Stump Camping Supplies</td>
<td></td>
<td>7,219</td>
<td>2,646,226.32</td>
<td>1,262,948.46</td>
<td>462,817.16</td>
<td>4,192,708.97</td>
</tr>
<tr>
<td>Blue Mountains Cycling Company</td>
<td></td>
<td>8,287,037.18</td>
<td>69</td>
<td>1,707,644.23</td>
<td></td>
<td>10,954,780.41</td>
</tr>
<tr>
<td>Can't Beat The Bush Supplies</td>
<td></td>
<td>44,577.27</td>
<td></td>
<td></td>
<td></td>
<td>1,049,035.84</td>
</tr>
<tr>
<td>Gate Bluff Pty.</td>
<td></td>
<td>318,222.29</td>
<td>27,493.88</td>
<td>110,789.47</td>
<td>310,539.01</td>
<td>2,155,243.84</td>
</tr>
<tr>
<td>Harbour Pty Ltd.</td>
<td></td>
<td>49,467.26</td>
<td>417,913.2</td>
<td>1,312,644.02</td>
<td></td>
<td>1,893,404.48</td>
</tr>
<tr>
<td>Jackals Trading Co.</td>
<td></td>
<td>114,726.36</td>
<td>75,004.83</td>
<td>5,711,060.21</td>
<td></td>
<td>6,519,790.58</td>
</tr>
<tr>
<td>Kangaroo Camper</td>
<td></td>
<td>235,156.62</td>
<td>7,765,870.22</td>
<td>11,535,544.63</td>
<td>8,594,327.64</td>
<td>28,318,066.91</td>
</tr>
<tr>
<td>Outback Pty</td>
<td></td>
<td>112,841.24</td>
<td>3,032,847.67</td>
<td>2,528,262.2</td>
<td>4,031,208.43</td>
<td>6,975,239.44</td>
</tr>
<tr>
<td>Southern Cross Pty.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7,589,488.86</td>
<td>13,681.08</td>
</tr>
<tr>
<td>Too End Equipment</td>
<td></td>
<td>24,596.64</td>
<td>111,254.47</td>
<td>230,314.07</td>
<td>1,678,231.65</td>
<td>2,153,066.83</td>
</tr>
<tr>
<td>Western's Golf Supplies</td>
<td></td>
<td>3,105.72</td>
<td>0</td>
<td>1,006,992.96</td>
<td></td>
<td>4,906,032.12</td>
</tr>
</tbody>
</table>

3. Click Edit in authoring.

Task 2. Add parameters to the report.

You will add parameters so that this report will only display data for the Country or Countries and the Product line from which the users drill through.

1. Click the container handle for the crosstab.
2. On the toolbar, click Filters, click Edit Filters, and then click Add.
3. Create, and then validate, the following expression:
   [Retailer country] in ?Retailer country?
   Hint: Drag Retailer country from the Data Items tab.
   Validate using Australia.

4. Click **OK** to close the **Detail filter expression** dialog box.

5. Click **Add**.

6. Create and validate the following expression:
   [Product line] = ?Product line?
   Hint: Drag Product line from the Data Items tab.
   Validate using Camping Equipment.

7. Click **OK** to close the **Detail filter expression** dialog box, and then click **OK** to close the **Filters** dialog box.

**Task 3. Make prompt selections and save the report.**

1. Run the report in **HTML**.

2. For **Product line**, select **Camping Equipment**, for **Retailer country**, select **Australia**, and then click **OK**.

A section of the results appear as follows:

3. Close the rendered report tab.

4. Click the down arrow next to **Save**.

5. Click **Save as**, and then click **My content**.

6. In the **Save as** box, type **Total Revenue by Country and Product line**.

7. Click **Save**.

8. From the drop-down menu in the center of the **Application** bar, click **Welcome**.
Task 4. Create a drill-through definition for the GO Data Warehouse (query) package.

You will create a drill-through definition so that users can drill through to this report from reports that were created with the GO Data Warehouse (query) package.

1. At the left, click New, and then click Other.
2. In the Companion applications list, click Drill-Through Definitions.
3. Click Legacy Samples > Samples_Drillthrough > Models.
4. Click GO Data Warehouse (query), and then from the top right corner of the interface, click New Drill-through Definition from the toolbar.
5. In the Name box, type Total Revenue by Country Definition, and then click Next.

It is important to create a logical name for each drill-through definition that describes the contents of the drill-through target report. This helps organize drill-through definitions.

Your target report has a Product line parameter; therefore, you want to limit the scope of this drill-through definition so that users can drill through only from cells in source reports that have Product line as their context.

6. Click Set the scope.
7. Navigate to Sales and Marketing (query) > Sales (query) > Products.
8. Click Product line, and then click OK.
9. Click Set the target.
10. In the navigation path at the top, click Cognos, and then click My Folders.
11. Click the **Total Revenue by Country and Product line** radio button, and then click **OK**.

![Specify a scope and target - New Drill-through Definition wizard](image)

12. Click **Next**.

The Specify the target details page appears, and displays the Product line and Retailer country parameters from the Total Revenue by Country and Product line report, as follows:

![Parameter mapping](image)

13. For the **Product line** parameter, in the **Source metadata item** column, click **Set the value for Product line**.

14. Navigate to **Sales and Marketing (query) > Sales (query) > Products**.

15. Click **Product line**, and then click **OK**.

16. For the **Retailer country** parameter, in the **Source metadata item** column, click **Set the value for Retailer country**.

17. Navigate to **Sales and Marketing (query) > Sales (query) > Retailers**.
18. Click Retailer country, and then click OK.

![Map a source metadata item to each specified target parameter so values can be passed to the target when the drill through action occurs.](image)

19. Click Finish.

20. Close the Drill-through Definitions tab, and return to the Welcome screen.

**Task 5. Test the drill-through definition.**

You will test this drill-through definition by drilling through from a report created which uses the GO Data Warehouse (query) package.

1. Create a New report. The report should use the Legacy Samples > Samples_Drillthrough > Models > GO Data Warehouse (query) package.
   The report should also create a List.
2. From the Side pane, click Data.
3. Expand Sales and Marketing (query) > Sales (query) > Retailers.
4. Ensure that the List is selected.
5. Double-click Retailer country to add it to the list.
6. Expand Sales fact, and then double-click Revenue to add it to the list.
7. Click Show properties.
8. Click Show Ancestor, and then click Report.
9. In the RUNNING & VALIDATING section, change the value for Run with full interactivity to No.
10. Run the report in HTML.
    You will attempt to drill through to view more data about revenue in Australia.

**Task 6. Use the related links feature.**

1. In the report, in the Australia row, right-click the revenue cell 109,299,969.14, point to Go To, and then click Related Links.
   Note: if you click in the cell, but not on the actual value in the cell, the menu many not appear. Be sure to click the value (character) that appears in the cell.
   The drill-through definition that you created does not appear in the list of available links. This is because you limited the scope of this drill-through definition to the Product line data item. The Australia row does not have a specific product line as its context.
2. Close the Go to tab.
3. Close the Reporting tab.
4. Click the **Navigate** button at the left.
5. Click **Page1** to return to the design screen.
6. Click `<Retailer country>`, and then press the **Delete** key.

   You will add product line data to this report, and then drill through to view more data about revenue generated by the camping equipment product line.
7. Click the **Data** tab at the left.
8. Expand **Products**, and then drag **Product line** to the left of the **Revenue** column.
9. Run the report in **HTML**.

**Task 7. Testing drill through definitions.**

1. In the report, in the **Camping Equipment** row, right-click the revenue cell **1,589,036,664.03**, point to **Go To**, and then click **Related Links**.
2. Under Available links, click **Total Revenue by Country Definition**.
3. In the **Retailer country** prompt, click **Australia** and then click **OK**.

   IBM Cognos drills through to the Total Revenue by Country target report.
   The report contains data for only Australia, and for only the Camping Equipment product line.
   A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Revenue</th>
<th>Camping Equipment</th>
<th>Total(Product line)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asia Pacific</td>
<td>Australia</td>
<td>Beach Beds Pty Ltd.</td>
</tr>
<tr>
<td>Australia</td>
<td>Black Stump Camping Supplies</td>
<td>1,262,948.49</td>
</tr>
<tr>
<td>Australia</td>
<td>Can't Beat The Bush Supplies</td>
<td>1,049,058.84</td>
</tr>
<tr>
<td>Australia</td>
<td>Gone Bush Supplies</td>
<td>1,107,959.92</td>
</tr>
<tr>
<td>Australia</td>
<td>Harbour Pty Ltd.</td>
<td>1,352,640.02</td>
</tr>
<tr>
<td>Australia</td>
<td>Jackos Enviro Shop</td>
<td>5,731,660.31</td>
</tr>
<tr>
<td>Australia</td>
<td>Kanga Kampers</td>
<td>11,352,544.63</td>
</tr>
<tr>
<td>Australia</td>
<td>OutBack Pty</td>
<td>2,598,252.2</td>
</tr>
<tr>
<td>Australia</td>
<td>Southern Cross Pty</td>
<td>13,681.08</td>
</tr>
<tr>
<td>Australia</td>
<td>Top End Equipment</td>
<td>1,678,931.65</td>
</tr>
<tr>
<td><strong>Total(Retailer country)</strong></td>
<td></td>
<td><strong>41,935,932.19</strong></td>
</tr>
</tbody>
</table>

   The drill-through definition works as expected.
4. Close the **Total Revenue by Country** tab.
5. In the report, in the **Camping Equipment** row, right-click the revenue cell **1,589,036,664.03**, point to **Go To**, and then click **Related Links**.
6. Click **View passed source values**.

   The results appear as follows:

   ![Image of drill-through definition](image)

   You can see the items and values that are passed. If you want, you can click Total Revenue by Country Definition, to view the target report.

   For further troubleshooting, you can click the More link. The information found there is typically sent to customer support for investigation.

7. Click **Cancel**, and then close the **Reporting** rendered tab.

8. On the **Application** bar, click **Save**, and then navigate to **My content**.

9. In the **Save as** box, type **Revenue by Product line - Source**, and then click **Save**.

   This report will be used in the next demonstration.

10. Using the dropdown menu at the top, return to the **Welcome** screen.

### Results:

You created a drill-through definition for the GO Data Warehouse (query) package to let users drill through to a target report containing detailed information about sales of each product line. To let users focus on specific areas of interest, you set up drill-through access so that when users drill through, they will see details for only the product line they are interested in. Finally, you enabled the Drill Through Assistant and viewed the values that were passed.
Dynamic drill-through

Dynamic filtering eliminates the need for pre-authored drill-through prompts and parameters previously required for drill-through reports.

Dynamic drill-through simplifies the process of creating reports for drill through and allows administrators to create reliable drill-through definitions between any reports provided they have common items with conformed values.

You can use dynamic drill-through alone, or combine with parameterized drill-through when reports expect parameters. Non-parameterized items would be filtered dynamically, whereas the parameterized items would be predefined.

In the example, the source Analysis Studio report, based on a PowerCube, is configured to drill through to a report, based on a relational model. Both the source and the target contain an item called Country. The value from the PowerCube is converted into a string value (based on caption) which matches the string value found in the relational target, in this case, France.
Drill-through functionality also applies to PowerPlay Studio (PPS).

Note: In this release, if the drill-through target is a PowerCube, it will always open in PowerPlay Studio (not Analysis Studio). If PPS is not installed, you cannot select a PowerCube as a target for dynamic drill-through. In short PowerCubes, not just reports, can be targets when PPS is installed.
### Dynamic drill-through - matching criteria

- Dynamic drill through matches source model item names to either:
  - target model item name
  - target report data item name
- No match, source item is ignored.

First, IBM Cognos tries to match model item names. This is the most reliable match since column names may be renamed during the design process of potential target reports.

For example, if both the source and target reports had items called Product line, renaming Product line in the target report to something else would still work since the model names match.

When the model names do not match, then the target report data item name must match that of the source report.

