

Microsoft 70-461 Exam Questions & Answers

Number: 70-461
Passing Score: 700
Time Limit: 300 min
File Version: 42.2



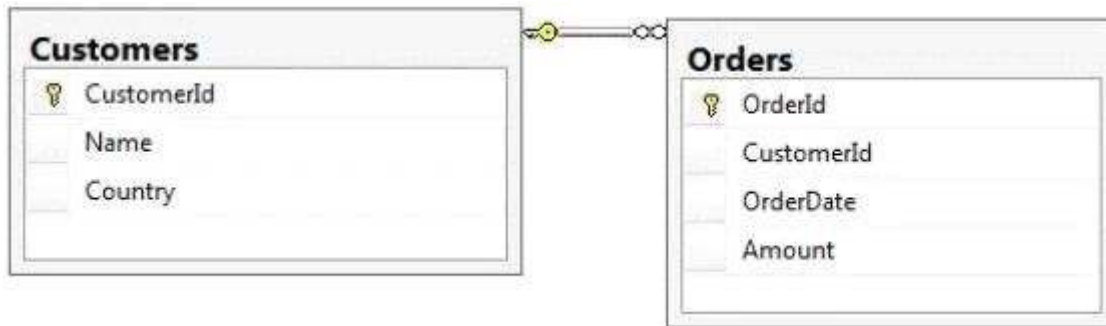
Microsoft 70-461 Exam Questions & Answers

Exam Name: Querying Microsoft SQL Server 2012

Exam A

QUESTION 1

You administer a Microsoft SQL Server 2012 database named ContosoDb. Tables are defined as shown in the exhibit. (Click the Exhibit button.)



You need to display rows from the Orders table for the Customers row having the CustomerId value set to 1 in the following XML format:

```
<row OrderId="1" OrderDate="2000-01-01T00:00:00" Amount="3400.00" Name="Customer A" Country="Australia" />
<row OrderId="2" OrderDate="2001-01-01T00:00:00" Amount="4300.00" Name="Customer A" Country="Australia" />
```

Which Transact-SQL query should you use?

- A.

```
SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId
WHERE Customers.CustomerId = 1
FOR XML RAW
```
- B.

```
SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId
WHERE Customers.CustomerId = 1
FOR XML RAW, ELEMENTS
```
- C.

```
SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId
WHERE Customers.CustomerId = 1
FOR XML AUTO
```
- D.

```
SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML AUTO, ELEMENTS
```
- E.

```
SELECT Name, Country, OrderId, OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML AUTO
```
- F.

```
SELECT Name, Country, OrderId, OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML AUTO, ELEMENTS
```
- G.

```
SELECT Name AS '@Name', Country AS '@Country', OrderId, OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML PATH ('Customers')
```
- H.

```
SELECT Name AS 'Customers/Name', Country AS 'Customers/Country', OrderId,
OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
```

```
WHERE Customers.CustomerId= 1
FOR XML PATH ('Customers')
```

Correct Answer: A

Section: (none)

Explanation

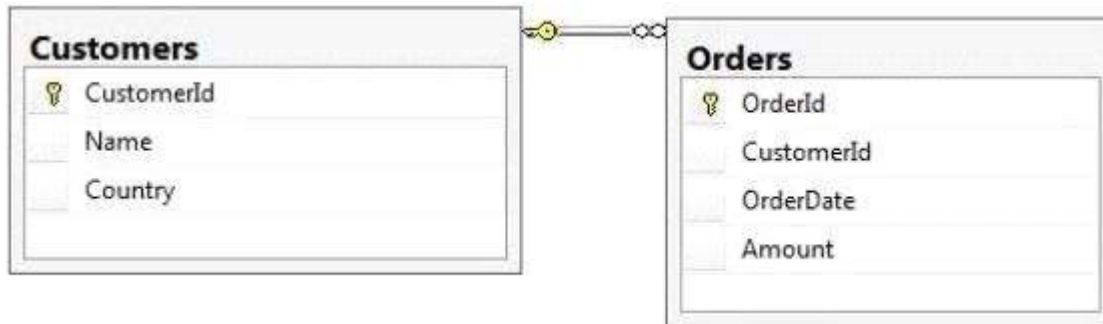
Explanation/Reference:

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/bb510464.aspx>

QUESTION 2

You administer a Microsoft SQL Server 2012 database named ContosoDb. Tables are defined as shown in the exhibit. (Click the Exhibit button.)



