**Lab 1:**

Users report that insert statements against the view named **Production.vExpensiveExpiredProduct** do not succeed. You confirm that the insert statements that are being used are valid.

You need to ensure that insert statements can complete successfully. Use SSMS

- Open up SSMS.

- Goto Server **\Databases \Adventureworks\ Views\ Production.vExpensiveExpiredProduct\ Triggers**

- Select the trigger: **trgvvvExpensiveExpiredProductInsert**

- Right-click and select Delete on menu

- A delete object window will appear. Select **OK**

- In the Question window click on **DONE**

**Lab 2:**

Users report that they receive an error message when they execute a CLR user-defined function named: **Production.ufnGetProductionInfo** in the **Adventureworks** database. The error is occurring because the function is being denied access to the local file system. You must ensure that the least permissions possible to solve the problem are applied. Use SSMS

- Open up SSMS

- Goto Server **\Databases \Adventureworks \Programmability \Functions \Scalar-valued Functions**

- Select **Production.ufnGetProductionInfo**

- Right-click and select **View Dependencies** from menu

- When the window opens, select the bullet beside **Objects on which [ufnGetProductionIngo] depends**

- In the field below select **AWAssembly2** and then click on OK button

- In SSMS expand the **Assemblies** folder and select **AWAssembly2**

- Right click and select Properties from the menu. The Properties window will appear.

- In the Permission set field, select **External Access**

- Click on **OK**.

- In the Question window click on **DONE**

**Lab 3:**

You are a database administrator for your company. The only non-system database on the server is **AdventureWorks**.

A new written company policy states that nested trigger execution is not allowed.

You need to ensure that this policy is implemented by modifying the appropriate server and database settings in the database properties dialog box, or both of these dialog boxes, in SQL Server Management Studio (SSMS).

Perform the appropriate actions in the simulation. When you are finished, click Done.

- In Server properties window under Advanced page select **Allow Triggers to Fire Others** to **False**.

- Click **OK** in Server properties window.

- Click **OK** in **AdventureWorks** Database properties window. **\*\*\* Do NOT alter ‘Recursive triggers enabled’ in Database properties \*\*\***

- In the Question window click on **DONE**

**Lab 4:**

You are a database administrator for your company.

Car license plate data is no longer required to be stored in the **AdventureWorks** database.

You need to identify the tables that contain columns referencing the user-defined data type named **LicencePlate**, and you must remove those columns. You should not remove the actual **LicencePlate** user-defined data type.

You need to perform this task by using SQL Server Management Studio (SSMS).

Perform the appropriate actions in the simulation. When you are finished, click Done.

- In the SSMS window select **Databases \AdventureWorks \Programmability \Types\ User-defined Data Types** and select **dbo.LicencePlate(char(7),null)**

- Check Dependecies to see which tables use that type.

- In the SSMS window select **Databases \AdventureWorks \Tables \HumanResources.Employee \Columns**

- Right click **PrimaryCar LicencePlate(char(7),null)** and select delete.

- In Delete Object window click **OK**

- In the Question window click on **DONE**

**Lab 5:**

You are a database administrator for your company.

There is an uncommitted transaction in the **AdventureWorks** database. You must find the process that is associated with this transaction, and you must kill that process. You must not kill any other processes.

You need to kill the process by using the **Activity Monitor** dialog how within SQL Server Management Studio (SSMS).

Perform the appropriate actions in the simulation. When you are finished, click **Done**.

- Goto **Server \Management \Activity Monitor**. Double click to open up the Activity Monitor window

- In Activity Monitor on Server 1 window under **Process Info** page right click on **Process ID 56** and select **Kill Process**.

- Click **Yes**.

- Click on **Refresh**

- Click on Close

- In the Question window click on **DONE**

**Lab 6:**

You are a database administrator for your company. You need to create a new database named Sales.

The new database must meet the following requirements.

\* The data file must be 10,000 MB in size

\* The log file must be 3,000 MB in size.

\* Both files should have the default settings for automatic file growth and maximum file size.

\* The transaction log must be stored on a fault-tolerant volume.

\* The data file, the transaction log file, and the Windows installation must all be located on different volumes.

\* The database file must be stored in a folder named SQL in the root of each volume.

\* The transaction log must be set to be automatically truncated The available volumes are shown in the exhibit.

The available volumes are shown in the following table.

|  |  |  |
| --- | --- | --- |
| Volume | RAID level | Free space |
| С: (Windows volume) | RAID1 | 20 GB |
| D: | RAID1 | 20 GB |
| E: | RAID0 | 20 GB |

You need to create the **Sales** database by using the **New Database** dialog box.

Perform the appropriate actions in the simulation. When you are finished, click **Done**.