If no match is found for the model or data item name, then the item is ignored for the drill-through.
## Demonstration 2

Configure dynamic drill-through and set measure scope

### Total Revenue by Country For Product Line

<table>
<thead>
<tr>
<th>Country</th>
<th>Golf Equipment</th>
<th>Outdoor Protection</th>
<th>Personal Accessories</th>
<th>Camping Equipment</th>
<th>Mountaineering Equipment</th>
<th>Total(Product Line)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Asia Pacific</strong></td>
<td>4,137,151.17</td>
<td>239,119.01</td>
<td>5,300,995.74</td>
<td>15,789,253.05</td>
<td>462,817.16</td>
<td>25,599,436.73</td>
</tr>
<tr>
<td>Australia</td>
<td>1,860,782.68</td>
<td>1,551.18</td>
<td>1,227,033.78</td>
<td>1,227,033.78</td>
<td>462,817.16</td>
<td>5,015,021.60</td>
</tr>
<tr>
<td>Beach Bikes Pty Ltd</td>
<td>4,137,151.17</td>
<td>239,119.01</td>
<td>5,300,995.74</td>
<td>15,789,253.05</td>
<td>462,817.16</td>
<td>25,599,436.73</td>
</tr>
<tr>
<td>Black Stump Canopy Supplies</td>
<td>7,719</td>
<td>2,464,224.32</td>
<td>1,252,946.48</td>
<td>462,817.16</td>
<td>4,197,926.92</td>
<td></td>
</tr>
<tr>
<td>Blue Mountain Golf Company</td>
<td>8,257,037.18</td>
<td>60</td>
<td>1,797,644.33</td>
<td>1,054,794.03</td>
<td>1,093,633.11</td>
<td></td>
</tr>
<tr>
<td>Can't Beat The Bush Supplies</td>
<td>44,577.27</td>
<td>1,093,633.11</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Game Bush Supplies</td>
<td>318,232.24</td>
<td>216,786.47</td>
<td>11,107,099.52</td>
<td>479,896.83</td>
<td>2,155,254.84</td>
<td></td>
</tr>
<tr>
<td>Hermiton Pty Ltd</td>
<td>69,407.28</td>
<td>471,913.2</td>
<td>1,252,946.48</td>
<td>1,093,633.11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jack's Enclave Shop</td>
<td>114,723.39</td>
<td>673,094.88</td>
<td>5,731,660.31</td>
<td>6,519,399.58</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kangaroo Campers</td>
<td>253,156.42</td>
<td>7,752,876.22</td>
<td>11,352,564.73</td>
<td>38,958,397.64</td>
<td>28,310,968.91</td>
<td></td>
</tr>
<tr>
<td>Outback Pty</td>
<td>122,841.24</td>
<td>3,082,807.57</td>
<td>2,593,232.2</td>
<td>5,785,229.64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Southern Cross Pty</td>
<td>7,399,488.86</td>
<td>13,681.98</td>
<td>7,463,554.94</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Top End Equipment</td>
<td>24,586.64</td>
<td>211,334.47</td>
<td>238,314.07</td>
<td>6,170,031.65</td>
<td>2,155,254.84</td>
<td></td>
</tr>
<tr>
<td>Weyl's Golf Supplies</td>
<td>3,135,754.6</td>
<td>3,292.96</td>
<td>1,808,993.56</td>
<td>4,968,032.12</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>China</strong></td>
<td>70,015.85</td>
<td>5,668,108.98</td>
<td>5,152,608.57</td>
<td>396,268.54</td>
<td>11,250,222.94</td>
<td></td>
</tr>
<tr>
<td>COMET/WENYI</td>
<td>127,346.03</td>
<td>4,900,094.78</td>
<td>4,272,722.56</td>
<td>1,074,171.74</td>
<td>6,779,949.13</td>
<td></td>
</tr>
<tr>
<td>CROSS/迪欧</td>
<td>352,183.86</td>
<td>310,065.63</td>
<td>3,056,594.42</td>
<td>3,548,005.69</td>
<td></td>
<td></td>
</tr>
<tr>
<td>WISS/威斯/威斯</td>
<td>303,337.08</td>
<td>3,888,404.59</td>
<td>5,889,002.82</td>
<td>574,014.05</td>
<td>10,355,406.55</td>
<td></td>
</tr>
<tr>
<td>上海展览中心</td>
<td>626,236.92</td>
<td>7,677,546.74</td>
<td>6,715,756.87</td>
<td>3,763,966.48</td>
<td>18,383,518.8</td>
<td></td>
</tr>
<tr>
<td>上海国际礼品及包装展览会</td>
<td>310,015.08</td>
<td>3,202,561.83</td>
<td>3,973,931.38</td>
<td>7,492,996.34</td>
<td></td>
<td></td>
</tr>
<tr>
<td>上海第一天国际印刷展</td>
<td>104,455.32</td>
<td>3,140,031.18</td>
<td>5,274,404.77</td>
<td>6,695,818.48</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Demonstration 2:
Configure dynamic drill-through and set measure scope

Purpose:
You have been asked to configure a package to enable dynamic drill-through from source reports based on a package to a target report that provides revenue details.

In addition, you will also set the scope of the target report to be available only if the source report contains the Revenue measure.

Note: Demonstration 1 needs to be completed before starting this demonstration.

Portal: http://vclassbase:9300/bi
User/Password: hirschb/Education1

Task 1. Examine the target report.

In this task you will open an existing report, remove its filters, and then save the report with a different name. Be sure to log in as Branka Hirsch.

1. From the left margin of the Welcome screen, click My content.
2. Click the action menu for Total Revenue by Country and Product line (you may need to hover over Total Revenue by Country and Product line).
3. Click Edit report.
   This is the crosstab report created in the previous demonstration.
4. Click the container handle for the crosstab.
5. On the toolbar, click Filters, and then click Remove all Filters.
6. Run the report in HTML.
   A section of the report appears as follows:

<p>|</p>
<table>
<thead>
<tr>
<th>Revenue</th>
<th>Golf Equipment</th>
<th>Outdoor Protection</th>
<th>Personal Accessories</th>
<th>Camping Equipment</th>
<th>Mountainering Equipment</th>
<th>Total(Revenue)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asia Pacific, Australia</td>
<td>3,186,759.6</td>
<td>1,551.16</td>
<td>1,327,677.99</td>
<td>15,788.05</td>
<td>15,788.05</td>
<td>5,015,354.54</td>
</tr>
<tr>
<td>Beach Bites Pty Ltd</td>
<td>4,137,155.17</td>
<td>293,110.01</td>
<td>5,330,905.74</td>
<td>15,788.05</td>
<td>15,788.05</td>
<td>25,549,434.97</td>
</tr>
<tr>
<td>Beach Stamps Camping Supplies</td>
<td>7,719</td>
<td>2,464,224.32</td>
<td>2,464,224.32</td>
<td>1,262,944.49</td>
<td>462,817.16</td>
<td>4,197,708.57</td>
</tr>
<tr>
<td>Bus Mountaineering Company</td>
<td>8,257,037.18</td>
<td>40</td>
<td>1,797,844.23</td>
<td>10,054,750.41</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notice that all data is returned for each of the data items on the report (region country, retailer name, and product line).
7. Click Page down to advance through the report.
8. Close the rendered report tab.
9. Click the down arrow next to the Save button, and then click Save as.
10. Click My content.
11. In the **Save as** box, type *Demonstration2_Total Revenue by Country*, and then click **Save**.
12. Return to the **Welcome** screen.

**Task 2. Create a drill-through definition.**

You will create a drill-through definition with dynamic filtering, and use it to drill through to the target report, and then you will identify a filter error.

1. From the **Welcome** screen, click **New > Other**.
2. From the **Companion applications** list, click **Drill-Through Definitions**.
3. Navigate to **Legacy Samples > Samples_Drillthrough > Cubes**.
4. Click **Sales and Marketing (cube)**.
5. In the upper right, click **New Drill-through Definition**.
6. In the **Name** box, type **Revenue Details Drill-through Definition**, and then click **Next**.
   
   For now, you will not set any scope on this drill-through definition. Users will be allowed to drill through on any level in any dimension. Later, you will restrict the scope based on a measure.
7. Click **Set the target**, and then navigate to **Cognos > My Folders**.
8. Click the **Demonstration2_Total Revenue by Country** report radio button, and then click **OK**.
9. Click **Next**, and then from the **Action** list, click **Run the report using dynamic filtering**.
   
   Notice that there are no parameters under Parameter mapping because the target report has no parameters. If the target report did have parameters, you would need to map those parameters, but could still leverage dynamic filtering on other common, non-parameterized items. In other words, you can combine the two methods if required.
10. Click **Finish**, and then close the **Drill-through Definitions** tab.

**Task 3. View passed source values and drill-through definition.**

1. From the **Welcome** screen, click **Team content**.
2. Navigate to **Legacy Samples > Samples_Drillthrough > Cubes > Sales and Marketing (cube) > Report Studio Report Samples**.
3. Click **Top Retailers by Country**.
4. When prompted to make a selection of retailers, click **Select all**, and then click **Finish**.
5. Right-click the intersection of **Netherlands/Extra Sport** (rows) and **Prior YTD/Camping Equipment** (columns)

6. Point to **Go To**, and then click **Related Links**.

7. Click **View passed source values**.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Selection context</th>
<th>Display value</th>
<th>Use value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revenue</td>
<td>40/1999-3</td>
<td>[sales_and_marketing].[Measures].[Revenue]</td>
</tr>
<tr>
<td>top 10</td>
<td>Extra Sport</td>
<td>[sales_and_marketing].[Retailers].[Retailer name] &gt; [PC]</td>
</tr>
<tr>
<td>Retailer country</td>
<td>Netherlands</td>
<td>[sales_and_marketing].[Retailers].[Retailers].[Retailer country] &gt; [PC]</td>
</tr>
<tr>
<td>Product Set</td>
<td>Camping Equipment</td>
<td>[sales_and_marketing].[Products].[Products].[Product line] &gt; [PC]</td>
</tr>
<tr>
<td>Time Set</td>
<td>Prior YTD</td>
<td>[sales_and_marketing].[Time].[Prior YTD] &gt; [PC]</td>
</tr>
</tbody>
</table>

You can see the data items used, their display values, and the values that are used to filter the target report. The Use value may be converted at run time depending on the source and target report data sources.

For example, if the source report is based on a PowerCube and the target report is based on a relational model, then the member unique name (MUN) value from the source report may be converted to a string representation of the member caption in order to conform to the relational data value.

Be aware that one of the use values being passed is Retailer name, as highlighted in the screen capture above.

8. Under **Available links**, click the **Revenue Details Drill-through Definition** link.

A section of the report appears as follows:

<table>
<thead>
<tr>
<th>Revenue</th>
<th>Camping Equipment</th>
<th>Total(Product line)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Northern Europe</td>
<td>Netherlands</td>
<td>Beter Buitenleven</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cornelius' buitensport</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Eurobal</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Extra Sport</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Revenue</th>
<th>Camping Equipment</th>
<th>Total(Product line)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Northern Europe</td>
<td>Netherlands</td>
<td>Beter Buitenleven</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cornelius' buitensport</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Eurobal</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Extra Sport</td>
</tr>
</tbody>
</table>

The source and target reports have country, retailer name and product line in common, yet the report is only filtered on country and product line.

The item names must match. In this case the source item name does not match the target item name. The use value for the item from the source report is Retailer name, as seen with the View passed source values feature. The value for the item from the target report is Retailer name (multiscript).

9. Close the **Demonstration2_Total Revenue by Country** tab.

1. Use the dropdown menu at the top to navigate back to design mode for Demonstration2_Total Revenue by Country.
   You will need to verify the item name in the target report, and retest the drill-through.

2. From the Demonstration2_Total Revenue by Country report, in the crosstab, click `<#Retailer name (multiscript)#>` to select it.
   Notice that the name does not match the item name in the cube. In the relational source it is called Retailer name (multiscript); in the cube source it is called Retailer name.

3. In the Properties pane, under DATA ITEM, change the Name property to Retailer name, and then save the report.

4. In design mode of the Top Retailers by Country source report, run the report in HTML.

5. When prompted to make a selection of retailers, click Select all, and then click Finish.

6. Right-click the intersection of Netherlands/Extra Sport (rows) and Prior YTD/Camping Equipment (columns), point to Go To, and then click Related Links.

7. Under Available links, click the Revenue Details Drill-through Definition link.
   A section of the report appears as follows:

   ![Revenue Details Drill-through Definition](image)

   The report is now filtered on country, retailer name, and product line.

8. Close the rendered report tab.

Task 5. Set measure scope.

You will now only permit drill through for this definition when the source report includes the Planned revenue measure.

1. From the Welcome screen, click New > Other.

2. Click Drill-through Definitions.

3. Navigate to Legacy Samples > Samples_Drillthrough > Cubes > Sales and Marketing (cube).

4. In the Actions column, beside Revenue Details Drill-through Definition, click Set properties.

5. Click the Target tab, and then click Set the scope.
6. Expand **Measures**, and then click **Planned revenue**.
7. Click **OK** to close the **Set the scope** dialog box, and then click **OK** to close the **Set properties** dialog box.
8. Close the **Drill-through Definitions** tab.
9. In the **Reporting** tab, right-click the intersection of **Netherlands/Extra Sport** (rows) and **Prior YTD/Camping Equipment** (columns), point to **Go To**, and then click **Related Links**.
   
   The Revenue Details link does not appear any longer because Planned revenue is not in the analysis.
10. Close the web browser.

---

**Results:**

By configuring a dynamic drill-through definition and ensuring that the common item names in the source and target reports matched, you were able to achieve a dynamic drill-through.

You also ensured that the target report would only be available if the source report contained the Revenue measure.
Unit summary

- Discuss parameter-driven drill through
- Discuss dynamic drill through
- Set up package-based drill-through definitions
- Set scope
- Use the Drill Through Assistant
Exercise 1: Configure dynamic drill-through

<table>
<thead>
<tr>
<th>Product type</th>
<th>Base product</th>
<th>Quantity</th>
<th>Return quantity</th>
<th>% Returned</th>
<th>Lost revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binoculars</td>
<td>Seeker 50</td>
<td>159,701</td>
<td>2,282</td>
<td>1.43%</td>
<td>$211,267.56</td>
</tr>
<tr>
<td><strong>Summary</strong></td>
<td></td>
<td>159,701</td>
<td>2,282</td>
<td>1.43%</td>
<td>$211,267.56</td>
</tr>
</tbody>
</table>
Exercise 1: Configure dynamic drill-through

Consumers using the Sales and Marketing (cube) package for analysis would like to review actual and planned revenue for order methods, using the Actual vs. Planned Revenue report, and be able to get details on lost revenue for specific products displayed in the report. The GO Data Warehouse (analysis) package has lost revenue information and can be used to provide the details consumers require.

A list report has been created based on the GO Data Warehouse (analysis) package that retrieves the following items: Product type, Product, Quantity, Return quantity, % Returned, and Lost revenue. The report is called Returns by Product Type and is located in the Team Content\Samples\Models\GO Data Warehouse (analysis)\Query Studio Report Samples folder. This will provide a starting point for your target report.

