Microsoft

OFFICIAL MICROSOFT LEARNING PRODUCT

20768C

Developing SQL Data Models

Product Number: 20768C

Part Number (if applicable):

Released: 02/2018

Module 1

Introduction to Business Intelligence and Data Modeling

Contents:

Lesson 1

Overview of Business Intelligence and Data Models

Contents:

Question and Answers

Question: True or false? A data model, which is created by using a technology such as SQL Server Analysis Services, is a required component of all BI projects.

( ) True

( ) False

Answer:

( ) True

(√) False

Lesson 2

Microsoft Business Intelligence Platform

Contents:

Question and Answers

Question: You are part of a team that is in the early stages of planning a BI project. You have identified that a key requirement for the project will be the ability to use historical data to make predictions about future scenarios, which the marketing department will use to help to create new campaigns. Which type of SQL Server Analysis Services instance will you need to include so that you can meet this requirement?

( ) An instance of SQL Server Analysis Services in tabular mode.

( ) An instance of SQL Server Analysis Services in multidimensional and data mining mode.

( ) An instance of SQL Server Analysis Services in PowerPivot mode.

Answer:

( ) An instance of SQL Server Analysis Services in tabular mode.

(√) An instance of SQL Server Analysis Services in multidimensional and data mining mode.

( ) An instance of SQL Server Analysis Services in PowerPivot mode.

Module Review and Takeaways

Review Question(s)

Question: In your experience, what are the factors that affect the success or failure of a BI project?

Answer: Answers will vary depending on the experience of the students.

Lab Review Questions and Answers

Lab: Exploring a Business Intelligence Solution

Question and Answers

Lab Review

Question: Which tools have you used to view BI data? How do these tools differ and what are the advantages and disadvantages of these tools?

Answer: Answers will vary depending on students' knowledge, experience, and opinions. Tools might include Excel, Report Viewer, Report Builder, Power View, and other Microsoft and third-party tools. Some tools, such as Excel, offer a data-centric view; others, such as Power View, use visualizations to represent the data.

Module 2

Creating Multidimensional Databases

Contents:

Lesson 1

Introduction to Multidimensional Analysis

Contents:

Question and Answers

Match the definition to the correct Analysis Services component.

Items	
1	An object that filters a data source and specifies a subset of the tables to include.
2	An aggregated numerical value, such as a sales total.
3	An object that contains data, such as product or customer information, that provides context for data analysis.

Category 1		Category 2		Category 3
Data source view		Measure		Dimension

Answer:

Category 1		Category 2		Category 3
Data source view		Measure		Dimension
An object that filters a data source and specifies a subset of the tables to include.		An aggregated numerical value, such as a sales total.		An object that contains data, such as product or customer information, that provides context for data analysis.

Lesson 2

Data Sources and Data Source Views

Contents:

Question and Answers

Question: True or false? You are investigating data sources for a planned BI project that includes Analysis Services multidimensional cubes. You have identified an Oracle database that contains important data, but a colleague says that you cannot import data from an Oracle database into an Analysis Services cube.

( ) True

( ) False

Answer:

( ) True

(√) False

Demonstration: Creating a Data Source and a Data Source View

Demonstration Steps

Create an Analysis Services Project

1. Ensure that the **20768C-MIA-DC** and **20768C-MIA-SQL** virtual machines are both running, and then log on to **20768C-MIA-SQL** as **ADVENTUREWORKS\Student** with the password **Pa55w.rd**.
2. In the **D:\Demofiles\Mod02** folder, run **Setup.cmd** as **Administrator**.
3. Start **Visual Studio**, and then click **New Project**.
4. In the **New Project** dialog box, in the **Business Intelligence** templates section, create a new **Analysis Services Multidimensional and Data Mining Project** named **Demo** in the **D:\Demofiles\Mod02** folder.

Create a Data Source

1. In Solution Explorer, right-click **Data Sources**, and then click **New Data Source**.
2. On the **Welcome to the Data Source Wizard** page, click **Next**.
3. On the **Select how to define the connection** page, click **New**.
4. In the **Server name** field, type **localhost**, in the **Select or enter a database name** list, click **AdventureWorksDW**, click **OK**, and then click **Next**.
5. On the **Impersonation Information** page, click **Use a specific Windows user name and password**, enter the user name **ADVENTUREWORKS\ServiceAcct** and the password **Pa55w.rd**, and then click **Next**.
6. In the **Data source name** field, type **Adventure Works Demo DS**, and then click **Finish**.

Create a Data Source View

1. In Solution Explorer, right-click **Data Source Views**, and then click **New Data Source View**.
2. On the **Welcome to the Data Source View Wizard** page, click **Next**.
3. On the **Select a Data Source** page, ensure that **Adventure Works Demo DS** is selected, and then click **Next**.
4. On the **Select Tables and Views** page, select the **DimDate**, **DimProduct**, **DimProductCategory**, **DimProductSubcategory**, **DimReseller**, **DimSalesTerritory** and **FactResellerSales** tables. Add these tables to **Included objects**, and then click **Next**.
5. On the **Completing the Wizard** page, in the **Name** field, type **Adventure Works Demo DSV**, and then click **Finish**.

Modify a Data Source View

1. If it is not already open, in Solution Explorer, right-click **Adventure Works Demo DSV**, and then click **Open**.
2. In Data Source Designer, click the table name of the **FactResellerSales** table and press F4.
3. In the Properties pane, change the **FriendlyName** property to **Reseller Sales**.
4. Repeat the previous steps to rename the **DimSalesTerritory** table to **Sales Territory**.
5. Right-click the **DimProduct** table, and then click **New Named Calculation**.
6. In the **Column name** field, type **Product Size**.
7. In the **Expression** field, type **[Size] + ' ' + [SizeUnitMeasureCode]**, and then click **OK**.
8. Right-click the **DimProduct** table and click **Explore Data**, then scroll down until you see a value in the **Product Size** column (the last column in the table) that shows the size and measurement unit.
9. Close the Explore DimProduct Table window then, on the **File** menu, click **Save All**.
10. Keep Visual Studio open. You will return to it in a later demonstration.

Lesson 3

Cubes

Contents:

Question and Answers

Question: You have created a new cube by using SQL Server Data Tools, and you want to test it by browsing the cube by using a PivotTable in Excel. What is the easiest way to do this?

( ) Open Excel, connect to the cube, create a PivotTable, and then add measures and attribute values to it.

( ) In the Cube Browser window in SQL Server Data Tools, use the Analyze in Excel options to open an Excel workbook, and then add measures and attribute values to it.

Answer:

( ) Open Excel, connect to the cube, create a PivotTable, and then add measures and attribute values to it.

(√) In the Cube Browser window in SQL Server Data Tools, use the Analyze in Excel options to open an Excel workbook, and then add measures and attribute values to it.

Demonstration: Creating and Browsing a Cube

Demonstration Steps

Create a Cube

1. Ensure that you have completed the previous demonstration in this module.
2. Maximize Visual Studio, which should be open with the **Demo** project loaded.
3. In Solution Explorer, right-click **Cubes**, and click **New Cube**.
4. On the **Welcome to the Cube Wizard** page, click **Next**.
5. On the **Select Creation Method** page, ensure that **Use existing tables** is selected, and click **Next**.
6. On the **Select Measure Group Tables** page, click **Suggest** then note that the wizard identifies **DimReseller** and **Reseller Sales** as potential measure group tables. Clear **DimReseller** so that only **Fact**R**eseller Sales** is selected, and then click **Next**.
7. On the **Select Measures** page, clear the **FactReseller Sales** check box to clear all selections, select the **Total Product Cost** and **Sales Amount** measures, and then click **Next**.
8. On the **Select New Dimensions** page, clear the **Reseller Sales** dimension, and then click **Next**.
9. On the **Completing the Wizard** page, change the cube name to **SalesDemo** and click **Finish**. The cube is created and opened in the Cube Designer.

Deploy and Browse a Cube

1. In Solution Explorer, right-click **Demo** and click **Deploy**. If prompted, enter the password **Pa55w.rd** for the **ADVENTUREWORKS\ServiceAcct** user.
2. After the deployment completes, in the Cube Designer, click the **Browser** tab.
3. In the Metadata pane, expand **Measures** and expand **Reseller Sales**. Note that the wizard has created the measures you selected.
4. Drag the **Sales Amount** measure to the **Drag levels or measures here to add to the query** area. The total sales amount for all reseller sales is shown.
5. In the Metadata pane, note that the wizard has created dimensions for the **DimReseller** and **DimProduct** tables. It has also determined that there are three relationships defined between the **Reseller Sales** table and the **DimDate** table, and so has a dimension for each related column (**Due Date**, **Order Date**, and **Ship Date**).

6. Expand each dimension and note that the wizard has only created attributes for key columns. You will need to modify the dimensions to add meaningful attributes for analysis.
7. Drag the **Product Key** attribute from the **Dim Product** dimension to the left of the **Sales Amount** value in the grid. The browser now shows the aggregated sales amount totals for each product key.

Customize Measures and Dimensions

1. In the Cube Designer, click the **Cube Structure** tab, and then, in the Measures pane, expand **Reseller Sales**.
2. Right-click the **Total Product Cost** measure and click **Rename**, then change the name of the measure to **Cost**.
3. In the Dimensions pane, right-click **Dim Reseller**, click **Rename** and change the name of the dimension to **Reseller**.
4. Repeat the previous step to rename **Dim Product** to **Product**.
5. Right-click the **Product** dimension and click **Edit Dimension**. This opens the Dimension Designer for the **Dim Product** dimension (the dimension is named **Product** in the **SalesDemo** cube, but still named **Dim Product** in the database).
6. In the Data Source View pane, in the **DimProduct** table, drag the **EnglishProductName** field to the Attributes pane.
7. In the Attributes pane, right-click **English Product Name** and click **Rename**, then rename the attribute to **Product Name**.
8. In Solution Explorer, right-click **Demo** and click **Deploy**. If prompted, enter the password **Pa55w.rd** for the **ADVENTUREWORKS\ServiceAcct** user.
9. After the deployment completes, click the **SalesDemo.cube [Design]** tab to return to the Cube Designer, and then click the **Browser** tab.
10. On the **Browser** tab, click the **Reconnect** button in the upper left corner. If the grid contains any data from previous queries, right-click in the grid and click **Clear Grid**.
11. In the Metadata pane, expand **Measures** and **Reseller Sales**, and drag **Cost** to the **Drag levels or measures here to add to the query** area.
12. Expand the **Product** dimension, and then drag **Product Name** to the left of the **Cost** value. The total cost associated with sales for each product is shown.
13. On the **File** menu, click **Save All**.
14. Keep Visual Studio open. You will use it again in a later demonstration.

Lesson 4

Overview of Cube Security

Contents:

Question and Answers

Question: You wish to assign permissions in Analysis Services to meet the following criteria:

User A needs access to the entire cube to browse and analyze the data.

User B needs access to the entire cube to browse and analyze the data, and also needs to perform predictive analysis against a data mining model.

How should you configure these users' roles and permissions?

( ) Create a single role and assign permissions that enable access to the cube and the data source. Assign both user accounts to this role.

( ) Create a single role and assign permissions that enable access to the cube. Assign both user accounts to this role.

( ) Create a single role and assign permissions that enable access to the data source. Assign both user accounts to this role.