- In the New Database window under General page type in Database name as **Sales**.

- Select Initial Size to 10000 MB for Sales and 3000 MB for **Sales\_Log**.

- Type in Path as **E:\SQL** for Sales and **D:\SQL** for **Sales\_Log**.

- In the New Database window under Options page select Recovery Model as **Simple**. **\*\*\*Do NOT alter CHECKSUM \*\*\***

- Click OK.

- In the Question window click on DONE

**Lab 7:**

You work as a database administrator for your company. A SQL Server Login named Tom is used only within the **AdventureWorks** database.

A new company security policy is being implemented. This new policy states that SQL Server Authentication cannot be used. In preparation for implementation of the policy, you must remove the Tom Login.

The new policy states that users associated with SQL Server Logins must be removed, and the ownership of anything that is owned by these users must be transferred to **dbo**.

You need to remove the Tom Login by using SQL Server Management Studio (SSMS).

Perform the appropriate actions in the simulation. When you are finished, click **Done**.

- In the SSMS window select **Databases \AdventureWorks \Security \Schemas\Tom**. Right click and select **Properties**.

- In Schemas properties – Tom window, under **General** page type in **Schema Owner** as **dbo** and click **OK**

- In the SSMS window select **Security \Users\Tom**. Right-click and select **Delete**.

- In the Delete Object window click **OK**. Again click on **OK**

- In the SSMS window select **Security \Logins\Tom**. Right-click and select **Delete**.

- In the Delete Object window click **OK**. Again click on **OK**

- In the Question window click on **DONE.**

**Lab 8:**

You are a database developer for your company. Users report that queries that are executed against some views are returning incorrect results.

You investigate and discover that the problem is being caused by one or more views. You want to ensure that each of these views returns the results described in the following table.

|  |  |
| --- | --- |
| View name | Required result |
| HumanResources.vEmployeeTopVacationHours | Returns the employees that have the highest value in the **VacationHours** field |
| HumanResources.vEmployeeAverageOrMoreVacationHours | Returns the employees that have a value in the  **VacationHours** field that is greater than or equal  to the average |
| Production.vProductMediumPrice | Returns the products that have a value in the **ListPrice** field that is greater than or equal to 10 and less than 20. |
| Poduct.vProductMini | Returns the products that have a value in the **Name** Field that starts with the text **Mini** using a case sensitive comparison. |

You need to examine the query in each оf the four views and then delete the view or views that return incorrect results. You must not delete views that return correct results.

Perform the appropriate actions in the simulation. When you are finished, click **Done**.

- In SSMS select Databases **\AdventureWorks \Views \Humanresources.vEmployeeTopVacationHours**. Right-click and select **Modify.** \*\*\* As a precautionary measure you may want to view the other ‘views’, in case Microsoft tracks your actions. \*\*\*

**-** right click on **Humanresources.vEmployeeTopVacationHours.** Select **Delete.**

-In the Delete Object window click **OK**

- In the Question window click on **DONE**

**Lab 9:**

You are a database administrator for your company.

You need to back up all the changes that have occurred in the **AdventureWorks** database since the last full database backup. The size of the backup file must be kept to a minimum. The database is currently using log shipping, which must not be disrupted.

The backup file must be named AW.bak, and must be placed in the D:\Backup folder.

You need to perform the backup by using the **Back Up Database** dialog box.

Perform the appropriate actions in the simulation. When you are finished, click **Done**.

- Select **Databases \Adventureworks**. Right-click and select **Tasks \Back Up**.

- In the Back-up Database window under **General page** select **Backup Type** as **Differential**

- Click on **Remove** button to remove the current path and click on **Add** button to add a new path.

- In the **Select Backup Destination** type in the path **D:\Backup\AW.bak** and click on **OK**.

- In the Question window click on **DONE**

**Lab 10:**

You are a database administrator for your company.

You need to create a maintenance plan to rebuild indexes in the **Adventureworks** database. The indexes must be rebuilt using a fill factor of 90. The fill factor must be applied to both the leaf level and the balanced tree of the index. While the maintenance plan is running, all tables must be available for queries to be run.

You need to accomplish this task by completing the **Define Rebuild Index Task** page in the Maintenance Wizard.

Perform the appropriate actions in the simulation. When you are finished, click **Done**.

- Click bullet for **Change free space per page percentage to**. Enter **10** into the percentage field.

- Put a check mark in the **Pad Index** box

- Put a checkmark in **Sort results in tempdb**

- Put a check mark in **Keep index online while reindexing**

**Lab 11:**

You are a database administrator for your company.

You need to back up all the changes that have occurred in the **AdventureWorks** database since the last transaction log backup. The size of the backup file must be kept to a minimum.

