



OFFICIAL MICROSOFT LEARNING PRODUCT

# 6158C

**Updating Your Microsoft® SQL Server®  
2005 Skills to SQL Server 2008**

*Companion Content*

Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The names of manufacturers, products, or URLs are provided for informational purposes only and Microsoft makes no representations and warranties, either expressed, implied, or statutory, regarding these manufacturers or the use of the products with any Microsoft technologies. The inclusion of a manufacturer or product does not imply endorsement of Microsoft of the manufacturer or product. Links may be provided to third party sites. Such sites are not under the control of Microsoft and Microsoft is not responsible for the contents of any linked site or any link contained in a linked site, or any changes or updates to such sites. Microsoft is not responsible for webcasting or any other form of transmission received from any linked site. Microsoft is providing these links to you only as a convenience, and the inclusion of any link does not imply endorsement of Microsoft of the site or the products contained therein.

© 2008 Microsoft Corporation. All rights reserved.

Microsoft and the trademarks listed at <http://www.microsoft.com/about/legal/en/us/IntellectualProperty/Trademarks/EN-US.aspx> are trademarks of the Microsoft group of companies. All other marks are property of their respective owners.

Product Number: 6158C

Released: 11/2008

## MICROSOFT LICENSE TERMS

### OFFICIAL MICROSOFT LEARNING PRODUCTS COURSEWARE – STUDENT EDITION – Pre-Release and Final Versions

---

These license terms are an agreement between Microsoft Corporation and you. Please read them. They apply to the licensed content named above, which includes the media on which you received it, if any. The terms also apply to any Microsoft

- updates,
- supplements,
- Internet-based services, and
- support services

for this licensed content, unless other terms accompany those items. If so, those terms apply.

**By using the licensed content, you accept these terms. If you do not accept them, do not use the licensed content.**

---

**If you comply with these license terms, you have the rights below.**

#### 1. OVERVIEW.

**Licensed Content.** The licensed content includes software, printed materials, academic materials (online and electronic), and associated media.

**License Model.** The licensed content is licensed on a per copy per device basis.

#### 2. INSTALLATION AND USE RIGHTS.

- Licensed Device.** The licensed device is the device on which you use the licensed content. You may install and use one copy of the licensed content on the licensed device.
- Portable Device.** You may install another copy on a portable device for use by the single primary user of the licensed device.
- Separation of Components.** The components of the licensed content are licensed as a single unit. You may not separate the components and install them on different devices.
- Third Party Programs.** The licensed content may contain third party programs. These license terms will apply to your use of those third party programs, unless other terms accompany those programs.

#### 3. PRE-RELEASE VERSIONS.

If the licensed content is a pre-release ("beta") version, in addition to the other provisions in this agreement, then these terms also apply:

- Pre-Release Licensed Content.** This licensed content is a pre-release version. It may not contain the same information and/or work the way a final version of the licensed content will. We may change it for the final, commercial version. We also may not release a commercial version. You will clearly and conspicuously inform any Students who participate in an Authorized Training Session and any Trainers who provide training in such Authorized Training Sessions of the foregoing; and, that you or Microsoft are under no obligation to provide them with any further content, including but not limited to the final released version of the Licensed Content for the Course.
- Feedback.** If you agree to give feedback about the licensed content to Microsoft, you give to Microsoft, without charge, the right to use, share and commercialize your feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software, licensed content, or service that includes the feedback. You will not give feedback that is subject to a license that requires Microsoft to license its software or documentation to third parties because we include your feedback in them. These rights survive this agreement.
- Confidential Information.** The licensed content, including any viewer, user interface, features and documentation that may be included with the licensed content, is confidential and proprietary to Microsoft and its suppliers.
  - Use.** For five years after installation of the licensed content or its commercial release, whichever is first, you may not disclose confidential information to third parties. You may disclose confidential information only to your employees and consultants who need to know the information. You must have written agreements with them that protect the confidential information at least as much as this agreement.
  - Survival.** Your duty to protect confidential information survives this agreement.

iii. **Exclusions.** You may disclose confidential information in response to a judicial or governmental order. You must first give written notice to Microsoft to allow it to seek a protective order or otherwise protect the information. Confidential information does not include information that

- becomes publicly known through no wrongful act;
- you received from a third party who did not breach confidentiality obligations to Microsoft or its suppliers; or
- you developed independently.

d. **Term.** The term of this agreement for pre-release versions is (i) the date which Microsoft informs you is the end date for using the beta version, or (ii) the commercial release of the final release version of the licensed content, whichever is first ("beta term").

e. **Use.** You will cease using all copies of the beta version upon expiration or termination of the beta term, and will destroy all copies of same in the possession or under your control.

f. **Copies.** Microsoft will inform Authorized Learning Centers if they may make copies of the beta version (in either print and/or CD version) and distribute such copies to Students and/or Trainers. If Microsoft allows to such distribution, you will follow any additional terms that Microsoft provides to you for such copies and distribution.

#### 4. ADDITIONAL LICENSING REQUIREMENTS AND/OR USE RIGHTS.

a. **Media Elements and Templates.** You may use images, clip art, animations, sounds, music, shapes, video clips and templates provided with the licensed content solely for your personal training use. If you wish to use these media elements or templates for any other purpose, go to [www.microsoft.com/permission](http://www.microsoft.com/permission) to learn whether that use is allowed.

b. **Academic Materials.** If the licensed content contains academic materials (such as white papers, labs, tests, datasheets and FAQs), you may copy and use the academic materials. You may not make any modifications to the academic materials and you may not print any book (either electronic or print version) in its entirety. If you reproduce any academic materials, you agree that:

- The use of the academic materials will be only for your personal reference or training use
- You will not republish or post the academic materials on any network computer or broadcast in any media;
- You will include the academic material's original copyright notice, or a copyright notice to Microsoft's benefit in the format provided below:

##### **Form of Notice:**

© 2008 Reprinted for personal reference use only with permission by Microsoft Corporation. All rights reserved.

Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the US and/or other countries. Other product and company names mentioned herein may be the trademarks of their respective owners.

c. **Distributable Code.** The licensed content may contain code that you are permitted to distribute in programs you develop if you comply with the terms below.

i. **Right to Use and Distribute.** The code and text files listed below are "Distributable Code."

- REDIST.TXT Files. You may copy and distribute the object code form of code listed in REDIST.TXT files.
- Sample Code. You may modify, copy, and distribute the source and object code form of code marked as "sample."
- Third Party Distribution. You may permit distributors of your programs to copy and distribute the Distributable Code as part of those programs.

ii. **Distribution Requirements.** For any Distributable Code you distribute, you must

- add significant primary functionality to it in your programs;
- require distributors and external end users to agree to terms that protect it at least as much as this agreement;
- display your valid copyright notice on your programs; and
- indemnify, defend, and hold harmless Microsoft from any claims, including attorneys' fees, related to the distribution or use of your programs.

**iii. Distribution Restrictions.** You may not

- alter any copyright, trademark or patent notice in the Distributable Code;
- use Microsoft's trademarks in your programs' names or in a way that suggests your programs come from or are endorsed by Microsoft;
- distribute Distributable Code to run on a platform other than the Windows platform;
- include Distributable Code in malicious, deceptive or unlawful programs; or
- modify or distribute the source code of any Distributable Code so that any part of it becomes subject to an Excluded License. An Excluded License is one that requires, as a condition of use, modification or distribution, that
  - the code be disclosed or distributed in source code form; or
  - others have the right to modify it.

- 5. INTERNET-BASED SERVICES.** Microsoft may provide Internet-based services with the licensed content. It may change or cancel them at any time. You may not use these services in any way that could harm them or impair anyone else's use of them. You may not use the services to try to gain unauthorized access to any service, data, account or network by any means.
- 6. SCOPE OF LICENSE.** The licensed content is licensed, not sold. This agreement only gives you some rights to use the licensed content. Microsoft reserves all other rights. Unless applicable law gives you more rights despite this limitation, you may use the licensed content only as expressly permitted in this agreement. In doing so, you must comply with any technical limitations in the licensed content that only allow you to use it in certain ways. You may not
- disclose the results of any benchmark tests of the licensed content to any third party without Microsoft's prior written approval;
  - work around any technical limitations in the licensed content;
  - reverse engineer, decompile or disassemble the licensed content, except and only to the extent that applicable law expressly permits, despite this limitation;
  - make more copies of the licensed content than specified in this agreement or allowed by applicable law, despite this limitation;
  - publish the licensed content for others to copy;
  - transfer the licensed content marked as 'beta' or 'pre-release' to any third party;
  - allow others to access or use the licensed content;
  - rent, lease or lend the licensed content; or
  - use the licensed content for commercial licensed content hosting services.
  - Rights to access the server software that may be included with the Licensed Content, including the Virtual Hard Disks does not give you any right to implement Microsoft patents or other Microsoft intellectual property in software or devices that may access the server.
- 7. BACKUP COPY.** You may make one backup copy of the licensed content. You may use it only to reinstall the licensed content.
- 8. TRANSFER TO ANOTHER DEVICE.** You may uninstall the licensed content and install it on another device for your personal training use. You may not do so to share this license between devices.
- 9. TRANSFER TO A THIRD PARTY.** You may not transfer those versions marked as 'beta' or 'pre-release' to a third party. For final versions, these terms apply: The first user of the licensed content may transfer it and this agreement directly to a third party. Before the transfer, that party must agree that this agreement applies to the transfer and use of the licensed content. The first user must uninstall the licensed content before transferring it separately from the device. The first user may not retain any copies.
- 10. EXPORT RESTRICTIONS.** The licensed content is subject to United States export laws and regulations. You must comply with all domestic and international export laws and regulations that apply to the licensed content. These laws include restrictions on destinations, end users and end use. For additional information, see [www.microsoft.com/exporting](http://www.microsoft.com/exporting).
- 11. NOT FOR RESALE SOFTWARE/LICENSED CONTENT.** You may not sell software or licensed content marked as "NFR" or "Not for Resale."



**12. ACADEMIC EDITION.** You must be a "Qualified Educational User" to use licensed content marked as "Academic Edition" or "AE." If you do not know whether you are a Qualified Educational User, visit [www.microsoft.com/education](http://www.microsoft.com/education) or contact the Microsoft affiliate serving your country.

**13. ENTIRE AGREEMENT.** This agreement, and the terms for supplements, updates, Internet-based services and support services that you use, are the entire agreement for the licensed content and support services.

**14. APPLICABLE LAW.**

- a. **United States.** If you acquired the licensed content in the United States, Washington state law governs the interpretation of this agreement and applies to claims for breach of it, regardless of conflict of laws principles. The laws of the state where you live govern all other claims, including claims under state consumer protection laws, unfair competition laws, and in tort.
- b. **Outside the United States.** If you acquired the licensed content in any other country, the laws of that country apply.

**15. LEGAL EFFECT.** This agreement describes certain legal rights. You may have other rights under the laws of your country. You may also have rights with respect to the party from whom you acquired the licensed content. This agreement does not change your rights under the laws of your country if the laws of your country do not permit it to do so.

**16. DISCLAIMER OF WARRANTY. THE LICENSED CONTENT IS LICENSED "AS-IS." YOU BEAR THE RISK OF USING IT. MICROSOFT GIVES NO EXPRESS WARRANTIES, GUARANTEES OR CONDITIONS. YOU MAY HAVE ADDITIONAL CONSUMER RIGHTS UNDER YOUR LOCAL LAWS WHICH THIS AGREEMENT CANNOT CHANGE. TO THE EXTENT PERMITTED UNDER YOUR LOCAL LAWS, MICROSOFT EXCLUDES THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.**

**17. LIMITATION ON AND EXCLUSION OF REMEDIES AND DAMAGES. YOU CAN RECOVER FROM MICROSOFT AND ITS SUPPLIERS ONLY DIRECT DAMAGES UP TO U.S. \$5.00. YOU CANNOT RECOVER ANY OTHER DAMAGES, INCLUDING CONSEQUENTIAL, LOST PROFITS, SPECIAL, INDIRECT OR INCIDENTAL DAMAGES.**

This limitation applies to

- anything related to the licensed content, software, services, content (including code) on third party Internet sites, or third party programs; and
- claims for breach of contract, breach of warranty, guarantee or condition, strict liability, negligence, or other tort to the extent permitted by applicable law.

It also applies even if Microsoft knew or should have known about the possibility of the damages. The above limitation or exclusion may not apply to you because your country may not allow the exclusion or limitation of incidental, consequential or other damages.

**Please note: As this licensed content is distributed in Quebec, Canada, some of the clauses in this agreement are provided below in French.**

**Remarque : Ce le contenu sous licence étant distribué au Québec, Canada, certaines des clauses dans ce contrat sont fournies ci-dessous en français.**

**EXONÉRATION DE GARANTIE.** Le contenu sous licence visé par une licence est offert « tel quel ». Toute utilisation de ce contenu sous licence est à votre seule risque et péril. Microsoft n'accorde aucune autre garantie expresse. Vous pouvez bénéficier de droits additionnels en vertu du droit local sur la protection des consommateurs, que ce contrat ne peut modifier. La ou elles sont permises par le droit locale, les garanties implicites de qualité marchande, d'adéquation à un usage particulier et d'absence de contrefaçon sont exclues.

**LIMITATION DES DOMMAGES-INTÉRÊTS ET EXCLUSION DE RESPONSABILITÉ POUR LES DOMMAGES.** Vous pouvez obtenir de Microsoft et de ses fournisseurs une indemnisation en cas de dommages directs uniquement à hauteur de 5,00 \$ US. Vous ne pouvez prétendre à aucune indemnisation pour les autres dommages, y compris les dommages spéciaux, indirects ou accessoires et pertes de bénéfices.

Cette limitation concerne:

- tout ce qui est relié au le contenu sous licence , aux services ou au contenu (y compris le code) figurant sur des sites Internet tiers ou dans des programmes tiers ; et
- les réclamations au titre de violation de contrat ou de garantie, ou au titre de responsabilité stricte, de négligence ou d'une autre faute dans la limite autorisée par la loi en vigueur.

Elle s'applique également, même si Microsoft connaissait ou devrait connaître l'éventualité d'un tel dommage. Si votre pays n'autorise pas l'exclusion ou la limitation de responsabilité pour les dommages indirects, accessoires ou de quelque nature que ce soit, il se peut que la limitation ou l'exclusion ci-dessus ne s'appliquera pas à votre égard.

**EFFET JURIDIQUE.** Le présent contrat décrit certains droits juridiques. Vous pourriez avoir d'autres droits prévus par les lois de votre pays. Le présent contrat ne modifie pas les droits que vous confèrent les lois de votre pays si celles-ci ne le permettent pas.

# Module 1

## An Introduction to SQL Server 2008

### Contents:

Lab Answer Keys

**2**

## Lab Answer Keys

### Lab: Reviewing the SQL Server Installation Center and SQL Server Books Online

#### Exercise 1: Exploring the SQL Server Installation Center

##### ► Task 1: Explore the Planning page

1. In the **6158C** Lab Launcher window, in the **6158C-MIA-SQL-01** virtual machine section, click **Launch**.
2. Log on to the virtual machine as Student, with a password of Pa\$\$w0rd:
  - a. Press ALT GR+DELETE, and then click **Student**.
  - b. In the **Password** box, type **Pa\$\$w0rd** and then press ENTER.
3. Open the SQL Server Installation Center:
  - a. Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, click **Configuration Tools**, and then click **SQL Server Installation Center**.
  - b. In the **User Account Control** dialog box, click **Continue**.
4. On the **Planning** page, review the links that are available.

Note that all of the links on this page, excluding the System Configuration Checker and the Install Upgrade Advisor links, are links to the relevant information on the MSDN® Web site. This computer is not connected to the Internet, so these links will not function correctly. You will browse some of the SQL Server documentation later in this lab.
5. Launch the System Configuration Checker, specifying the E:\Evaluation folder as the location of the installation media:
  - a. On the **Planning** page, click **System Configuration Checker**.
  - b. In the **Browse For Folder** dialog box, expand **Computer**, expand **Allfiles (E:)**, click **Evaluation**, and then click **OK**.
  - c. Wait for the **SQL Server 2008 Setup** dialog box to appear.
6. On the **Setup Support Rules** page, review the information that the System Configuration Checker returns and verify that no failures have occurred. The information on the page identifies problems that might occur when you install SQL Server on this computer.
7. Review the Microsoft .NET Application Security warning:
  - a. In the table, on the **Microsoft .NET Application Security** row, click the **Warning** link.
  - b. Review the information in the **Rule Check Result** dialog box, and then click **OK**.
8. Close the SQL Server 2008 Setup dialog box:
  - In the **SQL Server 2008 Setup** dialog box, click **OK**.

##### ► Task 2: Explore the Installation page

- In the SQL Server Installation Center window, review the links on the Installation page:
  - a. In the SQL Server Installation Center window, in the left pane, click **Installation**.
  - b. On the **Installation** page, review the links that are available.



Note that all of the links on this page, excluding the Search for product updates link, launch installation wizards to install or upgrade instances of SQL Server. The Search for product updates link launches Windows® Update.

► **Task 3: Explore the Maintenance page**

- In the SQL Server Installation Center window, review the links on the Maintenance page:
  - a. In the SQL Server Installation Center window, in the left pane, click **Maintenance**.
  - b. On the **Maintenance** page, review the links that are available.

Note that all of the links on this page launch installation wizards to upgrade, repair, or remove instances of SQL Server.

► **Task 4: Explore the Tools page**

1. In the SQL Server Installation Center window, review the links on the Tools page:
  - a. In the SQL Server Installation Center window, in the left pane, click **Tools**.
  - b. On the **Tools** page, review the links that are available.

Note that the System Configuration Checker link here launches the same tool that you reviewed in Task 1.

2. Review the information that is displayed in the Microsoft SQL Server 2008 Setup Discovery Report. This report describes all of the components of SQL Server that are installed on this computer:
  - On the **Tools** page, click **Installed SQL Server features discovery report**.
3. Close the Microsoft SQL Server 2008 Setup Discovery Report window.

► **Task 5: Explore the Resources page**

1. In the SQL Server Installation Center window, review the links on the Resources page:
  - a. In the SQL Server Installation Center window, in the left pane, click **Resources**.
  - b. On the **Resources** page, review the links that are available.

Note that all of the links on this page, excluding the License Agreement link, are links to the relevant information on the MSDN Web site. This computer is not connected to the Internet, so these links will not function correctly.

2. Review the SQLServer2008\_License\_DEV\_1033 License Terms:
  - a. On the **Resources** page, click **License Agreement**.
  - b. In the License Terms Windows Explorer window, double-click **SQLServer2008\_License\_DEV\_1033**.
  - c. Review the license terms.
3. Close WordPad and then close the License Terms Windows Explorer window.

► **Task 6: Explore the Advanced page**

1. In the SQL Server Installation Center window, review the links on the Advanced page:
  - a. In the SQL Server Installation Center window, in the left pane, click **Advanced**.
  - b. On the **Advanced** page, review the links that are available.

Note that all of the links on this page are used to perform advanced SQL Server installations.

2. Close the SQL Server Installation Center window.

## Exercise 2: Reviewing Tutorials in SQL Server Books Online

### ► Task 1: Review the Writing Transact-SQL Statements tutorial

1. Start SQL Server Books Online:
  - Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, click **Documentation and Tutorials**, and then click **SQL Server Books Online**.
2. Review the contents of the Tutorial: Writing Transact-SQL Statements tutorial, and decide whether it is suitable for new or experienced database administrators and developers:
  - a. In the **Contents** pane, expand **SQL Server 2008 Books Online**, and then click **SQL Server Tutorials**.
  - b. In the topic pane, in the **Database Engine** section, in the **Database Engine tutorials** list, click **Tutorial: Writing Transact-SQL Statements**.
  - c. Review the contents of the tutorial and decide whether it is suitable for new or experienced database administrators and developers.

### ► Task 2: Review the Using the hierarchyid Data Type tutorial

1. Review the contents of the Tutorial: Using the hierarchyid Data Type tutorial and decide whether it is suitable for new or experienced database administrators and developers:
  - a. In the **Contents** pane, click **SQL Server Tutorials**.
  - b. In the topic pane, in the **Database Engine** section, in the **Database Engine tutorials** list, click **Tutorial: Using the hierarchyid Data Type**.
  - c. Review the contents of the tutorial and decide whether it is suitable for new or experienced database administrators and developers.
2. Leave SQL Server Books Online open; you will use it in the next exercise.

## Exercise 3: Reviewing Topics in SQL Server Books Online

### ► Task 1: Review the What's New topic in SQL Server Books Online

- In SQL Server Books Online, review the Database Engine|Getting Started|What's New|Manageability Enhancements topic:
  - a. In SQL Server Books Online, click **Database Engine**, click **Getting Started (Database Engine)**, click **What's New (Database Engine)**, and then in the table, click **Manageability Enhancements (Database Engine)**.
  - b. Review some of the sections of the **Manageability Enhancements (Database Engine)** topic.