( ) Create two roles, named Role X and Role Y. Assign permissions that enable access to the cube on Role X and assign permissions that enable access to the cube and the data source on Role Y. Assign User A to Role X and User B to Role Y.

( ) Create two roles, named Role X and Role Y. Assign permissions that enable access to the cube on Role X and assign permissions that enable access to the cube and the data source on Role Y. Assign User A to Role Y and User B to Role X.

Answer:

( ) Create a single role and assign permissions that enable access to the cube and the data source. Assign both user accounts to this role.

( ) Create a single role and assign permissions that enable access to the cube. Assign both user accounts to this role.

( ) Create a single role and assign permissions that enable access to the data source. Assign both user accounts to this role.

(√) Create two roles, named Role X and Role Y. Assign permissions that enable access to the cube on Role X and assign permissions that enable access to the cube and the data source on Role Y. Assign User A to Role X and User B to Role Y.

( ) Create two roles, named Role X and Role Y. Assign permissions that enable access to the cube on Role X and assign permissions that enable access to the cube and the data source on Role Y. Assign User A to Role Y and User B to Role X.

Demonstration: Implementing Cube Security

Demonstration Steps

Create a Role

1. Ensure you have completed the previous demonstration in this module. Visual Studio should be open with the **Demo** project loaded.
2. In Solution Explorer, right-click **Roles**, and click **New Role**. In Solution Explorer, right-click the **Role.role** icon that has been added, click **Rename**, and change the name to **Restricted User.role**. When prompted to change the object name, click **Yes**.
3. In the role designer, on the **General** tab, note that by default this role has no database permissions.
4. On the **Membership** tab, note that this is where you can add Windows users and groups to the role.

5. On the **Data Sources** tab, note that by default the role has no access to any data sources.
6. On the **Cubes** tab, change the **Access** value for **SalesDemo** to **Read**. This allows the role to access the cube and read its data.
7. On the **Cell Data** tab, note that this is where you can specify MDX statements that define cells in the cube that you want to permit or deny this role to access.
8. On the **Dimensions** tab, note that the role has **Read** access to all dimensions in the database. In the **Select Dimension Set** list at the top of the page, click **SalesDemo cube dimensions** and note that these permissions are inherited by the dimensions in the **Demo Cube** cube.
9. On the **Dimension Data** tab, in the **Dimension** list, under **SalesDemo**, select **Measures Dimension** and click **OK**. On the **Basic** tab, clear the check box for the **Cost** measure.
10. In the **Dimension** list, under **SalesDemo**, select **Product** and click **OK**. On the **Basic** tab, in the **Attribute hierarchy** list, select **Product Name**.
11. Select **Deselect all members** then scroll to the bottom of the list of product names and select all of the products that begin with the word **Touring**.
12. On the **File** menu, click **Save All**.

Test Permissions

1. In Solution Explorer, right-click **Demo** and click **Deploy**. If prompted, enter the password **Pa55w.rd** for the **ADVENTUREWORKS\ServiceAcct** user.
2. When deployment is complete, in Solution Explorer, double-click **SalesDemo.cube** to return to the Cube Designer.
3. On the **Browser** tab, click the **Reconnect** button at the top left. If the grid contains any data from previous queries, right-click in the grid and click **Clear Grid**.
4. In the Metadata pane, expand **Measures** and **Reseller Sales**, and drag **Sales Amount** and **Cost** to the **Drag levels or measures here to add to the query** area. Expand the **Product** dimension and drag **Product Name** to the left of the measure values. The browser shows the cost and sales amount for all products.
5. Right-click in the grid and click **Clear Grid**.
6. In the top left area of the browser window, click the **Change User** button. In the **Security Context – Demo Cube** dialog box, select **Roles**, and in the drop-down list, select **Restricted User** and click **OK**. Click **OK** again to return to the browser window.
7. Note that the grid has been cleared then, in the Metadata pane, expand **Measures** and **Reseller Sales**, and drag **Sales Amount** to the **Drag levels or measures here to add to the query** area. Note that members of this role cannot access the **Cost** measure.
8. Expand the **Product** dimension and drag **Product Name** to the left of the measure values. The browser only shows the sales amount for the **Touring** products.
9. On the **File** menu, click **Save All**, and then close Visual Studio.

Lesson 5

Configuring SSAS

Contents:

Question and Answers

Question: You are configuring an Analysis Services instance to coexist on the same server as a SQL Server relational database instance. You want to ensure that Analysis Services leaves enough memory for the relational instance to be always running. What setting should you configure?

( ) Set the HardMemoryLimit value to 0.

( ) Set the TotalMemoryLimit value to 0.

( ) Set the HardMemoryLimit value to 95.

( ) Set the HardMemoryLimit value to 45.

( ) Set the LowMemoryLimit value to 45.

Answer:

( ) Set the HardMemoryLimit value to 0.

( ) Set the TotalMemoryLimit value to 0.

( ) Set the HardMemoryLimit value to 95.

(√) Set the HardMemoryLimit value to 45.

Module Review and Takeaways

Question: What should you consider when assigning friendly names for objects in a multidimensional database?

Answer: There is no single correct answer. However, a major factor is the need to discuss object names with business users and, where possible, use terminology that is consistent with the way users interact with business entities across the organization.

Lab Review Questions and Answers

Lab: Creating a Multidimensional Database

Question and Answers

Lab Review

Question: In the final exercise, in which you added a dimension to the cube, you edited the data source view to include the table that the dimension was based on. Why was it not necessary to also edit the data source object?

Answer: It was not necessary to edit the data source because it references the entire database, and so already includes all of the tables. The data source view includes only a subset of the tables, so you need to add the extra table to this subset by editing the data source view.

Module 3

Working with Cubes and Dimensions

Contents:

Lesson 1

Configuring Dimensions

Contents:

Question and Answers

Question: MOLAP dimensions provide better performance than ROLAP but which of the following is a reason for choosing ROLAP?

( ) ROLAP is easier to implement than MOLAP.

( ) When real-time OLAP is required.

( ) Querying ROLAP generally involves simpler queries.

( ) ROLAP is more efficient with small dimensions.

Answer:

( ) ROLAP is easier to implement than MOLAP.

(√) When real-time OLAP is required.

( ) Querying ROLAP generally involves simpler queries.

( ) ROLAP is more efficient with small dimensions.

Lesson 2

Defining Attribute Hierarchies

Contents:

Question and Answers

Question: Excluding organizational charts, where else might you use parent-child hierarchies?

> **Answer:** There is no correct or incorrect answer. Typically, if you can assign a fact to a non-leaf member, it needs to be a parent-child hierarchy.

Demonstration: Creating a Parent-Child Hierarchy

Demonstration Steps

Create a Parent-Child Hierarchy

1. Ensure that the **20768C-MIA-DC** and **20768C-MIA-SQL** virtual machines are both running, and log on to **20768C-MIA-SQL** as **ADVENTUREWORKS\Student** with the password **Pa55w.rd**. In the **D:\Demofiles\Mod03** folder, run **Setup.cmd** as **Administrator**.
2. Start Visual Studio and open **Demo.sln** in the **D:\Demofiles\Mod03** folder.
3. In Solution Explorer, expand **Data Source Views** and double-click **Adventure Works Demo DSV.dsv**.
4. On the **Data Source View** menu, click **Add/Remove Tables**. In the **Add/Remove Tables** dialog box, add the **DimEmployee** table to the **Included objects** list and click **OK**.
5. In the data source view diagram, right-click the **DimEmployee** table and click **New Named Calculation**. In the **Create Named Calculation** dialog box, create a named calculation with the column name **Full Name** and the expression **[FirstName] + ' ' + [LastName]**, and then click **OK**.
6. In Solution Explorer, right-click **Dimensions** and click **New Dimension**.
7. On the **Welcome to the Dimension Wizard** page, click **Next**.
8. On the **Select Creation Method** page, select **Use an existing table** and click **Next**.
9. On the **Specify Source Information** page:
   - In the **Main table** box, select **DimEmployee**.
   - In the **Key columns** list, ensure that **EmployeeKey** is selected.
   - In the **Name column** box, select **Full Name**.
   - Click **Next**.
10. On the **Select Related Tables** page, clear the **Sales Territory** check box, and click **Next**.
11. On the **Select Dimension Attributes** page, click **Next**.
12. On the **Completing the Wizard** page, in the **Name** box, type **Employee** and click **Finish**.
13. Leave Visual Studio open for the next demonstration.

Name the Levels in a Parent-Child Hierarchy

1. In the **Employee.dim** Dimension Designer, on the **Dimension Structure** tab, in the **Attributes** section, click **Parent Employee Key**.
2. In the Properties pane, note that the **Usage** property for this attribute has been set to **Parent**.
3. In the **NamingTemplate**, property value, click the ellipses (**...**).
4. In the **Level Naming Template** dialog box, in the first blank **Name** cell, type **Executive**.
5. In the next blank **Name** cell, type **Middle Management**.

6. In the next blank **Name** cell, type **Senior Employee**.
7. In the next blank **Name** cell, type **Junior Employee** then click **OK**.
8. In the Properties pane, change the **MembersWithData** property value to **NonLeafDataHidden**.
9. In the Attributes pane, click **Employee** dimension, and set the **UnknownMember** property to **None.** Select the **Employee Key** attribute, expand its **KeyColumns** property, expand the **DimEmployee.EmployeeKey** column, and set the **NullProcessing** property to **Automatic**.
10. In Solution Explorer, right-click **Demo** and click **Deploy**. If prompted for a password, type **Pa55w.rd** for the **ADVENTUREWORKS\ServiceAcct** user, and then click **OK**.
11. When deployment has completed, in the **Employee.dim** Dimension Designer, in the **Browser** tab, expand the employees in the hierarchy. Note that, when you select an employee, the hierarchy level name is displayed at the top of the browser area.
12. Leave Visual Studio open for the next demo.

Lesson 3

Implementing Sorting and Grouping for Attributes

Contents:

Question and Answers

Question: When might you need to use a secondary attribute for sorting data?

> **Answer:** There are many possible answers. One example could be to sort employee by last name, even though the key column is employee_id and the name column is calculated.

Demonstration: Using Sorting and Grouping

Demonstration Steps

Use Sorting

1. In Visual Studio, in Solution Explorer, expand **Dimensions**, and then double-click **Dim Date.dim**.
2. In the **Browser** tab, ensure **Date** is selected in the **Hierarchy** drop-down and explore the **Date** dimension, noting that the month numbers are not in numeric order.
3. On the **Dimension Structure** tab, in the **Attributes** section, click **Month Number Of Year**.
4. In the Properties pane, change the **OrderBy** property to **Key**.
5. On the **Dimension** menu, click **Process**. If you receive a message about out-of-date content, click **Yes**, and then click **Yes** to confirm you want to continue with deployment. If you are prompted for a password, type **Pa55w.rd** and click **OK**.
6. In the **Process Dimension – Dim Date** dialog, click **Run**, and when the build process has completed, in the **Process Progress dialog**, click **Close**.
7. In the **Process Dimension – Dim Date** dialog, click **Close**.
8. In the **Browser** tab, ensure **Date** is selected in the **Hierarchy** menu and then, on the **Dimension** menu, click **Reconnect**.
9. Browse the **Date** dimension noting the month numbers are now sorted correctly.