You need to display rows from the Orders table for the Customers row having the CustomerId value set to 1 in the following XML format.

```
<Orders OrderId="1" OrderDate="2000-01-01T00:00:00" Amount="3400.00">
  <Customers Name="Customer A" Country="Australia" />
</Orders>
<Orders OrderId="2" OrderDate="2001-01-01T00:00:00" Amount="4300.00">
  <Customers Name="Customer A" Country="Australia" />
</Orders>
```

Which Transact-SQL query should you use?

- A.

```
SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId
WHERE Customers.CustomerId = 1
FOR XML RAW
```
- B.

```
SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId
WHERE Customers=CustomerId = 1
FOR XML RAW, ELEMENTS
```
- C.

```
SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId
WHERE Customers.CustomerId = 1
FOR XML AUTO
```
- D.

```
SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML AUTO, ELEMENTS
```
- E.

```
SELECT Name, Country, OrderId, OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
```

```
WHERE Customers.CustomerId= 1
FOR XML AUTO
```

- F.

```
SELECT Name, Country, OrderId, OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML AUTO, ELEMENTS
```
- G.

```
SELECT Name AS '@Name', Country AS '@Country', OrderId, OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML PATH ('Customers')
```
- H.

```
SELECT Name AS 'Customers/Name', Country AS 'Customers/Country', OrderId,
OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML PATH ('Customers')
```

Correct Answer: C

Section: (none)

Explanation

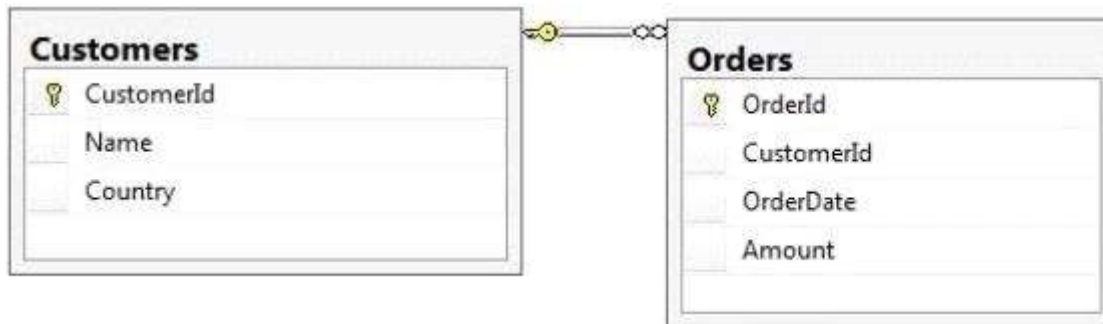
Explanation/Reference:

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms188273.aspx>

QUESTION 3

You administer a Microsoft SQL Server 2012 database named ContosoDb. Tables are defined as shown in the exhibit. (Click the Exhibit button.)



You need to display rows from the Orders table for the Customers row having the CustomerId value set to 1 in the following XML format.

```
<CUSTOMERS Name="Customer A" Country="Australia">
  <ORDERS OrderID="1" OrderDate="2001-01-01" Amount="3400.00" />
  <ORDERS OrderID="2" OrderDate="2002-01-01" Amount="4300.00" />
</CUSTOMERS>
```

Which Transact-SQL query should you use?

- A.

```
SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId
WHERE Customers.CustomerId = 1
FOR XML RAW
```
- B.

```
SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId
WHERE Customers.CustomerId = 1
FOR XML RAW, ELEMENTS
```
- C.

```
SELECT OrderId, OrderDate, Amount, Name, Country
```

```
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId
WHERE Customers.CustomerId = 1
FOR XML AUTO
```

- D.

```
SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML AUTO, ELEMENTS
```
- E.

```
SELECT Name, Country, OrderId, OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML AUTO
```
- F.

```
SELECT Name, Country, OrderId, OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML AUTO, ELEMENTS
```
- G.

```
SELECT Name AS '@Name', Country AS '@Country', OrderId, OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML PATH ('Customers')
```
- H.

```
SELECT Name AS 'Customers/Name', Country AS 'Customers/Country', OrderId, OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
WHERE Customers.CustomerId= 1
FOR XML PATH ('Customers')
```