The backup file must be named AW.bak, and must be placed in the D:\Backup folder.

You need to perform the backup by using the **Back Up Database** dialog box.

Perform the appropriate actions in the simulation. When you are finished, click Done.

- In the Back-up Database window under General page select **Backup Type** as **Transaction Log**

- Click on **Remove** button to remove the current path and click on **Add** button to add a new path.

- In the **Select Backup Destination** window, type in D:\Backup\AW.bak and click on **OK**.

- In the Question window click on **DONE**

**Lab 12:**

You are a database administrator for your company.

You need to move the **AdventureWorks** database to a different SQL Server instance. You must perform a backup that contains all of the changes since the last transaction log backup. You must ensure that no changes can be applied to the database after this backup completes.

The backup file must be named AW.bak, and must be placed in the D:\Backup folder.

You need to perform the backup by using the **Back Up Database** dialog box.

Perform the appropriate actions in the simulation. When you are finished, click **Done**.

- In the **Back-up Database** window under General page select **Backup Type** as **Transaction Log**

- Click on **Remove** button to remove the current path and click on **Add** button to add a new path.

- In the **Select Backup Destination** window, type in the desired path D:\Backup\AW.bak and click **OK**.

- In the Back-up Database window under **Options** page, select **Back up the tail of the log, and leave the database in the restoring state** and click on **OK**.

- In the Question window click on **DONE**

**Lab 13:**

You are a database administrator for your company. The company has two SQL Server 2005 servers named Server1 and Server2.

You need to add Server2 as a linked server on Server1. Only the login named App1 should be able to access Server2 through Server1. A login named App2 with the password **password** must be used for the linked server connections to Server2. You must apply the fewest possible settings.

You need to create the linked server by using the **New Linked Server** dialog box on Server1.

Perform the appropriate actions in the simulation. When you are finished, click **Done**.

- In the New Linked Server window under **General** page type **Linked Server** as **Server2.** Select **SQL Server** for **Server Type**.

- In the New Linked Server window under **Security** page select **Be made using this security context** radio button.

- Remote login: **App2**. With password: **password**. Click on **OK**

- In the Question window click on **Done**

**Lab 14:**

You are a database administrator for your company. The **AdventureWorks** database is the only non-system database on the server. Transaction log backups are part of the backup routine for the **AdventureWorks** database.

Each night, several SQL Server Integration Services (SSIS) packages are run to import 5 million rows of data.

You need to minimize the log space that is used by these import operations. You must modify the appropriate server and database settings by using the **Database Properties** dialog box at the **Server Properties** dialog box, or both of these dialog boxes, in SQL Server Management Studio (SSMS).

Perform the appropriate actions in the simulation. When you are finished, click **Done**.

- In the Database properties under **Options** page select Recovery Model as **Bulked-Logged**.

- Under **Other Options \Automatic \Auto Shrink** select **True**

- Under **Other Options \Recovery \Page Verify** select **TORN\_PAGE\_DETECTION**

- In the Question window click on **DONE**

**Lab 15:**

You are a database administrator for your company.

A query against the **Sales.SalesOrderDetail** table is performing poorly. You must create an index that will cause the cost of this query to be as low as possible. You must minimize the space that is used by the index. The index must contain as few columns as possible.

There is an existing clustered index on the table. This index must not be dropped.

Because the table is on a production server, it is not possible to create different indexes and test their performance.

You need to create the required index by using the **New Index** dialog box.

Perform the appropriate actions in the simulation. When you are finished, click **Done**.

- In the New Index window under **General** page give the Index name as **productid**. Click the **Add** button.

- In the Select columns from **Sales.SalesOrderDetail** check mark **PRODUCTID** and click OK.

- In the New Index window under **Options** page check mark **Set fill factor** and type in **0**% and click **OK**.

- In the Question window click on **DONE**

**Lab 16:**

You are a database administrator at your company. The company has two SQL Server 2005 Servers named Server1 and Server2.

You need to add Server2 as a linked server on Server1. All the logins should be able to access Server2 through Server1. A login named App2 with the password “password” must be used for the linked server connection to Server2. You must apply the fewest possible settings.

You need to create the linked server by using the New Linked Server dialog box on Server1.

Perform the appropriate actions in the simulation. When you are finished, click **Done**.

- In the New Linked Server window under **General** page type Linked Server as **Server2** and select **SQL Server** for Server Type.

- In the New Linked Server window under **Security** page select **Not be made** radio button.

- Click on the **Add** button. Under **Local Login** select **App1**. Under Remote User type **App2**. Under Remote Password type **password**.

- In the Question window click on **DONE**