Use Grouping

1. In Solution Explorer, expand **Dimensions**, and then double-click **Dim Reseller.dim**.
2. On the **Browser** tab, ensure **Resellers** is selected in the **Hierarchy** drop-down, and then browse the **Resellers** hierarchy, noting the data is presented as one long list of reseller IDs. To enable simpler location of a specific retailer ID, these will be grouped.
3. On the **Dimension Structure** tab, in the Attributes pane, click **Reseller Key**, in the Properties pane change the **DiscretizationMethod** property to **Automatic**, and then change the **DiscretizationBucketCount** to **10**.
4. On the **Dimension** menu, click **Process**. If you receive a message about out-of-date content, click **Yes**, and then click **Yes** to confirm you want to continue with deployment. If you are prompted for a password, type **Pa55w.rd** and click **OK**.
5. In the **Process Dimension – Dim Reseller** dialog, click **Run**, and when the build process has completed, in the **Process Progress** dialog, click **Close**.
6. In the **Process Dimension – Dim Reseller** dialog, click **Close**.
7. On the **Browser** tab, ensure **Resellers** is selected in the **Hierarchy** drop-down and then, on the **Dimension** menu, click **Reconnect**.
8. Browse the Reseller dimension noting Reseller IDs are now grouped.

Lesson 4

Slowly Changing Dimensions

Contents:

Question and Answers

Question: You are implementing a Customer dimension in a cube. Should you implement this as a fixed dimension, a Type 1 SCD, or a Type 2 SCD?

> **Answer:** This answer is open to discussion. Customer information, such as the home address, marital status, and company name, is subject to change so you should implement this dimension as an SCD. If the class feels that it is not necessary to maintain old values of these attributes, then a Type 1 SCD is appropriate. Otherwise, the administrators should implement a Type 2 SCD.

Resources

The Slowly Changing Dimension Wizard

 **Best Practice:** When you design and select key columns in dimension tables, ensure that they are unchanging. For example, in a product database, a product number or part number is often used as a primary key because it is unique to the product. However, a product number or part number can change and so should be an attribute of an SCD. If you want to store both old and new product numbers for later analysis, you need a Type 2 SCD and two rows with different product numbers. Ensure that the Slowly Changing Dimension Transformation can identify these records as relating to the same product by using a different identity value for the primary key in the dimension table.

Module 4

Working with Measures and Measure Groups

Contents:

Lesson 1

Working with Measures

Contents:

Question and Answers

Question: Give an example of scenarios where you would use an additive measure, a semiadditive measure, and a nonadditive measure.

Answer: Answers will vary—some suggested answers include:

- Use an additive sales amount measure in a sales measure group, which can be aggregated by **Sum** across all dimensions.
- Use a semiadditive balance measure in an accounts measure group, which can be aggregated by **Sum** across customers, but by **LastNonEmpty** when aggregated by time.
- Use a nonadditive profit margin measure in a sales measure group, which cannot be meaningfully aggregated across any dimension.

Lesson 2

Working with Measure Groups

Contents:

Question and Answers

Question: You are configuring the storage settings for a measure group. The data in the group does not need to be updated regularly. However, the data must be available at all times, and queries that access the data must execute with the best possible performance. What is the most suitable storage setting to apply to the measure group?

( ) Real Time ROLAP

( ) Real Time HOLAP

( ) MOLAP

( ) Automatic MOLAP

( ) Medium Latency MOLAP

Answer:

( ) Real Time ROLAP

( ) Real Time HOLAP

(√) MOLAP

( ) Automatic MOLAP

( ) Medium Latency MOLAP

Demonstration: Defining Relationships Between Dimensions and Measure Groups

Demonstration Steps

1. Ensure that the **20768C-MIA-DC** and **20768C-MIA-SQL** virtual machines are both running, and then log on to **20768C-MIA-SQL** as **ADVENTUREWORKS\Student** with the password **Pa55w.rd**.
2. In the **D:\Demofiles\Mod04** folder, run **Setup.cmd** as Administrator.
3. Start **SQL Server Data Tools**, and then open **Demo.sln** in the **D:\Demofiles\Mod04** folder.
4. On the **Build** menu, click **Deploy Solution**. If you are prompted, specify the user name **ADVENTUREWORKS\ServiceAcct** and the password **Pa55w.rd**.
5. Wait for the **Deployment Completed Successfully** message in the Deployment Progress – Demo window, and then close the Deployment Progress – Demo window.
6. In Solution Explorer, double-click **SalesDemo.cube**, and on the **Browser** tab, expand **Measures**, expand **Reseller Sales**, and then drag the **Revenue** measure to the query results area. Expand the **Geography** dimension, and then drag **Country-Region** to the left of the **Revenue** value. Note that the values for each region are the same. The aggregation is incorrect.
7. In Solution Explorer, expand dimensions, right-click **Geography.dim**, and then click **View Designer**. Note that this dimension is based on the **Geography** table and has a **Country-Region** attribute.
8. In Solution Explorer, right-click **Reseller.dim**, and then click **View Designer**. Note that this dimension is based on the **DimReseller** table, which includes a **GeographyKey** attribute that relates it to the **Geography** table.
9. Click the tab for the **SalesDemo** cube, and on the **Cube Structure** tab, in the Data Source View pane, note that there is no direct relationship between the **Reseller Sales** fact table and the **Geography** dimension table.

10. On the **Dimension Usage** tab, click the intersection of the **Reseller Sales** measure group and the **Geography** dimension, and then click the ellipsis **(…)** button.
11. In the **Select relationship type** list, select **Referenced**.
12. In the **Intermediate dimension** list, select **Reseller**.
13. In the **Reference dimension attribute** list, select **Geography Key**, in the **Intermediate dimension attribute** list, select **Geography Key**, and then click **OK**.
14. On the **Dimension Usage** tab, in the **Dimensions** list, right-click **Geography**, and then click **Rename**.
15. Change the name of this cube dimension to **Reseller Geography**, and then press Enter to make this name change take effect.
16. On the **Build** menu, click **Deploy Solution**. If you are prompted, specify the user name **ADVENTUREWORKS\ServiceAcct** and the password **Pa55w.rd**.
17. Wait for the **Deployment Completed Successfully** message in the Deployment Progress – Demo window, and then close the Deployment Progress – Demo window.
18. On the **Browser** tab for the **SalesDemo** cube, click the **Reconnect** button.
19. Right-click the query results area, and then click **Clear Grid**.
20. Expand **Measures**, expand **Reseller Sales**, and then drag the **Revenue** measure to the query results area. Expand the **Reseller Geography** dimension, and then drag **Country-Region** to the left of the **Revenue** value. Note that the values for each region are now correct.

Module 5

Introduction to MDX

Contents:

Lesson 1

MDX Fundamentals

Contents:

Question and Answers

Question: You want to run 10 queries against the **Adventure Works** cube, but you are only interested in summarizing results from 2015 and 2016. Which of the following techniques should you use to execute queries most efficiently?

( ) Specify the 2015 and 2016 members in each SELECT query.

( ) Use the FILTER function in each query to display only the 2015 and 2016 members.

( ) Use the LOOKUPCUBE function.

( ) Create a subcube that includes only the 2015 and 2016 members.

( ) Specify a slicer axis by using the WHERE clause.

Answer:

( ) Specify the 2015 and 2016 members in each SELECT query.

( ) Use the FILTER function in each query to display only the 2015 and 2016 members.

( ) Use the LOOKUPCUBE function.

(√) Create a subcube that includes only the 2015 and 2016 members.

( ) Specify a slicer axis by using the WHERE clause.

Resources

Establishing Cube Context

 **Best Practice:** Although they are useful, queries that use the LOOKUPCUBE function to integrate two cubes are likely to perform less well than queries against a single cube. If you find that you use LOOKUPCUBE often, consider redesigning your cubes to include all of the relevant data.

Demonstration: Querying a Cube

Demonstration Steps

Deploy and Query a Cube

1. Ensure that the **20768C-MIA-DC** and **20768C-MIA-SQL** virtual machines are both running, and then log on to **20768C-MIA-SQL** as **ADVENTUREWORKS\Student** with the password **Pa55w.rd**. In the **D:\Demofiles\Mod05\Starter** folder, run **Setup.cmd** as Administrator.
2. On the Start screen, type **SQL Server Data Tools 2015**, and then press Enter.
3. In SQL Server Data Tools, on the **File** menu, point to **Open**, and then click **Project/Solution**.
4. In the **Open Project** dialog box, browse to the **D:\Demofiles\Mod05\Starter** folder, and then double-click **Adventure Works OLAP.sln**.
5. On the **Build** menu, click **Deploy Solution**.
6. If the **Microsoft Visual Studio** dialog box appears, click **Yes**.
7. If you are prompted, specify the user name **ADVENTUREWORKS\ServiceAcct** and the password **Pa55w.rd**, and then click **OK**.

8. Wait for the **Deployment Completed Successfully** message in the **Deployment Progress – Adventure Works OLAP** window, and then close the Deployment Progress – Adventure Works OLAP window.
9. Close Microsoft Visual Studio®.
10. Start **SQL Server Management Studio**.
11. In the **Connect to Server** dialog box, in the **Server type** drop-down list, select **Analysis Services**.
12. In the **Server name** text box, select **MIA-SQL**, and then click **Connect**.
13. In Object Explorer, expand **Databases**, right-click **Adventure Works OLAP**, point to **New Query**, and then click **MDX**.
14. In the query pane, type the following code:

```
SELECT
{
[Measures].[Internet Sales Count],
[Measures].[Reseller Sales Count]
} ON COLUMNS,
{
[Order Date].[Calendar Date].[Calendar Year]
} ON ROWS
FROM [Sales]
```

15. Click **Execute**, and then examine the results.
16. To specify a slicer axis, add the following line of code to the previous query:

```
WHERE [Customer].[State Province Name].&[Ohio]
```

17. Click **Execute**, and then examine the results.
18. Close SQL Server Management Studio, without saving any changes.

Lesson 2

Adding Calculations to a Cube

Contents:

Question and Answers

Categorize each item into the appropriate category. Indicate your answer by writing the category number to the right of each item.

Items	
1	PARENT
2	PARALLELPERIOD
3	CHILDREN
4	NAME
5	SIBLINGS
6	CURRENTMEMBER
7	FIRSTSIBLING
8	HIERARCHY
9	LASTCHILD
10	LEVEL
11	ANCESTOR
12	NONEMPTY

Category 1		Category 2
Family MDX functions		Non-family MDX functions

Answer:

Category 1		Category 2
Family MDX functions		Non-family MDX functions
PARENT CHILDREN SIBLINGS FIRSTSIBLING LASTCHILD ANCESTOR		PARALLELPERIOD NAME CURRENTMEMBER HIERARCHY LEVEL NONEMPTY

Demonstration: Adding a Calculated Member to a Cube

Demonstration Steps

Add a Calculated Member

1. Ensure that you have completed the previous demonstration in this module.
2. On the Start screen, type **SQL Server Data Tools 2015**, and then press Enter.
3. In SQL Server Data Tools, on the **File** menu, point to **Open**, and then click **Project/Solution**.
4. In the **Open Project** dialog box, browse to the **D:\Demofiles\Mod05\Starter** folder, and then double-click **Adventure Works OLAP.sln**.
5. In Solution Explorer, under **Cubes**, double-click **Sales.cube**.
6. In the Cube Designer, click the **Calculations** tab.
7. Right-click in the **Script Organizer** pane, and then click **New Calculated Member**.
8. In the **Name** text box, type **[Total Sales Count]**.
9. In the **Parent hierarchy** list, ensure that **Measures** is selected.
10. In the **Expression** text box, type the following MDX expression:

```
[Measures].[Internet Sales Count] + [Measures].[Reseller Sales Count]
```

11. In the **Format string** drop-down list, select **"0"**.
12. In the **Non-empty behavior** list, select **Internet Sales Count** and **Reseller Sales Count**, and then click **OK**.
13. On the **File** menu, click **Save All**.
14. On the **Build** menu, click **Deploy Solution**. If you are prompted to authenticate, use the password **Pa55w.rd**, and then click **OK**.
15. When the deployment is complete, close Visual Studio.