As Branka Hirsch, the administrator, you will create a drill-through definition called Exercise 1_Returns by Product Type Definition that allows consumers to drill through from the Sales and Marketing (cube) package to the target report. Consumers should be able to drill through any report in the package only if the Product type level is available. To accomplish this, you will need to:

- set the scope in the drill-through definition at the Product type level
- ensure that all item names match between the source report and the target report
- create parameterized drill through that will dynamically filter the target report

You will save the target report as Exercise 1_Returns by Product Type Target report, to keep the original sample report unchanged.

You will save the source report as Exercise 1_Drill-through Source_Actual vs. Planned Revenue, to keep the original sample report unchanged.

For more information about where to work and the exercise results, refer to the Tasks and Results section that follows. If you need more information to complete a task, refer to earlier demonstrations for detailed steps.
Exercise 1: Tasks and Results

Task 1. Examine the target report.

- **Browser**: Log on to IBM Cognos Analytics using the following credentials: hirschb/Education1.

- **Search**: Do a search for Returns by Product Type_DQ. (Note: If the report does not come up in the Search, browse to the following path: Team content\Legacy Samples\Sample_DQ\Models\GO Data Warehouse (analysis)\Query Studio Report Samples\)

- **Open**: Click the report to see it open in IBM Cognos Query Studio.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Product type</th>
<th>Base product</th>
<th>Quantity</th>
<th>Return quantity</th>
<th>% Returned</th>
<th>Lost revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insect Repellents</td>
<td>BugShield Lotion</td>
<td>773,324</td>
<td>81,189</td>
<td>10.50%</td>
<td>$189,170.37</td>
</tr>
<tr>
<td>Navigation</td>
<td>Trail Star</td>
<td>65,146</td>
<td>5,461</td>
<td>8.38%</td>
<td>$483,691.20</td>
</tr>
<tr>
<td>Insect Repellents</td>
<td>BugShield Lotion Lite</td>
<td>384,513</td>
<td>14,171</td>
<td>3.69%</td>
<td>$26,641.48</td>
</tr>
<tr>
<td>Sunscreen</td>
<td>Sun Shield</td>
<td>991,486</td>
<td>32,382</td>
<td>3.27%</td>
<td>$89,574.32</td>
</tr>
<tr>
<td>Lanterns</td>
<td>EverGlow Lamp</td>
<td>965,019</td>
<td>29,434</td>
<td>3.05%</td>
<td>$434,454.32</td>
</tr>
<tr>
<td>Insect Repellents</td>
<td>BugShield Extreme</td>
<td>2,666,714</td>
<td>72,255</td>
<td>2.71%</td>
<td>$174,857.10</td>
</tr>
</tbody>
</table>

Notice the first two columns are Product type and Base product. This report provides information about product returns and lost revenue.

- **Toolbar**: Save the report to My Folders, as: Exercise 1_Returns by Product Type Target.

  - From the Welcome screen, under My content, with the vertical ellipse, use **Convert to report** on Exercise 1_Returns by Product Type Target.

  - Save the report as Unit 10 Exercise 1_Returns, under My content.
Task 2. Add parameters to Unit 10 Exercise 1_Returns report.

- **Toolbar**: Create a custom filter based on **Product type**.
- **Filter condition dialog**: Select **Prompt for values when report is run in viewer**.
  - Add all values to the **Selected values** pane.
- **Filters dialog box**: Add a combined filter based on **Product**.
- **Filter condition dialog**: Select **Prompt for values when report is run in viewer**.
  - Add all values to the **Selected values** pane.
- **Toolbar**: Save the report, and then close the browser tab.

Task 3. Create the drill-through definition.

- **Welcome**: click **New > Other > Drill Through Definitions**.
- **Team content**: Navigate to **Team content\Legacy Samples\Samples_Drillthrough\Cubes\Sales and Marketing (cube)**.
- **Toolbar**: Click **New Drill-through Definition**.
- **Name box**: Name **Unit 10 Exercise 1_Returns**, and then click **Next**.
- **Set scope and target**:
  - Set the scope to **Product** from **Products**.
  - Set the target to **Unit 10 Exercise 1_Returns** (located in **My Folders**).
  - **Specify the target details**: From the **Action** list, select **Run the report using dynamic filtering**.
• Under **Parameter mapping**, click **map to metadata** for the target parameter **Product type**, expand **Products**, and then click the **Product type**.

• Under **Parameter mapping**, click **map to metadata** for the target parameter **Product**, expand **Products**, and then click the **Product**.

The results appear as follows:

![Parameter mapping screen](image)

• Click **Finish**.

• **Toolbar**: Return to the **Welcome** screen.

**Task 4. View the passed source values.**

• **Welcome**: navigate to:
  Team content\Legacy Samples\Samples_Drillthrough\Cubes\Sales and Marketing (cube)\Report Studio Report Samples.

• Run **Actual vs. Planned Revenue** report.

• **Prompt page**:
  • From the **pMethod** list: select **Web**.
  • From the **Year** list: select **2012**.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Americas</th>
<th>2012</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order method type</td>
<td>Product type</td>
</tr>
<tr>
<td>Web</td>
<td>Binoculars</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Binoculars</td>
<td></td>
</tr>
</tbody>
</table>
• Product column: Go To - Related Links, for Seeker 50.
• Go to window: Click View passed source values (right side of window).

The results appear as follows (notice Product):

<table>
<thead>
<tr>
<th>Selection context</th>
<th>Display value</th>
<th>Use value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product</td>
<td>Seeker 50</td>
<td>[sales_and_marketing],[Products],[Pro]</td>
</tr>
<tr>
<td>Order method type</td>
<td>Web</td>
<td>[sales_and_marketing],[Order method],[Pro]</td>
</tr>
<tr>
<td>Product type</td>
<td>Binoculars</td>
<td>[sales_and_marketing],[Products],[Pro]</td>
</tr>
<tr>
<td>Revenue</td>
<td>1,484,847.00</td>
<td>[sales_and_marketing],[Measures],[Re]</td>
</tr>
<tr>
<td>Planned revenue</td>
<td>1,563,052.05</td>
<td>[sales_and_marketing],[Measures],[Pl]</td>
</tr>
<tr>
<td>Difference</td>
<td>78,205.05</td>
<td>78205.05</td>
</tr>
<tr>
<td>Year</td>
<td>2012</td>
<td>[sales_and_marketing],[Time],[Time]</td>
</tr>
</tbody>
</table>

Task 5. Fix and test source report.

• Welcome: Open the Actual vs. Planned Revenue report (under Team content\Legacy Samples\Samples_Drillthrough\Cubes\Sales and Marketing (cube)\Report Studio Report Samples)
• File menu: click Edit in authoring.
• File menu: In My content, save the report as Unit 10 Exercise 1_Drill
• Work area: Change Name property of Product column header to Base product.
• Toolbar: Run the report.
• Prompt page:
  • From the pMethod list: select Web.
  • From the Year list: select 2012.
• Product column: Go To - Related Links, for Seeker 50.
• Go to window: Click View passed source values.

The results appear as follows (notice Base product):

<table>
<thead>
<tr>
<th>Selection context</th>
<th>Display value</th>
<th>Use value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base product</td>
<td>Seeker 50</td>
<td>[sales_and_marketing],[Products],[Pro]</td>
</tr>
<tr>
<td>Order method type</td>
<td>Web</td>
<td>[sales_and_marketing],[Order method],[Pro]</td>
</tr>
<tr>
<td>Product type</td>
<td>Binoculars</td>
<td>[sales_and_marketing],[Products],[Pro]</td>
</tr>
<tr>
<td>Revenue</td>
<td>1,484,847.00</td>
<td>[sales_and_marketing],[Measures],[Re]</td>
</tr>
<tr>
<td>Planned revenue</td>
<td>1,563,052.05</td>
<td>[sales_and_marketing],[Measures],[Pl]</td>
</tr>
<tr>
<td>Difference</td>
<td>78,205.05</td>
<td>78205.05</td>
</tr>
<tr>
<td>Year</td>
<td>2012</td>
<td>[sales_and_marketing],[Time],[Time]</td>
</tr>
</tbody>
</table>
• **Available links**: click **Unit 10 Exercise 1_Returns**.

The results appear as follows:

![Returns by Product Type](image)

Because the item names now match at the Product level, the report filters as expected, and the lost revenue details are displayed for the product you selected.

- Close the rendered report tab.
- Sign out of **IBM Cognos Analytics**.
- Close all browser windows.
Unit 11  Enhance report layout

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2016
Course materials may not be reproduced in whole or in part without the written permission of IBM.
Unit objectives

- Force page breaks in reports
- Modify existing report structures
- Apply horizontal formatting
- Specify print options for PDF reports
- Format data and report objects
View the structure of the report

- To view your report in a different way and see how objects are organized, view the page structure.

View the page structure to:

- view the entire contents of a report page in a tree structure
- move objects quickly from one area of a page to another
- modify object properties
- view the page structure, on the toolbar, click View, and then click Page Structure. A tree structure is useful for locating the objects in a page and troubleshooting problems with nested objects.
- view a complex layout, it may be difficult to select, cut, and paste objects in the layout view. Objects are easier to locate in the page structure view. This view can also be helpful if you want to modify an object but are not sure where the object is located within the report structure. Once you know where an object it placed, you can select it and modify its properties.

Objects can be changed in either view, depending on your preference. For example, you can group and sort list columns in the page structure view. Any changes made in the page structure view will also be visible in the page design view.
Force page breaks in reports

- Page sets let you associate report pages with a query structure to force page breaks.

To force page breaks based on a data item, you must associate the page set with a query and then define a grouping structure for the page set.

You can add multiple detail pages to a page set.

You can also create nested page sets, and can define a master-detail relationship between them to see data in the nested page set that is related to the data in the parent page set. For example, you have a page set that shows pages of product line information. The page set contains a nested page set that shows pages of product type information.

You can use a page set to create a report that contains detail pages displaying data for each order method. Each order method type will begin on a new page.

In the slide example, Page Set1 has been grouped by Region. This page set will begin a new list for each region.
Demonstration 1:
Create a report structured on data items

<table>
<thead>
<tr>
<th>Country</th>
<th>Employee name</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Americas</strong></td>
<td></td>
</tr>
<tr>
<td>Brazil</td>
<td>Alexandre Pereira</td>
<td>34,720,977.7</td>
</tr>
<tr>
<td></td>
<td>Beatrix Couto</td>
<td>3,842,910.29</td>
</tr>
<tr>
<td></td>
<td>Eduardo Guimarães</td>
<td>48,839,028.63</td>
</tr>
<tr>
<td></td>
<td>Morela Castro</td>
<td>3,131,988.79</td>
</tr>
<tr>
<td>Canada</td>
<td>Brendon Pike</td>
<td>24,827,214.69</td>
</tr>
<tr>
<td></td>
<td>Carole Claudel</td>
<td>15,723,893.35</td>
</tr>
</tbody>
</table>

Demonstration 1: Create a report structured on data items
Demonstration 1: 
Create a report structured on data items

Purpose:
You have been asked to create a report showing sales rep revenues generated in each region with each sales region on a separate page. You will need to design a title page for the report and make changes to the report using the structure view.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Template: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the list.
1. Open a new List template, using the GO data warehouse (query) package.
2. From the Source tab, add the following query items to the new list data container:
   - Employee by region: Branch region, Country, Employee name
   - Sales fact: Revenue

<table>
<thead>
<tr>
<th>Branch region</th>
<th>Country</th>
<th>Employee name</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Branch region&gt;</td>
<td>&lt;Country&gt;</td>
<td>&lt;Employee name&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Branch region&gt;</td>
<td>&lt;Country&gt;</td>
<td>&lt;Employee name&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Branch region&gt;</td>
<td>&lt;Country&gt;</td>
<td>&lt;Employee name&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
</tbody>
</table>

3. Click <Branch region>, Ctrl-click <Country>, and then on the toolbar, click Group / Ungroup.
4. Click <Branch region>, on the toolbar, click Headers & footers, and then click Create header.
5. With <Branch region>column still selected, on the toolbar, click Delete to remove the redundant column.
6. Run the report in **HTML**.
A section of the results appear as follows:

![Table of results](image)

7. Click **Page down**.
Multiple sales regions are displayed on the same page. You want each sales region to display on a different page.

8. Close the rendered report tab.

**Task 2. Add page sets to the report.**
1. On the **Navigate** tab, click **Report pages**.
2. From the **Toolbox**, double-click **Page set** to add it to the **Report pages** pane.
3. In the **Properties** pane, under **DATA**, in the **Query** list, select **Query1**.
   This will associate the query to the page set.

**Task 3. Define the grouping structure for the page set.**
1. In the **Properties** pane, under **DATA**, double-click **Grouping and sorting**.
2. From the **Data items** pane, drag **Branch region** to the **Groups** folder, and then click **OK**.
3. In the **Report pages** pane, drag **Page1** onto the **Detail Pages** folder.

4. Run the report in **HTML** and then click **Page down** to examine multiple pages.
The different sales regions are now on separate pages.
5. Close the rendered report tab.
Task 4. Add a cover page title to the report.