### ► Task 2: Review the Resource Governor content

1. In the **Manageability Enhancements** topic, review the content in the Resource Governor section:
  - In the topic pane, scroll down to the **Resource Governor** section and review the content.
2. Review the introductory text in the Managing SQL Server Workloads with Resource Governor topic:

- 
- Click **Managing SQL Server Workloads with Resource Governor**, and then review the topic introduction.
  3. Review the Introducing Resource Governor topic:
    - In the topic pane, click the **Introducing Resource Governor** link, and then review the information in the topic pane. This subject matter will be covered in detail later in the course.
  4. Close SQL Server Books Online.
  5. In the 6158C-MIA-SQL-01 on localhost - Virtual Machine Remote Control window, click **Close**.
  6. In the **Close** dialog box, click **Turn off machine and discard changes**, and then click **OK**.

# Module 2

## SQL Server 2008 Manageability Enhancements

### Contents:

Lab Answer Keys

**2**

# Lab Answer Keys

## Lab: Managing SQL Server 2008

### Exercise 1: Creating a Central Management Server

#### ► Task 1: Create a Central Management Server

1. In the **6158C** Lab Launcher window, in the **6158C-MIA-SQL-02** virtual machine section, click **Launch**.
2. Log on to the virtual machine as Student, with a password of Pa\$\$w0rd:
  - a. Press ALT GR+DELETE, and then click **Student**.
  - b. In the **Password** box, type **Pa\$\$w0rd** and then press ENTER.
3. Start SQL Server Management Studio. When you are prompted, connect to the MIAMI instance of the database engine by using Windows® authentication:
  - a. Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, and then click **SQL Server Management Studio**.
  - b. In the **Connect to Server** dialog box, verify the following settings, and then click **Connect**:
    - **Server type**: Database Engine
    - **Server name**: MIAMI
    - **Authentication**: Windows Authentication
4. Open the Registered Servers window, and then create a new Central Management Server with the server name of MIAMI and using Windows authentication:
  - a. On the **View** menu, click **Registered Servers**.
  - b. In the Registered Servers window, expand **Database Engine**, right-click **Central Management Servers**, and then click **Register Central Management Server**.
  - c. In the **New Server Registration** dialog box, on the **General** tab, in the **Server name** box, type **MIAMI**.
  - d. In the **Authentication** box, verify that **Windows Authentication** is selected, and then click **Save**.
5. Verify that the MIAMI Central Management Server has been created:
  - In the Registered Servers window, expand the **Central Management Servers** folder, and then verify that the MIAMI Central Management Server has been created.

#### ► Task 2: Create a server group

- On the MIAMI server, create a new server group named Management Group:
  - a. In the Registered Servers window, right-click the **MIAMI** server, and then click **New Server Group**.
  - b. In the **New Server Group Properties** dialog box, in the **Group name** box, type **Management Group**, and then click **OK**.

Notice the new folder that is titled Management Group under the MIAMI Central Management Server.



**Note:** You can also use the Local Server Groups folder to query and manage multiple servers. This folder is unique to each logon.

► **Task 3: Add SQL Server instances to a server group**

1. Register MIAMI\SQLINSTANCE2 as part of the Management Group server group:
  - a. In the Registered Servers window, right-click the **Management Group** folder, and then click **New Server Registration**.
  - b. In the **New Server Registration** dialog box, on the **General** tab, in the **Server name** box, type **MIAMI\SQLINSTANCE2**, and then click **Save**.
2. Register MIAMI\SQLINSTANCE3 as part of the Management Group server group:
  - a. In the Registered Servers window, right-click the **Management Group** folder, and then click **New Server Registration**.
  - b. In the **New Server Registration** dialog box, on the **General** tab, in the **Server name** box, type **MIAMI\SQLINSTANCE3**, and then click **Save**.

► **Task 4: Execute multiserver queries**

1. Open a new query window that is connected to the MIAMI server:
  - In Object Explorer, right-click **MIAMI**, and then click **New Query**.
2. In the Query Editor, write and execute a query to list all of the databases on the server:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE master
SELECT * FROM sysdatabases
```

3. Review the results. Notice that the results include only databases on the MIAMI server.
4. Close the Query Editor without saving changes.
5. Open a new query window that is connected to the Management Group:
  - In the Registered Servers window, right-click the **Management Group** folder, and then click **New Query**.
6. In the Query Editor, write and execute a query to list all of the databases on the server:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE master
SELECT * FROM sysdatabases
```

7. Review the results.

**Note:** When you execute a query across multiple servers, the results are merged into a single result set and the **ServerName** and **Login** columns are added. You can change this default behavior by using the **Options** dialog box.

8. Change the default display options for the multiserver query results so that a separate results pane will be displayed for each server:
  - a. On the **Tools** menu, click **Options**.

- b. In the **Options** dialog box, expand **Query Results**, expand **SQL Server**, and then click **Multiserver Results**.
  - c. In the details pane, change the **Merge results** value to **False**, and then click **OK**.
9. Return to the Query Editor, and execute the multiserver query again. Notice that the results are now displayed as two separate result sets.
10. Close the Query Editor without saving changes.
11. Keep SQL Server Management Studio open; you will use it in the next exercise.

## Exercise 2: Creating and Applying a Policy

### ► Task 1: Create policy conditions

1. In Object Explorer, review the preconfigured facets that are available:
  - In Object Explorer, expand **Management**, expand **Policy Management**, expand **Facets**, and then review the preconfigured facets that are available.
2. Use the Facet Properties dialog box for the Surface Area Configuration facet to review the properties of the facet:
  - a. Right-click the **Surface Area Configuration** facet, and then click **Properties**.
  - b. Review the properties that are a part of this facet.
  - c. Close the **Facet Properties – Surface Area Configuration** dialog box.
3. Create a new condition for the Surface Area Configuration facet named Restrict SQL Mail. Set the condition expression to be @SqlMailEnabled = False:
  - a. In Object Explorer, right-click **Conditions**, and then click **New Condition**.
  - b. In the **Create New Condition** dialog box, on the **General** page, in the **Name** box, type **Restrict SQL Mail**.

**Tip:** The red circle with the white X at the top of the window provides hints about how to resolve errors in the current window.

- c. In the **Facet** box, click **Surface Area Configuration**.
  - d. In the **Expression** pane, in the **Field** box, click **@SqlMailEnabled**, and then verify that the **Operator** is set to **=**.
  - e. In the **Value** box, click **False**, and then click **OK**.
4. Create a new condition for the View Options facet named Encrypt View Definitions. Set the condition expression to be @IsEncrypted = True:
  - a. In Object Explorer, right-click **Conditions**, and then click **New Condition**.
  - b. In the **Create New Condition** dialog box, on the **General** page, in the **Name** box, type **Encrypt View Definitions**.
  - c. In the **Facet** box, click **View Options**.
  - d. In the **Expression** pane, in the **Field** box, click **@IsEncrypted**, and then verify that the **Operator** is set to **=**.
  - e. In the **Value** box, click **True**, and then click **OK**.

► **Task 2: Create and test a server-wide policy**

1. Create a new policy named SQL Mail Disabled that uses the Restrict SQL Mail condition. Store the policy in a new category named Surface Area Policies:
  - a. In Object Explorer, right-click the **Policies** folder, and then click **New Policy**.
  - b. In the **Create New Policy** dialog box, on the **General** page, in the **Name** box, type **SQL Mail Disabled**.
  - c. In the **Check condition** box, click **Restrict SQL Mail**.

**Note:** The drop-down list is organized into facets. The **Restrict SQL Mail** condition is under the **Surface Area Configuration** facet.

- d. In the **Create New Policy** dialog box, click the **Description** page.
  - e. Next to the **Category** box, click **New**.
  - f. In the **Create new category** dialog box, type **Surface Area Policies**, and then click **OK**.
  - g. In the **Create New Policy** dialog box, click **OK**.
2. Evaluate the SQL Mail Disabled policy, review the results, and then view the target details:
  - a. In Object Explorer, expand **Policies**, right-click the **SQL Mail Disabled** policy, and then click **Evaluate**.
  - b. In the **Evaluate Policies - SQL Mail Disabled** dialog box, review the results, and then in the **Details** column, click the **View** hyperlink.

Notice that the policy verified that the **Actual Value** for the **@SqlMailEnabled** field was **False** and matched the **Expected Value**.

3. Close all open dialog boxes:
  - In the **Results Details View** dialog box, click **Close**, and then in the **Evaluate Policies - SQL Mail Disabled** dialog box, click **Close**.

► **Task 3: Create and test a policy to which a single database subscribes**

1. Create and enable a new policy named Encrypted Views that uses the Encrypt View Definition condition. Set the policy to run against only the AdventureWorks2008 database by creating a new condition named AdventureWorks DB that checks that the name of the database is 'AdventureWorks2008'. Set the evaluation mode to automatically prevent policy violations and store the policy in a new category named AWDDatabase Policies:
  - a. In Object Explorer, right-click the **Policies** folder, and then click **New Policy**.
  - b. In the **Create New Policy** dialog box, on the **General** page, in the **Name** box, type **Encrypted Views**.
  - c. In the **Check condition** box, click **Encrypt View Definitions**.

**Note:** The Encrypt View Definitions condition is under the View Options facet.

- d. In the **Against targets** box, locate **in Every Database** on the second line, click the arrow next to **Every**, and then click **New condition**.
  - e. In the **Create New Condition** dialog box, in the **Name** box, type **AdventureWorks DB**, and then verify that the **Facet** is **Database**.

- f. In the **Expression** pane, in the **Field** box, click **@Name**, and then verify that the **Operator** is set to **=**.
- g. In the **Value** box, type **'AdventureWorks2008'**.
- h. Press ENTER, and then click **OK**.

**Tip:** Include the single quotes around AdventureWorks2008.

- i. In the **Evaluation Mode** box, click **On change: prevent**.
  - j. Select the **Enabled** check box.
  - k. Click the **Description** page.
  - l. On the **Description** page, next to the **Category** box, click **New**.
  - m. In the **Create new category** dialog box, type **AWDatabase Policies**, and then click **OK**.
  - n. In the **Create New Policy** dialog box, click **OK**.
2. Evaluate the **Encrypted Views** policy:
- a. In Object Explorer, right-click the **Encrypted Views** policy, and then click **Evaluate**.
  - b. In the **Evaluate Policies - Encrypted Views** dialog box, review the results.

**Note:** The result is a failure for all nonencrypted views.

3. View the target details. Note that only views in the AdventureWorks2008 database were analyzed and that several of them are unencrypted:
- a. In the **Target details** section, point to the **Target** column, and then view the information in the ScreenTip that appears. Note the name of the server, database, and views that the policy has analyzed.
  - b. In the second row, in the **Details** column, click the **View** hyperlink. The **Actual Value** column is **False**, so we know that the view definition is not encrypted.
4. Close all open dialog boxes:
- In the **Results Detailed View** dialog box, click **Close**, and then in the **Evaluate Policies - Encrypted Views** dialog box, click **Close**.

#### ► Task 4: Verify policy subscription and functionality

1. In Object Explorer, right-click the **Policy Management** folder, and then click **Manage Categories**.
2. Configure the Surface Area Policies category to be enforced on all databases, configure the AWDatabase Policies category not to be enforced on all databases, and then click OK:
  - a. In the **Manage Policy Categories** dialog box, in the **Mandate Database Subscriptions** column, verify that the check box is selected for the **Surface Area Policies** category.
  - b. Clear the **Mandate Database Subscriptions** check box for the **AWDatabase Policies** category, and then click **OK**.

**Note:** The **Mandate Database Subscriptions** setting will automatically apply this policy to all target object types that are defined in the policy.

3. In Object Explorer, expand **Databases**, right-click **AdventureWorks2008**, point to **Policies**, and then click **Categories**.
4. Review the Surface Area Policies category for the AdventureWorks2008 database:
  - In the **Categories** dialog box, expand **Surface Area Policies**.

Notice that the policy has a check showing that the database is subscribed to the policy, but you cannot change the **Subscribed** setting. This is because the check box in the **Mandate Database Subscriptions** column is selected for this category.

5. Review and subscribe to the AWDatabase Policies category:
  - a. Expand the **AWDatabase Policies** category, and then review the policy settings.
  - b. Select the **Subscribed** check box for the **AWDatabase Policies** category, and then click **OK**.
6. Open a new query window connected to the AdventureWorks2008 database:
  - In Object Explorer, right-click the **AdventureWorks2008** database, and then click **New Query**.
7. In the Query Editor, write and execute a query to create an encrypted view named **TestEncrypt** that returns the following fields:
  - Production.Product.Name
  - Production.ProductReview.ReviewDate
  - Production.ProductReview.ReviewerName
- In the Query Editor, type the code in the following code example, and then click **Execute**:

**Note:** This code is also available in the E:\folder\Labfiles\Solution\Ex2.sql file.

```
USE AdventureWorks2008;
GO
CREATE VIEW TestEncrypt WITH ENCRYPTION
AS
SELECT Production.Product.Name,
Production.ProductReview.ReviewDate,
Production.ProductReview.ReviewerName
FROM Production.Product INNER JOIN Production.ProductReview
ON Production.Product.ProductID =
Production.ProductReview.ProductID
```

8. Verify that the **TestEncrypt** view has been created:
  - In Object Explorer, expand the **AdventureWorks2008** database, expand **Views**, and then verify the creation of the view.
9. Open a new query window connected to the AdventureWorks2008 database:
  - In Object Explorer, right-click the **AdventureWorks2008** database, and then click **New Query**.
10. In the Query Editor, write and execute a query to create a view named **TestNoEncrypt** that returns the following fields:
  - Production.Product.Name
  - Production.ProductReview.ReviewDate



- Production.ProductReview.ReviewerName
- In the Query Editor, type the code in the following code example, and then click **Execute**:

**Note:** This code is also available in the E:\folder\Labfiles\Solution\Ex2.sql file.

```
USE AdventureWorks2008;
GO
CREATE VIEW TestNoEncrypt
AS
SELECT Production.Product.Name,
Production.ProductReview.ReviewDate,
Production.ProductReview.ReviewerName
FROM Production.Product INNER JOIN Production.ProductReview
ON Production.Product.ProductID =
Production.ProductReview.ProductID
```

Notice that the batch was aborted because it violates the **Encrypted Views** policy.

11. In Object Explorer, right-click **MIAMI**, point to **Policies**, and then click **View**. Notice that the **SQL Mail Disabled** policy setting is not enabled because the policy is set for manual execution.
12. Review the history of the **SQL Mail Disabled** policy:
  - In the **History** column, click the **History** link, and then review the information that is displayed. This history was created when you tested the policy earlier in the lab.
13. Close all dialog boxes, and then close SQL Server Management Studio without saving changes:
  - a. In the **Log File Viewer** dialog box, click **Close**.
  - b. In the **View Policies - MIAMI** dialog box, click **Close**.
  - c. In SQL Server Management Studio, on the **File** menu, click **Exit**. In the **Microsoft SQL Server Management Studio** dialog box, click **No**.
14. In the 6158C-MIA-SQL-02 on localhost - Virtual Machine Remote Control window, click **Close**.
15. In the **Close** dialog box, click **Turn off machine and discard changes**, and then click **OK**.

# Module 3

## SQL Server 2008 Performance Enhancements

### Contents:

Lab Answer Keys

**2**

# Lab Answer Keys

## Lab: Optimizing SQL Server 2008

### Exercise 1: Using Resource Governor

#### ► Task 1: Prepare the lab environment

1. In the **6158C** Lab Launcher window, in the **6158C-MIA-SQL-03** virtual machine section, click **Launch**.
2. Log on to the virtual machine as Student, with a password of Pa\$\$w0rd:
  - a. Press ALT GR+DELETE, and then click **Student**.
  - b. In the **Password** box, type **Pa\$\$w0rd** and then press ENTER.
3. Run the Setup.bat file located in the E:\Labfiles\Starter folder. This batch file creates a view and a stored procedure that you will use in this lab:
  - a. Click **Start**, click **Computer**, and then navigate to the **E:\Labfiles\Starter** folder.
  - b. Double-click **Setup.bat**.
4. Close Windows® Explorer.

#### ► Task 2: Create a resource pool

1. Start SQL Server Management Studio. When you are prompted, connect to the MIAMI instance of the database engine by using Windows® authentication:
  - a. Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, and then click **SQL Server Management Studio**.
  - b. In the **Connect to Server** dialog box, verify the following settings, and then click **Connect**:
    - **Server type**: Database Engine
    - **Server name**: MIAMI
    - **Authentication**: Windows Authentication
2. Enable the Resource Governor:
  - a. In Object Explorer, expand the **Management** node.
  - b. Right-click **Resource Governor**, and then click **Enable**.
3. Create a new resource pool named **Low Importance Pool**. Set the properties of Low Importance Pool so that it uses a minimum of 0 percent CPU usage, a maximum of 10 percent CPU usage, a minimum of 0 percent memory usage, and a maximum of 10 percent memory usage:
  - a. Right-click **Resource Governor**, and then click **New Resource Pool**.
  - b. In the **Resource Governor Properties** dialog box, in the first empty line of the **Resource pools** grid, enter the following values, and then click **OK**:
    - **Name**: Low Importance Pool
    - **Minimum CPU %**: 0
    - **Maximum CPU %**: 10
    - **Minimum Memory %**: 0

- **Maximum Memory %:** 10

**Note:** The resource pool enables you to create limits for resource usage. These limits are applied if other applications require the resources. If no other applications are running, the limits are not applied.

► **Task 3: Create a workload group**

- Create a new workload group named **Low Importance Group** for the Low Importance Pool resource pool. Set the properties of the group so that it uses an importance setting of low, a maximum request value of 0, a CPU time of 0, a 25 percent memory grant, a grant time-out of 0, and a 0 degree of parallelism:
  - a. In Object Explorer, expand **Resource Governor**, expand **Resource Pools**, and then expand **Low Importance Pool**.
  - b. Right-click **Workload Groups**, and then click **New Workload Group**.
  - c. In the **Resource Governor Properties** dialog box, in the **Workload groups for resource pool** grid, enter the following values, and then click **OK**:
    - **Name:** Low Importance Group
    - **Importance:** Low
    - **Maximum requests:** 0
    - **CPU Time (sec):** 0
    - **Memory Grant %:** 25
    - **Grant Time-out (sec):** 0
    - **Degree of Parallelism:** 0

► **Task 4: Create a classifier function**

1. Open a new query window that is connected to the master database:
  - In Object Explorer, expand **Databases**, expand **System Databases**, rightclick **master**, and then click **New Query**.
2. In the Query Editor, write and execute a query to create a function that classifies client applications that have application names that contain the text "Low Importance Application" into the Low Importance Group workload group:
  - a. In the Query Editor, type the code in the following code example, and then click **Execute**:

**Note:** This code is also available from the E:\Labfiles\Starter\Function.sql file.

```
CREATE FUNCTION dbo.fn_ClassifyApps() RETURNS sysname
WITH SCHEMABINDING
AS
BEGIN
    DECLARE @ret sysname
    IF (APP_NAME() LIKE '%Low Importance Application%')
        SET @ret='Low Importance Group'
    RETURN @ret
END
GO
```

```
ALTER RESOURCE GOVERNOR
WITH (CLASSIFIER_FUNCTION = dbo.fn_ClassifyApps)
ALTER RESOURCE GOVERNOR RECONFIGURE
```

**Note:** Classifier functions group applications into workload groups so that you can then apply limits to their resource usage.

- b. Close the Query Editor without saving changes.

► **Task 5: Add the Resource Pool Stats counter to Performance Monitor**

1. Open the Performance Monitor Tool and delete all of the default counters:
  - a. On the **Start** menu, point to **Administrative Tools**, and then click **Reliability and Performance Monitor**.
  - b. In the **User Account Control** dialog box, click **Continue**.
  - c. In the **Reliability and Performance Monitor** dialog box, click **Performance Monitor**.
  - d. In the Performance Monitor tool, on the toolbar, click the **Delete** button to remove each of the default counters.
2. Use the Performance Monitor Tool to add the Low Importance Pool instance of the CPU usage % counter in SQLServer:Resource Pool Stats group:
  - a. On the toolbar, click the **Add** button to add a counter to the Performance Monitor display.
  - b. In the **Add Counters** dialog box, in the list of counters, locate and expand **SQLServer:Resource Pool Stats**.
  - c. In the list of counters, click **CPU usage %**.
  - d. In the list of instances, click **Low Importance Pool**.
  - e. Click **Add**, and then click **OK**.

► **Task 6: Review the configuration of the workload group and resource pool**

1. Run the LowImportanceApp.vbs application in the E:\Labfiles\Starter folder:
  - a. Click **Start**, click **Computer**, and then navigate to the **E:\Labfiles\Starter** folder.
  - b. Double-click **LowImportanceApp.vbs**, and then click **OK** to start the application.
2. In the Reliability and Performance Monitor window, review the resource usage.