Query a Calculated Member

1. Start **SQL Server Management Studio**.
2. In the **Connect to Server** dialog box, in the **Server type** drop-down list, click **Analysis Services**.
3. In the **Server name** drop-down list, click **MIA-SQL**, and then click **Connect**.
4. In Object Explorer, expand **Databases**, right-click **Adventure Works OLAP**, point to **New Query**, and then click **MDX**.
5. Type the following code in the query pane:

```
SELECT
{
[Measures].[Internet Sales Count],
[Measures].[Reseller Sales Count],
[Measures].[Total Sales Count]
} ON COLUMNS,
[Product].[Categorized Products].[Category] ON ROWS
FROM [Sales]
```

6. Execute the query, and then examine the results.
7. Close SQL Server Management Studio, without saving any changes.

Lesson 3

Using MDX to Query a Cube

Contents:

Question and Answers

Question: You are creating a connection string for a data source in SQL Server Reporting Services. You want to ensure that ragged hierarchies include placeholders for missing members. Which of the following properties should you use for this setting?

( ) Data source

( ) Initial catalog

( ) Provider

( ) Character encoding

( ) MDX compatibility

Answer:

( ) Data source

( ) Initial catalog

( ) Provider

( ) Character encoding

(√) MDX compatibility

Demonstration: Querying a Cube from Excel

Demonstration Steps

Connect to SQL Server Analysis Services

1. On the Start screen, type **Excel 2016**, and then press Enter.
2. On the Welcome screen, click **Blank workbook**.
3. On the ribbon, click the **Data** tab.
4. Click **Get External Data**, click **From Other Sources**, and then click **From Analysis Services**.
5. In the **Server name** text box, type **MIA-SQL.adventureworks.msft**.
6. Select **Use the following User name and Password**, type the following, and the click **Next**:
   a. **User Name**: adventureworks\Student
   b. **Password**: Pa55w.rd
7. Ensure that the **Adventure Works OLAP** database and the **Sales** cube are selected, and then click **Next**.
8. Click **Finish**.
9. In the **Import Data** dialog box, ensure that **PivotTable Report** is selected, and then click **OK**.

Formulate an MDX Query in Excel

1. In the **PivotTable Fields** box, under **Internet Sales**, select **Internet Sales Count**.
2. Under **Reseller Sales**, select **Reseller Sales Count**.
3. Under **Values**, select **Total Sales Count**.
4. Under **Order Date**, select **Order Date.Calendar Date**.
5. In the **COLUMNS** box, drag **Order Date.Calendar Date** to the **ROWS** box.

6. Examine the data in the PivotTable.
7. Close all open windows, without saving any changes.

Module Review and Takeaways

Question: Users often want to aggregate data from Europe and North America in their reports and spreadsheets. Users who have good MDX skills have been able to do this by editing MDX queries in their client programs. How can you help users who have no MDX skills to display this data?

Answer: Use the Cube Designer to add a named set to the cube. Include North America and Europe members in the named set. Users can select this named set from the cube in the same way as they select any other dimension member.

Question: You are testing the **Sales** cube by writing MDX queries in SQL Server Management Studio. You would like to compare sales for 2015 with data from the **Archived Sales** cube, which includes sales data from 2008 to 2010. How can you add this data to your query?

Answer: Use the LOOKUPCUBE statement to add the interesting members from the **Archived Sales** cube. Then, select the measures and dimension as normal.

Module 6

Customizing Cube Functionality

Contents:

Lesson 1

Implementing Key Performance Indicators

Contents:

Question and Answers

Question: What KPIs are used in your business? Do you have KPIs in your department? Does the KPI generally have a positive effect on employee behavior? Can you think of any KPIs that have had unwanted results?

> **Answer:** KPIs that are focused on sales turnover and profit are typical within a sales department. Unwanted behavior might include booking sales before the contract has been signed, or selling at a reduced margin.
>
> KPIs in an IT environment might include the number of service calls that have been closed with positive feedback, or the number of outstanding change requests from users.
>
> Each KPI has an impact on how people behave. You should also consider the data that needs to be collected to accurately measure the KPI.

Resources

Introducing KPIs

 **Best Practice:** Getting a KPI measurement right is a business decision rather than an IT decision. A KPI must genuinely reflect an outcome that the business wants to achieve. Care must be taken that the KPI does not encourage damaging behavior. For example, measuring sales turnover by quarter could result in salespeople booking work before client contracts have been signed, or agreeing sales at reduced margins. A KPI that measures repeat purchases, on the other hand, might be a better indicator of customer satisfaction and future business success.

Demonstration: Creating a KPI

Demonstration Steps

Create a KPI

1. Ensure that the **20768C-MIA-DC** and **20768C-MIA-SQL** virtual machines are both running, and then log on to **20768C-MIA-SQL** as **ADVENTUREWORKS\Student** with the password **Pa55w.rd**.
2. In the **D:\Demofiles\Mod06** folder, run **Setup.cmd** as Administrator.
3. In the **User Account Control** dialog box, click **Yes**.
4. Start **Microsoft® Visual Studio®**, and then in the **D:\Demofiles\Mod06** folder, open the **Adventure Works OLAP.sln** solution.
5. On the **Build** menu, click **Deploy Solution**.
6. If the **Microsoft Visual Studio** dialog box appears, click **Yes**.
7. If the **Account Password** dialog box appears, enter the password **Pa55w.rd** for **ADVENTUREWORKS\ServiceAcct**, and then click **OK**.
8. After deployment is complete, close the **Deployment Progress** window.
9. In Solution Explorer, double-click **Sales.cube** to open it in the Cube Designer, and then click the **Calculations** tab.
10. In the **Script Organizer** pane, select **[Reseller Profit]**, and then note that this calculated member calculates reseller gross margin as a percentage of revenue.
11. In the Cube Designer, click the **KPIs** tab, and then, on the **Cube** menu, click **New KPI**.

12. In the **Name** box, type **Reseller Margin**, and then in the **Associated measure group** list, select **Reseller Sales**.
13. In the **Calculation Tools** pane, on the **Metadata** tab, expand **Measures**, expand **Reseller Sales**, and then drag the **Reseller Profit** measure to the **Value Expression** box.
14. In the **Goal Expression** box, type **0.1** (this sets a gross margin target of 10 percent).
15. Check that **Gauge** is selected in the **Status indicator** list, and then type the following MDX expression in the **Status expression** box (you can copy this from New_KPI_Demo.txt in the demo folder):

```
Case
   When
   KpiValue("Reseller Profit")/KpiGoal("Reseller Profit")>=.95
Then 1
When
  KpiValue("Reseller Profit")/KpiGoal("Reseller Profit")<.95
   And
  KpiValue("Reseller Profit")/KpiGoal("Reseller Profit")>=.5
   Then 0
  Else -1
End
```

16. Check that **Standard arrow** is selected in the **Trend indicator** list, and then type the following expression in the **Trend expression** box (you can copy this from New_KPI_Demo.txt in the demo folder):

```
Case
  When IsEmpty(ParallelPeriod([Order Date].[Calendar Date].[Calendar Year],
      1, [Order Date].[Calendar Date].CurrentMember))
    Then 0
  When [Measures].[Reseller Profit] = (ParallelPeriod([Order Date].[Calendar
Date].[Calendar Year],
      1, [Order Date].[Calendar Date].CurrentMember), [Measures].[Reseller Profit])
    Then 0
  When [Measures].[Reseller Profit] >
    (ParallelPeriod([Order Date].[Calendar Date].[Calendar Year],
      1, [Order Date].[Calendar Date].CurrentMember), [Measures].[Reseller Profit])
    Then 1
  Else -1
End
```

17. On the **File** menu, click **Save All**.

Browse a KPI in the Cube Designer

1. On the **Build** menu, click **Deploy Solution**. If you are prompted for impersonation credentials, enter the password **Pa55w.rd** for **ADVENTUREWORKS\ServiceAcct**.
2. After deployment is complete, close the **Deployment Progress** window.
3. In the cube browser, click the **KPIs** tab, and then on the **Cube** menu, point to **Show KPIs in**, and then click **Browser**. The KPI is shown based on the total overall margin.
4. In the top pane of the KPI browser, in the **Dimension** list, select **Sales Territory**, in the **Hierarchy** list, select **Sales Territory**, in the **Operator** list, select **Equal**, and then in the **Filter Expression** list, expand **All**, select **North America**, and then click **OK**.
5. Click anywhere in the KPI browser pane to update the values for the Reseller Margin KPI. The updated values indicate the overall margin for sales in North America.

6. Amend the KPI criteria again. In the top pane of the KPI browser, in the **Dimension** list, select **Order Date**, in the **Hierarchy** list, click **Calendar Date**, in the **Operator** list, select **Equal**, and then in the **Filter Expression** list, expand **All**, select **2006**, and then click **OK**.
7. Click anywhere in the KPI browser pane to update the values for the Reseller Margin KPI. The updated values indicate the overall Reseller margin for sales in 2006.

Browse a KPI in Excel

1. In the Cube Designer, click the **Browser** tab, and then on the **Cube** menu, click **Analyze in Excel**. If a security notice is displayed, click **Enable**.
2. In **Microsoft Excel®**, in the **PivotTable Fields** pane, under **Reseller Sales**, select **Reseller Profit**. The overall gross margin is shown in the PivotTable.
3. In the **PivotTable Fields** pane, under **Order Date**, drag **Order Date.Calendar Date** to the **Rows** area. The gross margin for each year is shown.
4. In the **PivotTable Fields** pane, expand **KPIs**, expand **Reseller Margin**, and then select **Goal**. The Reseller Margin goal is displayed in the PivotTable.
5. Close Excel without saving the workbook.
6. Close Visual Studio.

Lesson 2

Implementing Actions

Contents:

Question and Answers

Question: True or false? Actions are client-side processes that return data to the server.

( ) True

( ) False

Answer:

( ) True

(√) False

Lesson 3

Implementing Perspectives

Contents:

Question and Answers

Question: How many perspectives can each cube contain?

( ) Each cube can contain no more than one perspective.

( ) Each cube can contain many perspectives.

( ) Each cube must contain one perspective for each user.

( ) The number of perspectives depends on the security rules of the cube.

( ) Cubes do not support perspectives in SQL Server Analysis Services.

Answer:

( ) Each cube can contain no more than one perspective.

(√) Each cube can contain many perspectives.

( ) Each cube must contain one perspective for each user.