2. In the Properties pane, under MISCELLANEOUS, in the Name property, type Cover, and then press Enter.
3. In the Report pages pane, double-click Cover.
4. From the Toolbox tab, drag a Table onto the work area, with a size of 1 column and 3 rows.
5. Ctrl-click each of the table cells, and then from the toolbar, click Center.
6. Drag a Text Item into the top table cell, naming it Total Revenue by Sales Representatives, and then click OK.
7. Drag a Text Item into the bottom table cell, naming it Sales Report, and then click OK.
8. Click only (not the cell) Total Revenue by Sales Representative, and then Ctrl-click Sales Report.
9. On the toolbar, change the font to Arial Black, the font size to 20pt, and the foreground color to Navy.

Task 5. Add an image to the cover page.

1. From the Toolbox tab, expand LAYOUT.
2. Drag an Image object to the middle table cell, click the Image, and then in the Properties pane, under URL SOURCE, double-click the URL property.
3. In the Image URL dialog box, click Browse, select cover1.jpg, and then click OK to close the Image Brower dialog box.
4. Click OK to close the Image URL dialog box.
5. Click the work area and then, from the toolbar, click Middle.
6. Run the report in **HTML**.
The results appear as follows:

![Total Revenue by Sales Representatives]

You now have a cover page to your report.

7. **Page down** to see the details in the report.

8. Close the rendered report page.

**Task 6. View the report structure and make changes to the report using the structure view.**

1. On the **Navigate** tab, click **Page 1**.

2. From the **Application** bar, click **Page views** 🎧, and then click **Page structure**.
   All the objects of the report can be reviewed in a tree structure. Here you can quickly move and modify objects within the page of the report.

3. Expand **Page - Page1**.
   The page header, page body, and page footer of the report page are displayed.
   You want to view the structure of your list and quickly modify the format of all the list column titles in the report.

4. Expand **Page body**, and then expand **List**.
5. Click **List columns title style**. The results appear as follows:

6. In the **Properties** pane, under **FONT & TEXT**, double-click the **Font** property.
7. Change the font to **Arial Black**, **12pt**, **Italic**, and then click **OK**.
8. Click **Page views**, and then click **Page design**. The list column titles are changed to reflect the modifications you made to the page structure.

9. Run the report in **HTML**.
10. Click **Page down** to view the other pages of the report that now reflect the formatting you implemented.

A section of the results appear as follows:

```
<table>
<thead>
<tr>
<th>Country</th>
<th>Employee name</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Americas</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Alexandre Pereira</td>
<td>34,720,977.7</td>
</tr>
<tr>
<td></td>
<td>Beatriz Couto</td>
<td>3,842,910.29</td>
</tr>
<tr>
<td></td>
<td>Eduardo Guimarães</td>
<td>48,839,028.63</td>
</tr>
<tr>
<td></td>
<td>Morela Castro</td>
<td>3,131,988.79</td>
</tr>
<tr>
<td>Canada</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Brendon Pike</td>
<td>24,827,214.69</td>
</tr>
<tr>
<td></td>
<td>Carole Claudel</td>
<td>15,728,893.35</td>
</tr>
</tbody>
</table>
```

11. Close the rendered report tab.

12. Leave the report authoring tab open for the next demonstration.

**Results:**
You created a report showing sales rep revenues generated in each region with each sales region on a separate page. You created and designed a title page for the report. You also made changes to the report using the structure view.
Horizontal pagination

Horizontal Pagination enables you to span wide reports across multiple PDF pages with the appropriate page number.

You can only use horizontal pagination with list and crosstab reports.
Add horizontal page numbers

- There are three options for adding horizontal page numbers:
  - Preset (using Number Style)
  - Custom Number Style
  - Report Layout Functions

Add an existing page number from the preset list of styles.

By using the custom style option, you can create your own page number style.

Use a Layout Calculation from the toolbox tab to create an expression that determines a page number style.
## Demonstration 2

Format a report for horizontal viewing

<table>
<thead>
<tr>
<th>Date</th>
<th>Outdoor Protection</th>
<th>First Aid</th>
<th>Aloe Relief</th>
<th>2012</th>
<th>Telephone</th>
<th>1001</th>
<th>3.23</th>
<th>37.32</th>
<th>104.43</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mar 5, 2012</td>
<td>Outdoor Protection</td>
<td>First Aid</td>
<td>Aloe Relief</td>
<td>2012</td>
<td>Web</td>
<td>200</td>
<td>1.92</td>
<td>3.23</td>
<td>37.32</td>
</tr>
<tr>
<td></td>
<td>Outdoor Protection</td>
<td>First Aid</td>
<td>Aloe Relief</td>
<td>2013</td>
<td>E-mail</td>
<td>238</td>
<td>1.92</td>
<td>3.23</td>
<td>37.32</td>
</tr>
<tr>
<td></td>
<td>Outdoor Protection</td>
<td>First Aid</td>
<td>Aloe Relief</td>
<td>2013</td>
<td>Fax</td>
<td>7</td>
<td>1.92</td>
<td>3.23</td>
<td>37.32</td>
</tr>
<tr>
<td></td>
<td>Outdoor Protection</td>
<td>First Aid</td>
<td>Aloe Relief</td>
<td>2013</td>
<td>Sales visit</td>
<td>311</td>
<td>1.92</td>
<td>3.23</td>
<td>37.32</td>
</tr>
<tr>
<td></td>
<td>Outdoor Protection</td>
<td>First Aid</td>
<td>Aloe Relief</td>
<td>2013</td>
<td>Telephone</td>
<td>157</td>
<td>1.92</td>
<td>3.23</td>
<td>37.32</td>
</tr>
<tr>
<td></td>
<td>Outdoor Protection</td>
<td>First Aid</td>
<td>Aloe Relief</td>
<td>2013</td>
<td>Web</td>
<td>9226</td>
<td>1.92</td>
<td>3.23</td>
<td>37.32</td>
</tr>
</tbody>
</table>
Demonstration 2: Format a report for horizontal viewing

Purpose:
Management has asked you to create a list report that fits on one page. They have also asked to create a report with certain columns that repeat on each page. You will also need to apply different page numbering formats using horizontal page numbering and report layout functions.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Template: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the list.
1. Open a new List template using the GO data warehouse (query) package.
2. From the Source tab, add the following query items to the new list report object:
   - Products: Product line, Product type, and Product
   - Time: Year
   - Order method: Order method type
   - Sales fact: Quantity, Unit cost, Unit price, Revenue, Gross profit, Product cost, Planned revenue
   - Sales (query): Gross margin

3. Run the report in PDF.
   Notice that the list is too wide for one page, so it is split across two pages.
4. Close the rendered report tab.
Task 2. Fit the list to the page width.

1. Click the list Container Selector.
2. In the Properties pane, under GENERAL, click Pagination, and then click the ellipsis.

Notice that Allow horizontal pagination is selected. This ensures that new reports allow horizontal pagination.

The results appear as follows:

3. Clear Allow horizontal pagination, and then click OK.
4. Run the report in PDF.

Notice that the entire list displays across the width of the page, but there is more horizontal scrolling.

5. Close the rendered report tab.
6. On the Application bar, click Undo to once again enable the Allow horizontal pagination property.

Task 3. Repeat columns on multiple pages.

When a report is split across pages, it is useful to repeat columns to carry context across. The author determines which list columns repeat. In this report, you will repeat Product name, Year, and Order method type.

1. Click Page views, and then click Page structure.
2. Expand Page - Page1, expand Page body, expand List, and then expand List columns.
3. Click <Product>, and then Ctrl-click <Year> and <Order method type>.
4. In the Properties pane, under GENERAL, click Pagination, and then click the ellipsis.
5. Select **Repeat every page**, and then click **OK**.  
This can only be set in the Page Structure view.

6. Run the report in **PDF**.

7. Scroll down to **Page 2**.

The Product, Year, and Order method type columns repeat on each page and provide the reader with sufficient content to understand the report. In a list, you can repeat any column.

8. Close the rendered report tab.

9. Click **Page views**, and then click **Page design**.

**Task 4. Add horizontal page numbering.**

Currently, the page numbering in your report is 1, 2, 3, and so on. Reporting provides three methods to create the page numbering: preset, report layout functions, and custom number style.

1. To use a preset number, locate the **Page number** object in the page footer.

2. Double-click the **Page number** object, select 1A, click **OK**, and then run the report in **PDF**.

3. Scroll down, to view several of the page numbers in the footer. The pages are now numbered: 1A, 1B, 2A, 2B, and so on.

4. Close the rendered report tab.

5. To create a number using **Custom number style**, double-click the **Page number** object in the footer, and then click **Edit**.

6. In the **Custom number style** dialog, incorporate the following formatting:

- under **Vertical**, add **3 periods** (…) in the box to the right of **Number**.
- under **Horizontal**, add a **left square bracket** ([ ) in the box to the left of **Letter (upper case)**, and then add a **right square bracket** ( ] ) in the box to the right of **Letter (upper case)**.

The results appear as follows:
7. Click OK to close all dialog boxes.
8. Run the report in PDF.
   The custom page number style is applied.
9. Close the rendered report tab.

Task 5. **Create numbers using report layout functions (optional).**

1. To create a number using report layout functions, delete the Page number object in the footer.
2. From the Toolbox tab, expand TEXTUAL, and then drag a Layout calculation to the middle cell of the footer.
3. Create and validate the following expression:
   
   ```
   if (HorizontalPageCount() = 1) then (number2string(PageNumber())) else (number2string(ceiling(PageNumber()/HorizontalPageCount()))) + '...(number2string(mapNumberToLetter('A', HorizontalPageNumber()-1)+ ''))
   ```
   
   Hint: HorizontalPageCount(), PageNumber(), and mapNumberToLetter are found under the Functions tab/Report Functions folder; number2string is found in the Data Type Casting Functions, under the Report Functions folder.
4. Click OK, and then run the report in PDF.
   The page numbers are now 1…(A), 1…(B), 2…(A), 2…(B), and so on.
5. Close the rendered report tab.
6. Leave the report authoring tab open for the next demonstration.

**Results:**
You created a list report with columns too wide for one page and then modified it to fit on one page. You modified a report so that certain columns repeated on each page. You also applied different page numbering formats using horizontal page numbering and report layout functions.
Modify structures

By unlocking the cells, you can add multiple items to a single column to tailor a report to your needs.

Unlock cells to manipulate an object’s contents. Once they are unlocked, you can change the text and add objects inside existing objects. This feature is useful for displaying related information in a single column, or for renaming a column. You can add additional rows to a list report to add extra information. You can add additional rows to a list report using the Structure menu.

Once a new row is added you can merge the cells by selecting one or more cells and then click the Merge cells button.
Demonstration 3: Create a condensed list report

<table>
<thead>
<tr>
<th>City</th>
<th>Employee name</th>
<th>Position name</th>
<th>Contact Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Australia</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Melbourne</td>
<td>Thomas Faraday</td>
<td>Warehouse Worker</td>
<td>Work phone: +(61) 03 2982 4242 ext. 8223 Email: <a href="mailto:TFaraday@grtd123.com">TFaraday@grtd123.com</a> Date hired: Feb 16, 2011</td>
</tr>
<tr>
<td>Melbourne</td>
<td>Andrea Samuel</td>
<td>Payroll Clerk</td>
<td>Work phone: +(61) 03 2982 4242 ext. 8224 Email: <a href="mailto:ASamuels@grtd123.com">ASamuels@grtd123.com</a> Date hired: Apr 25, 2011</td>
</tr>
<tr>
<td>Melbourne</td>
<td>Cindy Sandles</td>
<td>Product Technician</td>
<td>Work phone: +(61) 03 2982 4242 ext. 8225 Email: <a href="mailto:CSandles@grtd123.com">CSandles@grtd123.com</a> Date hired: Apr 11, 2011</td>
</tr>
<tr>
<td>Melbourne</td>
<td>Jonathan Farrell</td>
<td>Level 1 Sales Rep</td>
<td>Work phone: +(61) 03 2982 4242 ext. 8226 Email: <a href="mailto:JFarrel@grtd123.com">JFarrel@grtd123.com</a> Date hired: Feb 2, 2013</td>
</tr>
<tr>
<td>Melbourne</td>
<td>Ken Wilson</td>
<td>Customer Service Rep</td>
<td>Work phone: +(61) 03 2982 4242 ext. 8227 Email: <a href="mailto:KWilson@grtd123.com">KWilson@grtd123.com</a> Date hired: Apr 19, 2011</td>
</tr>
</tbody>
</table>
Demonstration 3: Create a condensed list report

Purpose:
The Human Resources department has requested a list of detailed sales rep information for each city. To reduce the number of columns in the report, you will combine information into one column.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Template: List
Folder: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Create the list.

1. Open a new List template using GO data warehouse (query) package.
2. From the Source tab, add the following query items to the new list report object:
   - Employee by region: Country, City, Employee name, Position name, Work phone, Extension, Email, Date hired.

3. Click <Country>, and then on the toolbar, click Group / Ungroup.
4. With <Country> selected, on the toolbar click Headers and footers, and then click Create header.
5. On the toolbar, click Delete to remove the redundant <Country>.
6. Click the `<Country>` header, and then click **Center**.

   The results appear as follows:

   ![Table example](image)

7. Run the report in **HTML**.

   The report data spreads out across the page. You want to condense it so that some of the data appears in a single column.

8. Close the rendered report tab.

**Task 2. Unlock cells and condense report data.**

1. On the **Application** bar, click **More**, and then click **Locked** to unlock the cells of the report.

   Once cells are unlocked, query items can be added directly from the Content tabs.