Notice that CPU usage for the **Low Importance Pool** instance rises and is not limited by Resource Governor because there are no workloads from other resource pools.
3. Run the MissionCriticalApp.vbs application in the **E:\Labfiles\Starter** folder:
  - a. In Windows Explorer, double-click **MissionCriticalApp.vbs**, and then click **OK** to start the application.

Notice that, whenever the default workload CPU usage increases, Resource Governor limits the CPU usage of the Low Importance Pool workload.

**Note:** You can use Resource Governor to limit resources for specific workloads; this enables more important workloads to achieve optimum performance.



- b. Wait for the two Microsoft Visual Basic®, Scripting Edition applications to finish, and in the message boxes that are displayed, click **OK**.
4. Close the Reliability and Performance Monitor window, and then close Windows Explorer.

## Exercise 2: Using the Data Collector

### ► Task 1: Configure a management data warehouse

1. In SQL Server Management Studio, start the SQL Server Agent service:
  - a. In SQL Server Management Studio, in Object Explorer, right-click **SQL Server Agent**, and then click **Start**.
  - b. In the **User Account Control** dialog box, click **Continue**.
  - c. In the **Microsoft SQL Server Management Studio** dialog box, click **Yes** to confirm that you want to start the SQLSERVERAGENT service on MIAMI.
2. Start the Configure Management Data Warehouse Wizard:
  - In Object Explorer, expand the **Management** node, right-click **Data Collection**, and then click **Configure Management Data Warehouse**.
3. Use the Configure Management Data Warehouse Wizard to create a new management data warehouse for storing data collection set results. Create a new database named MgmtDW to store collected performance data. Map the built-in Administrators group to the mdw\_admin role, and then close the Configure Management Data Warehouse Wizard:
  - a. On the **Welcome to the Configure Management Data Warehouse Wizard** page, click **Next**.
  - b. On the **Select configuration task** page, click **Create or upgrade a management data warehouse**, and then click **Next**.
  - c. On the **Configure Management Data Warehouse Storage** page, click **New**.
  - d. In the **New Database** dialog box, in the **Database name** box, type **MgmtDW** and then click **OK**.
  - e. On the **Configure Management Data Warehouse Storage** page, click **Next**.
  - f. On the **Map Logins and Users** page, click **New Login**.
  - g. In the **Login – New** dialog box, click **Search**.
  - h. In the **Select User or Group** dialog box, click **Advanced**.
  - i. In the **Select User or Group** dialog box, click **Find Now**.
  - j. In the **Search Results** box, click **Administrator**, and then click **OK**.
  - k. In the **Select User or Group** dialog box, click **OK**.
  - l. In the **Login – New** dialog box, click **OK**.
  - m. On the **Map Logins and Users** page, in the **Users mapped to this login** list, select the **MIAMI\Administrators** check box.
  - n. In the **Database role membership for: MgmtDW** list, select the **mdw\_admin** check box, and then click **Next**.
  - o. On the **Complete the Wizard** page, review the configuration, and then click **Finish**. When configuration is complete, click **Close**.

4. Start the Configure Management Data Warehouse Wizard again:
  - In Object Explorer, right-click **Data Collection**, and then click **Configure Management Data Warehouse**.
5. Use the Configure Management Data Warehouse Wizard to set up a data collection to the MgmtDW management data warehouse by using the MIAMI server. Specify that you want the cache directory to be located in the E:\Labfiles\Starter\Mgmt folder:
  - a. On the **Welcome to the Configure Management Data Warehouse Wizard** page, click **Next**.
  - b. On the **Select configuration task** page, click **Set up data collection**, and then click **Next**.
  - c. On the **Configure Management Data Warehouse Storage** page, click the ellipsis (...) next to the **Server name** box, and then click **Connect** to connect to the MIAMI server.
  - d. In the **Database name** box, click **MgmtDW**, in the **Cache directory** box, type **E:\Labfiles\Starter\Mgmt**, and then click **Next**.
  - e. On the **Complete the Wizard** page, review the configuration, and then click **Finish**. When configuration is complete, click **Close**.

► **Task 2: Modify the configuration of the System Data collection sets**

1. In Object Explorer, under the **Management** node, expand **Data Collection**, and then expand **System Data Collection Sets**.
2. In the Disk Usage properties, specify that you want to retain data in the management data warehouse for 90 days:
  - a. Right-click **Disk Usage**, and then click **Properties**.
  - b. In the **Data Collection Set Properties** dialog box, in the **Retain data for** box, type **90** and then click **OK**.

► **Task 3: View collected data**

1. Collect and upload the data for the disk usage set:
  - a. In Object Explorer, right-click **Disk Usage**, and then click **Collect and Upload Now**.
  - b. When the collection and uploading is complete, click **Close**.
2. View the disk usage report for the AdventureWorks2008 database:
  - a. Right-click **Data Collection**, point to **Reports**, point to **Management Data Warehouse**, and then click **Disk Usage Summary**.
  - b. Click **AdventureWorks2008**, and then examine the graphical view of disk usage.

**Note:** In a live environment, you would need to wait for several hours for the Data Collector to collect meaningful data.

3. Close SQL Server Management Studio:
  - a. On the **File** menu, click **Exit**.
  - b. If you are prompted to save changes, click **No**.
4. In the 6158C-MIA-SQL-03 on localhost - Virtual Machine Remote Control window, click **Close**.
5. In the **Close** dialog box, click **Turn off machine and discard changes**, and then click **OK**.

# Module 4

## SQL Server 2008 Security Enhancements

### Contents:

Lab Answer Keys

**2**

# Lab Answer Keys

## Lab: Securing and Auditing a SQL Server 2008 Database

### Exercise 1: Using Transparent Data Encryption

#### ► Task 1: Back up the AdventureWorks2008 database

1. In the **6158C** Lab Launcher window, in the **6158C-MIA-SQL-04** virtual machine section, click **Launch**.
2. Log on to the virtual machine as **Student**, with a password of **Pa\$\$w0rd**:
  - a. Press ALT GR+DELETE, and then click **Student**.
  - b. In the **Password** box, type **Pa\$\$w0rd** and then press ENTER.
3. Start SQL Server Management Studio. When you are prompted, connect to the MIAMI instance of the database engine by using Windows® authentication:
  - a. Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, and then click **SQL Server Management Studio**.
  - b. In the **Connect to Server** dialog box, verify the following settings, and then click **Connect**:
    - **Server type**: Database Engine
    - **Server name**: MIAMI
    - **Authentication**: Windows Authentication
4. Back up the AdventureWorks2008 database to the E:\Labfiles\Starter\AdventureWorks.bak folder and remove existing backup locations that are listed:
  - a. In Object Explorer, expand **Databases**, right-click **AdventureWorks2008**, point to **Tasks**, and then click **Back Up**.
  - b. In the **Destination** section, click **Remove** to remove any existing backup destinations, and then click **Add**.
  - c. In the **Select Backup Destination** dialog box, in the **File name** box, type **E:\Labfiles\Starter\AdventureWorks.bak**, and then click **OK**.
  - d. Click **OK** to back up the database, and then click **OK** when the backup operation has completed successfully.

#### ► Task 2: Restore the AdventureWorks2008 database on another instance of SQL Server

1. Connect to the MIAMI\SQLInstance2 instance of the database engine by using Windows authentication:
  - a. In Object Explorer, click **Connect**, and then click **Database Engine**.
  - b. In the **Connect to Server** dialog box, verify the following settings, and then click **Connect**:
    - **Server type**: Database Engine
    - **Server name**: MIAMI\SQLInstance2
    - **Authentication**: Windows Authentication

2. Restore the AdventureWorks2008 database from the E:\Labfiles\Starter\AdventureWorks.bak backup file to a new database named AdventureWorksNonEncrypt on MIAMI\SQLInstance2. You should overwrite the existing database and use file destinations as shown in the following table:

Original file name	Restore as
AdventureWorks2008_Data	E:\Labfiles\Starter\ AdventureWorksNonEncrypt.mdf
AdventureWorks2008_Log	E:\Labfiles\Starter\ AdventureWorksNonEncrypt_log.ldf
FileStreamDocuments	E:\Labfiles\Starter\ AdventureWorksNonEncrypt_2.Documents

- a. On the MIAMI\SQLInstance2 connection, right-click **Databases**, and then click **Restore Database**.
- b. In the **Restore Database** dialog box, in the **Destination for restore** section, in the **To database** box, type **AdventureWorksNonEncrypt**.
- c. In the **Source for restore** section, click **From device**, and then click the ellipsis (...) button.
- d. In the **Specify Backup** dialog box, click **Add**, navigate to the **E:\Labfiles\Starter** folder, double-click **AdventureWorks.bak**, and then click **OK**.
- e. In the **Select the backup sets to restore** grid, in the **Restore** column, select the **AdventureWorks2008-Full Database Backup** check box.
- f. On the **Options** page, select the **Overwrite the existing database (WITH REPLACE)** check box.
- g. In the **Restore the database files as** grid, type the **Restore As** file destinations as shown in the previous table.
- h. Click **OK** to restore the database, and then click **OK** when the restore operation has completed successfully.

### ► Task 3: Access the copied data

1. Open a new query window that is connected to the MIAMI\SQLInstance2 server:
  - In Object Explorer, right-click **MIAMI\SQLInstance2**, and then click **New Query**.
2. In the Query Editor, write and execute a query that uses the AdventureWorksNonEncrypt database to display all fields in the HumanResources.Employee table to confirm that the table is accessible:
  - a. In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE AdventureWorksNonEncrypt
GO
SELECT * FROM HumanResources.Employee
```

- b. Verify that the query returns the confidential data.

**Key point:** Stolen backup media is a security risk because data can be restored and accessed on a remote system.



► **Task 4: Configure transparent data encryption**

1. Open a new query window that is connected to the MIAMI server:
  - In Object Explorer, right-click **MIAMI**, and then click **New Query**.

**Tip:** To see the completed code for the following steps, open the E:\Labfiles\Solution\Encrypt.sql file.

2. In the Query Editor, write and execute a query to create a master key that has the password **Pa\$\$w0rd**:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
-- Create Master Key

USE master
GO
CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'Pa$$w0rd'
GO
```

3. In the Query Editor, write and execute a query to create a certificate named **ServerCertificate** with the subject **Server level certificate**:
  - In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
-- Create Server Certificate

CREATE CERTIFICATE ServerCertificate WITH SUBJECT = 'Server
level certificate'
GO
```

4. In the Query Editor, write and execute a query to create a database encryption key that uses the **AES\_128** algorithm and is encrypted by using the server certificate **ServerCertificate**:
  - In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
-- Create Database Encryption Key

USE AdventureWorks2008
GO
CREATE DATABASE ENCRYPTION KEY
WITH ALGORITHM = AES_128
ENCRYPTION BY SERVER CERTIFICATE ServerCertificate
GO
```

**Note:** You will receive a warning message that you should back up the certificate and the private key that is associated with the certificate. If the certificate ever becomes unavailable, or if you must restore or attach the database on another server, you must have backups of both the certificate and the private key or you will not be able to open the database.

5. In the Query Editor, write and execute a query to alter the AdventureWorks2008 database and set the **Encryption** option to **On**:
  - In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
-- Encrypt Database  
  
ALTER DATABASE AdventureWorks2008  
SET ENCRYPTION ON  
GO
```

► **Task 5: Access the encrypted database from a client application**

1. Start the Microsoft Office Excel® 2007 spreadsheet software:
  - Click **Start**, click **All Programs**, click **Microsoft Office**, and then click **Microsoft Office Excel 2007**.
2. Import the Employee table from the AdventureWorks2008 database on the MIAMI instance of SQL Server:
  - a. On the **Data** tab of the ribbon, in the **Get External Data** section, click **From Other Sources**, and then click **From SQL Server**.
  - b. On the **Connect to Database Server** page, in the **Server name** box, type **MIAMI**, and then click **Next**.
  - c. On the **Select Database and Table** page, in the **Select the database that contains the data you want** drop-down box, click **AdventureWorks2008**.
  - d. In the **Connect to a specific table** list, click **Employee**, and then click **Next**.
  - e. Click **Finish**, and then click **OK** to import the data.

Notice that SQL Server has transparently decrypted the data without any changes to the client application.

3. Close Office Excel without saving changes:
  - a. Click the **Office** button, and then click **Exit Excel**.
  - b. In the **Microsoft Office Excel** dialog box, click **No**.

**Key point:** Transparent data encryption (TDE) requires no configuration or programming on the client application.

► **Task 6: Back up the encrypted AdventureWorks2008 database**

- Back up the encrypted AdventureWorks2008 database on the MIAMI instance to the E:\Labfiles\Starter\AdventureWorksEncrypt.bak folder and remove any existing backup locations that are listed:
  - a. In SQL Server Management Studio, in Object Explorer, in the connection to the MIAMI database engine instance, expand **Databases**, right-click **AdventureWorks2008**, point to **Tasks**, and then click **Back Up**.
  - b. In the **Destination** section, click **Remove** to remove any existing backup destinations, and then click **Add**.
  - c. In the **Select Backup Destination** dialog box, in the **File name** box, type **E:\Labfiles\Starter\AdventureWorksEncrypt.bak**, and then click **OK**.
  - d. Click **OK** to back up the database, and then click **OK** when the backup operation has completed successfully.

► **Task 7: Attempt to restore the encrypted backup**

1. Restore the encrypted AdventureWorks2008 database from the E:\Labfiles\Starter\AdventureWorksEncrypt.bak backup file to a new database named AdventureWorksEncrypt on MIAMI\SQLInstance2:
  - a. In Object Explorer, in the connection to the MIAMI\SQLInstance2 database engine instance, right-click the **Databases** folder, and then click **Restore Database**.
  - b. In the **Restore Database** dialog box, in the **Destination for restore** section, in the **To database** box, type **AdventureWorksEncrypt**.
  - c. In the **Source for restore** section, click **From device**, and then click the ellipsis (...) button.
  - d. In the **Specify Backup** dialog box, click **Add**, navigate to the **E:\Labfiles\Starter** folder, double-click **AdventureWorksEncrypt.bak**, and then click **OK**.
  - e. In the **Select the backup sets to restore** grid, in the **Restore** column, select the **AdventureWorks2008-Full Database Backup** check box.
  - f. Click **OK** to restore the database.

Notice that the restore operation fails because the correct server certificate cannot be found and the encrypted data is inaccessible.
  - g. In the **Microsoft SQL Server Management Studio** dialog box, click **OK**.

Notice that the restore operation fails because the correct server certificate cannot be found and the encrypted data is inaccessible.
2. In the **Restore Database – AdventureWorksEncrypt** dialog box, click **Cancel**.
3. Leave SQL Server Management Studio open; you will use it in the next exercise.

**Key point:** Although the encryption is completely transparent to client applications, the data is encrypted and backups cannot be restored to other servers without the server master key.

## Exercise 2: Implementing Auditing by Using Transact-SQL

► **Task 1: Create the Audit object**

1. Open a new query window that is connected to the MIAMI server:
  - In Object Explorer, right-click **MIAMI**, and then click **New Query**.
2. In the Query Editor, write and execute a query that uses the master database to create an **Audit** object named **UpdatePerson\_Audit**. Set the **Queue Delay** property to **2000** and set the **On Failure** property so that the **Continue** action will be taken if the audit information cannot be written to the destination:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE master
GO
CREATE SERVER AUDIT UpdatePerson_Audit
    TO APPLICATION_LOG
    WITH ( QUEUE_DELAY = 2000, ON_FAILURE = CONTINUE)
GO
```

### ► Task 2: Create the audit specification

- In the Query Editor, write and execute a query that uses the AdventureWorks2008 database to create a database audit specification named **UpdatePerson\_Audit\_Specification** for the **UpdatePerson\_Audit** audit. Specify that any updates that the **dbo** user makes on the Person.Person table should be recorded in the audit and enable the audit specification:
  - In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
USE AdventureWorks2008
GO
CREATE DATABASE AUDIT SPECIFICATION
UpdatePerson_Audit_Specification
FOR SERVER AUDIT UpdatePerson_Audit
ADD (UPDATE
      ON Person.Person
      BY dbo)
WITH (STATE = ON)
GO
```

### ► Task 3: Enable the server audit

- In the Query Editor, write and execute a query that uses the master database to enable the **UpdatePerson\_Audit** audit:
  - In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
USE master
GO

ALTER SERVER AUDIT UpdatePerson_Audit
WITH (STATE = ON)
GO
```

### ► Task 4: Test the SQL Server Audit object

1. In the Query Editor, write and execute a query to update the Person.Person table in the AdventureWorks2008 database so that it sets **LastName** to **Davolio** where **LastName** is **Duffy**:
  - In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
USE AdventureWorks2008
GO

UPDATE Person.Person
SET LastName = 'Davolio'
WHERE LastName = 'Duffy'
GO
```

2. Use the Windows Event Viewer to examine the information about the audited event in the Windows Application log:
  - a. Click **Start**, point to **Administrative Tools**, and then click **Event Viewer**.
  - b. In the **User Account Control** dialog box, click **Continue**.
  - c. In the Event Viewer window, expand **Windows Logs**, and then click **Application**.

- d. In the Windows Application log, examine the information about the audited event in the log.
  - e. Close the Event Viewer window.
3. Close SQL Server Management Studio without saving any changes:
  - a. On the **File** menu, click **Exit**.
  - b. If you are prompted to save changes, click **No**.
4. In the 6158C-MIA-SQL-04 on localhost - Virtual Machine Remote Control window, click **Close**.
5. In the **Close** dialog box, click **Turn off machine and discard changes**, and then click **OK**.

# Module 5

## SQL Server 2008 Database Development Enhancements

### Contents:

Lab Answer Keys

2

# Lab Answer Keys

## Lab: Developing Databases with SQL Server 2008

### Exercise 1: Using Date and Time Data Types

► **Task 1: Create a database with tables for date and time data**

1. In the **6158C** Lab Launcher window, in the **6158C-MIA-SQL-05** virtual machine section, click **Launch**.
2. Log on to the virtual machine as Student, with a password of Pa\$\$w0rd:
  - a. Press ALT GR+DELETE, and then click **Student**.
  - b. In the **Password** box, type **Pa\$\$w0rd** and then press ENTER.
3. Start Microsoft® SQL Server® 2008 Management Studio. When you are prompted, connect to the MIAMI instance of the database engine by using Windows® authentication:
  - a. Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, and then click **SQL Server Management Studio**.
  - b. In the **Connect to Server** dialog box, verify the following settings, and then click **Connect**:
    - **Server type**: Database Engine
    - **Server name**: MIAMI
    - **Authentication**: Windows Authentication
4. Open a new query window that is connected to the MIAMI server:
  - In Object Explorer, right-click **MIAMI**, and then click **New Query**.
5. In the Query Editor, write and execute a query that creates a new database named **AWSales**. Use the AWSales database to create a table named **SalesOrder** that contains the following fields:
  - SalesOrderID int IDENTITY PRIMARY KEY
  - StoreID int
  - SalesPersonID int
  - ProductNumber int
  - CustomerID int
  - OrderDate date
  - OrderTime time
- In the Query Editor, type the code in the following code example, and then click **Execute**:

```
CREATE DATABASE AWSales
GO

USE AWSales
GO

-- Create a table for sales orders
CREATE TABLE SalesOrder
(SalesOrderID int IDENTITY PRIMARY KEY,
StoreID int,
SalesPersonID int,
```

```
ProductNumber int,  
CustomerID int,  
OrderDate date,  
OrderTime time)  
GO
```

### ► Task 2: Insert date and time data into the table

1. In the Query Editor, write and execute a query that inserts the following data into the **SalesOrder** table:
  - StoreID = 4
  - SalesPersonID = 1
  - ProductNumber = 12
  - CustomerID = 23
  - OrderDate = SYSDATETIME()
  - OrderTime = SYSDATETIME()
- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
INSERT INTO SalesOrder  
(StoreID, SalesPersonID, ProductNumber, CustomerID, OrderDate,  
OrderTime)  
VALUES  
(4, 1, 12, 23, SYSDATETIME(), SYSDATETIME())
```

2. Save the query in the E:\Labfiles\Starter\Ex1 folder as OrderData.sql:
  - a. On the **File** menu, click **Save SQLQuery1.sql**.
  - b. Save the file as **OrderData.sql** in the **E:\Labfiles\Starter\Ex1** folder.
3. Select the top one thousand rows from the SalesOrder table and verify that the date component has been added to the OrderDate column and the time component has been added to the OrderTime column:
  - a. In Object Explorer, expand **Databases**, expand **AWSales**, expand **Tables**, right-click **dbo.SalesOrder**, and then click **Select Top 1000 Rows**.
  - b. In the **Results** grid, verify that the INSERT statement has added only the date component to the **OrderDate** column and the time component to the **OrderTime** column.
  - c. On the **File** menu, click **Close**.