( ) The number of perspectives depends on the security rules of the cube.

( ) Cubes do not support perspectives in SQL Server Analysis Services.

Lesson 4

Implementing Translations

Contents:

Question and Answers

Question: What do you think are the advantages and disadvantages of implementing translations in a cube?

Answer: Answers will vary.

Module Review and Takeaways

Question: Can you think of scenarios in your own business where the use of the cube enhancements discussed in this module would help business users?

Answer: Some suggested answers, and ideas for discussion, include:

- A business might use KPIs to track critical performance metrics, which might include financial measurements such as revenue and profit, but also other metrics that help executives to evaluate the health of the business, such as the number of new customers per month.
- Actions can be used to automate business processes or extend analysis. For example, a business might analyze a cube to identify geographical locations that contain a large volume of clusters, and then use an action to show that location in Bing Maps and identify suitable sites for a new store.
- Perspectives are useful when a cube contains a large number of measures and dimensions that reflect the business as a whole, but the organization includes subsets of users who only need to focus on specific areas. For example, an employee from the marketing department might not need to use measures and dimensions that relate to product manufacturing, so a perspective can be used to simplify the cube.
- Translations are useful when the cube is used in multiple geographies—or to generate reports that must be sent to international customers or business partners.

Lab Review Questions and Answers

Lab: Customizing a Cube

Question and Answers

Lab Review

Question: When you apply the translation feature to a cube, what appears in the specified language?

Answer: Row and column captions appear in the specified language. You can also specify translations for certain columns, such as month. Textual data is not translated into the specified language.

Question: What does a drill-through action do?

Answer: Drill-through actions enable users to access the lowest level of detail of a cube's data.

Module 7

Implementing a Tabular Data Model by Using SQL Server Analysis Services

Contents:

Lesson 1

Introduction to Tabular Data Models in SQL Server Analysis Services

Contents:

Question and Answers

Question: What are the advantages and disadvantages of tabular models when compared to multidimensional models?

Answer: Tabular models are easier to create for most users and do not require as much specialist knowledge as multidimensional cubes. They can also be easily created from PowerPivot workbooks, meaning that business users can use Excel for self-service analysis, and the models that they create can then be refined and brought under IT management as they mature.

On the other hand, multidimensional data modeling is a well-understood professional BI technique, and there are many tools and client applications that can take advantage of it. There are some capabilities of multidimensional models that are not supported, or are difficult to implement in a tabular data model. These limitations include the ability to define specific dimension types such as account dimensions, implement translations, or perform write-back analysis.

Lesson 2

Creating a Tabular Data Model

Contents:

Question and Answers

Question: You are implementing an order date and a ship date role-playing dimension based on a date dimension table. You can achieve this by importing the fact table and the date dimension table once, and then creating a relationship between the order date key in the fact table and the date key in the date dimension table, and also creating a relationship between the ship date key in the fact table and the date key in the date dimension table.

( ) True

( ) False

Answer:

( ) True

(√) False

Demonstration: Creating a Tabular Data Model Project

Demonstration Steps

Create a Tabular Data Model Project

1. Ensure that the **20768C-MIA-DC** and **20768C-MIA-SQL** virtual machines are both running, and then log on to **20768C-MIA-SQL** as **ADVENTUREWORKS\Student** with the password **Pa55w.rd**.
2. In the **D:\Demofiles\Mod07** folder, run **Setup.cmd** as **Administrator**.
3. In the **User Account Control** dialog box, click **Yes**.
4. Start **SQL Server Data Tools 2015**, and then on the **File** menu, point to **New**, and then click **Project**.
5. In the **New Project** dialog box, type the following values, and then click **OK**:
   - **Project Template**: Analysis Services Tabular Project
   - **Name**: TabularDemo
   - **Location**: D:\Demofiles\Mod07
6. If the **Tabular model designer** dialog box is displayed, type the following values, and then click **OK**:
   - **Workspace server**: localhost\SQL2
   - **Compatibility level**: SQL Server 2017 / Azure Analysis Services (1400)
7. On the **Tools** menu, click **Options**.
8. In the **Options** dialog box, expand **Analysis Services Tabular Designers**, and then click **Workspace Database**. Note the name of the default workspace server that is used to host the data model during development, and then click **Cancel**.

Import Tables into a Tabular Model

1. On the **Model** menu, click **Import From Data Source**.
2. In the **Get Data** window, select **SQL Server database**, and then click **Connect**.
3. On the **SQL Server database** page, type the following values, and then click **OK**:
   - **Server**: MIA-SQL
   - **Database (optional)**: AdventureWorksDW

4. On the Impersonation Information page, change the **Impersonation Mode** to **Impersonate Service Account**, then click **Connect**.
5. In the **Encryption Support** dialog box, click **OK**
6. On the **Navigator** page, select the following source tables, and then click **OK**:
   - **DimCustomer**
   - **DimDate**
   - **DimGeography**
   - **DimProduct**
   - **DimProductCategory**
   - **DimProductSubcategory**
   - **FactInternetSales**
7. In the Query Editor, in the **Queries** list, click **DimCustomer** table and change the **Name** property to **Customer**.
8. On the **Columns** menu, click **Choose Columns**, clear the check box for each of the following columns, and then click **OK**:
   - Title
   - MiddleName
   - NameStyle
   - Suffix
   - SpanishEducation
   - FrenchEducation
   - SpanishOccupation
   - FrenchOccupation
9. In the **Queries** list, click **DimDate** and change the **Name** property to **Date**.
10. On the **Columns** menu, click **Choose Columns**, clear the check box for each of the following columns, and then click **OK**:
    - SpanishDayNameOfWeek
    - FrenchDayNameOfWeek
    - SpanishMonthName
    - FrenchMonthName
    - CalendarSemester
    - FiscalSemester
11. In the **Queries** list, click **DimGeography** and change the **Name** property to **Geography**.
12. In the **Queries** list, click **DimProduct** and change the **Name** property to **Product**.
13. In the **Queries** list, click **DimProductCategory** and change the **Name** property to **Product Category**.

14. In the **Queries** list, click **DimProductSubCategory** and change the **Name** property to **Product Subcategory**.
15. In the **Queries** list, click **FactInternetSales** and change the **Name** property to **Internet Sales**.
16. On the toolbar, click **Import**, and then wait for the data to be imported. When the data has been imported successfully, click **Close**.
17. Click each of the tabs in the model designer to see the data that has been imported into each table.

Define Measures in a Data Model

1. Click the **Internet Sales** tab, and then click the first empty cell in the measure grid under the **SalesAmount** column.
2. On the **Column** menu, point to **AutoSum**, and then click **Sum**.
3. In the formula bar, modify the expression that has been generated to change the name of the measure to **Revenue**, as shown in the following example:

```
Revenue:=SUM([SalesAmount])
```

4. Widen the **SalesAmount** column, and then note that the measure is calculated and displayed as a currency value. The formatting has been inherited from the column.
5. Select the cell that contains the **Revenue** measure, and then press F4 to display the properties pane. View the **Format** property, which is set to **Currency**.
6. Create a second measure named **Cost** in the measure grid under the **TotalProductCost** column based on the following expression, and format the **Cost** measure as currency:

```
Cost:=Sum([TotalProductCost])
```

7. On the **File** menu, click **Save All**.

Test the Data Model in Excel

1. On the **Model** menu, click **Analyze in Excel**.
2. In the **Analyze in Excel** dialog box, ensure that **Current Windows User** is selected and that the **(Default)** perspective is specified, and then click **OK**.
3. When Excel opens, in the **PivotTable Fields** pane, note that the measures that you defined are displayed under an **Internet Sales** measure group. The **Internet Sales** table is also shown in this pane, and contains the columns in the table, including those on which the aggregated measures are based. Some users might find this confusing so, in a future refinement of the data model, the columns that are not useful for analysis should be hidden.
4. Select the **Revenue** measure so that it is summarized in the PivotTable.
5. In the **Date** table, select **CalendarYear** so that the revenue in the PivotTable is aggregated by year. Note that there is no indication in the **Date** table that tells the user if the values that are displayed represent the order date, the due date, or the ship date that is associated with the order.
6. In the **Date** table, select **EnglishMonthName** so that the revenue is further broken down into monthly totals. Note that the months are shown in alphabetical order instead of chronological order.
7. Close Excel without saving the workbook.
8. Keep SQL Server Data Tools open for the next demonstration.

Demonstration: Managing Relationships and Columns

Demonstration Steps

Manage Relationships

1. Ensure that you have completed the previous demonstration in this module.
2. In SQL Server Data Tools, with the **Model.bim** pane visible, on the **Model** menu, point to **Model View**, and then click **Diagram View**. The tables are shown as a schema diagram, with lines between them to denote relationships. Note that there are three relationships between the **Internet Sales** table and the **Date** table.
3. Double-click the solid line between the **Internet Sales** and **Date** tables. Note the columns that are used to define this relationship, and that this is an active relationship, and then click **Cancel**.
4. Double-click each of the dotted lines between the **Internet Sales** and **Date** tables, and then click **Cancel**. In each case, note the columns that are used to define these relationships and that these are inactive relationships, and then click **Cancel**.
5. Right-click each dotted relationship line in turn, and then click **Delete**. When you are prompted to permanently delete the relationship from the model, click **Delete from Model**.
6. Double-click the remaining solid line between the **Internet Sales** and **Date** tables.
7. In the **Edit Relationship** dialog box, in the **Columns** list for the **Internet Sales** table, select **OrderDateKey** and then click **OK** (if it is already selected, click **Cancel**), This ensures that the active relationship is based on the order date, not the shipping date or due date.
8. Right-click the **Date** table title bar, click **Rename**, and then rename the table to **Order Date**.
9. In the Tabular Model Explorer, expand **Data Sources**, right-click **SQL\mia-sql:AdventureWorksDW**, and then click **Import New Tables**.
10. On the **Navigator** page, select the **DimDate** table, and then click **OK**.
11. In the Query Editor, in the **Queries** list, click **DimDate** table and change the **Name** to **Ship Date**.
12. On the **Columns** menu, click **Choose Columns**, clear the following columns, and then click **OK**:
    - SpanishDayNameOfWeek
    - FrenchDayNameOfWeek
    - SpanishMonthName
    - FrenchMonthName
    - CalendarSemester
    - FiscalSemester
13. Click **Import**, and then wait for the table to be imported. When it has been imported successfully, click **Close**.
14. Arrange the diagram so that you can see the **Ship Date** and **Internet Sales** tables, and then drag the **ShipDateKey** column from the **Internet Sales** table to the **DateKey** column in the **Ship Date** table to create the relationship. If the **Ship Date** table does not appear, refresh the page.

Rename and Hide Columns

1. Click the title bar of the **Order Date** table, and then click its **Maximize** icon.
2. In the maximized **Order Date** table, click the **DateKey** column, hold down CTRL, click the **DayNumberOfWeek** and **MonthNumberOfYear** columns to select them, right-click any of the selected columns, and then click **Hide from Client Tools**.
3. In the maximized **Order Date** table, right-click the **FullDateAlternateKey** column, click **Rename**, and then rename the column to **Date**.
4. Repeat the previous step to rename the following columns:
   - EnglishDayNameOfWeek (rename to **Weekday**)
   - EnglishMonthName (rename to **Month**)
5. Click the **Restore** icon for the **Order Date** table.
6. Maximize the **Internet Sales** table, hide all columns other than the **Revenue** and **Cost** measures that you created in the previous exercise, and then restore the **Internet Sales** table.