2. From the **Toolbox**, drag a **Table** to the **Work phone** column, to the right of `<Work phone>`.

3. Set the table with 1 column and 3 rows, and then click **OK**.

   A section of the results appear as follows:

   ![Table example](image)

4. From the list data container, drag `<Work phone>` into the first row of the table, drag `<Email>` into the second row, and then `<Date hired>` into the third row.
5. Drag `<Extension>` into the top table cell, to the right of `<Work phone>`.
The results appear as follows:

<table>
<thead>
<tr>
<th>City</th>
<th>Employee name</th>
<th>Position name</th>
<th>Work phone</th>
<th>Extension</th>
<th>Email</th>
<th>Date hired</th>
</tr>
</thead>
<tbody>
<tr>
<td>City</td>
<td>Employee name</td>
<td>Position name</td>
<td>Work phone</td>
<td>Extension</td>
<td>Email</td>
<td>Date hired</td>
</tr>
<tr>
<td>City</td>
<td>Employee name</td>
<td>Position name</td>
<td>Work phone</td>
<td>Extension</td>
<td>Email</td>
<td>Date hired</td>
</tr>
</tbody>
</table>

You can add text in front of the data items, to identify them.

6. From the **Toolbox**, drag a **Text item** to the left `<Work phone>`.
7. In the **Text** box type **Work phone:**, press the spacebar, and then click **OK**.
8. Repeat steps 6 and 7 to add the following text items to the left of `<Email>` and `<Date hired>` respectively: **Email:** and **Date hired:**.
9. Drag a **Text item** between `<Work phone>` and `<Extension>`, press the spacebar, type `ext.`, press the spacebar, and then click **OK**.

A section of the results appear as follows:

<table>
<thead>
<tr>
<th>Work phone:</th>
<th>Work phone: ext.</th>
<th>Extension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Email:</td>
<td>Email:</td>
<td></td>
</tr>
<tr>
<td>Date hired:</td>
<td>Date hired:</td>
<td></td>
</tr>
</tbody>
</table>

10. Click the list Container Selector, from the Application toolbar click **More**, and then click **Unlocked** to lock the cells of the report,
11. Click the **Extension** column header, Ctrl-click the **Email** and **Date hired** column headers, and then on the toolbar, click **Delete**.
12. In the list column, click the **Work phone** list column title, in the **Properties** pane, under **DATA ITEM**, in the **Label** box, type **Contact Information**, and then press **Enter**.
13. Run the report as **HTML**.

All key contact information is consolidated under a single column.
15. Leave the report authoring tab open for the next demonstration.

**Results:**
You created a list of detailed sales rep information in each city. To reduce the number of columns in the report, you combined information in one column.
Change PDF page orientation to suit report objects

You can set the page orientation and size for each page in the report independently.
Set PDF security options

• You can secure PDF reports when you run the report with options.
• You can set a password to secure the document.
Demonstration 4: Change a PDF page from portrait to landscape orientation
Demonstration 4: Change a PDF page from portrait to landscape orientation

Purpose:
You have been asked to build a PDF report that contains a list report and a crosstab report. You will use PDF Page Setup properties to display individual report pages as portrait or landscape. You will then create a secured version of the report.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1
Package: Team content\Samples\Models\GO data warehouse (query)
Report Type: List
Folder: Sales and Marketing (query)

Task 1. Create the list.
1. Open a new List template using the GO data warehouse (query) package.
2. From the Source tab, add the following query items to the new list report object:
   - Sales order: Order number
   - Retailers: Retailer name
   - Time: Year
   - Sales fact: Revenue

<table>
<thead>
<tr>
<th>Order number</th>
<th>Retailer name</th>
<th>Year</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Order number&gt;</td>
<td>&lt;Retailer name&gt;</td>
<td>&lt;Year&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Order number&gt;</td>
<td>&lt;Retailer name&gt;</td>
<td>&lt;Year&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
<tr>
<td>&lt;Order number&gt;</td>
<td>&lt;Retailer name&gt;</td>
<td>&lt;Year&gt;</td>
<td>&lt;Revenue&gt;</td>
</tr>
</tbody>
</table>

3. Double-click the text in the header to edit, change the current text to Revenue by Order Number, and then click OK.
4. Left justify the text within its block.
   You will make a copy of this page so that you have a page header and footer on the second page.
6. In the Report pages pane, right-click Page1, and then click Copy.
7. Right-click the white space below Page1, and then click Paste.
8. Double-click the new Page2.
9. Click the list Container Selector, and then from the toolbar, click Delete.
Task 2. Create the crosstab.

1. On the work area, click Add, click Crosstab, and then click OK to close the Object and query name dialog box and accept the defaults.
2. From the Data/Source tab, add the following query items to the new crosstab:
   - **Rows** area:
     - Time: Year
   - **Columns** area:
     - Products: Product line
       - Nested under Product line
     - Sales fact: Revenue, Quantity
     - Gross margin

3. Click the <#Gross margin#> fact cells, and then in the Properties pane, under DATA, double-click Data format.
4. From the Format type list, select Percent, in the Properties pane, click Number of decimal places, and then select 2 from the list.
5. Click OK.
6. Click the <#Revenue #> fact cells, and then in the Properties pane, under DATA, double-click Data format.
7. From the Format type list, select Number, in the Properties pane, click Number of decimal places, and then select 0.
8. Click OK.
9. Click the list, on the Properties header, click Select Ancestor, and then click Report.
10. In the Properties pane, under REPORT, double-click PDF page setup. The Orientation is set to Portrait by default. This is the setting for the entire report.
11. Click OK, and then run the report in PDF.
12. Scroll down to the last two pages. The crosstab gets split across two pages, because the page is not wide enough to display all data together, when portrait page orientation is used.
13. Close the rendered report tab.
Task 3. Change the page orientation from portrait to landscape.

1. On Page2, click anywhere below the crosstab to select the entire page.
2. On the Properties pane header, click Select Ancestor, and then click the Page object.
3. In the Properties pane, under GENERAL, double-click the PDF page setup property.
4. Select Override the page setup for this page, click Landscape, and then click OK.
5. Click Page views, and then click Page structure.
6. Expand Page - Page2, then Page body, and then click Crosstab to select it.
7. From the Properties pane, under GENERAL, double-click Pagination.
8. Clear Allow horizontal pagination, and then click OK.
9. Run the report as PDF.
10. Scroll down to the last page.
    The crosstabs now fit on a single page. You can vary the orientation by page.
11. Close the rendered report tab.
12. Save the report to My content, naming it Demonstration 4_Enhance Report Layout.
    This report will be used in the next tasks, as well as in Demonstration 5.

Task 4. Explore an unsecured PDF version of the report.

In this task, you will open and copy some text from a PDF report and paste it into a Notepad document to prove that there is no security on the PDF document. In the next task, you will create a PDF output which will require a password to open.

1. From the Welcome to IBM Cognos Analytics screen, click My content.
2. For Demonstration 4_Enhanced Report Layout, click the vertical ellipsis icon to the right.
3. Click Run as, and then enable Run in background (checkmark appears).
4. In the list of file options, select PDF, and then clear HTML.
5. Click the Advanced header, and then click > next to Delivery.
6. Verify Save the report on the system is selected.
7. Click Done, and then click Run.
8. Wait about five minutes for the report to be generated.
Task 5. Opening the PDF report.
1. From the Welcome screen, click My content, and then click the vertical ellipsis icon to the right of the Demonstration 4_Enhanced Report Layout.
2. Click View Versions. You should see a timestamp from about five minutes ago.
3. Click the most recent timestamp. You should see the symbol PDF.
4. Click the PDF to open it.
5. Once the PDF is visible, click and drag the cursor over the report to highlight some of the text, right-click the highlighted text, and then click Copy to copy the text to the clipboard.
6. From the Start menu, click All Programs, click Accessories, and then click Notepad.
7. Paste the text into the Notepad document.
This proves that you can copy the content from the PDF document into another document.
8. Close Notepad without saving the document, and then go back to IBM Cognos Analytics.

1. From the Welcome screen of IBM Cognos Analytics, click My Content.
2. Click the vertical ellipsis icon at the right of Demonstration 4_Enhance Report Layout.
3. Click Run as, and then click the switch for Run in background to enable it.
4. Select the PDF checkbox, and then clear the HTML checkbox.
5. Click Advanced and then click the > to the right of PDF.
6. Select the checkbox for Requires a password to open the report.
7. In the Password and Confirm Password textboxes, type Education1
8. Click the < Back tab at the top, and then click Run.
9. Wait about five minutes for the report to be generated.

Task 7. A secured PDF report.
1. From the Welcome screen, click My content.
2. Click the vertical ellipsis icon at the right of Demonstration 4_Enhance Report Layout.
3. Click View versions.
4. Open the most recent time stamp and then open the PDF.
5. When the PDF appears, you are prompted to type a password because you secured this version of the report.
6. Type the password **Education1**, and then click **OK**.
7. Close the rendered tab.

**Results:**
You created a PDF report that contains a list report and a crosstab report. You used PDF Page Setup properties to display individual report pages as portrait or landscape. You then created a secured version of the report that required a password to access.
Format objects across a report

- Format reports quickly and consistently using Cascading Style Sheet (CSS) classes.

CSS classes are used in reports and templates.

To determine what class an object uses, select the object and view the Classes property. An object also inherits the classes set for its parent objects.
Demonstration 5

Format objects across a report (optional)

<table>
<thead>
<tr>
<th>Order number</th>
<th>Retailer</th>
<th>Year</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>100001</td>
<td>Kavanagh Sports</td>
<td>2010</td>
<td>18,036.24</td>
</tr>
<tr>
<td>100002</td>
<td>Ar fresco</td>
<td>2010</td>
<td>58,826.44</td>
</tr>
<tr>
<td>100003</td>
<td>Universo Aconcango</td>
<td>2010</td>
<td>41,255.35</td>
</tr>
<tr>
<td>100004</td>
<td>Ao ar livros</td>
<td>2010</td>
<td>228,447.45</td>
</tr>
<tr>
<td>100005</td>
<td>Galáctica do esporte</td>
<td>2010</td>
<td>71,237.12</td>
</tr>
<tr>
<td>100090</td>
<td>Mundo saudável</td>
<td>2010</td>
<td>35,015.7</td>
</tr>
<tr>
<td>100007</td>
<td>Tomarock Outfitter Rentals</td>
<td>2010</td>
<td>94,859.1</td>
</tr>
<tr>
<td>100085</td>
<td>Husky Outfitters</td>
<td>2010</td>
<td>833,586.64</td>
</tr>
<tr>
<td>100099</td>
<td>Sporting Goods Direct</td>
<td>2010</td>
<td>149,054.11</td>
</tr>
<tr>
<td>100010</td>
<td>Game On! Sports</td>
<td>2010</td>
<td>296,228.09</td>
</tr>
</tbody>
</table>

This is an optional demonstration; however, to complete it, you must first complete Demonstration 4.
Demonstration 5: Format objects across a report (optional)

Purpose:
In Reporting, you will override a global style to modify the way report title objects appear in the report. You will also add a local style and will use it to format the report footer text. You will then create a report and observe how these style changes affect it.

You will need to complete Demonstration 4 before starting this demonstration.

Portal: http://vclassbase:9300/bi
User/Password: brettonf/Education1

Task 1. Explore Global Class Extensions.
1. Click My content, and then to the right of Demonstration 4_Enhance Report Layout, click the vertical ellipsis.
2. Click Edit report.
3. Click Page views, and then click Page design.
4. Click the report title text in the page header, and then in the Properties pane, under MISCELLANEOUS, ensure that the Classes property is set to Report title text.
5. In the Properties pane, click Select Ancestor, click Block, and then under MISCELLANEOUS, verify that the Classes property is set to Report title area.
6. Click Navigate, and then from Page explorer, click Classes.
7. In the Global Class Extensions list, select Report title text.
Report title text class corresponds to the class name you set on Page1.
In the Preview pane, Sample Text is underlined. You can explore the Report title area class to preview its style.

Task 2. Override a class style definition, and then add a new class.
You will now change the report title text style. The changes you make will apply only to instances of the style in this report.
1. With the Report title text style selected, in the Properties pane, under FONT & TEXT, double-click the Font property.
2. Click Foreground Color, click Blue, and then click OK.
3. Click **OK** to close the **Font** dialog box, then within the **Page explorer**, click **Page1**.

   The change you made to the report title text style has been applied. You will now format the text in the footer of the report.

4. On the **Page explorer** tab, click **Classes**.

5. From the **Toolbox**, drag a **Class** object to the **Local Classes** pane.

6. In the **Properties** pane, under **GENERAL**, modify the **Label** property to be **ReportFooterText**, and then press **Enter**.

7. In the **Properties** pane, under **FONT & TEXT**, double-click **Font**, modify the properties to **Tahoma, 10pt, Bold, Underline**, the **Foreground Color** as **Purple**, and then click **OK**.

8. Click **OK** to close the **Font** dialog box.

   Notice that the preview window shows the changes that have been made.

9. In the **Properties** pane, under **FONT & TEXT**, click the **Horizontal alignment** property, and then from the list, select **Left**.

   ![Image showing font and text properties]

**Task 3. Apply the new class to the report, add details, and run the report.**

1. From **Navigate**, on the **Page explorer** tab, click **Page1**.

2. In the page footer, click the first table cell, and then Ctrl-click the two remaining cells.

3. In the **Properties** pane, under **MISCELLANEOUS**, double-click the **Classes** property.

4. From the **Local classes** list, click **cls1:ReportFooterText**, click **Add to select classes** (right arrow), and then click **OK**.

   The style from the Report footer text class you created has been applied.
5. Run the report as **HTML**.
A section of the results appear as follows:

![Revenue by Order Number](image)

The formatting that you applied using the named styles appears in the report.