### ► Task 3: Insert UTC data into the table

1. In the Query Editor, write and execute a query that inserts the following fields into the **SalesOrder** table:
  - StoreID = 4
  - SalesPersonID = 1
  - ProductNumber = 12
  - CustomerID = 23



- OrderDate = SYSUTCDATETIME ()
- OrderTime = SYSUTCDATETIME ()
- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**.

```
INSERT INTO SalesOrder
(StoreID, SalesPersonID, ProductNumber, CustomerID, OrderDate,
OrderTime)
VALUES
(4, 1, 12, 23, SYSUTCDATETIME(), SYSUTCDATETIME())
```

2. Save the **OrderData.sql** query script file, and then close the OrderData.sql query window.
3. Select the top one thousand rows of the **SalesOrder** table and verify that the INSERT statement has entered the Coordinated Universal Time (UTC) time, which is ahead of the system time:
  - a. In Object Explorer, right-click **dbo.SalesOrder**, and then click **Select Top 1000 Rows**.
  - b. Verify that the INSERT statement has entered the UTC time, which is ahead of the system time.
4. Close the SalesOrder table window.

## Exercise 2: Using Spatial Data

### ► Task 1: Create tables for geographic data

1. Open a new query window that is connected to the MIAMI server:
  - In Object Explorer, right-click **MIAMI**, and then click **New Query**.
2. In the Query Editor, write and execute a query that uses the AWSales database to create a table named **Store** that contains the following fields and properties of the fields:
  - StoreID int IDENTITY PRIMARY KEY
  - StoreName nvarchar(200)
  - StoreAddress nvarchar(200)
  - StorePhone nvarchar(25)
  - StoreRegion varchar(2)
  - StoreLocation geography
- In the Query Editor, type the code in the following code example, and then click **Execute**:

**Note:** This code is also available in the E:\Labfiles\Starter\Ex2\CreateTables.sql file.

```
USE AWSales
GO

-- Create a table for stores
CREATE TABLE Store
(StoreID int IDENTITY PRIMARY KEY,
StoreName nvarchar(200),
StoreAddress nvarchar(200),
StorePhone nvarchar(25),
StoreRegion varchar(2),
```

```
StoreLocation geography)
GO
```

3. In the Query Editor, write and execute a query that creates a table named **SalesPerson** that contains the following fields and properties of the fields:
  - SalesPersonID int IDENTITY PRIMARY KEY
  - FirstName nvarchar(20)
  - LastName nvarchar(20)
  - Email nvarchar(35)
  - SalesRegionName nvarchar(25)
  - SalesRegion geography
- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
-- Create a table for salespeople
CREATE TABLE SalesPerson
(SalesPersonID int IDENTITY PRIMARY KEY,
 FirstName nvarchar(20),
 LastName nvarchar(20),
 Email nvarchar(35),
 SalesRegionName nvarchar(25),
 SalesRegion geography)
GO
```

## ► Task 2: Insert records that include point values

1. In the Query Editor, write and execute a query that inserts the following values into the Store table by using the **Parse** method of the **geography** data type:
  - StoreName = 'Parts Supply'
  - StoreAddress = '2532 Fairgrounds Road, West Kingston, RI 02892'
  - StorePhone = '623-555-0153'
  - StoreLocation = geography::Parse('POINT(-71.548531 41.497478)')
- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
INSERT INTO Store
(StoreName, StoreAddress, StorePhone, StoreLocation)
VALUES
('Parts Supply',
 '2532 Fairgrounds Road, West Kingston, RI 02892',
 '623-555-0153',
 geography::Parse('POINT(-71.548531 41.497478)'))
```

2. In the Query Editor, write and execute a query that inserts the following values into the Store table by using the **Point** method of the **geography** data type:
  - StoreName = 'Two Bike Shops'
  - StoreAddress = '4567 9th Street SW, Puyallup, WA 98371'

- StorePhone = '724-555-0161'
- StoreLocation = geography::Point('47.156622', '-122.30383', 4326)
- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
INSERT INTO Store
  (StoreName, StoreAddress, StorePhone, StoreLocation)
VALUES
  ('Two Bike Shops',
   '4567 9th Street SW, Puyallup, WA 98371',
   '724-555-0161',
   geography::Point('47.156622', '-122.30383', 4326))
```

**Note:** The code for steps 3, 4, and 5 is available in the E:\Labfiles\Starter\Ex2\Inserts.sql file.

3. In the Query Editor, write and execute a query that inserts the following values into the Store table by using the **STGeomFromText** method of the **geography** data type:

- StoreName = 'Eastside Department Store'
- StoreAddress = '9992 Whipple Rd, Union City, CA 94587'
- StorePhone = '926-555-0164'
- StoreLocation = geography::STGeomFromText('POINT(-122.084552 37.603936)', 4326)
- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
INSERT INTO Store
  (StoreName, StoreAddress, StorePhone, StoreLocation)
VALUES
  ('Eastside Department Store',
   '9992 Whipple Rd, Union City, CA 94587',
   '926-555-0164',
   geography::STGeomFromText('POINT(-122.084552 37.603936)',
   4326))
```

4. In the Query Editor, write and execute a query that inserts the following values into the Store table by using the **STGeomFromWKB** method of the **geography** data type:

- StoreName = 'Fitness Hotel'
- StoreAddress = '7901 France Ave S, Edina, MN 55410'
- StorePhone = '377-555-0132'
- StoreLocation = geography::STGeomFromWKB(0x0101000000DC4603780B5557C0EF535568206E4640, 4326)
- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
INSERT INTO Store
  (StoreName, StoreAddress, StorePhone, StoreLocation)
VALUES
  ('Fitness Hotel',
   '7901 France Ave S, Edina, MN 55410',
   '377-555-0132',
   geography::STGeomFromWKB(0x0101000000DC4603780B5557C0EF535568206
```

```
E4640, 4326))
```

5. In the Query Editor, write and execute a query that inserts the following values into the Store table by using the **GeomFromGml** method of the **geography** data type:

- StoreName = 'Aerobic Exercise Company'
- StoreAddress = '39933 Mission Oaks Blvd, Camarillo, CA 93010'
- StorePhone = '244-555-0112'
- StoreLocation = geography::GeomFromGml('<Point xmlns="http://www.opengis.net/gml"><pos>34.235504 - 118.988734</pos> </Point>', 4326)
- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
INSERT INTO Store
(StoreName, StoreAddress, StorePhone, StoreLocation)
VALUES
('Aerobic Exercise Company',
'39933 Mission Oaks Blvd, Camarillo, CA 93010',
'244-555-0112',
geography::GeomFromGml('<Point
xmlns="http://www.opengis.net/gml">
<pos>34.235504 -118.988734</pos>
</Point>', 4326))
```

6. Use any of the techniques that you have seen so far in this procedure to create and execute appropriate INSERT statements for the following stores:

Store data
<b>Store name:</b> Exotic Bikes <b>Store address:</b> 6900 William Richardson Ct, South Bend, IN 46628 <b>Store telephone number:</b> 415-555-0147 <b>Latitude and Longitude location:</b> -86.298173 41.733388
<b>Store name:</b> Healthy Activity Store <b>Store address:</b> 4460 Newport Center Drive, Newport Beach, CA 92625 <b>Store telephone number:</b> 911-555-0165 <b>Latitude and Longitude location:</b> -117.87749 33.617021
<b>Store name:</b> Metropolitan Sports Supply <b>Store address:</b> 482505 Warm Springs Blvd, Fremont, CA 94539 <b>Store telephone number:</b> 828-555-0186 <b>Latitude and Longitude location:</b> -121.914168 37.464038

7. In the Query Editor, write and execute a query that selects the **StoreName** and **StoreLocation.ToString()** values from the Store table:

- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
SELECT StoreName, StoreLocation.ToString()
FROM Store
```

► **Task 3: Insert records that include polygon values**

1. In Windows Explorer, use Notepad to view the contents of the WesternUS.txt text file in the E:\Labfiles\Starter\Ex2\ folder. This file contains the wellknown text (WKT) definition for the Adventure Works Western United States sales region:
  - a. Click **Start**, and then click **Computer**.
  - b. In Windows Explorer, navigate to the **E:\Labfiles\Starter\Ex2** folder.
  - c. Right-click **WesternUS.txt**, point to **Open With**, and then click **Notepad**.
2. Copy the contents of the WesternUS.txt file:
  - a. In Notepad, on the **Edit** menu, click **Select All**.
  - b. On the **Edit** menu, click **Copy**.
  - c. Close Notepad.
3. Paste the contents of the WesternUS.txt file into the Query Editor:
  - In SQL Server Management Studio, on the **Edit** menu, click **Paste**.
4. In Windows Explorer, use Notepad to view the content of the EasternUS.txt XML file in the E:\Labfiles\Starter\Ex2 folder. This file contains the Geographic Markup Language (GML) definition for the Adventure Works Eastern United States sales region:
  - a. In Windows Explorer, navigate to the **E:\Labfiles\Starter\Ex2** folder.
  - b. Right-click **EasternUS.txt**, point to **Open With**, and then click **Notepad**.
5. Copy the contents of the EasternUS.txt file:
  - a. In Notepad, on the **Edit** menu, click **Select All**.
  - b. On the **Edit** menu, click **Copy**.
  - c. Close Notepad.
6. Paste the contents of the EasternUS.txt file into the Query Editor:
  - In SQL Server Management Studio, on the **Edit** menu, click **Paste**.
7. Execute the two INSERT statements that you have just created:
  - In the Query Editor, select the two INSERT statements, and then click **Execute**.
8. In the Query Editor, write and execute a query that selects the **SalesRegionName** and **SalesRegion.ToString()** values from the **SalesPerson** table:
  - In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
SELECT SalesRegionName, SalesRegion.ToString()  
FROM SalesPerson
```

The results show the points that define the boundary of the Western and Eastern sales regions.

9. Save the query in the E:\Labfiles\Starter\Ex2 folder as SalesData.sql, and then close the Query Editor:
  - a. On the **File** menu, click **Save SQL Query As**.
  - b. Save the file as **SalesData.sql** in the **E:\Labfiles\Starter\Ex2** folder.

- c. On the **File** menu, click **Close**.

► **Task 4: Find the area of a geography instance**

1. Open a new query window that is connected to the MIAMI server:
  - In Object Explorer, right-click **MIAMI**, and then click **New Query**.
2. In the Query Editor, write and execute a query that uses the AWSales database to calculate the area of the sales regions by using the **STArea** method of the **geography** instance:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE AWSales
GO

-- Find the area of the sales regions
SELECT SalesRegionName, SalesRegion.STArea()/1000000 AreaInSqKm
FROM SalesPerson
```

The results show the area in square kilometers of each sales region.

► **Task 5: Find geography instances that intersect another geography instance**

- In the Query Editor, write and execute a query to calculate the intersection of stores and sales regions by using the **STIntersects** method of the **geography** instance.

Return the following fields from the Store and Salesperson tables:

- Store.StoreName
- Store.StoreAddress
- SalesPerson.SalesRegionName
- SalesPerson.FirstName + ' ' + SalesPerson.LastName
- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
-- Find which stores are in which sales region
SELECT s.StoreName,
       s.StoreAddress,
       sp.SalesRegionName,
       sp.FirstName + ' ' + sp.LastName Salesperson
FROM Store s, SalesPerson sp
WHERE sp.SalesRegion.STIntersects(s.StoreLocation) = 1
```

The results show which stores are within which sales regions, along with the salesperson for the region.

► **Task 6: Find the union of two geography instances**

- In the Query Editor, write and execute a query to create a new **geography** instance that represents the combination of the Eastern and Western U.S. sales regions by using the **STUnion** method of the **geography** instance:
- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
-- Create the union of the sales regions
DECLARE @westUS geography
```

```

DECLARE @eastUS geography
DECLARE @totalUS geography
SELECT @westUS = SalesRegion FROM SalesPerson WHERE
SalesRegionName = 'Western US'
SELECT @eastUS = SalesRegion FROM SalesPerson WHERE
SalesRegionName = 'Eastern US'
SET @totalUS = @westUS.STUnion(@eastUS)
SELECT @totalUS.ToString()

```

The results show a text representation of the union of the two sales regions.

### ► Task 7: Find geography instances near a geographical feature

1. In Windows Explorer, use Notepad to view the contents of the I5.xml XML file in the E:\Labfiles\Starter\Ex2 folder. This file contains a GML definition for the Interstate 5 highway:
  - a. Click **Start**, click **Computer**, and then navigate to the **E:\Labfiles\Starter\Ex2** folder.
  - b. Right-click **I5.xml**, point to **Open With**, and then click **Notepad**.
  - c. Review the contents of the file and then close Notepad.
2. In SQL Server Management Studio, in the Query Editor, write and execute a query to find stores within 100 kilometers of the I5 road by using the **STDistance** method of the **geography** instance:
  - In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

**Note:** This code is also available in the E:\Labfiles\Starter\Ex2\QuerySpatial.sql file.