Configure Column Sort Order

1. On the **Model** menu, point to **Model View**, and then click **Data View**.
2. On the **Order Date** tab, click the **Weekday** column heading.
3. On the **Column** menu, point to **Sort**, and then click **Sort by Column**.
4. In the **Sort by Column** dialog box, in the **Sort** column, ensure that **Weekday** is selected, in the **By** column, select **DayNumberOfWeek**, and then click **OK**.
5. Click the **Month** column heading.
6. On the **Column** menu, point to **Sort**, and then click **Sort by Column**.
7. In the **Sort by Column** dialog box, in the **Sort** column, ensure that **Month** is selected, in the **By** column, select **MonthNumberOfYear**, and then click **OK**.
8. On the **File** menu, click **Save All**.

Review Your Changes in Excel

1. On the **Model** menu, click **Analyze in Excel**.
2. In the **Analyze in Excel** dialog box, ensure that **Current Windows User** is selected and that the **(Default)** perspective is specified, and then click **OK**.
3. When Excel opens, in the **PivotTable Fields** pane, note that the measures that you defined are still displayed under an **Internet Sales** measure group, but that the **Internet Sales** table is no longer shown here because all other columns have been hidden.
4. Select the **Revenue** measure so that it is summarized in the PivotTable.
5. In the **Order Date** table, select **CalendarYear** so that the revenue in the PivotTable is aggregated by year based on the order date.
6. In the **Order Date** table, select **Month** so that the revenue is further broken down into monthly totals. Note that the months are now shown in chronological order.
7. In the **Order Date** table, select **CalendarQuarter**, and then note that the quarter is shown under each month.

8. In the **PivotTable Fields** pane, in the **Rows** area, drag **CalendarQuarter** so that it is above **Month**, and the PivotTable shows the correct hierarchy for year, quarter, and month. This user experience could be improved by enabling users to select a hierarchy that automatically enables them to drill down to the correct levels.
9. Close Excel without saving the workbook.
10. Keep SQL Server Data Tools open for the next demonstration.

Demonstration: Creating a Hierarchy

Demonstration Steps

Create a Hierarchy

1. Ensure that you have completed the previous demonstrations in this module.
2. In SQL Server Data Tools, with the **Model.bim** pane visible, on the **Model** menu, point to **Model View**, and then click **Diagram View**.
3. Maximize the **Order Date** table, and then on its title bar, click the **Create Hierarchy** icon. Name the new hierarchy **Calendar Date**.
4. Drag the **CalendarYear** column to the **Calendar Date** hierarchy.
5. Drag the **CalendarQuarter** column to the **Calendar Date** hierarchy.
6. Drag the **Month** column to the **Calendar Date** hierarchy.
7. Drag the **Date** column to the **Calendar Date** hierarchy.
8. Restore the **Order Date** table, and then on the **File** menu, click **Save All**.

Browse the Hierarchy in Excel

1. On the **Model** menu, click **Analyze in Excel**.
2. In the **Analyze in Excel** dialog box, ensure that **Current Windows User** is selected and that the **(Default)** perspective is specified, and then click **OK**.
3. When Excel opens, select the **Revenue** measure so that it is summarized in the PivotTable.
4. In the **Order Date** table, select **Calendar Date** so that the revenue in the PivotTable is aggregated by year.
5. In the PivotTable, expand the years to show the revenue by quarter, expand the quarters to view revenue by month, and then expand the months to view revenue for individual dates.
6. Close Excel without saving the workbook.
7. Keep SQL Server Data Tools open for the next demonstration.

Lesson 3

Using a SQL Server Analysis Services Tabular Model in an Enterprise BI Solution

Contents:

Question and Answers

Question: Which of the following statements regarding DirectQuery mode is correct?

( ) A tabular data model that uses DirectQuery mode will always have a better performance than a tabular data model that uses in-memory storage.

( ) A tabular data model that uses in-memory storage will always have a better performance than a tabular data model that uses DirectQuery mode.

( ) A tabular data model that uses DirectQuery mode will always contain up-to-date information.

( ) A tabular data model that uses DirectQuery mode will only support a SQL Server relational database data source.

( ) You cannot use MDX in a tabular data model that uses DirectQuery mode.

Answer:

( ) A tabular data model that uses DirectQuery mode will always have a better performance than a tabular data model that uses in-memory storage.

( ) A tabular data model that uses in-memory storage will always have a better performance than a tabular data model that uses DirectQuery mode.

(√) A tabular data model that uses DirectQuery mode will always contain up-to-date information.

( ) A tabular data model that uses DirectQuery mode will only support a SQL Server relational database data source.

( ) You cannot use MDX in a tabular data model that uses DirectQuery mode.

Demonstration: Deploying a Tabular Data Model

Demonstration Steps

Deploy a Tabular Data Model Project

1. Ensure that you have completed the previous demonstrations in this module.
2. In SQL Server Data Tools, in Solution Explorer, right-click the **TabularDemo** project, and then click **Properties**.
3. In the **TabularDemo Property Pages** dialog box, type the following values, and then click **OK**:
   - **Server**: localhost\SQL2
   - **Database**: DemoDB
   - **Model Name**: Internet Sales
4. On the **Build** menu, click **Deploy TabularDemo**. If you are prompted for impersonation credentials, type the user name **ADVENTUREWORKS\ServiceAcct** and the password **Pa55w.rd**, and then click **OK**.
5. In the **Deploy** dialog box, when deployment has completed, click **Close**.
6. Close SQL Server Data Tools.

Use a Tabular SQL Server Analysis Services Database in Excel

1. Start Excel, and then create a new blank workbook.
2. On the **Data** tab, in **Get External Data** area, in the **From Other Sources** drop-down list, select **From Analysis Services**.

3. In the **Data Connection Wizard** dialog box, in the **Server name** box, type **MIA-SQL\SQL2**, ensure that **Use Windows Authentication** is selected, and then click **Next**.
4. On the **Select Database and Table** page, ensure that the **DemoDB** database and the **Internet Sales** cube are selected, and then click **Next**.
5. On the **Save Data Connection File and Finish** page, click **Finish**.
6. In the **Import Data** dialog box, ensure that the **Existing worksheet** option is selected, and then click **OK**.
7. In the **PivotTable Fields** pane, under **Internet Sales**, select **Revenue**.
8. In the **PivotTable Fields** pane, under **Order Date**, select the **Calendar Date** hierarchy.
9. In the **PivotTable Fields** pane, under **Geography**, drag **EnglishCountryRegionName** to the **Columns** area.
10. Explore the data in the PivotTable. When you have finished, close Excel without saving the workbook.

Module 8

Introduction to Data Analysis Expressions (DAX)

Contents:

Lesson 1

DAX Fundamentals

Contents:

Question and Answers

Question: When a DAX expression is evaluated, which contexts apply?

( ) The row context.

( ) The filter context.

( ) Both the row context and the filter context.

Answer:

( ) The row context.

( ) The filter context.

(√) Both the row context and the filter context.

Lesson 2

Using DAX to Create Calculated Columns and Measures in a Tabular Data Model

Contents:

Question and Answers

Question: You are creating a calculated column in a table, based on the values in a column in another table. The two tables have a one-to-many relationship, based on a key column. Which DAX function should you use to perform the lookup in the most efficient way?

( ) USERELATIONSHIP

( ) RELATEDTABLE

( ) RELATED

( ) LOOKUPVALUE

Answer:

( ) USERELATIONSHIP

( ) RELATEDTABLE

(√) RELATED

( ) LOOKUPVALUE

Demonstration: Creating and Using Measures and Calculated Columns

Demonstration Steps

Create Calculated Columns

1. Ensure that the **20768C-MIA-DC** and **20768C-MIA-SQL** virtual machines are both running, and log on to the **20768C-MIA-SQL** virtual machine as **ADVENTUREWORKS\Student** with the password **Pa55w.rd**.
2. In the **D:\Demofiles\Mod08** folder, right-click **Setup.cmd** and click **Run as administrator**. When prompted, click **Yes**.
3. Start Visual Studio 2015 and open the **TabularDemo.sln** solution in the **D:\Demofiles\Mod08** folder.
4. If the **Tabular model designer** dialog box appears, in the **Workspace server** list, specify **localhost\SQL2** and then click **OK**.
5. If the **Tabular Model Designer** dialog box appears, click **Yes**.
6. In Solution Explorer, double-click **Model.bim** to open the data model designer.
7. On the **Model** menu, point to **Process**, and click **Process All**. If you are prompted for impersonation credentials, specify the user name **ADVENTUREWORKS\ServiceAcct** with the password **Pa55w.rd** and click **OK**.
8. In the **Data Processing** dialog box, when all of the tables have been processed, click **Close**.
9. On the **Customer** tab, in the first empty column after the existing columns, double-click **Add Column** and name the new column **Full Name**.
10. With the **Full Name** column selected, in the formula bar, enter the following DAX expression:

```
=CONCATENATE([FirstName], " " & [LastName])
```

    Wait for the table to finish updating, and note the calculated values in the new column.

11. Click the **FirstName** column heading, hold **Shift** and click the **LastName** column heading, and then right-click either of the selected column headings and click **Hide from Client Tools**.

12. On the **Internet Sales** tab, in the first empty column after the existing columns, double-click **Add Column** and name the new column **SalesProfit**.
13. With the **SalesProfit** column selected, in the formula bar, enter the following DAX expression:

```
=[SalesAmount]-[TotalProductCost]
```

Wait for the table to finish updating, and note the calculated values in the new column.

Create Measures

1. On the **Internet Sales** tab, click the first empty cell in the measure grid under the **SalesProfit** column.
2. In the formula bar, enter the following DAX expression:

```
Profit:=SUM([SalesProfit])
```

3. Select the cell containing the Profit measure, and press F4 to display the properties pane. Then set the **Format** property to **Currency**.
4. Click the empty cell in the measure grid under the **Profit** measure.
5. In the formula bar, enter the following DAX expression:

```
Margin:=[Profit]/[Revenue]
```

6. Select the cell containing the **Margin** measure, and press F4 to display the properties pane. Then set the **Format** property to **Percentage**.
7. Right-click the **SalesProfit** column header and click **Hide from Client Tools**.
8. On the **File** menu, click **Save All**.

Create a KPI

1. On the **Internet Sales** tab, click the empty cell in the measure grid under the **Margin** measure.
2. In the formula bar, enter the following DAX expression:

```
Target Margin:=(SUM('Product'[ListPrice]) - SUM('Product'[StandardCost])) /
SUM('Product'[ListPrice])
```

3. Select the cell containing the **Margin** measure, and press F4 to display the properties pane. Then set the **Format** property to **Percentage**.
4. On the **Internet Sales** tab, right-click the cell in the measure grid containing the **Margin** measure and click **Create KPI**.
5. In the **Key Performance Indicator (KPI)** dialog box, note that the **KPI base measure (value)** is defined by the **Margin** measure.
6. Under **Target**, ensure that **Measure** is selected and select **Target Margin**.
7. Set the first status threshold to **65%** and the second to **90%**.
8. Note the default icon style, and click **OK**.
9. On the **File** menu, click **Save All**.