6. Close the rendered report tab.

**Results:**
In Reporting, you overrode a global class style and added a new local class style to the report.
Unit summary

- Force page breaks in reports
- Modify existing report structures
- Apply horizontal formatting
- Specify print options for PDF reports
- Format data and report objects
### Exercise 1

Analyze retailer contacts by country

<table>
<thead>
<tr>
<th>City</th>
<th>Retailer name</th>
<th>Contact first name</th>
<th>Contact last name</th>
<th>Contact phone number</th>
<th>Contact extension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Melbourne</td>
<td>Beach Boys Pty Ltd</td>
<td>Paul</td>
<td>Baranov</td>
<td>+61 7 5183 2022</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Kanga Kamers</td>
<td>Linda</td>
<td>Bullf</td>
<td>+61 7 6137 1007</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Blue Mountains Golfing Company</td>
<td>Adam</td>
<td>Baile</td>
<td>+61 7 3120 1027</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Beach Boys Pty Ltd</td>
<td>Scott</td>
<td>Crawford</td>
<td>+61 7 62 3221 3237</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Kanga Kamers</td>
<td>Doug</td>
<td>Cross</td>
<td>+61 7 7238 3005</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Top End Equipment</td>
<td>Cynthia</td>
<td>Garand</td>
<td>+61 7 7444 4567</td>
<td>3225</td>
</tr>
<tr>
<td></td>
<td>Gene Bush Supplies</td>
<td>Kenneth</td>
<td>Gardner</td>
<td>+61 7 82 8183 7772</td>
<td>3757</td>
</tr>
<tr>
<td></td>
<td>4 Golf only</td>
<td>Paul</td>
<td>Gaspar</td>
<td>+61 7 53 6000 0874</td>
<td>557</td>
</tr>
<tr>
<td></td>
<td>Kanga Kamers</td>
<td>David</td>
<td>Glodey</td>
<td>+61 7 6137 3704</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Beach Boys Pty Ltd</td>
<td>Steve</td>
<td>Givens</td>
<td>+61 7 35 6000 4870</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Black Stallion Supplies</td>
<td>Emily</td>
<td>Gooden</td>
<td>+61 7 70 2399 9480</td>
<td>577</td>
</tr>
<tr>
<td></td>
<td>Cutback Plt</td>
<td>William</td>
<td>Home</td>
<td>+61 7 70 2399 4681</td>
<td>9223</td>
</tr>
<tr>
<td></td>
<td>Southern Cross Pty</td>
<td>Enid</td>
<td>Jackson</td>
<td>+61 7 70 2399 8464</td>
<td>125</td>
</tr>
<tr>
<td></td>
<td>Harbour Pty Ltd</td>
<td>Karen</td>
<td>Jacksonson</td>
<td>+61 7 60 6058 9038</td>
<td>741</td>
</tr>
<tr>
<td></td>
<td>Jacks Ems Shop</td>
<td>Lesn</td>
<td>James</td>
<td>+61 7 2498 7570</td>
<td>147</td>
</tr>
<tr>
<td></td>
<td>Kanga Kamers</td>
<td>Norman</td>
<td>Janec</td>
<td>+61 7 6137 3210 4602</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Beach Boys Pty Ltd</td>
<td>Jake</td>
<td>Jenkins</td>
<td>+61 7 7238 3237</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Watsons Golf Supplies</td>
<td>Julie</td>
<td>Marchan</td>
<td>+61 7 39 2775 860</td>
<td>877</td>
</tr>
<tr>
<td></td>
<td>Can't Beat The Bush Supplies</td>
<td>Tony</td>
<td>Marriott</td>
<td>+61 7 2498 5068</td>
<td>0558</td>
</tr>
</tbody>
</table>

**Exercise 1: Analyze retailer contacts by country**
Exercise 1: Analyze retailer contacts by country

You have been asked to prepare a report that contains retailer contact information for each retailer for every country. The report must be broken into separate sections for each country so that the country appears as a section at the top, and only that country's contacts are displayed on each page.

To create the report, you must perform the following high-level tasks:

- Add the following query items to a new list using GO data warehouse (query)\Sales (query):
  - Employee by region: Country
  - Employee by region: City
  - Retailers: Retailer name
  - Retailers: Contact first name
  - Retailers: Contact last name
  - Retailers: Contact phone number
  - Retailers: Contact extension
- Section on Country; group on City.
- Apply page sets to display all contacts per Country per page.

For more information about where to work and the exercise results, refer to the Tasks and Results section that follows. If you need more information to complete a task, refer to earlier demonstrations for detailed steps.
Exercise 1: Tasks and Results

Portal:  http://vclassbase:9300/bi
User/Password:  brettonf/Education1
Package:  Team content\Samples\Models\GO data warehouse (query)
Report Template:  List
Folder:  Sales and Marketing (query)
Namespace:  Sales (query)

Task 1. Create the list.

- **Toolbar:** Open a new *List* template, using the *GO data warehouse (query)* package.
- **Source tab:** Navigate to *Sales and Marketing (query)/Sales (query)/Employee by region*.
  - Add *Country* and *City* to the list report object.
- **Source tab:** Navigate to *Sales and Marketing (query)/Sales (query)/Retailers*.
  - Add *Retailer name*, *Contact first name*, *Contact last name*, *Contact phone number*, *Contact extension*.

The results appear as follows:

<table>
<thead>
<tr>
<th>Country</th>
<th>City</th>
<th>Retailer name</th>
<th>Contact first name</th>
<th>Contact last name</th>
<th>Contact phone number</th>
<th>Contact extension</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Country&gt;</td>
<td>&lt;City&gt;</td>
<td>&lt;Retailer name&gt;</td>
<td>&lt;Contact first name&gt;</td>
<td>&lt;Contact last name&gt;</td>
<td>&lt;Contact phone number&gt;</td>
<td>&lt;Contact extension&gt;</td>
</tr>
</tbody>
</table>

- **Toolbar:** Section *<Country>*.
  - Group *<City>*.
  - Run the report in **HTML**.

A section of the results appear as follows:

```
Country: Australia

<table>
<thead>
<tr>
<th>City</th>
<th>Retailer name</th>
<th>Contact first name</th>
<th>Contact last name</th>
<th>Contact phone number</th>
<th>Contact extension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Melbourne</td>
<td>Beach Beds Pty Ltd.</td>
<td>Paul</td>
<td>Balesri</td>
<td>+61 02 5138 2622</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Kanga Kapers</td>
<td>Linda</td>
<td>Bafe</td>
<td>+61 02 9437 1067</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Blue Mountains Golfing Company</td>
<td>Adam</td>
<td>Bala</td>
<td>+61 03 7210 5227</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Beach Beds Pty Ltd.</td>
<td>Scott</td>
<td>Crawford</td>
<td>+61 09 9321 3237</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Kanga Kapers</td>
<td>Doug</td>
<td>Crease</td>
<td>+61 07 7238 1095</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Top End Equipment</td>
<td>Cynthia</td>
<td>Garland</td>
<td>61-7-54445677</td>
<td>3226</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Gone Bush Supplies</td>
<td>Kenneth</td>
<td>Gartner</td>
<td>61-52-5483779</td>
<td>8787</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4 Golf only</td>
<td>Paul</td>
<td>Gasper</td>
<td>61-39-6508784</td>
<td>557</td>
</tr>
<tr>
<td></td>
<td>Kanga Kapers</td>
<td>David</td>
<td>Gidley</td>
<td>+61 02 9437 5784</td>
<td></td>
</tr>
</tbody>
</table>
```

- Close the rendered report tab.
Task 2. **Add page sets to the report.**

- **Navigate:** Open Report pages.
- **Toolbox:** Add a Page set to the Report pages pane.
- **Properties pane:** Associate the Page set to Query1.

Task 3. **Define the grouping structure for the page set.**

- **Properties pane:** Add Country to the Groups folder, under Grouping & sorting.
- **Report Pages pane:** Drag Page 1 into the Detail Pages folder.
- **Toolbar:** Run the report in HTML.

A section of the results appear as follows:

![Report screenshot](image)

You have created a report that contains retailer contact information for each retailer for each country. The report has been sectioned for each country so that the country appears as a section header at the top, and each country's contacts are displayed on a separate page.

- Close the rendered report tab.
- Sign out of **IBM Cognos Analytics**.
- Close all browser windows.
The content presented in this unit is a subset of a high-level overview of IBM Cognos Analytics available in Overview of IBM Cognos Analytics (v11.0).

Students who want to gain a more detailed overview of IBM Cognos Analytics can visit IBM training (www.ibm.com/services/learning) to enroll in the overview course: Overview of IBM Cognos Analytics (v11.0).
IBM Cognos Analytics is a complete web-based BI and analytics solution, offering a unified user experience that works the same on the web, the desktop, or on mobile devices.

The smarter self-service offered by Cognos Analytics allows users across your organization to navigate to relevant data, interact with information to gain insight, and to create reports and dashboards without requiring access to a technical expert.

The Cognos Analytics environment is designed to be used at all levels of business, from departmental reporting and analysis, to broader enterprise reporting.
Redefined Business Intelligence

- Smarter self-service
  - guided
  - built-in intelligence
- Amplify and Act
  - make findings available to others
  - re-use and augment
  - increase collective intelligence
- Security and Confidence
  - user profiles and access to curated data
  - data cleansing
  - security and authentication
  - scalable single environment

The built-in intelligence provided by Cognos Analytics helps to guide users with an intuitive interface and intelligent analysis of your data to recommend appropriate visualizations to represent your data.

The single, shared solution allows sharing and re-use of information across self-service and enterprise reporting to make findings available to others.

IBM Cognos security protects your data by controlling access to data and bringing data from multiple sources together into a single, trusted data source.
IBM Cognos Analytics supports consumers who want to understand their data, through static and interactive reports, intuitive navigation, and search capabilities.

Self-service reporting allows users to create powerful dashboards and customized Data Modules based on keywords, without requiring the assistance of an IT specialist.

Power users are able to create complex data models, and detailed, professional reports to draw insight from varied data sources.
Navigate to content in IBM Cognos Analytics

- A single, unified web experience on all devices
- Smart search (in context)
- Recently-viewed

Cognos Analytics provides a unified experience across all devices, whether you are on a desktop system or a mobile device, even disconnected from your network. Navigation to locate data, reports, and dashboards functions the same regardless of how you are using Cognos Analytics.

Context-sensitive search provides targeted, relevant results based on your current context, allowing you to quickly locate reports, dashboards and data.

The Recently-viewed feature in Cognos Analytics helps you quickly find content you work with on a regular basis.
Interact with the user interface

- On-demand toolbars and menus
- Intuitive and interactive UI
- Personalization

On-demand toolbars and menus give you access to tools and options in context, so that power users and casual users can improve productivity within the same interface.

An intuitive user interface anticipates user intent and automates processes to improve user performance and reduce training requirements.

The user interface can also be personalized by individual users to meet their own specific requirements.
Model data with IBM Cognos Analytics

- Easy access to data
- Easily combine Data Modules
- Intent-driven modeling
- Accelerated authoring

Users can easily access data from a variety of sources, including enterprise data, as well as personal data sources.

Data Modules allow users to combine data from different sources into a single data source for reporting, and an automated model generation system allows self-service users to quickly generate a data module based on keywords.

By enabling users to access data on-demand, authors can quickly generate reports to gain insights into their data.
IBM Cognos Analytics components

IBM Cognos Analytics capabilities provide reporting, analysis, scorecarding, dashboard creation, business event management, and data integration from a wide array of corporate and personal data sources. IBM Cognos Analytics includes:

- IBM Cognos Portal, which is the Web portal for BI content presentation, management, and administration.
- Web and desktop reporting and analysis tools to author and analyze corporate data.
- Metadata modeling tools, including Framework Manager, Dynamic Cube Designer, Transformer, as well as Data Modules.

You can access IBM Cognos Administration by clicking Manage in the bottom right corner, and then clicking Administration console. You must have the required permissions to access IBM Cognos Administration. You can also perform some basic administrative tasks in the IBM Cognos portal.

Use Dashboards to create personal dashboards. Packages cannot be used in dashboards in Cognos Analytics v11.0.
Use IBM Cognos Analytics - Reporting to view reports, and perform both self-service reporting and advanced reporting to build sophisticated reports against multiple data sources, including personal data sources.

Use Framework Manager to create basic query packages, relationally based packages, and dimensional analysis packages.

Use Dynamic Cube Designer to create, edit, import, export, and deploy virtual cube models over a relational warehouse schema.

Use Transformer to create PowerCubes for dimensional analysis.

Existing customers can enable legacy support to use additional tools from previous IBM Cognos BI releases:

- Use IBM Cognos Viewer to view reports.
- Use Query Studio to perform ad hoc querying and quickly answer a focused question.
- Use Analysis Studio to perform analyses of data to discover trends, risks, and opportunities.
- Use Event Studio to create agents which notify users of key operational or performance-related events in their business.
- Use PowerPlay Studio to perform multidimensional analysis using IBM Cognos PowerCubes.
- Use Metric Studio to manage performance by monitoring and analyzing metrics.
- Use Metric Designer to create scorecard applications for use in Metric Studio.
Create reports with IBM Cognos Analytics - Reporting

- Unified environment for report authoring and analysis
- Intuitive search and navigation of data
- Data visualizations
- Templates and themes to accelerate report development
- Report subscriptions

IBM Cognos Analytics - Reporting is a unified environment for reporting and analysis. Consumers can view reports to analyze data, and report creators and power users can seamlessly transition to editing reports without needing a separate authoring environment.