```

-- Load Interstate 5 GML from XML file
DECLARE @i5Gml xml
SET @i5Gml =(
SELECT * FROM OPENROWSET(
    BULK 'E:\Labfiles\Starter\Ex2\I5.xml',
    SINGLE_BLOB
) AS i5GML)
-- Find stores within 100 km of Interstate 5
DECLARE @i5 geography;
SET @i5 = geography::GeomFromGml(@i5Gml, 4326)
SELECT StoreName,
       StoreAddress,
       StoreLocation.STDistance(@i5)/1000 DistanceFromI5inKM
FROM Store
WHERE StoreLocation.STDistance(@i5) < 100000
ORDER BY DistanceFromI5inKM

```

The results show all stores within 100 kilometers of I5 in ascending order of distance.

3. Close SQL Server Management Studio without saving any changes:
  - a. On the **File** menu, click **Exit**.
  - b. If you are prompted to save changes, click **No**.
4. In the 6158C-MIA-SQL-05 on localhost - Virtual Machine Remote Control window, click **Close**.
5. In the **Close** dialog box, click **Turn off machine and discard changes**, and then click **OK**.

# Module 6

## SQL Server 2008 Availability Enhancements

### Contents:

Lab Answer Keys

**2**



# Lab Answer Keys

## Lab: Maintaining High Availability

### Exercise 1: Configuring Peer-to-Peer Replication

#### ► Task 1: Start SQL Server Agent on all instances

1. In the **6158C** Lab Launcher window, in the **6158C-MIA-SQL-06** virtual machine section, click **Launch**.
2. Log on to the virtual machine as Student, with a password of Pa\$\$w0rd:
  - a. Press ALT GR+DELETE, and then click **Student**.
  - b. In the **Password** box, type **Pa\$\$w0rd**, and then press ENTER.
3. Start SQL Server Management Studio. When you are prompted, connect to the MIAMI instance of the database engine by using Windows® authentication:
  - a. Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, and then click **SQL Server Management Studio**.
  - b. In the **Connect to Server** dialog box, verify the following settings, and then click **Connect**:
    - **Server type**: Database Engine
    - **Server name**: MIAMI
    - **Authentication**: Windows Authentication
4. Use Object Explorer to connect to the MIAMI\SQLINSTANCE2 instance of the database engine by using Windows authentication:
  - a. In Object Explorer, click **Connect**, and then click **Database Engine**.
  - b. In the **Connect to Server** dialog box, configure the following settings, and then click **Connect**:
    - **Server type**: Database Engine
    - **Server name**: MIAMI\SQLINSTANCE2
    - **Authentication**: Windows Authentication
5. Start the SQL Server Agent service on the MIAMI instance:
  - a. In Object Explorer, on the MIAMI instance, right-click **SQL Server Agent**, and then click **Start**.
  - b. In the **User Account Control** dialog box, click **Continue**.
  - c. In the **Microsoft SQL Server Management Studio** dialog box, click **Yes** to confirm that you want to start the SQLSERVERAGENT service.
6. Start the SQL Server Agent service on the MIAMI\SQLINSTANCE2 instance:
  - a. In Object Explorer, on the MIAMI\SQLINSTANCE2 instance, right-click **SQL Server Agent**, and then click **Start**.
  - b. In the **User Account Control** dialog box, click **Continue**.
  - c. In the **Microsoft SQL Server Management Studio** dialog box, click **Yes** to confirm that you want to start the SQLSERVERAGENT\$SQLINSTANCE2 service.

► **Task 2: Back up the AdventureWorks2008 database and restore it to the SQLINSTANCE2 instance**

1. Back up the AdventureWorks2008 database on the MIAMI server. Remove any existing backup destinations and add E:\Labfiles\Starter\AdventureWorks.bak as the new backup destination:
  - a. On the MIAMI instance, expand **Databases**, right-click **AdventureWorks2008**, point to **Tasks**, and then click **Back Up**.
  - b. In the **Destination** section, click **Remove**, and then click **Add**.
  - c. In the **Select Backup Destination** dialog box, in the **File name** box, type **E:\Labfiles\Starter\Adventure Works.bak**, and then click **OK**.
  - d. In the **Back Up Database** dialog box, click **OK** to start the backup.
  - e. In the **Microsoft SQL Server Management Studio** dialog box, click **OK**.
2. On the MIAMI\SQLINSTANCE2 server, restore the Adventure Works.bak database from the E:\Labfiles\Starter folder to a new database named AdventureWorks2008. Restore the AdventureWorks2008\_Data file to C:\Program Files\Microsoft SQL Server\MSSQL10.SQLINSTANCE2\MSSQL\DATA\AdventureWorks.mdf and restore the AdventureWorks2008\_Log file to C:\Program Files\Microsoft SQL Server\MSSQL10.SQLINSTANCE2\MSSQL\DATA\AdventureWorks\_1.ldf.
  - a. On the MIAMI\SQLINSTANCE2 instance, right-click **Databases**, and then click **Restore Database**.
  - b. In the **Restore Database** dialog box, in the **To database** drop-down box type **AdventureWorks2008**.
  - c. In the **Source for restore** section, click **From device**, and then click the ellipsis (...).
  - d. In the **Specify Backup** dialog box, click **Add**.
  - e. In the **Locate Backup File** dialog box, in the **Selected path** box, type **E:\Labfiles\Starter**.
  - f. In the **File name** box, type **Adventure Works.bak**, and then click **OK**.
  - g. In the **Specify Backup** dialog box, click **OK**.
  - h. In the **Restore Database** dialog box, in the **Select the backup sets to restore** section, select the **AdventureWorks2008-Full Database Backup** check box.
  - i. In the **Select a page** pane, click **Options**.
  - j. In the **Restore the database files as** section, in the **AdventureWorks2008\_Data** row, click the ellipsis (...).
  - k. In the **Locate Database Files** dialog box, navigate to **C:\Program Files\Microsoft SQL Server\MSSQL10.SQLINSTANCE2\MSSQL\DATA**.
  - l. In the **File name** box, type **AdventureWorks.mdf**, and then click **OK**.
  - m. In the **Restore the database files as** section, in the **AdventureWorks2008\_Log** row, click the ellipsis (...).
  - n. In the **Locate Database Files** dialog box, navigate to **C:\Program Files\Microsoft SQL Server\MSSQL10.SQLINSTANCE2\MSSQL\DATA**.
  - o. In the **File name** box, type **AdventureWorks\_1.ldf**, and then click **OK**.
  - p. In the **Restore Database – AdventureWorks2008** dialog box, click **OK**.

- q. When the restore is complete, in the **Microsoft SQL Server Management Studio** dialog box, click **OK**.

► **Task 3: Configure distribution on all instances**

1. In Object Explorer, on the MIAMI instance, right-click the **Replication** folder, and then click **Configure Distribution**.
2. In the Configure Distribution Wizard, configure the MIAMI server so that it will act as its own distributor. Configure the SQL Server Agent service to automatically start and set the snapshot folder to be the E:\Labfiles\Starter\Snapshot folder. Use the default settings for the distribution database. Add MIAMI\SQLINSTANCE2 as a publisher and use Pa\$\$w0rd for the distributor password:
  - a. In the **Configure Distribution Wizard**, click **Next**.
  - b. On the **Distributor** page, click **Next** to use MIAMI as its own distributor.
  - c. On the **SQL Server Agent Start** page, click **Next** to ensure that SQL Server Agent starts automatically.
  - d. On the **Snapshot Folder** page, in the **Snapshot folder** box, type **E:\Labfiles\Starter\Snapshot**, and then click **Next**.
  - e. On the **Distribution Database** page, click **Next** to confirm the distribution database settings.
  - f. On the **Publishers** page, click **Add**, and then click **Add SQL Server Publisher** to add a publisher.
  - g. In the **Connect to Server** dialog box, in the **Server name** box, click **MIAMI\SQLINSTANCE2**, and then click **Connect**.
  - h. On the **Publishers** page, click **Next**.
  - i. On the **Distributor Password** page, in the **Password** and **Confirm password** boxes, type **Pa\$\$w0rd**, and then click **Next**.
  - j. On the **Wizard Actions** page, click **Next** to configure the distribution.
  - k. On the **Complete the Wizard** page, click **Finish**, and then when the configuration is complete, click **Close**.
3. In Object Explorer, on the MIAMI\SQLINSTANCE2 instance, right-click the **Replication** folder, and then click **Configure Distribution**.
4. In the Configure Distribution Wizard, configure the MIAMI server to be the distributor and set the distributor password to Pa\$\$w0rd:
  - a. In the **Configure Distribution Wizard**, click **Next**.
  - b. On the **Distributor** page, click **Use the following server as the Distributor**, and then click **Add**.
  - c. In the **Connect to Server** dialog box, in the **Server name** box, click **MIAMI**, and then click **Connect**.
  - d. On the **Distributor** page, click **Next**.
  - e. On the **Administrative Password** page, in the **Password** and **Confirm password** boxes, type **Pa\$\$w0rd**, and then click **Next**.
  - f. On the **Wizard Actions** page, click **Next** to configure the distribution.
  - g. On the **Complete the Wizard** page, click **Finish**, and then when the configuration is complete, click **Close**.

#### ► Task 4: Create a publication

1. In Object Explorer, on the MIAMI instance, expand **Replication**, right-click **Local Publications**, and then click **New Publication**.
2. In the New Publication Wizard, create a new transactional publication for the AdventureWorks2008 database. Publish all of the Tables objects and create the snapshot immediately. Use the MIAMI\SQLService account with a password of Pa\$\$w0rd for the snapshot agent and log reader agent. Name the publication AWPeerToPeer:
  - a. In the **New Publication Wizard**, click **Next**.
  - b. On the **Publication Database** page, click **AdventureWorks2008**, and then click **Next**.
  - c. On the **Publication Type** page, click **Transactional publication**, and then click **Next**.
  - d. On the **Articles** page, select the **Tables** check box, and then click **Next**.
  - e. On the **Filter Table Rows** page, click **Next** because you cannot filter peerto- peer transactional replication.
  - f. On the **Snapshot Agent** page, select the **Create a snapshot immediately and keep the snapshot available to initialize subscriptions** check box, and then click **Next**.
  - g. On the **Agent Security** page, click **Security settings**.
  - h. In the **Snapshot Agent Security** dialog box, in the **Process account** box, type **MIAMI\SQLService**.
  - i. In the **Password** and **Confirm Password** boxes, type **Pa\$\$w0rd**, and then click **OK**.
  - j. On the **Agent Security** page, click **Next**.
  - k. On the **Wizard Actions** page, click **Next** to create the publication.
  - l. On the **Complete the Wizard** page, in the **Publication name** box, type **AWPeerToPeer**, and then click **Finish**.
  - m. When the publication has been created, click **Close**.

#### ► Task 5: Enable peer-to-peer replication

- Use the Publication Properties - AWPeerToPeer dialog box to enable the Allow peer-to-peer subscriptions subscription option:
  - a. In Object Explorer, on the MIAMI instance, expand **Replication**, and then expand **Local Publications**.
  - b. Right-click the **[AdventureWorks2008]: AWPeerToPeer** publication, and then click **Properties**.
  - c. In the **Publication Properties - AWPeerToPeer** dialog box, in the **Select a page** pane, click **Subscription Options**.
  - d. Set the **Allow peer-to-peer subscriptions** property to **True**, and then click **OK**.

### Exercise 2: Adding a Replication Node

#### ► Task 1: Add a replication node

1. Refresh the Object Explorer view of the [AdventureWorks2008]: AWPeerToPeer publication:
  - a. On the MIAMI instance, expand **Replication**, and then expand **Local Publications**.

- b. Right-click the **[AdventureWorks2008]: AWPeerToPeer** publication, and then click **Refresh**.
  2. Right-click the **[AdventureWorks2008]: AWPeerToPeer** publication, and then click **Configure Peer-To-Peer Topology**.
  3. In the Configure Peer-To-Peer Topology Wizard, add a new peer node on the MIAMI\SQLINSTANCE2 instance. Using the AdventureWorks2008 database, connect the new peer to all displayed nodes and set the peer originator ID to 2. Configure the log reader agent and the distribution agent on all peers to use the MIAMI\SQLService account with a password of Pa\$\$w0rd:
    - a. In the **Configure Peer-To-Peer Topology Wizard**, click **Next**.
    - b. On the **Publication** page, in the **Databases and publications** list, click **AWPeerToPeer**, and then click **Next**.
    - c. On the **Configure Topology** page, right-click the blank design surface, and then click **Add a New Peer Node**.
    - d. In the **Connect to Server** dialog box, in the **Server name** box, click **MIAMI\SQLInstance2**, and then click **Connect**.
    - e. In the **Add a New Peer Node** dialog box, in the **Select Database** box, click **AdventureWorks2008**.
    - f. Select the **Connect to ALL displayed nodes** check box.
    - g. Change the **Peer Originator ID** to **2**, and then click **OK**.
    - h. On the **Configure Topology** page, click **Next**.
    - i. On the **Log Reader Agent Security** page, click the ellipsis (...).
    - j. In the **Log Reader Agent Security** dialog box, in the **Process account** box, type **MIAMI\SQLService**.
    - k. In the **Password** and **Confirm Password** boxes, type **Pa\$\$w0rd**, and then click **OK**.
    - l. On the **Log Reader Agent Security** page, click **Next**.
    - m. On the **Distribution Agent Security** page, in the **MIAMI** row, click the ellipsis (...).
    - n. In the **Distribution Agent Security** dialog box, in the **Process account** box, type **MIAMI\SQLService**.
    - o. In the **Password** and **Confirm Password** boxes, type **Pa\$\$w0rd**, and then click **OK**.
    - p. On the **Distribution Agent Security** page, select the **Use the first peer's security settings for all other peers** check box, and then click **Next**.
    - q. On the **New Peer Initialization** page, click **Next** to confirm that the peer database has been restored.
    - r. On the **Complete the Wizard** page, click **Finish**, and then when the configuration is complete, click **Close**.
  4. Verify that a transactional publication of the AdventureWorks2008 database on MIAMI has been created on the MIAMI\SQLINSTANCE2 instance that uses the settings that you specified:
    - a. On MIAMI\SQLINSTANCE2, expand **Replication**, and then expand **Local Publications**.
    - b. Right-click the **[AdventureWorks2008]: AWPeerToPeer** publication, and then click **Properties**.

- 
- c. In the **Publication Properties – AWPeerToPeer** dialog box, review the information in the **Description** box.
    - d. On the **Articles** page, verify that only the tables are being published.
    - e. On the **Snapshot** page, verify that the location of the snapshot files is the **E:\Labfiles\Starter\Snapshot** folder.
    - f. In the **Publication Properties – AWPeerToPeer** dialog box, click **Cancel**.
  5. Close SQL Server Management Studio without saving any changes:
    - a. On the **File** menu, click **Exit**.
    - b. If you are prompted to save changes, click **No**.
  6. In the 6158C-MIA-SQL-06 on localhost - Virtual Machine Remote Control window, click **Close**.
  7. In the **Close** dialog box, click **Turn off machine and discard changes**, and then click **OK**.

# Module 7

## SQL Server 2008 Data Warehousing Enhancements

### Contents:

Lab Answer Keys

2

# Lab Answer Keys

## Lab: Data Warehousing with SQL Server 2008

### Exercise 1: Using Change Data Capture

#### ► Task 1: Enable change data capture

1. In the **6158C** Lab Launcher window, in the **6158C-MIA-SQL-07** virtual machine section, click **Launch**.
2. Log on to the virtual machine as Student, with a password of Pa\$\$w0rd:
  - a. Press ALT GR+DELETE, and then click **Student**.
  - b. In the **Password** box, type **Pa\$\$w0rd**, and then press ENTER.
3. Start Microsoft® SQL Server® 2008 Management Studio. When you are prompted, connect to the MIAMI instance of the database engine by using Windows® authentication:
  - a. Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, and then click **SQL Server Management Studio**.
  - b. In the **Connect to Server** dialog box, verify the following settings, and then click **Connect**:
    - **Server type**: Database Engine
    - **Server name**: MIAMI
    - **Authentication**: Windows Authentication
4. Start the SQL Server Agent service on the MIAMI instance:
  - a. In Object Explorer, on the MIAMI instance, right-click **SQL Server Agent**, and then click **Start**.
  - b. In the **User Account Control** dialog box, click **Continue**.
  - c. In the **Microsoft SQL Server Management Studio** dialog box, click **Yes** to confirm that you want to start the SQLSERVERAGENT service.
5. Open a new query window that is connected to the MIAMI server:
  - In Object Explorer, right-click **MIAMI**, and then click **New Query**.
6. In the Query Editor, write and execute a query to enable change data capture (CDC) for the **AdventureWorksDW2008** database:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE AdventureworksDW2008;  
GO  
sys.sp_cdc_enable_db;
```

7. In the Query Editor, write and execute a query to enable **CDC** for the **FactInternetSales** table:
  - In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
EXEC sp_cdc_enable_table 'dbo', 'FactInternetSales', @role_name  
= NULL, @supports_net_changes = 0;  
GO
```



► **Task 2: View the results of enabling change data capture**

1. In the Query Editor, write and execute a query to view which of the fact tables currently have CDC enabled:

- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
SELECT name, is_tracked_by_cdc FROM sys.tables  
WHERE name LIKE ('fact%');
```

2. In Windows Explorer, use Notepad to view the contents of the Ex1.txt file in the E:\Labfiles\Starter\Ex1 folder:

- a. Click **Start**, and then click **Computer**.
- b. In Windows Explorer, navigate to the **E:\Labfiles\Starter\Ex1** folder.
- c. Right-click **Ex1.txt**, point to **Open With**, and then click **Notepad**.

3. Copy the Transact-SQL statements from the file:

- a. In Notepad, on the **Edit** menu, click **Select All**.
- b. On the **Edit** menu, click **Copy**.
- c. Close Notepad.

4. Paste the contents of the **Ex1.txt** file into the Query Editor:

- In SQL Server Management Studio, on the **Edit** menu, click **Paste**.

5. Execute the two INSERT statements that you have pasted into the Query Editor:

- Select the two INSERT statements, and then click **Execute**.

6. In the Query Editor, write and execute a query to update the **FactInternetSales** table, setting the **PromotionKey** to **2** where **SalesOrderNumber** is **SO75124**:

- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
UPDATE FactInternetSales  
SET PromotionKey = 2  
WHERE SalesOrderNumber = 'SO75124';
```

7. In the Query Editor, write and execute a query to view the captured changes from the **cdc.dbo\_FactInternetSales\_CT** table:

- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
SELECT * FROM cdc.dbo_FactInternetSales_CT;
```

8. Review the results. The third and fourth rows of the result set have the same log sequence number (LSN) information. The update operation generates two rows in the result set. The first row, which has an **\_\$operation** value of **3**, is the before image (image of the data before the update), and the **\_\$operation** value of **4** represents the after image (image of the data after the update).

9. Review the properties of the two SQL Server Agent jobs that were created to support CDC. Notice that the cleanup job is scheduled to run daily at 2 A.M. You can change this schedule to best fit your data warehousing needs:
  - a. In Object Explorer, expand **SQL Server Agent**, and then expand **Jobs**.
  - b. Right-click **cdc.AdventureWorksDW2008\_capture**, and then click **Properties**.
  - c. Review the properties of the capture job.
  - d. In the **Job Properties – cdc.AdventureWorksDW2008\_capture** dialog box, click **Cancel**.
  - e. Right-click **cdc.AdventureWorksDW2008\_cleanup**, and then click **Properties**.
  - f. Review the properties of the cleanup job. Notice that it is scheduled to run daily at 2 A.M. You can change this schedule to best fit your data warehousing needs.
  - g. In the **Job Properties – cdc.AdventureWorksDW2008\_cleanup** dialog box, click **Cancel**.
10. Save the query in the E:\Labfiles\Starter\Ex1 folder as Exercise1.sql, and then close the Query Editor:
  - a. On the **File** menu, click **Save SQLQuery1.sql**.
  - b. Save the file as **Exercise1.sql** in the **E:\Labfiles\Starter\Ex1** folder.
  - c. On the **File** menu, click **Close**.

## Exercise 2: Using the MERGE Statement

### ► Task 1: Create and populate the NewInternetSales table

1. Open the Ex2.ssmssln project that is located in the E:\Labfiles\Starter\Ex2 folder:
  - a. On the **File** menu, point to **Open**, and then click **Project/Solution**.
  - b. In the **Open Project** dialog box, navigate to the **E:\Labfiles\Starter\Ex2** folder, and then double-click **Ex2.ssmssln**.
2. Review and execute the CreateNewInternetSales.sql query to create the **NewInternetSales** table:
  - In Solution Explorer, double-click the **CreateNewInternetSales.sql** query, review the query, and then click **Execute**.

**Tip:** If Solution Explorer is not visible, on the **View** menu, click **Solution Explorer**.

3. Review and execute the SimulateDay1Inserts.SQL query to add two rows to the **NewInternetSales** table:
  - In Solution Explorer, double-click the **SimulateDay1Inserts.SQL** query, review the query, and then click **Execute**.

### ► Task 2: Merge rows from the NewInternetSales table into the FactInternetSales table

1. Create a new query in the project that uses the (local):MIAMI\Student connection:
  - In Solution Explorer, right-click **(local):MIAMI\Student**, and then click **New Query**.
2. Rename the SQLQuery1.sql file to Merge.sql:
  - a. In Solution Explorer, right-click **SQLQuery1.sql**, and then click **Rename**.
  - b. Type **Merge.sql** and then press ENTER.

3. In the Query Editor, write and execute a query that uses the MERGE statement to insert new rows from the **NewInternetSales** table into the **FactInternetSales** table:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE AdventureworksDW2008

MERGE FactInternetSales FIS -- target table
      USING NewInternetSales NIS -- source table
      ON FIS.SalesOrderNumber = NIS.SalesOrderNumber
      AND FIS.SalesOrderLineNumber = NIS.SalesOrderLineNumber
      WHEN NOT MATCHED THEN
        -- insert a row if it does not exist in the destination
        table
          INSERT ([ProductKey],[OrderDateKey],[DueDateKey]
                , [ShipDateKey],[CustomerKey],[PromotionKey],[CurrencyKey]
                , [SalesTerritoryKey],[SalesOrderNumber],
                [SalesOrderLineNumber]
                , [RevisionNumber],[OrderQuantity],[UnitPrice]
                , [ExtendedAmount],[UnitPriceDiscountPct],[DiscountAmount]
                , [ProductStandardCost],[TotalProductCost],[SalesAmount]
                , [TaxAmt],[Freight],[CarrierTrackingNumber]
                , [CustomerPONumber])
          VALUES ([ProductKey],[OrderDateKey],[DueDateKey]
                , [ShipDateKey],[CustomerKey],[PromotionKey],[CurrencyKey]
                , [SalesTerritoryKey],[SalesOrderNumber],
                [SalesOrderLineNumber]
                , [RevisionNumber],[OrderQuantity],[UnitPrice]
                , [ExtendedAmount],[UnitPriceDiscountPct],[DiscountAmount]
                , [ProductStandardCost],[TotalProductCost],[SalesAmount]
                , [TaxAmt],[Freight],[CarrierTrackingNumber]
                , [CustomerPONumber]);
```

**Tip:** You can copy the code for this query from E:\Labfiles\Starter\Ex2\Ex2.txt and paste it into the Query Editor.

Notice that two new rows are inserted into the FactInternetSales table.

4. In the Query Editor, write and execute a query to select rows from the **FactInternetSales** table that have a **SalesOrderNumber** greater or equal to **SO75130**. This will return the new rows that you added to the **NewInternetSales** table:
  - In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
SELECT * FROM FactInternetSales
WHERE SalesOrderNumber >= 'SO75130';
GO
```

Notice that the two new rows are returned.

### ► Task 3: Simulate additional sales

- Review and execute the SimulateDay2Inserts.sql query to add three rows to the **NewInternetSales** table:
- In Solution Explorer, double-click the **SimulateDay2Inserts.sql** query, review the query, and then click **Execute**.

► **Task 4: Merge additional sales**

1. Return to the Merge.sql query window, and then click **Execute**.
2. When the query has completed, view the results.

Notice that there are now five rows returned. If you had attempted to insert all of the rows from the NewInternetSales table, the UNIQUE constraint on the **SalesOrderNumber** and **SalesOrderLineNumber** columns would have been violated.

3. Close the Ex2 solution without saving changes:
  - a. On the **File** menu, click **Close Solution**.
  - b. In the **Microsoft SQL Server Management Studio** dialog box, click **No**.

### Exercise 3: Working with Partitioned Data

► **Task 1: Create and populate the partitioned table**

1. In SQL Server Management Studio, open the PartitionProcessing.ssmssln project that is located in the E:\Labfiles\Starter\Ex3 folder:
  - On the **File** menu, point to **Open**, and then click **Project/Solution**. Navigate to the **E:\Labfiles\Starter\Ex3** folder, click **PartitionProcessing.ssmssln**, and then click **Open**.
2. Review and execute the Files and Filegroups.sql query to create four new filegroups to support the partitioned table:
  - In Solution Explorer, double-click the **Files and Filegroups.sql** query, review the query, and then click **Execute**.

**Tip:** If Solution Explorer is not visible, on the **View** menu, click **Solution Explorer**.

3. To complete the process of creating and populating a partitioned table, review and execute each of the following scripts:
  - Create Partition Function.sql
  - Create Partition Scheme.sql
  - Create Table.sql
  - Load Data.sql
  - a. In Solution Explorer, double-click the **Create Partition Function.sql** query, review the query, and then click **Execute**.
  - b. In Solution Explorer, double-click the **Create Partition Scheme.sql** query, review the query, and then click **Execute**.
  - c. In Solution Explorer, double-click the **Create Table.sql** query, review the query, and then click **Execute**.
  - d. In Solution Explorer, double-click the **Load Data.sql** query, review the query, and then click **Execute**.

**Note:** Partitioning functions provide the boundaries between two filegroups. In this case, the values are the keys for January 1.

### ► Task 2: View the partitioned data

1. Create a new query in the project that uses the (local):MIAMI\Student connection:
  - In Solution Explorer, right-click **(local):MIAMI\Student**, and then click **New Query**.
2. Rename the SQLQuery1.sql file to View Partitioned Data.sql:
  - a. In Solution Explorer, right-click **SQLQuery1.sql**, and then click **Rename**.
  - b. Type **View Partitioned Data.sql**, and then press ENTER.
3. In the Query Editor, write and execute a query to display the **ProductKey**, **OrderDateKey**, and partition number for each row in the **FactInternetSalesPartitioned** table:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE AdventureWorksDW2008;
SELECT ProductKey,
       OrderDateKey,
       $PARTITION.pf_OrderDateKey (OrderDateKey) AS
PartitionNo
FROM FactInternetSalesPartitioned;
GO
```

4. Review the results that show that the rows are distributed across the three partitions.
5. In the Query Editor, write and execute a query to display the number of rows in each partition:
  - In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
SELECT $PARTITION.pf_OrderDateKey (OrderDateKey) AS PartitionNo,
COUNT(*) AS Rows
FROM FactInternetSalesPartitioned
GROUP BY $PARTITION.pf_OrderDateKey (OrderDateKey)
ORDER BY PartitionNo;
GO
```

6. Review the results that show the number of rows on each partition.

### ► Task 3: Create partition-aligned views

1. Review and execute the queries in Create Views.sql that create three views, each returning the data in one of the partitions of the **FactInternetSalesPartitioned** table:
  - In Solution Explorer, double-click **Create Views.sql**, review the queries, and then click **Execute**.
2. Review and execute the queries in Create Indexes.sql that create indexes on the three views created in Step 1:
  - In Solution Explorer, double-click **Create Indexes.sql**, review the queries, and then click **Execute**.

### ► Task 4: View query execution plans

1. Create a new query in the project that uses the (local):MIAMI\Student connection:
  - In Solution Explorer, right-click **(local):MIAMI\Student**, and then click **New Query**.
2. Rename the SQLQuery1.sql file to View Aligned View Execution Plan.sql:
  - a. In Solution Explorer, right-click **SQLQuery1.sql**, and then click **Rename**.

- b. Type **View Aligned View Execution Plan.sql**, and then press ENTER.
3. In the Query Editor, write a query to select the **ProductKey** and **OrderQuantity** fields from the **vw\_InternetSales2003** table where the **OrderDate** is between **01/01/2003** and **06/06/2003**:
  - In the Query Editor, type the code in the following code example:

```
USE AdventureWorksDW2008
GO

SELECT ProductKey, OrderQuantity
FROM vw_InternetSales2003
WHERE OrderDate BETWEEN '01/01/2003' AND '06/06/2003'
GO
```

**Note:** The views created above include a WHERE clause that restricts the view based on the **DueDateKey** column. The underlying table was partitioned on the **OrderDate** column. Because of this, there are rows that have an **OrderDate** that is before '01/01/2003' in the view.