Analyze Columns and Measures

1. On the **Model** menu, click **Analyze in Excel**.
2. In the **Analyze in Excel** dialog box, ensure that **Current Windows User** is selected and that the **(Default)** perspective is specified, and click **OK**.
3. In Excel, in the **PivotTable Fields** pane, in the **Internet Sales** table, select the **Profit** and **Margin** measures so that they are summarized in the **PivotTable**.
4. In the **PivotTable Fields** pane, in the **Customer** table, select **Full Name** so that the PivotTable shows profit and margin by customer.
5. In the **PivotTable Fields** pane, expand **KPIs**, expand **Margin**, and select **Status** so that an indicator shows the KPI status for each customer.
6. Close Excel without saving the workbook.
7. Keep Visual Studio open for the next demonstration.

Demonstration: Using a Specific Relationship in a Measure

Demonstration Steps

Retrieve a Value from an Unrelated Table

1. Ensure that you have completed the previous demonstration.
2. In Visual Studio, with the **Model.bim** pane open in data view, on the **Customer** tab, after the existing columns, double-click the **Add Column** header and enter the column name **First Fiscal Year**.
3. With the **First Fiscal Year** column selected, in the formula bar, enter the following DAX expression:

```
=LOOKUPVALUE('Order Date'[FiscalYear], [Date], [DateFirstPurchase])
```

   Wait for the table to finish updating, and note the calculated values in the new column.

Retrieve a Value from a Related Table

1. On the **Customer** tab, after the existing columns, double-click the **Add Column** header and enter the column name **Country-Region**.
2. With the **Country-Region** column selected, in the formula bar, enter the following DAX expression:

```
=RELATED('Geography'[CountryRegionCode])
```

   Wait for the table to finish updating, and note the calculated values in the new column.

Aggregate Related Values

1. On the **Customer** tab, after the existing columns, double-click the **Add Column** header and enter the column name **Order Count**.
2. With the **Order Count** column selected, in the formula bar, enter the following DAX expression:

```
=COUNTAX(RELATEDTABLE('Internet Sales'), [SalesOrderNumber])
```

   Wait for the table to finish updating, and note the calculated values in the new column.
3. On the **File** menu, click **Save All**.

Analyze Related Data

1. On the **Model** menu, click **Analyze in Excel**.
2. In the **Analyze in Excel** dialog box, ensure that **Current Windows User** is selected and that the **(Default)** perspective is specified, and click **OK**.
3. In Excel, in the **PivotTable Fields** pane, in the **Internet Sales** table, select the **Revenue** measure.
4. In the **PivotTable Fields** pane, in the **Customer** table, select **First Fiscal Year** so that the PivotTable shows sales revenue by customer, based on the first year in which customers made a purchase.
5. In the **PivotTable Fields** pane, in the **Customer** table, clear **First Fiscal Year** and select **Country-Region** so that the PivotTable shows sales revenue by customer country or region.
6. In the **PivotTable Fields** pane, in the **Customer** table, clear **Country-Region** and select **Order Count** so that the PivotTable shows sales revenue by customer, based on the number of orders placed.
7. Close Excel without saving the workbook.
8. Leave Visual Studio open for the next demonstration.

Demonstration: Using Time Intelligence

Demonstration Steps

Mark a Table As a Date Table

1. Ensure you have completed the previous demonstration.
2. In Visual Studio, with the **Model.bim** pane open in data view, click the **Order Date** tab.
3. On the **Table** menu, point to **Date** and click the **Mark As Date Table**.
4. In the **Mark As Date Table** dialog box, select the **Date** column, and click **OK**.

Create a Measure That Uses Time Intelligence

1. In the model designer, click the **Internet Sales** tab.
2. Click the first empty cell in the measure grid, then in the formula bar, enter the following DAX expression:

```
Previous Year Revenue:=CALCULATE(SUM([SalesAmount]), SAMEPERIODLASTYEAR('Order
Date'[Date]))
```

3. Select the cell containing the **Previous Year Revenue** measure, and press F4 to display the properties pane. Then set the **Format** property to **Currency**.
4. On the **File** menu, click **Save All**.

Analyze Data Using Time Intelligence

1. On the **Model** menu, click **Analyze in Excel**.
2. In the **Analyze in Excel** dialog box, ensure that **Current Windows User** is selected and that the **(Default)** perspective is specified, and click **OK**.
3. In Excel opens, in the **PivotTable Fields** pane, in the **Internet Sales** table, select the **Revenue** and **Previous Year Revenue** measures.
4. In the **PivotTable Fields** pane, in the **Order Date** table, drag **Calendar Date** to the **Rows** area so that the PivotTable shows sales revenue and previous year revenue for each year.

5. Expand **2007** and **2008**, and note that the revenue for each quarter in 2007 is shown as the previous year revenue for the same quarters in 2008.
6. Close Excel without saving the workbook.
7. Close Visual Studio.

Module Review and Takeaways

Review Question(s)

Question: Will time intelligence functionality be useful to you? What kinds of time-based analyses might you want to perform? Which are the relevant time intelligence functions that you might use to achieve this?

Answer: Answers will vary, depending on individuals and their experiences.

Lab Review Questions and Answers

Lab: Using DAX to Enhance a Tabular Data Model

Question and Answers

Lab Review

Question: Which DAX time intelligence function would you use to calculate a measure based on values for the end of the next quarter?

( ) ENDOFQUARTER

( ) NEXTQUARTER

( ) OPENINGBALANCEQUARTER

( ) PREVIOUSQUARTER

Answer:

( ) ENDOFQUARTER

(√) NEXTQUARTER

( ) OPENINGBALANCEQUARTER

( ) PREVIOUSQUARTER

Module 9

Performing Predictive Analysis with Data Mining

Contents:

Lesson 1

Overview of Data Mining

Contents:

Question and Answers

Put the following components of a data mining solution in order by numbering each to indicate the correct sequence from source to target.

	Steps
	Source data
	Analysis Services data source view
	Data Mining Structure
	Data Mining Model

Answer:

	Steps
1	Source data
2	Analysis Services data source view
3	Data Mining Structure
4	Data Mining Model

Lesson 2

Creating a Custom Data Mining Solution

Contents:

Question and Answers

Question: True or false? All the data mining algorithms implemented by default in Analysis Services can process all content types.

( ) True

( ) False

Answer:

( ) True

(√) False

Demonstration: Using the Data Mining Wizard

Demonstration Steps

Create a Data Mining Project in Visual Studio

1. Ensure that the **20768C-MIA-DC** and **20768C-MIA-SQL** virtual machines are both running, and then log on to **20768C-MIA-SQL** as **ADVENTUREWORKS\Student** with the password **Pa55w.rd**.
2. In the **D:\Demofiles\Mod09** folder, run **Setup.cmd** as **Administrator**.
3. In the **User Account Control** dialog box, click **Yes**, and then wait for the script to finish.
4. Start Visual Studio 2015 and on the **File** menu, point to **New**, and then click **Project**.
5. In the **New Project** dialog box, expand **Business Intelligence**, then click **Analysis Services**, then click **Analysis Services Multidimensional and Data Mining Project**.
6. In the **Name** box, type **Mine AW** then type **D:\Demofiles\Mod09** in the **Location** box, then click **OK**.
7. In Solution Explorer, right-click **Data Sources**, and then click **New Data Source**.
8. In the **Data Source Wizard**, on the **Welcome to the Data Source Wizard** page, click **Next**.
9. On the **Select how to define the connection** page, click **New**.
10. In the **Connection Manager** dialog box, in the **Server name** field, type **MIA-SQL**, in the **Select or enter a database name** drop-down list, click **AdventureWorksDW**, and then click **OK**.
11. On the **Select how to define the connection** page, click **Next**.
12. On the **Impersonation Information** page, click **Use a specific Windows user name and password**, then type the following credentials and click **Next**:
    - **User name**: ADVENTUREWORKS\ServiceAcct
    - **Password**: Pa55w.rd
13. On the **Completing the Wizard** page, click **Finish**.
14. In Solution Explorer, right-click **Data Source Views**, and then click **New Data Source View**.
15. In the **Data Source View Wizard**, on the **Welcome to the Data Source View Wizard** page, click **Next**.
16. On the **Select a Data Source** page, ensure that **Adventure Works DW** is selected, and then click **Next**.
17. On the **Select Tables and Views** page, in the **Available objects** list, click **ProspectiveBuyer (dbo)**, hold the **Ctrl** key and click **vTargetMail (dbo)**, and click the **>** button to move the selected objects to the **Included objects** list. Then click **Next**.

18. On the **Completing the Wizard** page, in the **Name** field, type **AW DW View**, and then click **Finish**.

Create a Data Mining Structure and a Data Mining Model Using the Data Mining Wizard

1. In Solution Explorer, right-click **Mining Structures**, click **New Mining Structure**.
2. In the **Data Mining Wizard**, on the **Welcome to the Data Mining Wizard** page, click **Next**.
3. On the **Select the Definition Method** page, ensure that **From existing relational database or data warehouse** is selected, and then click **Next**.
4. On the **Create the Data Mining Structure** page, ensure that **Create mining structure with mining model** is selected, ensure that **Microsoft Decision Trees** is selected, and then click **Next**.
5. On the **Select a Data Source View** page, select **AW DW View** and click **Next**.
6. On the **Specify Table Types** page, in the **vTargetMail** row, select the check box in the **Case** column, and then click **Next**.
7. On the **Specify the Training Data** page, in the **Mining Model Structure** table, select the following columns and click **Next**:
   - **BikeBuyer**: Predictable
   - **CustomerKey**: Key
   - **All other columns**: Input
8. On the **Specify Columns' Content and Data Type** page, click **Detect** and review the content type and data types found. Ensure that the content type of the **Bike Buyer** column is identified as **Discrete**. Then click **Next**.
9. On the **Create Testing Set** page, note that the **Percentage of data for testing** value is 30 percent, and then click **Next**.
10. On the **Completing the Wizard** page, in the **Mining structure name** box, type **Purchase Prediction**.
11. In the **Mining model name** box, type **Purchase Decision Tree**, and then click **Finish**.
12. On the **Build** menu, click **Deploy Mine AW**. When deployment completes, close the Deployment Progress – Mine AW window.
13. Leave Visual Studio open for the next demonstration.

Demonstration: Using the Data Mining Designer

Demonstration Steps

1. Ensure you have completed the previous demonstrations in this module.
2. In Visual Studio Solution Explorer, expand **Mining Structures** and double-click **Purchase Prediction.dmm**.
3. In the **Purchase Prediction.dmm [Design]** window, click **Mining Models**.
4. Right-click anywhere in the **Mining Models** tab, then click **New Mining Model**.
5. In the **New Mining Model** dialog box, type **Purchase Naive Bayes** in the **Model name** box.
6. In the **Algorithm name** box, select **Microsoft Naive Bayes**, then click **OK**.
7. In the **Microsoft Visual Studio** dialog box, read the information message about content types unsupported by the Microsoft Naïve Bayes algorithm, then click **Yes**.

8. In the **Mining Models** tab, in the **Purchase Naive Bayes** column, in the **Name Style** row, select **Ignore** to exclude the column from the model.
9. On the **Build** menu, click **Deploy Mine AW**. When deployment completes, close the Deployment Progress – Mine AW window.
10. Leave Visual Studio open for the next demonstration.