Search tools, and an intuitive navigation system help you to quickly find the data you are looking for, and data visualizations allow you to present data in a way that makes it easier to draw insights about the data.

Predefined templates and themes allow you to create a professional-looking report quickly, without needing a lot of authoring expertise.

You can subscribe to reports of interest to get regular updates as your data changes. You cannot subscribe to dashboards in Cognos Analytics v11.0.
Perform self-service analysis with Dashboards

IBM Cognos Dashboards lets you quickly and easily explore data to make decisions. Templates and system-recommended visualizations help you to better understand your data.

Data from personal files and Data Modules can be dragged and dropped onto the dashboard canvas to build dashboard content, and smart filtering allows you to focus the dashboard on data that is relevant to your analysis. Content from existing reports cannot be used in a dashboard in IBM Cognos Analytics v11.0.
IBM Cognos Analytics architecture (high level)

IBM Cognos Analytics is a Web-based architecture, which is separated into three tiers; Web server, applications, and data. For the web server tier, you have the option of installing the gateway component on your own web server, or using the default gateway component.

This architecture is scalable from a software and hardware perspective. For example, you can have several IBM Cognos servers for faster response times and load balancing.

IBM Cognos leverages existing corporate IT resources such as web servers, authentication providers, and application servers, and also supports multiple languages and locales in order to serve a global audience.

IBM Cognos is customizable to adopt your corporate look and feel and can be extended and integrated into other applications through the IBM Cognos SDK.
IBM Cognos Analytics security

- The IBM Cognos Analytics security model combines existing enterprise security solutions with IBM Cognos Analytics security to achieve:
  - Authentication - Who are you?
  - Authorization - What can you see/do?
  - Administration - What/where can you manage?

IBM Cognos authentication is based on the use of third party authentication providers. These define users, groups, and roles used for authentication. User names, IDs, passwords, regional settings, and personal preferences are some examples of information stored in the providers. Based on the type of installation, a built-in authentication provider may be available.

Authorization is the process of granting or denying access to content, and specifying the actions that can be performed on that content, based on a user identity. Authorization assigns permissions to users, groups, and roles that allow them to perform actions, such as read or write, on objects, such as folders and reports. Permissions can be granted to users, groups, or roles directly from authentication providers or through membership in Cognos namespace groups and roles.

The Cognos namespace is the built-in namespace from IBM Cognos. It contains the IBM Cognos objects, such as groups, roles, data sources, distribution lists, and contacts. During the content store initialization, built-in and predefined security entries are created in this namespace, and include default access to functionality.

You can configure and administer IBM Cognos security using IBM Cognos Configuration and IBM Cognos Administration. You can also configure some security settings using the Manage button in IBM Cognos Analytics.
IBM Cognos Analytics groups and roles

- IBM Cognos Analytics provides default groups and roles for security, such as:

- System Administrators
- Authors
- Query Users
- Analysis Users
- Consumers
- Readers

Take advantage of IBM Cognos groups and roles from the Cognos namespace to secure your IBM Cognos environment and content. The group or role to which a user belongs determines how much access the user has to the IBM Cognos environment and functionality.

Besides the default groups and roles, you can create new groups and roles that are specific to your IBM Cognos needs. Simply add users from your authentication source to specific groups and roles as required.

Not only can you use the groups and roles defined in the IBM Cognos namespace to control access to contents, you can use groups in your authentication provider as well.

Using the IBM Cognos namespace does not require the IT department and creates a more portable environment.

There are many different groups and roles the administrator can use to restrict what you can see, what you can do, etc.
When a user runs a report, interactively or in the background, the metadata and data in the report are accessed through a combination of the package from which the report was authored, and the data source from which the package was modeled. The data source includes a connection string to the database and may include a signon that allows access to the database. The data source is used to query the database and retrieve the appropriate data, and the result set is presented back to the user.

There may be multiple connections for a given data source and multiple signons for a given connection.

Each object (report, package, data module, data source, connection, and signon) may have security applied.
Create Data Modules

- Upload and model data
- Combine data
  - multiple data sources
  - uploaded and corporate data
- Guided modeling (intent panel)
  - Generate models based on keywords
- Try It feature
- Live data mode or snapshots

Data Modules allow you to upload data to IBM Cognos Analytics, and model data from different sources into a single, trusted data source for reporting and analysis.

Using guided modeling, keywords can be used to indicate the type of data you would like to model, and IBM Cognos Analytics will analyze your data and recommend a model based on your requirements.

Once you have modified the recommended model to meet your specific requirements, Cognos Analytics will automatically build the data model for you.

The Try It feature allows you to test out your newly-created model in a reporting environment to ensure the model works as intended.

By default, Data Modules run in live data mode. Queries are submitted directly to the underlying data server, and reports will reflect the most current data. If you encounter load issues, snapshots of Data Modules can be used to improve query performance and reduce database load during peak periods. Snapshots are cached copies of data for a data module. Their data is stored in a file system location accessible to the Cognos Analytics server.
Appendix A Introduction to IBM Cognos Analytics

Upload files

- Upload personal or external data
- Integrate with other data sources in a data model

Personal and external data, such as external data files, can be uploaded to the Cognos Analytics environment using the Upload Files feature.

IBM Cognos Analytics stores data from delimited text and Excel files in a high-speed columnar storage format on the file system, rather than in-memory or in an RDBMS. Uploading a file using this capability does not create a separate model or package that must be managed by the user.

Data uploaded from external sources can then be integrated with existing corporate data in a custom data model for self-service reporting.

In order to use content from uploaded files in a report, the content must be modeled into a Data Module.
Demonstration 1: Explore IBM Cognos Analytics
Demonstration 1: Explore IBM Cognos Analytics

Purpose:
You will obtain a high-level view of how IBM Cognos Analytics works by navigating through the system and tracing the lifecycle of a data item, from its appearance in a report to its existence in an underlying data source. You will take on different roles including a(n):
- Consumer who runs reports or performs analysis to answer business questions
- Author who creates reports using various data items from a metadata package
- Data Modeler who imports data from the underlying data source, models it, and publishes a metadata package to make it available for Authors
- Administrator who creates and manages data source connections
You will conclude by identifying how the data item from the report appears as a column in the underlying data source.

Server: localhost
User/Password: scottb/Education1
Packages: Go Data Warehouse (query), Go Data Warehouse (analysis)
Folders: Sales and Marketing (query)
Namespace: Sales (query)

Task 1. Run a report from the IBM Cognos Portal.
1. Open Internet Explorer, and then navigate to http://vclassbase:9300/bi.
2. Log on as scottb/Education1.

The Welcome page appears. Security is currently set in this environment and you have logged on as a member of the Consumers role. The interface displays only functionality that is available to a member of the Consumers role. As a Consumer, you have the ability to navigate to and run reports, and to create dashboards and Data Modules.

You will view report content in the IBM Cognos Portal.
3. Click **Search**.
The search panel appears. You are searching for a report titled Sales Growth Year Over Year.

4. Type **sales growth**.
The search results appear.

5. Click **Sales Growth Year Over Year**.
   You would like to save this search for future use.

6. Click **Save search**.
The search is saved.

7. Click **Clear**.
   Under Saved searches, you can now see the search you saved.
   You could open the report directly from the search results, but instead, you will navigate to the report manually.

8. Click **Team content**, and then navigate to **Samples/Reports/Standard reports**.
9. Click **Sales growth year over year**. The underlying data source is queried for data, and the report opens in Cognos Analytics. A section of the results appears as follows:

![Image of report showing sales growth](image_url)

This report contains a list to display query data, as well as two charts to provide a visual representation of the data. You will now create a dashboard.
Task 2. Upload a personal file and create a dashboard in IBM Cognos Analytics.

Note: If you have already uploaded SampleFile_GOSales.xlsx in Unit 1, Demonstration 3 of this course, you may not have to do steps 1 through 3 of this task.

1. On the Side panel, click Upload files.
2. In the Choose File to Upload dialog, browse to C:\Training\B6058\Instructor Files.
3. Click SampleFile_GOSales.xlsx, and then click Open.
   The system will take a few moments to load the file. A progress bar indicates the status of the upload. When it's finished, you will see a preview of the data, such as Retailer country, Order method type, Retailer type, etc.
4. On the Side panel, click New, and then click Dashboard.
5. Under Select a template, click Freeform, and then click OK.
6. Click Sources.
7. Next to Selected sources, click Add a source. Because we are creating a dashboard, we are only able to access uploaded data files and Data Modules. We do not have access to use packages.
8. In the Open dialog, navigate to My content, click SampleFile_GOSales.xlsx, and then click Open.
   The data is loaded and is ready to use in your dashboard.
9. Click Product line, Ctrl-click Product type, and then Ctrl-click Revenue. Each entry is highlighted with a bar to the left of the entry, and the text of each item is bolded, to show that it is part of the current selection.
10. Drag the selected items to the canvas. The results appear as follows:

IBM Cognos Analytics creates a default visualization to display the data.

11. Click the visualization so that its toolbar appears.
12. Click **Change visualization** [image], and then select **Column**.
The results appear as follows:

![Column Chart Example]

The visualization is updated to display a column chart.

13. From the available data pane, click **Product line**, and then Ctrl-click **Year**, **Quarter**, and **Gross profit**.
14. Drag the selection to the right of the column chart. The results appear as follows:

IBM Cognos Analytics displays a Tree Map to represent the data.
15. In the **Column chart**, click **Climbing Accessories**.

The results appear as follows:

You can see the total revenue for Climbing Accessories, as well as options to keep the data in the chart, or exclude it from the chart.

The Tree Map has also been updated, and now displays only the Mountaineering Equipment information. Both charts are linked automatically.

You will now create a second tab to display different information.

16. Click **Add a new tab** $+$, click **Freeform**, and then click **Use**.
17. Click **Retailer country**, and then Ctrl-click **Quarter**, **Revenue**, and **Planned revenue**.

18. Click **Add**, then click change the visualization to a **Grid**.

The results appear as follows:

![Grid Visualization with Retailer country, Quarter, Revenue, and Planned revenue](image)

19. Click **Tab 1**.

20. Click **Open data tray**.

21. In the **Data tray**, click **Retailer country**, and then click **Filter**.

22. Click **Canada**.

The charts now display data specific to Canada.

23. Click **Tab 2**.

The Retailer country column now only displays Canada, so the filter was applied across both tabs.

24. Click the arrow beside **New dashboard***, and then click **Remove** beside **Sales growth year over year** and **New dashboard***

25. Click **OK** to continue without saving.

You will now examine the report you ran in the previous task using Reporting.

Note that while you are able to edit an existing report in Reporting, you will not be able to save it as a member of the Consumers role. You will now log in as Frank Bretton, a member of the Authors role.

1. Click Bart Scott, and then click Sign out.
2. Log on as brettonf/Education1.

   The Welcome page appears. Security is currently set in this environment and you have logged on as a member of the Authors role. The interface displays only functionality that is available to a member of the Authors role. As an Author, you have the additional ability to create and save reports.

3. Click Team content, and then navigate to Samples/Reports/Standard reports.
4. Click Sales growth year over year.
5. On the Toolbar, click Edit.

   The icons and buttons on the page update to reflect a Reporting context.

6. On the Toolbar, click Page views, and then click Page structure.
7. Expand Page - Page 1, and then expand the rest of the hierarchy.

   The report includes a hierarchical object structure, beginning with a Page object, which includes Page Body and Page Footer objects. The Page Body hierarchy includes, Table objects, which in turn include Table row objects, which in turn includes Table cell objects, etc.

8. On the Toolbar, click Page views, and then click Page design.
9. At the bottom of the report, double-click <AsOfDate()%>.

   The Report expression dialog box opens showing the AsOfDate() expression. AsOfDate() is an embedded report function within the Reporting expression editor, which can be used to return and display the execution date for the report.

10. Click Cancel, and then repeat step 9 for PageNumber() and AsOfTime().

    PageNumber() returns the current page number.
    AsOfTime() returns the report execution time.

11. In the list report object, double-click the <Revenue> column body.

    The objects used to define this expression come from the GO data warehouse (analysis) package, as shown in the Available Components pane on the left. You can see objects from the package by navigating the hierarchy.
12. In the **Available Components** pane, expand **Sales and Marketing (analysis) > Sales > Sales fact** to locate the **Revenue** object.

The hierarchy you have navigated matches what is displayed in the Expression Definition pane. Note that the Sales and Marketing (query) object is a folder and is excluded from the expression. You will become familiar with this object hierarchy when you create a report in Task 4.

13. Click **Cancel**, and then repeat steps 11 and 12 to identify the expressions and objects used to define them for the following items in the report:

- **<Product line>**
- **<Product type>**
- **<Year>**

Next you will create a report using metadata objects from a package.

**Task 4. Create a report using Reporting.**

1. Click **New**, and then click **Report**.
2. In the **New** dialog, click **List**, and then click **OK**.
3. Click **Data**.
4. Click Add report data.
5. In the Open dialog, navigate to Team content > Samples > Models > GO data warehouse (query), and then click Open.

The Source pane on the left displays the contents of the GO data warehouse (query) package. This package has been published from IBM Cognos Framework Manager as a metadata source for authors to create reports.

The structure and organization has been defined in the IBM Cognos Framework Manager model. There are four folders in this package.

6. Expand **Sales and Marketing (query)**.

At this level you are viewing namespaces. A namespace provides containment and name qualification for child objects.