4. Execute the query, including the actual execution plan in the results:
  - On the toolbar, click **Include Actual Execution Plan**, and then click **Execute**.
5. Review the results, and then click the **Execution plan** tab.
6. In the resulting execution plan, verify that the **idx\_CL\_vw\_InternetSales2003** partitioned index is selected by the query optimizer.

**Note:** The optimizer chooses the smaller view index rather than the larger table index. Partition-aligned indexed views can be used to improve performance on systems with partitioned tables.

7. Create a new query in the project that uses the (local):MIAMI\Student connection:
  - In Solution Explorer, right-click **(local):MIAMI\Student**, and then click **New Query**.
8. Rename the SQLQuery1.sql file to View Partitioned Table Execution Plan.sql:
  - a. In Solution Explorer, right-click **SQLQuery1.sql**, and then click **Rename**.
  - b. Type **View Partitioned Table Execution Plan.sql** and then press ENTER.
9. In the Query Editor, write two queries. The first query should select the **ProductKey** and **OrderQuantity** fields from the **FactInternetSalesPartitioned** table where the **ProductKey** is **312**. The second query should select the **ProductKey** and **OrderQuantity** fields from the **FactInternetSalesPartitioned** table where the **OrderDateKey** is between **20011130** and **20020124**:
  - In the Query Editor, type the code in the following code example:

```
USE AdventureWorksDW2008
GO

SELECT ProductKey, OrderQuantity
FROM FactInternetSalesPartitioned
WHERE ProductKey = 312
GO

SELECT ProductKey, OrderQuantity
FROM FactInternetSalesPartitioned
WHERE OrderDateKey BETWEEN 20011130 AND 20020124
GO
```

10. Execute the query, including the actual execution plan in the results:
  - On the toolbar, click **Include Actual Execution Plan**, and then click **Execute**.
11. Review the results, and then click the **Execution plan** tab.
12. In the resulting execution plan, point to the **Clustered Index Scan** image in the first execution plan, and then wait for the ScreenTip to appear. Review the information that is presented.
13. Point to the **Clustered Index Scan** image in the second execution plan and wait for the ScreenTip to appear. Review the information that is presented. Notice the Seek Predicates section that is added with the SeekKeys that include the partition information. Also, notice that only two partitions are accessed to return the results.

**Note:** In the following procedures, you will implement a sliding window scenario by switching partitions into and out of the table.

► **Task 5: Create a partitioned table for archive data**

1. Create a new query in the project that uses the (local):MIAMI\Student connection:
  - In Solution Explorer, right-click **(local):MIAMI\Student**, and then click **New Query**.
2. Rename the SQLQuery1.sql file to Create Archive Partition Function.sql:
  - a. In Solution Explorer, right-click **SQLQuery1.sql**, and then click **Rename**.
  - b. Type **Create Archive Partition Function.sql** and then press ENTER.
3. In the Query Editor, write and execute a query to create a range right partition function named **pf\_OrderDateKeyArchive** for an integer column. Use a boundary value of **20020101**:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE AdventureWorksDW2008;  
  
CREATE PARTITION FUNCTION pf_OrderDateKeyArchive(int)  
AS RANGE RIGHT  
FOR VALUES(20020101);  
GO
```

4. Create a new query in the project that uses the (local):MIAMI\Student connection:
  - In Solution Explorer, right-click **(local):MIAMI\Student**, and then click **New Query**.
5. Rename the SQLQuery1.sql file to Create Archive Partition Scheme.sql:
  - a. In Solution Explorer, right-click **SQLQuery1.sql**, and then click **Rename**.
  - b. Type **Create Archive Partition Scheme.sql**, and then press ENTER.
6. In the Query Editor, write and execute a query to create a partition scheme named **ps\_OrderDateKeyArchive** that uses the **pf\_OrderDateKeyArchive** function to allocate data to the **fg2001**, **fg2002**, and **fg2003** partitions:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE AdventureWorksDW2008;  
  
CREATE PARTITION SCHEME ps_OrderDateKeyArchive  
AS PARTITION pf_OrderDateKeyArchive  
TO (fg2001, fg2002, fg2003);
```

```
GO
```

7. Review and execute the Create Archive Table.sql query that creates a new archive table for the archived Internet sales data:
  - In Solution Explorer, double-click the **Create Archive Table.sql** query, review the query, and then click **Execute**.

► **Task 6: View the partitioned data**

- Review and execute the View Archive Data.sql query that returns data from the **vw\_InternetSales2001** view and the **FactInternetSalesPartitioned** table:
  - In Solution Explorer, double-click the **View Archive Data.sql** query, review the query, and then click **Execute**.

Notice that there is data for the year 2001 in the vw\_InternetSales2001 view based on the FactInternetSalesPartitioned table, but no data in the FactInternetSalesArchive table at this time.

► **Task 7: Move data to the archive partition**

1. Create a new query in the project that uses the (local):MIAMI\Student connection:
  - In Solution Explorer, right-click **(local):MIAMI\Student**, and then click **New Query**.
2. Rename the SQLQuery1.sql file to Alter Archive Partition Function.sql:
  - a. In Solution Explorer, right-click **SQLQuery1.sql**, and then click **Rename**.
  - b. Type **Alter Archive Partition Function.sql**, and then press ENTER.
3. In the Query Editor, write and execute a query to add an extra boundary of **20030101** to the **pf\_OrderDateKeyArchive** partition function:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE AdventureWorksDW2008;  
  
ALTER PARTITION FUNCTION pf_OrderDateKeyArchive()  
SPLIT RANGE(20030101);  
GO
```

4. Create a new query in the project that uses the (local):MIAMI\Student connection:
  - In Solution Explorer, right-click **(local):MIAMI\Student**, and then click **New Query**.
5. Rename the SQLQuery1.sql file to Switch Partition.sql:
  - a. In Solution Explorer, right-click **SQLQuery1.sql**, and then click **Rename**.
  - b. Type **Switch Partition.sql**, and then press ENTER.
6. In the Query Editor, write and execute a query to move data from the first partition in the **FactInternetSalesPartitioned** table to the first partition in the **FactInternetSalesArchive** table:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE AdventureWorksDW2008;  
  
ALTER TABLE FactInternetSalesPartitioned SWITCH PARTITION 1 TO  
FactInternetSalesArchive PARTITION 1;
```



```
GO
```

**Note:** By using SWITCH, you can move data from a partition in one table to a partition in another. This can occur very quickly because there may be no actual data movement.

7. Create a new query in the project that uses the (local):MIAMI\Student connection:
  - In Solution Explorer, right-click **(local):MIAMI\Student**, and then click **New Query**.
8. Rename the SQLQuery1.sql file to Merge Partition.sql:
  - a. In Solution Explorer, right-click **SQLQuery1.sql**, and then click **Rename**.
  - b. Type **Merge Partition.sql**, and then press ENTER.
9. In the Query Editor, write and execute a query to alter the **pf\_OrderDateKey** partition function to merge the data on the partitions with a boundary value of **20020101**:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE AdventureWorksDW2008;  
  
ALTER PARTITION FUNCTION pf_OrderDateKey() MERGE  
RANGE(20020101);  
GO
```

10. In the Query Editor, write and execute a query to alter the **pf\_OrderDateKeyArchive** partition function to merge the data on the partitions with a boundary value of **20020101**:
  - In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
ALTER PARTITION FUNCTION pf_OrderDateKeyArchive () MERGE  
RANGE(20020101);  
GO
```

**Note:** By using **MERGE**, you can remove boundaries from partition functions.

## ► Task 8: Create a partition for new data

1. Create a new query in the project that uses the (local):MIAMI\Student connection:
  - In Solution Explorer, right-click **(local):MIAMI\Student**, and then click **New Query**.
2. Rename the SQLQuery1.sql file to Split Partition.sql:
  - a. In Solution Explorer, right-click **SQLQuery1.sql**, and then click **Rename**.
  - b. Type **Split Partition.sql**, and then press ENTER.
3. In the Query Editor, write and execute a query to add a partition to the **pf\_OrderDateKey** partition function that uses a boundary value of **915**:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE AdventureWorksDW2008;
```

```
ALTER PARTITION FUNCTION pf_OrderDateKey() SPLIT RANGE(915);
GO
```

4. Return to the View Archive Data.sql query window, and then click **Execute**.

Notice that there is no data for the year 2001 in the vw\_InternetSales2001 view based on the FactInternetSalesPartitioned table, but there is data in the FactInternetSalesArchive table.

**Note:** By using the sliding windows method, you have moved data to an archive table and provided a new partition function boundary for the next year's data.

5. Close the Ex3 solution without saving changes:
  - a. On the **File** menu, click **Close Solution**.
  - b. In the **Microsoft SQL Server Management Studio** dialog box, click **No**.

## Exercise 4: Compressing Data

### ► Task 1: Review the current compression type and setting of the FactFinance table

- Use the Table Properties – FactFinance dialog box for the FactFinance table in the AdventureWorksDW2008 database to view the storage properties of the table. Review the compression and space usage values, and then complete the following table:
  - a. In Object Explorer, expand **Databases**, expand **AdventureWorksDW2008**, and then expand **Tables**.
  - b. Right-click **dbo.FactFinance**, and then click **Properties**.
  - c. In the **Table Properties - FactFinance** dialog box, in the left pane, click **Storage**.
  - d. Review the compression and space usage values, and then complete the following table:

Property	Value
Compression type	
Index space	
Data space	

- e. In the **Table Properties - FactFinance** dialog box, click **Cancel**.

### ► Task 2: Review the estimated data compression savings

1. Open a new query window that is connected to the MIAMI server:
  - In Object Explorer, right-click **MIAMI**, and then click **New Query**.
2. In the Query Editor, write and execute a query to review the estimated data compression savings of implementing row compression on the **FactFinance** table:
  - In the Query Editor, type the code in the following code example, and then click **Execute**:

```
USE AdventureworksDW2008;
GO

EXEC sp_estimate_data_compression_savings 'dbo', 'FactFinance',
NULL, NULL, 'ROW';
```

```
GO
```

3. Review the compression and space usage values, and then complete the following table:

Property	Value
size_with_current_compression_setting	
size_with_requested_compression_setting	

4. In the Query Editor, write and execute a query to review the estimated data compression savings of implementing page compression on the **FactFinance** table:

- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
EXEC sp_estimate_data_compression_savings 'dbo', 'FactFinance',
NULL, NULL, 'PAGE';
GO
```

5. Review the compression and space usage values, and then complete the following table:

Property	Value
size_with_current_compression_setting	
size_with_requested_compression_setting	

### ► Task 3: Implement data compression on the FactFinance table by using Transact-SQL

1. In the Query Editor, write and execute a query to implement page compression on the **FactFinance** table:

- In the Query Editor, type the code in the following code example, select the code, and then click **Execute**:

```
ALTER TABLE FactFinance REBUILD PARTITION = ALL
WITH (DATA_COMPRESSION = PAGE);
GO
```

2. Use the Table Properties - FactFinance dialog box for the FactFinance table to view the storage properties of the table. Review the compression and space usage values, and then complete the following table:

- In Object Explorer, right-click **dbo.FactFinance**, and then click **Properties**.
- In the **Table Properties - FactFinance** dialog box, in the left pane, click **Storage**.
- Review the compression and space usage values, and then complete the following table:

Property	Value
Compression type	
Index space	
Data space	

3. Compare these values with those that were obtained before implementing the page compression and verify that the data space property value is reduced.
4. Close the **Table Properties – FactFinance** dialog box:
  - In the **Table Properties - FactFinance** dialog box, click **Cancel**.

► **Task 4: Implement data compression on the FactInternetSales table by using the Data Compression Wizard**

1. Use the Table Properties-FactInternetSales dialog box for the FactInternetSales table in the AdventureWorksDW2008 database to view the storage properties of the table. Review the compression and space usage values, and then complete the following table:
  - a. In Object Explorer, right-click **dbo.FactInternetSales**, and then click **Properties**.
  - b. In the **Table Properties - FactInternetSales** dialog box, in the left pane, click **Storage**.
  - c. Review the compression and space usage values, and then complete the following table:

Property	Value
Compression type	
Index space	
Data space	

- d. In the **Table Properties - FactInternetSales** dialog box, click **Cancel**.
2. In Object Explorer, right-click **dbo.FactInternetSales**, point to **Storage**, and then click **Manage Compression**.
3. Use the Data Compression Wizard - FactInternetSales Wizard to verify the compression that would be gained by implementing Row compression. Run the script immediately, and then when the compression is complete, click Close:
  - a. On the **Welcome to the Data Compression Wizard** page, click **Next**.
  - b. On the **Select Compression Type** page, in the **Compression type** column, click **Row**, and then click **Calculate**.
  - c. Verify that the **Requested compressed space** value is lower than the **Current space** value.
  - d. On the **Select Compression Type** page, click **Next**.
  - e. On the **Select an Output Option** page, click **Run immediately**, and then click Next.
  - f. On the **Data Compression Wizard Summary** page, click **Finish**.
  - g. On the **Compression Wizard Progress** page, click **Close**.
4. Use the Table Properties - FactInternetSales dialog box for the FactInternetSales table to view the storage properties of the table. Review the compression and space usage values, and then complete the following table:
  - a. In Object Explorer, right-click **dbo.FactInternetSales**, and then click **Properties**.
  - b. In the **Table Properties - FactInternetSales** dialog box, in the left pane, click **Storage**.
  - c. Review the compression and space usage values, and then complete the following table.

Property	Value
Compression type	
Index space	
Data space	

5. Compare these values with those that were obtained before implementing the row compression and verify that the data space property value is reduced.
6. Close the Table Properties – FactInternetSales dialog box:
  - In the **Table Properties - FactInternetSales** dialog box, click **Cancel**.
7. Close SQL Server Management Studio without saving any changes:
  - a. On the **File** menu, click **Exit**.
  - b. If you are prompted to save changes, click **No**.
8. In the 6158C-MIA-SQL-07 on localhost - Virtual Machine Remote Control window, click **Close**.
9. In the **Close** dialog box, click **Turn off machine and discard changes**, and then click **OK**.

# Module 8

## SQL Server 2008 Reporting Services Enhancements

### Contents:

Lab Answer Keys	2
-----------------	---

# Lab Answer Keys

## Lab: Using SQL Server 2008 Reporting Services

### Exercise 1: Creating a Report

#### ► Task 1: Create a report project

1. In the **6158C** Lab Launcher window, in the **6158C-MIA-SQL-08** virtual machine section, click **Launch**.
2. Log on to the virtual machine as Student, with a password of Pa\$\$w0rd:
  - a. Press ALT GR+DELETE, and then click **Student**.
  - b. In the **Password** box, type **Pa\$\$w0rd**, and then press **ENTER**.
3. Open Microsoft® SQL Server® Business Intelligence Development Studio:
  - Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, and then click **SQL Server Business Intelligence Development Studio**.
4. Create a new business intelligence project that uses the Report Server Project template named SalesReport. Save the project in the E:\Labfiles\Starter\Ex1 folder:
  - a. On the **File** menu, point to **New**, and then click **Project**.
  - b. In the **New Project** dialog box, in the **Project Types** list, click **Business Intelligence Projects**, and then in the **Templates** list, click **Report Server Project**.
  - c. In the **Name** text box, type **SalesReport**.
  - d. In the **Location** text box, type **E:\Labfiles\Starter\Ex1**, and then click **OK**.
5. Add a new report to the project named **SalesReport**:
  - a. In Solution Explorer, right-click **Reports**, point to **Add**, and then click **New Item**.
  - b. In the **Templates** list, click **Report**. In the **Name** text box, type **SalesReport**, and then click **Add**.

#### ► Task 2: Create a data source and a data set

1. Add a new data source to the report named AdventureWorksDW2008Source by using an embedded connection to SQL Server. Connect to the MIAMI server by using Windows® authentication, and then connect to the AdventureWorksDW2008 database:
  - a. In the **Report Data** pane, click **New**, and then click **Data Source**.
  - b. In the **Data Source Properties** dialog box, on the **General** page, in the **Name** box, type **AdventureWorksDW2008Source**, and then click **Embedded connection**.
  - c. In the **Type** box, click **Microsoft SQL Server**, and then click **Edit**.
  - d. In the **Connection Properties** dialog box, in the **Server name** box, type **MIAMI**.
  - e. In the **Log on to the server** section, click **Use Windows Authentication**, in the **Select or enter a database name** box, click **AdventureWorksDW2008**, and then click **OK**.
  - f. On the **General** page, click **OK**.
2. Add a new data set to the report named SalesReportDataSet by using the AdventureWorksDW2008Source data source. Import the query text from the Ex1.sql file in the E:\Labfiles\Starter\Ex1 folder:

- a. In the **Report Data** pane, click **New**, and then click **Dataset**.
- b. In the **Dataset Properties** dialog box, in the **Name** text box, type **SalesReportDataSet**, and then click **Import**.
- c. In the **Import Query** dialog box, browse to the **E:\Labfiles\Starter\Ex1** folder, click **Ex1.sql**, and then click **Open**.
- d. In the **Dataset Properties** dialog box, click **OK**.

► **Task 3: Create a matrix report**

1. In the **Design** pane, right-click in the upper left of the report section, point to **Insert**, and then click **Matrix**.
2. Add the SalesAmount field to the Data label:
  - a. In the **Design** tab, point to the **Data** label, and then rest the pointer there until the **Field List** icon appears.
  - b. Click the **Field List** icon, and then click the **SalesAmount** field.

**Tip:** You can also drag the **SalesAmount** field from the **Report Data** pane to the **Data** label in the matrix.

3. In the **Report Data** pane, drag the **ProductCategory** field to the **Row Groups** area at the bottom of the screen.

**Tip:** If the **Row Groups** area is not active, click in the matrix to activate the area.

4. In the **Report Data** pane, drag the **SubCategory** field to just below the **ProductCategory** field in the **Row Groups** area.
5. Add the Promotion field to the Column Groups area:
  - In the **Report Data** pane, drag the **Promotion** field to the **Column Groups** area.
6. Add a total row to the [Sum(SalesAmount)] field:
  - In the **Design** pane, right-click **[Sum(SalesAmount)]**, point to **Add Total**, and then click **Row**.
7. Add a total row after the [ProductCategory] field:
  - In the **Design** pane, right-click **[ProductCategory]**, point to **Add Total**, and then click **After**.
8. Add a total column after the [Promotion] field:
  - In the **Design** pane, right-click **[Promotion]**, point to **Add Total**, and then click **After**.
9. Add a group to the [Total] column that is on the right of the [Promotion] column. Configure it to be on the right of the column and group the results by [OrderYear]:
  - a. Right-click the **[Total]** column to the right of the **[Promotion]** column, point to **Add Group**, and then click **Adjacent Right**.
  - b. In the **Tablix Group** dialog box, in the **Group by** box, click **[OrderYear]**, and then click **OK**.
10. Use the Field List icon in the first cell below the [OrderYear] heading to add the SalesAmount field to each cell in the column:
  - a. In the **Design** tab, point to the first cell below the **[OrderYear]** heading, and then rest the pointer there until the **Field List** icon appears.



- b. Click the **Field List** icon, and then click the **SalesAmount** field.
  - c. Repeat steps a. and b. for all of the cells in the **[OrderYear]** column.
11. Add a total row after the [OrderYear] field:
  - In the **Design** pane, right-click **[OrderYear]**, point to **Add Total**, and then click **After**.
12. Preview the report:
  - Click the **Preview** tab.

► **Task 4: Format the matrix report**

1. In the Design pane, change the following cells so that they are bold and have a light-gray background color:
  - **[ProductCategory]**
  - **[SubCategory]**
  - **[Promotion]**
  - **[OrderYear]**
  - Each cell labeled **Total**
  - a. Click the **Design** tab.
  - b. In the **Design** pane, press and hold down the CTRL key while you select the following cells:
    - **[ProductCategory]**
    - **[SubCategory]**
    - **[Promotion]**
    - **[OrderYear]**
    - Each cell labeled **Total**
  - c. On the **Format** menu, point to **Font**, and then click **Bold**.
  - d. While the cells are still selected, on the **Format** menu, click **Background Color**.
  - e. In the **Choose Color** dialog box, click **Light Gray**, and then click **OK**.
2. Change the column and row headers of the four Total columns to bold:
  - a. In the **Design** pane, select the column headers of the two **Total** columns. While the columns are selected, on the **Format** menu, point to **Font**, and then click **Bold**.
  - b. In the **Design** pane, select the row headers of the two **Total** rows. While the rows are selected, on the **Format** menu, point to **Font**, and then click **Bold**.
3. Use the Text Box Properties dialog box for one of the [Sum(SalesAmount)] fields to change the format of the field so that it uses the Currency format with 0 decimal places and the 1000 separator (,):
  - a. Select one of the **[Sum(SalesAmount)]** fields, right-click the selected field, and then click **Text Box Properties**.
  - b. In the **Text Box Properties** dialog box, in the left pane, click **Number**. In the right pane, in the **Category** list, click **Currency**, and then in the **Decimal places** box, type **0**.
  - c. Select the **Use 1000 separator (,)** check box, and then click **OK**.