Demonstration: Using the Data Mining Model Viewer

Demonstration Steps

1. Ensure you have completed the previous demonstrations in this module.
2. Start **SQL Server Management Studio**.
3. In the **Connect to Server** dialog box, select **Analysis Services** in the **Server Type** box, then type **MIA-SQL** in the **Server name** box, then click **Connect**.
4. In Object Explorer, expand **Databases**, expand **Mine AW**, expand **Mining Structures**, right-click **Purchase Prediction**, then click **Browse**.
5. In the **Mining Model** drop-down list, ensure that **Purchase Naive Bayes** is selected, and on the **Dependency Network** tab, move the slider gradually from **All Links** to **Strongest Links** to see which factors are the strongest predictors that a customer will purchase a bike.
6. On the **Attribute Profiles** tab, view the color-coded indicators of the values for each column when compared to customers with a **Bike Buyer** value of **1** or **0**.
7. On the **Attribute Characteristics** tab, in the **Attribute** drop-down list, ensure that **Bike Buyer** is selected, and in the **Value** drop-down list, select **1.** Then view the probability for each other column value when the **Bike Buyer** value is **1**.
8. On the **Attribute Discrimination** tab, in the **Attribute** drop-down list, ensure that **Bike Buyer** is selected. In the **Value 1** drop-down list, select **1**, and in the **Value 2** drop-down list, select **0**. Then note how values for all other columns favor a particular **Bike Buyer** value.
9. In the **Mining Model** drop-down list, select **Purchase Decision Tree**, and on the **Decision Tree** tab, in the **Background** drop-down list, select **1**. Then view the decision tree to see how the other column values influence a value of **1** for **Bike Buyer**.
10. Close the Purchase Prediction [Browse] window, and leave SQL Server Management Studio open for the next demonstration.

Demonstration: Querying a Mining Model Using DMX

Demonstration Steps

1. In SSMS Object Explorer, right-click the **Purchase Prediction** mining structure and click **Build Prediction Query**.
2. In the query designer, in the **Mining Model** pane, click **Select Model**.
3. In the **Select Mining Model** dialog box, expand **Purchase Prediction**, click **Purchase Naive Bayes**, and click **OK**.
4. In the **Select Input Table(s)** pane, click **Select Case Table**.
5. In the **Select Table** dialog box, click **ProspectiveBuyer (dbo)**, and click **OK**.
6. Under the **Mining Model** pane, in the **Source** column, select **ProspectiveBuyer table**, and then in the **Field** column, select **EmailAddress**.

7. Under the row you just added, in the **Source** column, click **Purchase Naive Bayes** mining model, in the **Field** column, click **Bike Buyer**, and in the **Criteria/Argument** column, type **=1**.
8. Under the row you just added, in the **Source** column, click **Prediction Function**, in the **Field** column, click **PredictProbability**, in the **Alias** column, type **Purchase Probability**.
9. In the **Mining Model** pane, in the **Purchase Naïve Bayes** model, drag the **Bike Buyer** column to the **Criteria/Argument** column so it contains the value **[Purchase Naive Bayes].[Bike Buyer]**.
10. On the **Mining Model** menu, click **Query** to view the DMX code that has been generated.
11. On the **Mining Model** menu, click **Result** to view the query results. The query returns the email address of every prospective customer who is predicted to buy a bike, along with the probability (expressed as a percentage in fraction format) that this forecast is accurate.
12. Close the Purchase Prediction [Query] window, and leave SQL Server Management Studio open for the next demonstration.

Lesson 3

Validating a Data Mining Model

Contents:

Question and Answers

Question: Which data model accuracy chart type would you use to compare the accuracy of several data models?

( ) Scatter plot

( ) Classification matrix

( ) Lift chart

( ) Cross-validation report

( ) Profit chart

Answer:

( ) Scatter plot

( ) Classification matrix

(√) Lift chart

( ) Cross-validation report

( ) Profit chart

Demonstration: Viewing Accuracy Charts

Demonstration Steps

Create a Lift Chart

1. Ensure you have completed the previous demonstrations in this module.
2. In SSMS Object Explorer, right-click the **Purchase Prediction** mining structure and click **View Lift Chart**.
3. On the **Input Selection** tab, note that both mining models are selected with the **Bike Buyer** column as predictable, and that the test cases defined in the models themselves will be used for the validation.
4. Click the **Lift Chart** tab, and view the lift chart, which compares accuracy for the two mining models you have created against an ideal model by plotting the number of correct predictions against the number of cases in the overall sample. As the number of cases increases, the ideal model maintains an accuracy of 100 percent. However, the models you have created tend to become less accurate the more cases there are.
5. Review the scores in the Mining Legend pane to see which of your models is the most accurate for this test data.

Create a Profit Chart

1. In the **Chart type** drop-down list, select **Profit Chart**.
2. In the **Profit Chart Settings** dialog box, enter the following values to reflect a marketing campaign you are planning, and then click **OK**:
   - **Population**: 20,000 (this is the number of potential customers you plan to contact).
   - **Fixed cost**: 1,000 (this is the fixed cost of your marketing campaign).
   - **Individual cost**: 3 (this is the cost associated with contacting each customer).
   - **Revenue per individual**: 10 (this is the amount you expect a customer to spend if they respond positively to the campaign).

3. Review the chart and the **Mining Legend** pane to evaluate which mining model is likely to generate the most profitable marketing campaign based on the test data.

Create a Classification Matrix

1. Click the **Classification Matrix** tab.
2. Review the matrix, noting that for each model it shows the number of times the model predicted a **Bike Buyer** value of **1** or **0** on rows, with columns for the actual value of the **Bike Buyer** column in the test data.

Create a Cross Validation Report

1. Click the **Cross Validation** tab.
2. Enter the following values, and click **Get Results**:
   - **Fold Count**: 5 (this is the number of partitions used to group the data for analysis).
   - **Max Cases**: 5 (this is the number of cases to be analyzed).
   - **Target Attribute**: Bike Buyer (this is the predictable column to be evaluated).
   - **Target State**: 1 (this is the desired value for the target attribute).
   - **Target Threshold**: 0.1 (this is a value between 0 and 1 that indicates the level of accuracy required for a prediction to be considered correct).
3. View the resulting report, and note that for each mining model, the results include the following:
   - Classifications for true positives, false positives, true negatives, and false negatives.
   - The likely lift gained by using the model.
4. Close SQL Server Management Studio without saving any changes.

Lesson 4

Connecting to and Consuming a Data Mining Model

Contents:

Question and Answers

Question: True or false? You can include the results of multiple mining models in a single SSRS report.

( ) True

( ) False

Answer:

(√) True

( ) False

Demonstration: Using Data Mining Results in an SSRS Report

Demonstration Steps

1. Ensure you have completed the previous demonstrations in this module.
2. In Visual Studio, on the **File** menu, point to **New** and click **Project**.
3. In the **New Project** dialog box, expand **Business Intelligence**, select **Reporting Services**, select **Report Server Project Wizard**.
4. In the **Name** field, type **Purchase Prediction**, in the **Location** field, type **D:\Demofiles\Mod09**, and then click **OK**.
5. In the **Report Wizard**, on the **Welcome to the Report Wizard** page, click **Next**.
6. On the **Select the Data Source** page, select **New data source**, change the **Name** to **AWMine**, in the **Type** drop-down list, select **Microsoft SQL Server Analysis Services**, and click **Edit**.
7. In the **Connection Properties** dialog box, in the **Server name** field, type **MIA-SQL**, in the **Select or enter a database name** drop-down list, click **Mine AW**, and then click **OK**.
8. On the **Select the Data Source** page, click **Next**.
9. On the **Design the Query** page, click **Query Builder**.
10. In **Query Designer**, in the **Mining Model** pane, click **Select Model**.
11. In the **Select Mining Model** dialog box, expand **Purchase Prediction**, click **Purchase Naive Bayes**, and click **OK**.
12. In the **Select Input Table(s)** pane, click **Select Case Table**.
13. In the **Select Table** dialog box, click **ProspectiveBuyer (dbo)**, and click **OK**.
14. Under the **Mining Model** pane, in the **Source** column, select **ProspectiveBuyer table**, and then in the **Field** column, select **EmailAddress**.
15. Under the row you just added, in the **Source** column, click **Purchase Naive Bayes mining model**, in the **Field** column, click **Bike Buyer**, and in the **Criteria/Argument** column, type **=1**.
16. Under the row you just added, in the **Source** column, click **Prediction Function**, in the **Field** column, click **PredictProbability**, in the **Alias** column, type **Purchase Probability**.
17. in the **Mining Model** pane, from the **Purchase Naïve Bayes** model, drag the **Bike Buyer** column to the **Criteria/Argument** column so it contains the value **[Purchase Naive Bayes].[Bike Buyer]**, then click **OK**.
18. On the **Design the Query** page, click **Next**.
19. On the **Select the Report Type** page, verify that **Tabular** is selected, then click **Next**.
20. On the **Design the Table** page, in the **Available fields** box, click **EmailAddress**, then click **Details >**.

21. In the **Available fields** box, click **Purchase_Probability**, then click **Details >**. Click **Next**.
22. If the **Choose the Table Style** page appears, click **Next**.
23. If the **Choose the Deployment Location** page appears, click **Next**.
24. On the **Completing the Wizard** page, type **Likely Purchase Email Report** in the **Report name** box, then click **Finish**.
25. In the report designer, on the **Design** tab, click the table.
26. In the **Row Groups** pane, right-click the **(table1_Details_Group)** drop-down list and click **Group Properties**.
27. In the **Group Properties** dialog box, on the **Sorting** page, click **Add**, in the **Sort by** drop-down list, select **[Purchase_Probability]**, and in the **Order** drop-down list, click **Z to A**. Then click **OK**.
28. In the **Purchase Probability** column, right-click **[Purchase Probability]** and click **Text Box Properties**.
29. In the **Text Box Properties** dialog box, click **Number**, select **Percentage**, and click **OK**.
30. Click the **Preview** tab to review the report.
31. Close Visual Studio.

Module Review and Takeaways

Review Question(s)

Question: How have you seen, or can you envisage, data mining being used in organizations where you have worked?

Answer: Answers will vary. Data mining is increasingly used to predict values in scenarios, such as shopping cart analysis in an e-commerce site, or targeted advertising in social media and search services.

Lab Review Questions and Answers

Lab: Using Data Mining to Support a Marketing Campaign

Question and Answers

Lab Review

Question: Which type of data mining algorithm would you use to identify sequences in your data?

( ) A regression algorithm, such as the Microsoft Time Series algorithm.

( ) A sequence analysis algorithm, such as the Microsoft Sequence Clustering algorithm.

( ) A classification algorithm, such as the Microsoft Decision Trees algorithm.

( ) A clustering algorithm, such as the Microsoft Clustering algorithm.

( ) An association algorithm, such as the Microsoft Association algorithm.

Answer:

( ) A regression algorithm, such as the Microsoft Time Series algorithm.

(√) A sequence analysis algorithm, such as the Microsoft Sequence Clustering algorithm.

( ) A classification algorithm, such as the Microsoft Decision Trees algorithm.

( ) A clustering algorithm, such as the Microsoft Clustering algorithm.

( ) An association algorithm, such as the Microsoft Association algorithm.