7. Expand **Sales (query)**.

At this level you are viewing query subjects. A query subject is a set of query items that have an inherent relationship. In most cases, query subjects behave like tables. Query subjects produce the same set of rows regardless of which columns were queried.
8. Expand **Sales fact**.
   At this level you are viewing query items. A query item represents a single characteristic, such as the date that a product was introduced. Query items are contained in query subjects or dimensions (if using a dimensional data source). For example, a query subject that references an entire table contains query items that represent each column in the table.

9. Drag the following items into the list object:
   - **Product line** (from Products)
   - **Product type** (from Products)
   - **Revenue** (from Sales fact)

   The results appear as follows:

10. From the **Toolbar**, click **Run options**.
11. Click **Run HTML**.

   The report runs in a separate tab.

12. Close **IBM Cognos Analytics** without saving the report.

   Next you will take on the role of a modeler/developer to identify how the objects from the package, including the Revenue query item, are made available to authors to create their reports.

**Task 5. Examine a model in IBM Cognos Framework Manager.**

1. From the **Start** menu, click **All Programs** > IBM Cognos Framework Manager > IBM Cognos Framework Manager.
2. Click **Open a project**, and then open **great_outdoors_warehouse_dq.cpf** from C:\Program Files\IBM\cognos\samples\webcontent\samples\models\great_outdoors_warehouse_dq.
3. Log on as **admin/Education1**.

   In the current security environment, Admin Person is a member of the System Administrators role, and by default, has access to the entire IBM Cognos Analytics system, including IBM Cognos Framework Manager.

   You will examine what was published from IBM Cognos Framework Manager.
4. In the **Project Viewer** pane, expand **Packages**.
   The GO Data Warehouse (query) package appears. This is the package that you used in IBM Cognos Analytics to author the report.

5. Double-click the **GO Data Warehouse (query)** package.
   The Package Definition window displays which objects have been included and excluded from the package. The included objects are set to either visible or hidden. Hidden meaning they are included for publishing but will be hidden for authors.
   Note the four folders that have been included and set to visible. These match the folders you identified when you were viewing the package in Reporting (Task 4, step 9).

6. Expand the **Sales and Marketing (query)** folder > **Sales (query)** namespace. The results appear as follows:

   ![Image of Package Definition window](image)

   The Sales (query) namespace contains shortcuts to other objects in the model. The Sales fact shortcut points to a source object that is also included in the package; however, you cannot trace the source using this window. To do this, you will examine the model.
7. Click **Cancel**, and then in the **Project Viewer**, expand the **go_data_warehouse** namespace > **Sales and Marketing (query)** folder > **Sales (query)** namespace.

   Again, you can see the Sales fact shortcut, but now you can trace that shortcut back to its source.

8. Right-click **Sales fact**, and then click **Go To Target**.

   The results appear as follows:

   You are taken inside the Business View namespace (bolded text), to the Sales fact query subject (blue background).
9. Expand **Sales fact**.

This query subject includes the Revenue query item.

Without going into detail on the structure of this model, you should note that, as a proven practice, this sample model has been organized to include multiple namespaces. The Database View namespace is used to contain query subjects and query items that have been imported directly from the data source. The Business View namespace is used to contain query subjects and query items on which various modeling tasks have been performed, such as:

- combining query items from multiple query subjects
- creating calculated query items
- creating model filters and query item filters
- modifying and creating relationships between query subjects
- setting of various query subject and query item, properties

Objects that will be made available to authors are typically kept outside of these namespaces, for example the Sales and Marketing (query) folder.

The Sales and Marketing (query) folder contains only shortcuts to objects elsewhere in the model. Some of those objects, contained elsewhere in the model, have been included in the package. In the Package Definition, you noted that some objects were made hidden, including the Business View namespace. In Cognos Analytics, when the author creates the report and uses the Revenue item, they are using a shortcut to a hidden object in the package. In this case, it is the Revenue query item, from the Sales fact query subject, in the Business View namespace.

You will now identify the source for the Revenue query item from the Sales fact query subject.
10. Double-click the **Sales fact** query subject. 
   This is a model query subject. Model query subjects can be used to create a 
   more abstract, business-oriented view of a data source. For example, you can 
   add business objects such as calculations and filters and combine query items 
   from other query subjects, including other model query subjects. 
   The Measures pane displays the measures and their source that make up this 
   model query subject, including the Revenue measure. Also note that there are 
   two calculated items: Product cost, and Planned revenue. For Revenue, there 
   are two items to note:
   - The source is SLS_SALES_FACT.SALE_TOTAL.
   - The name is Revenue, indicating that the item was renamed from 
     SALE_TOTAL

   From this, you can conclude that Revenue is sourced from SALE_TOTAL. To 
   locate this object, you can search for it in the model.

11. Click **Cancel**, and then in the **Tools** pane on the right, click the **Search** tab.

12. Search for **SALE_TOTAL** using the **go_data_warehouse** model as the scope.
   
   Tip: Use the Search options button to define your search.

13. Click the first instance that contains **SALE_TOTAL**.
   Objects in the Project Viewer pane expand, and you can see the SALE_TOTAL 
   query item is located at go_data_warehouse > Database view > Sales and 
   marketing data > SLS_SALES_FACT.

14. Double-click **SLS_SALES_FACT**.
   This is a data source query subject. Data source query subjects directly 
   reference data in a single data source. IBM Cognos Framework Manager 
   automatically creates a relational data source query subject for each table and 
   view that you import into your model. It includes an SQL statement that will, at 
   runtime, retrieve all the columns from the table. You will test this behavior and in 
   turn locate the SALE_TOTAL column and its values.

15. Click the **Test** tab, click **Test Sample**, and then in the **Test results** window, 
   scroll to the right to locate **SALE_TOTAL**. 
   SALE_TOTAL is returned as a column.
16. Click the **Query Information** tab.

The results appear as follows:

This window also displays the SQL that is sent to the data source (Native SQL) and the SQL generated by the IBM Cognos query engine (Cognos SQL). In the Cognos SQL, you can see that the select statement includes all columns from the SLS_SALES_FACT table, including the SALE_TOTAL column. You can see that the "from" statement includes an object called great_outdoors_warehouse, and one called GOSALEDW. These objects represent the data source connections that are used at runtime. You will examine how these connections are defined in Task 6 when you view them in IBM Cognos Administration. For now, you will continue to examine how they are used in IBM Cognos Framework Manager.
17. Click **Cancel**, and then in the **Project Viewer**, expand **Data Sources**, and then click **go_data_warehouse**.

The results appear as follows:

From the information in the Properties pane, you can see that this model makes use of data source named `go_data_warehouse`, which uses a Content Manager data source named `great_outdoors_warehouse`. In this case, the modeler has opted to provide a different name for the model data source by editing the Name property in the Properties pane.

The Content Manager Data Source is the object through which:

- runtime data access is achieved
- the process of importing data source objects into the model is accomplished

You will examine the import process by running the Metadata Wizard.
18. In the **Project Viewer**, right-click **Database view**, and then click **Run Metadata Wizard**.

You can import from a wide range of data source types.

19. Leave **Data Sources** selected, and then click **Next**.

There are currently two or three data sources to choose from to perform the import. Clicking the New button will let you create a new data source provided you have the appropriate access rights to perform this action.

One of the data sources is named great_outdoors_warehouse, and is the data source that was previously used to import objects in to this model. The name also matches the Content Manager data source name identified at step 16.

20. Click **great_outdoors_warehouse**, and then click **Next**.

This data source points to a set of child data source objects, one of them being GOSALES DW. This name matches the object name identified when examining the SQL statement in step 16.

21. Expand **GOSALES DW > Tables**.

This data source object includes a set of tables, which can be imported into the model. Included in this set is the SLS_SALES_FACT table.

22. Expand **SLS_SALES_FACT**.

This table includes a set of columns, which can be imported into the model. Included in this set is the SALE_TOTAL column.

23. Click **Cancel**, and then close **IBM Cognos Framework Manager** without saving the project.

Up to this point you have identified how the Revenue query item:

- data values appear in existing reports
- is used in a new report created in IBM Cognos Analytics
- is made available to authors in a package that is published from IBM Cognos Framework Manager
- is sourced and modeled in the IBM Cognos Framework Manager model
- is imported into the IBM Cognos Framework model as the SALE_TOTAL column

### Task 6. Examine data sources in IBM Cognos Analytics.

You will now take on the role of the administrator to examine how data source connections are defined in IBM Cognos Administration.

1. Open **Internet Explorer**, and then navigate to **http://vclassbase:9300/bi**.
2. Log on as **admin/Education1**.
3. On the **Side panel**, click **Manage**.
4. Click **Administration console**.
IBM Cognos Administration is the portal that allows you to monitor and administer the IBM Cognos Analytics system including servers, security, capabilities, data source connections, and the deployment of content. In the current security environment, Admin is a member of the Directory Administrators role, which by default, provides access to the Directory pages of IBM Cognos Administration, including the ability to create and manage data sources.

5. Click the **Configuration** tab.
The first node selected is Data Source Connections. Here you can administer existing data sources and create new ones. Note: The New button on the toolbar provides the same user interface experience for creating a new data source that is available in IBM Cognos Framework Manager in the Metadata Wizard.

Existing data sources display; these are the same data sources that appeared when you ran the Metadata Wizard in IBM Cognos Framework Manager, at Task 5, steps 18 and 19.

6. Click the **great_outdoors_warehouse** data source.
This data source includes a single data source connection named great_outdoors_warehouse. Note: You can have multiple data source connections for a single data source. For example, if you have multiple databases with exactly the same structure (but different data), you can create one data source with multiple connections. The data source connection identifies which database you want to connect to.

7. Under **Actions**, click **Set properties - great_outdoors_warehouse**, and then click the **Connection** tab.
This connection is configured to connect to a DB2 database. There are many types available for creating connections.

8. Beside the **Connection string** box, click **Edit the connection string**. The connection is to a database named GS_DB, and under Signon, a signon has been configured for this connection. You will now examine the signon.

9. Click **Cancel** twice, and then click the **great_outdoors_warehouse** connection.
This connection includes a single signon named great_outdoors_warehouse. The database signon identifies the user's rights in the database. You can have multiple database signons that have access to different tables. Within the database, you can create sets of tables with different owners or schemas, and then provide access to these with the appropriate signon.
10. Under **Actions**, click **Set properties - great_outdoors_warehouse**, click the **Signon** tab, and then click **Edit the signon**.

This signon is configured to connect to the GOSALESDW schema using the credentials of the GOSALESDW user. The GOSALESDW schema is the object referenced in the generated SQL when you examined it in IBM Cognos Framework Manager (Task 5, step 16)

11. Close **Internet Explorer**, closing all tabs if prompted.

   You have identified how the connection is made to the underlying data source. Next, you will examine the required data source objects as they appear in IBM DB2.

**Task 7. Examine underlying data source objects.**

1. From the **Start** menu, navigate to **All Programs>IBM Data Studio>Data Studio 4.1.0.0 Client**.

2. Click **OK** to close the Workspace Launcher.

3. From the **Administration Explorer** pane, expand **localhost, 50000**, and then double-click **GS_DB [DB2 Alias]**.

4. In the **Properties for GS_DB** window, on the **General** tab, in the **User name** field type **db2admin**, and then in the **Password** field type **Education1**.

5. Click the **Save password** check box, and then click **OK**.

6. Double-click and expand **GS_DB**.

   The GS_DB database connection is active and folders are displayed. This is the database identified at Task 6, step 6.
7. From the Administration Explorer pane, click the Schemas folder.
The Schemas folder is displayed in the central pane. This is the schema that the
great_outdoors_warehouse data source is connecting to as identified in Task 6,
step 8.
A section of the results appear as follows:

8. From the Administration Explorer pane, click the Tables folder.
9. Scroll down to SLS_SALES_FACT (in the name column).
10. Right-click SLS_SALES_FACT, point to Data, and then click
New "Select" Script.

11. From the toolbar, click Run SQL.
The query executes and returns data from all the columns in the table.
12. From the Properties pane, at the bottom of the pane, click the Result1 tab on
the lower right pane.
13. Scroll to the right to locate the SALE_TOTAL column and its values.
You have traced the Revenue item in the Total Revenue by Country report all
the way back to its source in the underlying data source and have identified how
the IBM Cognos Analytics system works.
14. Close all open windows.

Results:
You obtained a high-level view of how IBM Cognos Analytics works by
navigating through the system and tracing the lifecycle of a data item, from its
appearance in a report to its existence in an underlying data source.
Extend IBM Cognos Analytics

- IBM Cognos provides a wide variety of ways to extend IBM Cognos Analytics.
- For more information, please visit the IBM Cognos Web site http://www-01.ibm.com/software/analytics/cognos/.

Cognos products that extend IBM Cognos Analytics include:

- IBM Cognos for Microsoft Office (integrate IBM Cognos content with MS Office)
- IBM Cognos Mobile (IBM Cognos content on mobile devices)
- IBM Cognos Analysis for Microsoft Excel (multidimensional analysis on IBM Cognos Analytics data in MS Excel spreadsheets)
- IBM Cognos Mashup Service

For developers, there is also IBM Cognos Virtual View Manager, which allows for access to an even wider variety of data sources, and the IBM Cognos SDK for customization and application development.
Unit summary

• Describe IBM Cognos Analytics and its position within an Analytics solution
• Describe IBM Cognos Analytics components
• Describe IBM Cognos Analytics at a high level
• Describe IBM Cognos Analytics security at a high level
• Explain how to extend IBM Cognos Analytics