4. Repeat step 3 for each of the **[Sum(SalesAmount)]** fields in the **Design** pane.
5. Preview the report:
  - Click the **Preview** tab.

► **Task 5: Add tablix to enable drill-down capabilities**

1. In the Design pane, use the Text Box Properties dialog box for the [ProductCategory] field to change the field name to Category:
  - a. Click the **Design** tab.
  - b. Right-click the **[ProductCategory]** cell, and then click **Text Box Properties**.
  - c. In the **Text Box Properties** dialog box, in the **Name** box, type **Category**, and then click **OK**.
2. Use the Text Box Properties dialog box for the [SubCategory] field to change the field name to SubCategory:
  - a. Right-click the **[Subcategory]** cell, and then click **Text Box Properties**.
  - b. In the **Text Box Properties** dialog box, in the **Name** box, type **SubCategory**, and then click **OK**.
3. Use the Group Properties dialog box for the SubCategory group to initially hide the field. Also configure the display to be switched on and off by the Category report item:
  - a. In the **Row Groups** section, right-click the **SubCategory** group, and then click **Group Properties**.
  - b. In the **Group Properties** dialog box, in the left pane, click **Visibility**.
  - c. On the **Visibility** page, click **Hide**, select the **Display can be toggled by this report item** check box, in the list, click **Category**, and then click **OK**.
4. Preview the report and verify that you can show and hide the subcategories:
  - a. Click the **Preview** tab.
  - b. Click the plus sign next to **Bikes** to expand the subcategories for the **Bikes** category.

► **Task 6: Create a chart**

1. In the Design pane, use the Tablix Properties dialog box of a row header to add a page break before the data region:
  - a. Click the **Design** tab.
  - b. Right-click a row header of the matrix, and then click **Tablix Properties**.
  - c. On the **General** page, select the **Add a page break before** check box, and then click **OK**.
2. Move the matrix so that it is located at the lower-right corner of the Design pane:
  - Click the upper-left corner of the matrix to activate the **Move** icon. Drag the matrix so that it is in the lower-right corner of the **Design** pane.
3. Insert a 3-D Exploded Pie chart and resize it to fill the width of the screen. The chart may overlap the matrix. If necessary, resize the Design pane by using the resize button that appears when you hold your mouse over the line between the Design pane and the footer section:
  - a. In the **Design** pane, right-click in the upper left of the report section, point to **Insert**, and then click **Chart**.

- b. In the **Select Chart Type** dialog box, click **3-D Exploded Pie**, and then click **OK**.
  - c. Resize the chart to fill the width of the screen. The chart may overlap the matrix. If necessary, resize the **Design** pane by using the resize button that appears when you hold your mouse over the line between the **Design** pane and the footer section.
4. Double-click the chart to activate the edit mode for the chart.
5. Drag the **OrderYear** field from the **Report Data** pane to the **Drop series fields here** area.
6. Drag the **ProductCategory** field to the **Drop category fields here** area.
7. Drag the **SalesAmount** field to the **Drop data fields here** area.
8. Preview the report. Notice that the matrix is now on page two of the report:
  - a. Click the **Preview** tab.
  - b. On the **Preview pane** toolbar, click **Next Page**.
9. In the Design pane, add a title to the chart named Sales by Year:
  - a. Click the **Design** tab.
  - b. Double-click in the **Chart Title** area, and then modify the title to **Sales by Year**.
10. Right-click an empty area within the chart, and then explore the options that are available.
11. Preview the report:
  - Click the **Preview** tab.
12. Save your report, and then close the Microsoft Visual Studio® development system:
  - a. On the **File** menu, click **Save All**.
  - b. On the **File** menu, click **Exit**.

## Exercise 2: Managing Reporting Services

### ► Task 1: View Reporting Services configuration settings

1. Open Reporting Services Configuration Manager, and then connect to the MSSQLSERVER Report Server instance on the MIAMI server:
  - a. Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, click **Configuration Tools**, and then click **Reporting Services Configuration Manager**.
  - b. In the **User Account Control** dialog box, click **Continue**.
  - c. When you are prompted, connect to the MSSQLSERVER Report Server instance on MIAMI.
2. Review the information in the **Report Server Status** pane. Note that you can use Reporting Services Configuration Manager to view and manage the status of the Report Server service.
3. In the **Connect** pane, click **Service Account** and note that you can use Reporting Services Configuration Manager to view and manage the service account that Reporting Services uses.
4. In the **Connect** pane, click **Web Service URL** and note that you can use Reporting Services Configuration Manager to view and manage the Web service URL that Reporting Services uses. Note that the Web service for this instance is configured to listen on Port 80 and that Secure Sockets Layer (SSL) encryption is not configured.

5. In the **Connect** pane, click **Database** and note that you can use Reporting Services Configuration Manager to view and manage the database that Reporting Services uses and the credentials that are used to connect to it. This is also where you can see the current Report Server Mode (**Native** or **SharePoint Integrated**).
6. In the **Connect** pane, click **Report Manager URL** and note that you can use Reporting Services Configuration Manager to view and manage the Web site that is used to host Report Manager, the Web-based report access and management tool for Reporting Services.
7. In the **Connect** pane, click **E-mail Settings** and note that you can use Reporting Services Configuration Manager to view and manage the Simple Mail Transfer Protocol (SMTP) settings that Reporting Services uses to send email messages.
8. In the **Connect** pane, click **Execution Account** and note that you can use Reporting Services Configuration Manager to view and manage the account that Reporting Services uses to access resources on remote servers or data sources that do not require credentials.
9. In the **Connect** pane, click **Encryption Keys** and note that you can use Reporting Services Configuration Manager to manage the encryption keys that Reporting Services uses to encrypt sensitive data.
10. In the **Connect** pane, click **Scale-out Deployment** and note that you can use Reporting Services Configuration Manager to join multiple reporting servers in a Web farm to maximize scalability.

► **Task 2: Back up Reporting Services encryption keys**

- Back up the Reporting Services encryption keys to a file named RSKeys.snk in the E:\Labfiles\Starter\Ex2 folder and use a password of Pa\$\$w0rd:
  - a. In the **Connect** pane, click **Encryption Keys**.
  - b. In the **Backup** section, click **Backup**.
  - c. Next to the **File Location** box, click the ellipsis (...).
  - d. In the **Save As** dialog box, browse to the **E:\Labfiles\Starter\Ex2** folder, in the **File name** box, type **RSKeys.snk**, and then click **Save**.
  - e. In the **Password** and **Confirm Password** boxes, type **Pa\$\$w0rd**, and then click **OK**.

**Note:** You should back up the encryption keys for your Reporting Services instances. You will require these keys if you move or migrate the Report Server installation to another computer.

► **Task 3: Configure Reporting Services e-mail settings**

1. Configure the Reporting Services e-mail settings to use the SQLService@adventure-works.com e-mail account on the localhost SMTP Server:
  - a. In the **Connect** pane, click **E-mail Settings**.
  - b. In the **Sender Address** box, type **SQLService@adventure-works.com**.
  - c. In the **SMTP Server** box, type **localhost**, and then click **Apply**.
2. Review the results, and then click **Exit**.

► **Task 4: Review IIS configuration**

1. Click **Start**, click **All Programs**, click **Administrative Tools**, and then click **Internet Information Services (IIS) Manager**.

2. In the **User Account Control** dialog box, click **Continue**.
3. In the **Connections** pane, expand **MIAMI (MIAMI\Student)**, and then expand **Sites**. Notice the absence of the Report Manager and Report Server Web sites. SQL Server 2008 Reporting Services is independent of IIS.
4. Close Internet Information Services (IIS) Manager.
5. In the 6158C-MIA-SQL-08 on localhost - Virtual Machine Remote Control window, click **Close**.
6. In the **Close** dialog box, click **Turn off machine and discard changes**, and then click **OK**.

# Module 9

## SQL Server 2008 Analysis Services Enhancements

### Contents:

Lab Answer Keys	2
-----------------	---

# Lab Answer Keys

## Lab: Using SQL Server 2008 Analysis Services

### Exercise 1: Implementing Multidimensional Analysis

#### ► Task 1: Open an existing project

1. In the **6158C** Lab Launcher window, in the **6158C-MIA-SQL-09** virtual machine section, click **Launch**.
2. Log on to the virtual machine as Student, with a password of Pa\$\$w0rd:
  - a. Press ALT GR+DELETE, and then click **Student**.
  - b. In the **Password** box, type **Pa\$\$w0rd**, and then press ENTER.
3. Open Microsoft® SQL Server® Business Intelligence Development Studio:
  - Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, and then click **SQL Server Business Intelligence Development Studio**.
4. Open Ex1.sln from the E:\Labfiles\Starter\Ex1 folder:
  - a. On the **File** menu, point to **Open**, and then click **Project/Solution**.
  - b. In the **Open Project** dialog box, browse to the **E:\Labfiles\Starter\Ex1** folder, click **Ex1.sln**, and then click **Open**.

#### ► Task 2: Create a cube by using the Cube Wizard

1. Use the Cube Wizard to create a new cube. Use the existing FactInternetSales and FactResellerSales tables and remove the Dim Date table from the list of dimensions to be created. Specify the name Sales for the cube:
  - a. In Solution Explorer, right-click **Cubes**, and then click **New Cube**.
  - b. On the **Welcome to the Cube Wizard** page, click **Next**.
  - c. On the **Select Creation Method** page, verify that **Use existing tables** is selected, and then click **Next**.
  - d. On the **Select Measure Group Tables** page, select the following tables, and then click **Next**:
    - **FactInternetSales**
    - **FactResellerSales**
  - e. On the **Select Measures** page, review the measure groups and measures, and then click **Next**.
  - f. On the **Select New Dimensions** page, review the dimensions that are selected. If the **Dim Date** dimension is included, clear the selection for the **Dim Date** dimension, and then click **Next**.

**Note:** When you clear the selection for the **Dim Date** dimension, the DimDate table is also cleared.

- g. On the **Completing the Wizard** page, in the **Cube name** box, type **Sales**, and then click **Finish**.
2. Review the cube and dimensions that the Cube Wizard has created.

### ► Task 3: Add a dimension by using the Dimension Wizard

- Use the Dimension Wizard to create a new dimension. Use the existing table, Dim Date, and use the FullDateAlternateKey for the Name column. Use the information in the following table to configure the dimension attributes:
  - a. In Solution Explorer, right-click **Dimensions**, and then click **New Dimension**.
  - b. On the **Welcome to the Dimension Wizard** page, click **Next**.
  - c. On the **Select Creation Method** page, click **Use an existing table**, and then click **Next**.

**Note:** Most cubes require a time dimension. If a time table does not already exist in the data source, you can create a new time table on the server or in the data source. To create a time table in the data source, you must have the appropriate permissions to create the table in the underlying source.

- d. On the **Specify Source Information** page, verify that the **Data source view** is **AdventureWorksDW2008**, in the **Main table** box, select **DimDate**, in the **Name column** box, select **FullDateAlternateKey**, and then click **Next**.
- e. On the **Select Dimension Attributes** page, select and enable browsing for the following attributes, select the appropriate **Attribute Type**, and then click **Next**.

**Note:** When you select an attribute, the **Enable Browsing** check box is automatically selected.

Attribute name	Attribute type
Date Key (already selected and enabled for browsing)	Regular
English Month Name	Date\Calendar\Month
Month Number of Year	Date\Calendar\Month of Year
Calendar Quarter	Date\Calendar\Quarter
Calendar Year	Date\Calendar\Year
Fiscal Quarter	Date\Fiscal\Fiscal Quarter
Fiscal Year	Date\Fiscal\Fiscal Year

- f. On the **Completing the Wizard** page, review the dimension name and attributes, and then click **Finish**.

### ► Task 4: Create hierarchies

1. On the Dimension Structure tab, review the AMO warning on the Dim Date dimension:
  - On the **Dimension Structure** tab, in the **Attributes** pane, point to the **Dim Date** dimension until the AMO warning message appears. Review the note. You will create hierarchies for this dimension in this task.
2. Rename the English Month Name attribute to Month:
  - On the **Dimension Structure** tab, in the **Attributes** pane, right-click **English Month Name**, and then click **Rename**. Type **Month**, and then press ENTER.
3. Create a new hierarchy for the Calendar Year attribute, and then rename the hierarchy to Calendar Date:



- a. Drag the **Calendar Year** attribute from the **Attributes** pane to the **To create a new hierarchy, drag an attribute here** label in the **Hierarchies** pane.
  - b. Right-click the **Hierarchy** heading, and then click **Rename**.
  - c. Type **Calendar Date**, and then press ENTER.
4. Add the Calendar Quarter and Month attributes to the Calendar Date hierarchy:
  - a. Drag the **Calendar Quarter** attribute from the **Attributes** pane to the **<new level>** area directly under **Calendar Year** in the **Hierarchies** pane.
  - b. Drag **Month** to the **<new level>** area directly under **Calendar Quarter** in the **Hierarchies** pane.
5. Create a second hierarchy that includes the Fiscal Year, Fiscal Quarter, and Month attributes, and then rename the hierarchy Fiscal Date:
  - a. Drag the **Fiscal Year** attribute from the **Attributes** pane to the **To create a new hierarchy, drag an attribute here** label in the **Hierarchies** pane.
  - b. Drag the **Fiscal Quarter** attribute from the **Attributes** pane to the **<new level>** area directly under **Fiscal Year** in the **Hierarchies** pane.
  - c. Drag **Month** to the **<new level>** area directly under **Fiscal Quarter** in the **Hierarchies** pane.
  - d. Right-click the **Hierarchy** heading, and then click **Rename**.
  - e. Type **Fiscal Date** and then press ENTER.

#### ► Task 5: Modify attribute relationships

1. On the Dimension Structure tab, review the AMO warning for the Calendar Date hierarchy. You will create relationships for the hierarchy in this task:
  - On the **Dimension Structure** tab, in the **Hierarchies** pane, point to the **Calendar Date** hierarchy name until the AMO warning message appears. Review the message.
2. On the Attribute Relationships tab, in the Attribute Relationships pane, delete all of the attribute relationships:
  - a. In the Dim Date.dim [Design] window, click the **Attribute Relationships** tab.
  - b. In the **Attribute Relationships** pane, select all of the relationships, rightclick the selected relationships, and then click **Delete**.
  - c. In the **Delete Objects** dialog box, click **OK**.

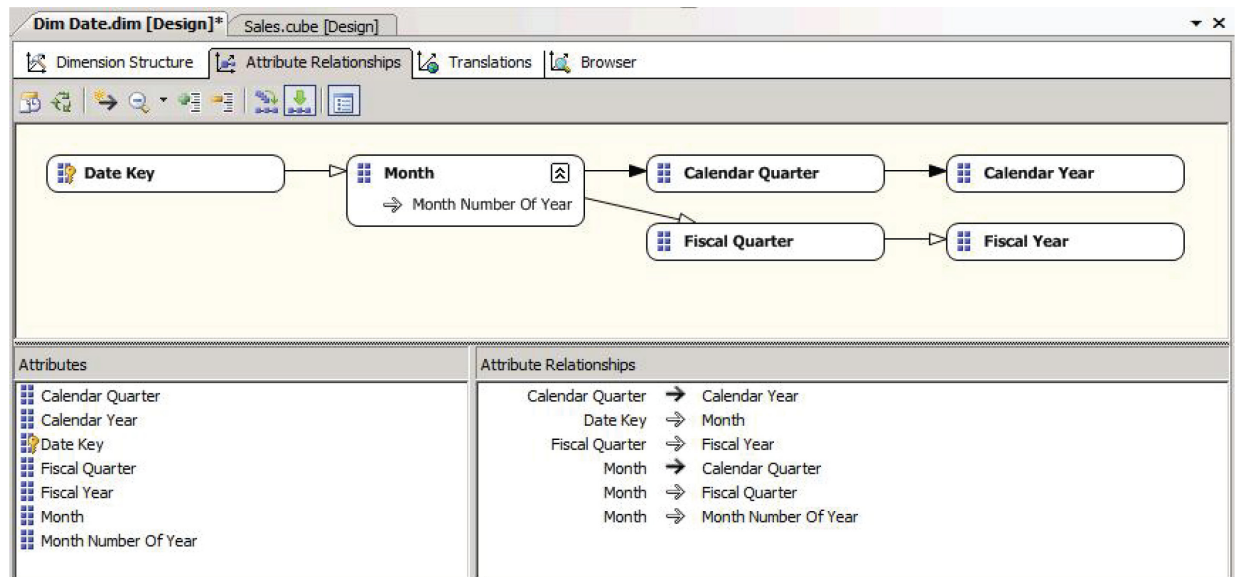
**Tip:** The **Attribute Relationships** pane is in the lower center of the window.

3. In the top pane, create a fiscal date hierarchy by creating relationships between Month and DateKey, Fiscal Quarter and Month, and Fiscal Year and Fiscal Quarter:
  - In the top pane, drag **Date Key** to **Month**, drag **Month** to **Fiscal Quarter**, and then drag **Fiscal Quarter** to **Fiscal Year**.
4. In the top pane, create a calendar date hierarchy by creating relationships between Calendar Quarter and Month, and Calendar Year and Calendar Quarter:
  - Drag **Month** to **Calendar Quarter**, and then drag **Calendar Quarter** to **Calendar Year** to create the attribute relationships for the **Calendar Date** hierarchy.

5. In the top pane, create a relationship between **Month** and **Month Number of Year**, and then use the **Properties** window to set the cardinality to **One**:
  - a. Drag **Month** to **Month Number of Year**. This creates a relationship between the **Month** and the **Month Number of Year** attributes.
  - b. In the **Attribute Relationships** pane, click the relationship between **Month** and **Month Number of Year**.
  - c. In the **Properties** window, change the **Cardinality** to **One**. There is always a one-to-one relationship between a month's date and the month's number of the year.
6. Change the relationship between **Calendar Quarter** and **Calendar Year** to a rigid relationship type:
  - In the **Attribute Relationships** pane, right-click the relationship between **Calendar Quarter** and **Calendar Year**, point to **Relationship Type**, and then click **Rigid**.
7. Change the relationship between **Month** and **Calendar Quarter** to a rigid relationship type:
  - In the **Attribute Relationships** pane, right-click the relationship between **Month** and **Calendar Quarter**, point to **Relationship Type**, and then click **Rigid**.

**Note:** Relationships are defined as rigid when the relationship will not change over time. For example, January is always part of the first quarter of a calendar year. The black arrow represents a rigid attribute relationship.

The figure below shows the completed attribute relationships:



8. Close SQL Server Business Intelligence Development Studio without saving changes:
  - a. On the **File** menu, click **Close Project**. If you are prompted to save changes, click **No**.
  - b. On the **File** menu, click **Exit**.

## ► Task 6: Design aggregations

1. Start SQL Server Business Intelligence Development Studio with administrative credentials:
  - a. Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, rightclick **SQL Server Business Intelligence Development Studio**, and then click **Run as administrator**.

- b. In the **User Account Control** dialog box, click **Continue**.

**Note:** In this task, you will be working with a solution that is located in the C:\Program Files folder. Elevated permissions are required to use files in this folder.

2. Open Adventure Works.sln from the C:\Program Files\Microsoft SQL Server\100\Tools\Samples\AdventureWorks 2008 Analysis Services Project\enterprise folder:
    - a. On the **File** menu, point to **Open**, and then click **Project/Solution**.
    - b. In the **Open Project** dialog box, browse to the **C:\Program Files\Microsoft SQL Server\100\Tools\Samples\AdventureWorks 2008 Analysis Services Project\enterprise** folder, click **Adventure Works.sln**, and then click **Open**.
  3. Open the Adventure Works.cube cube:
    - In Solution Explorer, right-click the **Adventure Works.cube** cube, and then click **Open**.
  4. On the **Aggregations** tab, right-click **Internet Sales (1 Aggregation Design)**, and then click **Design Aggregations**.
  5. In the Aggregation Design Wizard, modify the Internet\_Sales\_2004 partition. Update the count information, set the aggregation option of Performance gain reaches 30%, and then start the aggregation. Name the aggregation AggregationDesign2004, and then save the aggregation:
    - a. On the **Welcome to the Aggregation Design Wizard** page, click **Next**.
    - b. On the **Select Partitions to Modify** page, click **Internet\_Sales\_2004**, and then click **Next**.
    - c. On the **Review Aggregation Usage** page, review the current aggregation usage, and then click **Next**.
    - d. On the **Specify Object Counts** page, click **Count** to update the count information. When the count is completed, click **Next**.
    - e. On the **Set Aggregation Options** page, click **Performance gain reaches 30%**, and then click **Start**.
    - f. When the aggregation design finishes, review the results, and then click **Next**.
    - g. On the **Completing the Wizard** page, in the **Name** box, type **AggregationDesign2004**.
    - h. Verify that **Save the aggregations but do not process them** is selected, and then click **Finish**.
  6. In the center pane, review the two aggregation designs that are assigned to the **Internet Sales** measure group.
  7. On the **Aggregations** page, on the toolbar, click **Advanced View**.
  8. On the **Advanced View** page, click the **AggregationDesign2004** aggregation:
    - Verify that the **Measure Group** is **Internet Sales**, set the **Aggregation Design** to **AggregationDesign2004**, and then verify that **Sorting** is set to **(none)** and **Range** is set to **0 - 29**.
  9. Review the aggregations for the Date dimension:
    - In the center pane, expand the **Date** dimension, and then review the aggregations that have been created.
  10. In the **A0** column, select both the **Fiscal Year** and **Fiscal Quarter** check boxes.

11. Build the project:
  - On the **Build** menu, click **Build Adventure Works DW 2008**.
12. When the build is finished, review the warning in the Output window. Note that the aggregation should not include both the **Fiscal Year** and **Fiscal Quarter** attributes because there is a relationship between them:
  - a. On the **View** menu, click **Output**.
  - b. Review the warning in the Output window.
13. In the **AO** column, clear the **Fiscal Year** and **Fiscal Quarter** check boxes.
14. Close the project without saving changes:
  - a. On the **File** menu, click **Close Project**.
  - b. If you are prompted to save changes, click **No**.
15. Close Business Intelligence Development Studio:
  - On the **File** menu, click **Exit**.
16. If the Microsoft Visual Studio® development system stops responding, use Task Manager to end the task:
  - a. Right-click the taskbar, and then click **Task Manager**.
  - b. In the task list, click **Adventure Works – Microsoft Visual Studio (Administrator)**, and then click **End Task**.
  - c. In the **End Program** dialog box, click **End Now**.
  - d. In the **Windows Task Manager** dialog box, on the **File** menu, click **Exit Task Manager**.

## Exercise 2: Implementing Data Mining by Using the Microsoft Decision Trees Algorithm

### ► Task 1: Open an existing project

1. Open SQL Server Business Intelligence Development Studio:
  - Click **Start**, click **All Programs**, click **Microsoft SQL Server 2008**, and then click **SQL Server Business Intelligence Development Studio**.
2. Open Ex2.sln from the E:\Labfiles\Starter\Ex2 folder:
  - a. On the **File** menu, point to **Open**, and then click **Project/Solution**.
  - b. In the **Open Project** dialog box, browse to the **E:\Labfiles\Starter\Ex2** folder, click **Ex2.sln**, and then click **Open**.

### ► Task 2: Create a data-mining structure that includes a Decision Trees mining model

1. Use the Data Mining Wizard to create a new mining structure. Use an existing relational database or data warehouse and use the Microsoft Decision Trees mining model. Use the Adventure Works DM DSV data source view and the vTargetMail table for the Case input:
  - a. In Solution Explorer, right-click the **Mining Structures** folder, and then click **New Mining Structure**.
  - b. In the **Data Mining Wizard**, on the **Welcome to the Data Mining Wizard** page, click **Next**.

- c. On the **Select the Definition Method** page, verify that **From existing relational database or data warehouse** is selected, and then click **Next**.
  - d. On the **Create the Data Mining Structure** page, under the **Create mining structure with a mining model** option, verify that **Microsoft Decision Trees** is selected, and then click **Next**.
  - e. On the **Select Data Source View** page, in the **Available data source views** pane, verify that the **Adventure Works DM DSV** data source view is selected, and then click **Next**.
  - f. On the **Specify Table Types** page, in the **Input tables** grid, select the **Case** check box for the **vTargetMail** row, and then click **Next**.
2. On the **Specify the Training Data** page, in the **Mining model structure** grid, select the check boxes in the cells that are identified in the following table, and then click **Next**.

**Note:** Columns that are not listed in the table should not have any check boxes selected.

Table/column	Key	Input	Predictable
Age	No	Yes	No
Bike Buyer	No	No	Yes
CommuteDistance	No	Yes	No
CustomerKey	Yes	No	No
Gender	No	Yes	No
MaritalStatus	No	Yes	No
NumberCarsOwned	No	Yes	No
NumberChildrenAtHome	No	Yes	No
Region	No	Yes	No
Yearly Income	No	Yes	No

3. On the **Specify Columns' Content and Data Type** page, review the **Content Type** column for all numeric rows, and then click **Detect**.
4. When the detection is complete, notice that only the **Age** and **Yearly Income** fields remain **Continuous**, and then click **Next**.

**Note:** The **Specify Columns' Content and Data Type** page shows the data type that is determined from the source data. When you click **Detect**, Analysis Services scans numeric fields to determine whether they are continuous or discrete data. After the detection has occurred, the interface gives you the flexibility to manually edit both the **Data Type** and **Content Type** fields.

5. On the **Create Testing Set** page, ensure that **Percentage of data for testing** is set to **30%**, and then click **Next**.

6. Name the mining structure Target Mail, name the mining model Target Mail\_DT, enable drill-through, and then complete the wizard:
  - a. On the **Completing the Wizard** page, in the **Mining structure name** box, type **Target Mail**.
  - b. In the **Mining model name** box, type **Target Mail\_DT**.
  - c. Select the **Allow drill through** check box, and then click **Finish**. The **Mining Structure** designer will open.

► **Task 3: Create a filtered mining model**

1. On the Mining Models tab, create a related mining model named Pacific Target Mail\_DT by using the Microsoft Decision Trees algorithm:
  - a. In the Target Mail.dmm [Design] window, click the **Mining Models** tab.
  - b. On the **Mining Models** page, on the toolbar, click **Create a related mining model**.
  - c. In the **New Mining Model** dialog box, in the **Model name** box, type **Pacific Target Mail\_DT**.
  - d. Verify that the **Algorithm name** is set to **Microsoft Decision Trees**, and then click **OK**.
2. Verify that all of the columns for the Pacific Target Mail\_DT model are set to **Input** except for the **Bike Buyer** and **Customer Key** columns. **Bike Buyer** should be set to **PredictOnly** and **Customer Key** should be set to **Key**.
3. Set a model filter for the Pacific Target Mail\_DT model for the Pacific region:
  - a. On the **Mining Models** page, right-click the **Pacific Target Mail\_DT** model heading, and then click **Set Model Filter**.
  - b. In the **[Pacific Target Mail\_DT] Model Filter** dialog box, in the first row, in the **Mining Structure Column** box, click **Region**, and then verify that the **Operator** is **=**.
  - c. In the **Value** column, type **Pacific**, and then click **OK**.

► **Task 4: Deploy the Analysis Services solution**

1. Deploy the project. Set the file name to Ex2Deploy and use a password of Pa\$\$w0rd:
  - a. On the **Build** menu, click **Deploy Ex2**.
  - b. In the **Account Password** dialog box, in the **Password** box, type **Pa\$\$w0rd**, and then click **OK**.
2. Observe the deployment progress that is shown in the **Deployment Progress** pane (normally on the right side of Business Intelligence Development Studio). The **Deployment Progress** pane provides detailed information during the deployment process.

**Note:** Analysis Services may take a while to process the data-mining models.

► **Task 5: Compare the Target Mail\_DT and Pacific Target Mail\_DT models**

1. On the Mining Model Viewer tab, review the Decision Tree and Dependency Network tabs for each model:
  - a. In the Target Mail.dmm [Design] window, click the **Mining Model Viewer** tab.
  - b. If you are prompted to build and deploy the project, click **Yes**, and then wait for the deployment to finish.

- c. Review the **Decision Tree** and **Dependency Network** tabs for each mining model.
2. Notice that the third-level predictive columns vary between the Target Mail\_DT and Pacific Target Mail\_DT models:
  - a. Click the **Decision Tree** tab, and then in the **Mining Model** list, click **Target Mail\_DT**. Note the third-level column name for the **Number Cars Owned = 1** case.
  - b. Click the **Decision Tree** tab, and then in the **Mining Model** list, click **Pacific Target Mail\_DT**. Note that the third-level column name for the **Number Cars Owned = 1** case differs from the all-region scenario.
3. On the Mining Accuracy Chart tab, set the Predict Value to 1 for both of the models, and then on the Lift Chart tab, click the Profit Chart that uses the default values:
  - a. In the Target Mail.dmm [Design] window, click the **Mining Accuracy Chart** tab.
  - b. On the **Input Selection** tab, review the default values, and then change the **Predict Value** to **1** for both the Target Mail\_DT and Pacific Target Mail\_DT models.
  - c. On the **Lift Chart** tab, in the **Chart type** box, click **Profit Chart**.
  - d. In the **Profit Chart Settings** dialog box, review the default settings, and then click **OK**.
4. Compare the results for the Target Mail\_DT and Pacific Target Mail\_DT models.

#### ► Task 6: Perform cross-validation

1. On the **Cross Validation** tab, set the **Fold Count** to **5**, **Max Cases** to **5000**, **Target Attribute** to **Bike Buyer**, in the **Target State** box, type **1**, and then click **Get Results**.
2. Review the cross-validation results.

The first section of the report is for the Target Mail\_DT model. The first test results that are reported are for the Classification Test. For each fold, the report includes values that represent whether the model correctly predicted the Target State in each case.

A Target State of 1 was included in the definition, so the **Measure** column can be interpreted as follows:

- When the Test Case value is 1, and the model prediction is 1, the Measure is reported as a True Positive.
  - When the Test Case value is 1, and the model prediction is 0, the Measure is reported as a False Negative.
  - When the Test Case value is 0, and the model prediction is 1, the Measure is reported as a False Positive.
  - When the Test Case value is 0, and the model prediction is 0, the Measure is reported as a True Negative.
3. Close the project without saving changes:
    - a. On the **File** menu, click **Close Project**.
    - b. If you are prompted to save changes, click **No**.

## Exercise 3: Implementing Data Mining by Using the Microsoft Time Series Algorithm

### ► Task 1: Open an existing project

- Open Ex3.sln from the E:\Labfiles\Starter\Ex3 folder:
  - a. On the **File** menu, point to **Open**, and then click **Project/Solution**.
  - b. In the **Open Project** dialog box, browse to the **E:\Labfiles\Starter\Ex3** folder, click **Ex3.sln**, and then click **Open**.

### ► Task 2: Create a data-mining structure

1. Use the Data Mining Wizard to create a new mining structure. Use an existing relational database or data warehouse and use the Microsoft Time Series mining model. Use the Adventure Works DW2008 data source view and the vTimeSeries table for the Case input:
  - a. In Solution Explorer, right-click the **Mining Structures** folder, and then click **New Mining Structure**.
  - b. In the **Data Mining Wizard**, on the **Welcome to the Data Mining Wizard** page, click **Next**.
  - c. On the **Select the Definition Method** page, click **From existing relational database or data warehouse**, and then click **Next**.
  - d. On the **Create the Data Mining Structure** page, under the **Create mining structure with a mining model** option, in the **Which data mining technique do you want to use?** list, click **Microsoft Time Series**, and then click **Next**.
  - e. On the **Select Data Source View** page, in the **Available data source views** pane, verify that the **AdventureWorks DW2008** data source view is selected, and then click **Next**.
  - f. On the **Specify Table Types** page, in the **Input tables** grid, in the **vTimeSeries** row, select the **Case** check box, and then click **Next**.
2. On the **Specify the Training Data** page, in the **Mining model structure** grid, select the check boxes in the cells that are identified in the following table, and then click **Next**:

**Note:** Columns that are not listed in the table should not have any check boxes selected.

Table/column	Key	Input	Predictable
Model Region	Yes	No	No
Quantity	No	Yes	Yes
Time Index	Yes	No	No

3. On the **Specify Columns' Content and Data Type** page, review the default content and data types, and then click **Next**.
4. On the **Completing the Wizard** page, in the **Mining structure name** box, type **SalesForecast**, and then click **Finish**.

### ► Task 3: Modify a mining structure and model

1. On the Mining Model tab, open the **Algorithm Parameters** dialog box:
  - a. In the SalesForecast.dmm [Design] window, click the **Mining Models** tab.



- b. Right-click the **vTimeSeries** model heading, and then click **Set Algorithm Parameters**.
2. In the **Algorithm Parameters** dialog box, review the parameters and default values.
3. Set the **PERIODICITY\_HINT** value to **{12}**, and then click **OK**.

**Note:** The **PERIODICITY\_HINT** setting of **12** tells the algorithm that a pattern repeats itself every 12 periods (or months in this case).

#### ► Task 4: Deploy the Analysis Services solution

1. Deploy the project. Set the file name to Ex3Deploy and use a password of Pa\$\$w0rd:
  - a. On the **Build** menu, click **Deploy Ex3**.
  - b. In the **Account Password** dialog box, in the **Password** box, type **Pa\$\$w0rd**, and then click **OK**.
2. Observe the deployment progress that is shown in the **Deployment Progress** pane. The **Deployment Progress** pane provides detailed information during the deployment process.

**Note:** Analysis Services may take a while to process the data-mining models.

#### ► Task 5: View the Time Series mining model

1. On the **Mining Model Viewer** tab, review the chart that shows the sales over time.

**Tip:** Hide Solution Explorer and any other windows that are blocking the chart information by clicking the **Auto Hide** icon (the pushpin in the upper-right corner of each window).

The chart shows the quantity of bicycles that were sold by model and region. The values that are shown as solid lines are actual sales. The values that are shown in the darker shaded area and shown as dotted lines are predicted values.

2. Point to any one of the lines on the chart. A ScreenTip will appear and display information that is relevant to the location on the line.
3. Click in the chart where the chart background color changes to a darker shade of gray. The mining legend window that is located on the far right now displays the values that are related to that point in time. The timestamp at the top of the legend tells you the month that you are looking at. For example, Timestamp: 200407 is July 2004.
4. Compare the quantities of the M200 Europe: Quantity, R250 Europe: Quantity, and R750 Europe: Quantity sold:
  - In the right pane of the chart area, in the drop-down list, select only **M200 Europe: Quantity**, **R250 Europe: Quantity**, and **R750 Europe: Quantity**, and then click **OK**.
5. Click in the chart where the chart background color changes to a darker shade of gray. Notice that the mining legend is updated for the new chart view. This chart makes it easy to see that, in the European market, the outlook for the M200 model exceeds the other models that are currently being sold.
6. Change the **Prediction steps** to **20** to view predictions farther into the future.
7. Close SQL Server Business Intelligence Development Studio without saving changes:
  - On the **File** menu, click **Exit**. If you are prompted to save changes, click **No**.

## Exercise 4: Using the Data Mining Add-ins for Office 2007

### ► Task 1: Open an existing spreadsheet

1. Open the Microsoft Office Excel® spreadsheet software:
  - Click **Start**, click **All Programs**, click **Microsoft Office**, and then click **Microsoft Office Excel 2007**.
2. Open DMAddins\_SampleData.xlsx from the E:\Labfiles\Starter\Ex4 folder:
  - a. Click the **Office** button, and then click **Open**.
  - b. In the **Open** dialog box, browse to the **E:\Labfiles\Starter\Ex4** folder, click **DMAddins\_SampleData.xlsx**, and then click **Open**.

**Note:** You can access this file when you install the Data Mining Add-ins. It contains data that is extracted from the Adventure Works database and formatted as a table in Office Excel.

3. Review the two additional tabs that are available on the Ribbon: the Data Mining tab and the Analyze tab:
  - a. On the Ribbon, click the **Data Mining** tab. Review the buttons that are available on the tab.
  - b. On the Ribbon, click the **Analyze** tab. Review the buttons that are available on the tab.

**Tip:** If either of the tabs are not displayed on the Ribbon, click in Cell **B7** in the table to display them.

4. Create a connection to the DMAddinsDB database on the local server:
  - a. On the Ribbon, on the **Analyze** tab, in the **Connection** group, click **<No Connection>**.
  - b. In the **Analysis Services Connections** dialog box, click **New**.
  - c. In the **Server name** box, type **(local)**.
  - d. In the **Catalog name** box, click **DMAddinsDB**, and then click **OK**.
  - e. In the **Analysis Services Connections** dialog box, click **Close**.

### ► Task 2: Analyze factors that influence purchasing

1. Activate Cell A4 on the Table Analysis Tools Sample sheet, and then view the Analyze tab on the Ribbon:
  - a. Click the **Table Analysis Tools Sample** tab at the bottom of the window, and then click Cell **A4**.
  - b. On the Ribbon, click the **Analyze** tab.
2. Use the Prediction Calculator to analyze the data on the Table Analysis Tools Sample sheet to find which are the key factors that influence bike buyers. Target the Purchased Bike column for a value of Yes and exclude the Purchased Bike column from the analysis:
  - a. On the Ribbon, on the **Analyze** tab, in the **Table Analysis Tools** group, click **Prediction Calculator**.
  - b. In the **Prediction Calculator** dialog box, in the **Target** box, click **Purchased Bike**, and then in the **Exactly** box, click **Yes**.
  - c. Click the **Choose columns to be used for analysis** link.

- d. In the **Advanced Columns Selection** dialog box, note that ID is automatically not selected because it is a unique field, so it will not have a bearing on the results.
  - e. Scroll through the list, clear the **Purchased Bike** check box, and then click **OK**.
  - f. In the **Prediction Calculator** dialog box, click **Run**.
3. On the Prediction Report for Purcha sheet, change the value for a false positive cost to 5 and for a true positive profit to 50. Note that, as you change these values, the suggested threshold decreases:
  - a. Click the **Prediction Report for Purcha** tab at the bottom of the window.
  - b. Click in Cell **B6**, type in **5**, and then press ENTER.
  - c. Click in Cell **B8**, type in **50**, and then press ENTER.
4. In the Score Breakdown table, review the factors that have a high relative impact on the purchasing of a bike.
5. On the Prediction Calculator for Pu sheet, enter your details to determine whether you are a likely bike purchaser:
  - a. Click the **Prediction Calculator for Pu** tab at the bottom of the window.
  - b. Click in Cell **B7**, and then enter your marital status.
  - c. Click in Cell **B8**, and then enter your gender.
  - d. Click in Cell **B9**, and then enter your income.
  - e. Continue to complete the data entry in Cells **B10–B17**.
  - f. Review the total value in Cell **C18**. If this value is greater than the threshold, you are predicted to buy a bike. Verify that, based on your total figure, the prediction in Cell **C20** is correct.
6. On the Printable Calculator for Pur sheet, review the printable format for the questionnaire:
  - a. Click the **Printable Calculator for Pur** tab at the bottom of the window.
  - b. Click the **Office** button, point to **Print**, and then click **Print Preview**.
  - c. Review the printable format, and then on the Ribbon, click **Close Print Preview**.

### ► Task 3: Analyze multiple product purchasing

1. Activate Cell A4 on the Associate sheet, and then view the Analyze tab on the Ribbon:
  - a. Click the **Associate** tab at the bottom of the window, and then click in Cell **A4**.
  - b. On the Ribbon, click the **Analyze** tab.
2. Use the Shopping Basket Analysis tool to analyze which Product combinations are sold together:
  - a. On the Ribbon, on the **Analyze** tab, click **Shopping Basket Analysis**.
  - b. In the **Transaction ID** box, click **Order Number**.
  - c. In the **Item** box, click **Product**.
  - d. In the **Item Value (Optional)** box, click **Product Price**, and then click **Run**.
3. On the **Shopping Basket Bundled Item** sheet, review the bundles that have high **Overall value of Bundle** results. Note that high-selling bundles do not necessarily result in a high overall value if their individual value is low.

- 
4. On the Shopping Basket Recommendation sheet, review the recommendations for bundled items:
    - a. Click the **Shopping Basket Recommendation** tab at the bottom of the window.
    - b. Review the recommendations for bundled items.
  5. Close Office Excel without saving changes:
    - a. Click the **Office** button, and then click **Exit Excel**.
    - b. If you are prompted to save changes, click **No**.
  6. In the 6158C-MIA-SQL-09 on localhost - Virtual Machine Remote Control window, click **Close**.
  7. In the **Close** dialog box, click **Turn off machine and discard changes**, and then click **OK**.

# Send Us Your Feedback

You can search the Microsoft Knowledge Base for known issues at [Microsoft Help and Support](#) before submitting feedback. Search using either the course number and revision, or the course title.

---

**Note** Not all training products will have a Knowledge Base article – if that is the case, please ask your instructor whether or not there are existing error log entries.

---

## Courseware Feedback

Send all courseware feedback to [support@microsoft.com](mailto:support@microsoft.com). We truly appreciate your time and effort. We review every e-mail received and forward the information on to the appropriate team. Unfortunately, because of volume, we are unable to provide a response but we may use your feedback to improve your future experience with Microsoft Learning products.

## Reporting Errors

When providing feedback, include the training product name and number in the subject line of your e-mail. When you provide comments or report bugs, please include the following:

- Document or CD part number
- Page number or location
- Complete description of the error or suggested change

Please provide any details that are necessary to help us verify the issue.

---

**Important** All errors and suggestions are evaluated, but only those that are validated are added to the product Knowledge Base article.

---