Correct Answer: E

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

QUESTION 4

You develop a Microsoft SQL Server 2012 server database that supports an application. The application contains a table that has the following definition:

```
CREATE TABLE Inventory (
ItemID int NOT NULL PRIMARY KEY,
ItemsInStore int NOT NULL,
ItemsInWarehouse int NOT NULL)
```

You need to create a computed column that returns the sum total of the ItemsInStore and ItemsInWarehouse values for each row. The new column is expected to be queried heavily, and you need to be able to index the column. Which Transact-SQL statement should you use?

- A.

```
ALTER TABLE Inventory
ADD TotalItems AS ItemsInStore + ItemsInWarehouse
```
- B.

```
ALTER TABLE Inventory
ADD TotalItems AS ItemsInStore + ItemsInWarehouse PERSISTED
```
- C.

```
ALTER TABLE Inventory
ADD TotalItems AS SUM(ItemsInStore, ItemsInWarehouse) PERSISTED
```
- D.

```
ALTER TABLE Inventory
ADD TotalItems AS SUM(ItemsInStore, ItemsInWarehouse)
```

Correct Answer: B

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms174979.aspx>

QUESTION 5

You develop a Microsoft SQL Server 2012 database that has two tables named SavingAccounts and LoanAccounts. Both tables have a column named AccountNumber of the nvarchar data type. You use a third table named Transactions that has columns named TransactionId, AccountNumber, Amount, and TransactionDate. You need to ensure that when multiple records are inserted in the Transactions table, only the records that have a valid AccountNumber in the SavingAccounts or LoanAccounts are inserted. Which Transact-SQL statement should you use?

- A.

```
CREATE TRIGGER TrgValidateAccountNumber
ON Transactions
INSTEAD OF INSERT
AS
BEGIN
    INSERT INTO Transactions
    SELECT TransactionID, AccountNumber, Amount, TransactionDate FROM inserted
    WHERE AccountNumber IN
    (SELECT AccountNumber FROM LoanAccounts
    UNION SELECT AccountNumber FROM SavingAccounts))
END
```
- B.

```
CREATE TRIGGER TrgValidateAccountNumber
ON Transactions
FOR INSERT
AS
BEGIN
    INSERT INTO Transactions
    SELECT TransactionID, AccountNumber, Amount, TransactionDate FROM inserted
    WHERE AccountNumber IN
    (SELECT AccountNumber FROM LoanAccounts
    UNION SELECT AccountNumber FROM SavingAccounts))
END
```
- C.

```
CREATE TRIGGER TrgValidateAccountNumber
ON Transactions
INSTEAD OF INSERT
AS
BEGIN
    IF EXISTS (
        SELECT AccountNumber FROM inserted EXCEPT
        (SELECT AccountNumber FROM LoanAccounts
        UNION SELECT AccountNumber FROM SavingAccounts))
    BEGIN
        ROLLBACK TRAN
    END
END
```
- D.

```
CREATE TRIGGER TrgValidateAccountNumber
ON Transactions
FOR INSERT
AS
BEGIN
    IF EXISTS (
        SELECT AccountNumber FROM inserted EXCEPT
        (SELECT AccountNumber FROM LoanAccounts
        UNION SELECT AccountNumber FROM SavingAccounts))
    BEGIN
        ROLLBACK TRAN
    END
END
```

Correct Answer: A

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

QUESTION 6

You are a database developer of a Microsoft SQL Server 2012 database. The database contains a table named Customers that has the following definition:

```
CREATE TABLE Customer
(CustomerID INT NOT NULL PRIMARY KEY,
 CustomerName VARCHAR(255) NOT NULL,
 CustomerAddress VARCHAR(1000) NOT NULL)
```

You are designing a new table named Orders that has the following definition:

```
CREATE TABLE Orders
(OrderID INT NOT NULL PRIMARY KEY,
 CustomerID INT NOT NULL,
 OrderDescription VARCHAR(2000))
```

You need to ensure that the CustomerId column in the Orders table contains only values that exist in the CustomerId column of the Customer table. Which Transact-SQL statement should you use?

- A. ALTER TABLE Orders
ADD CONSTRAINT FX_Orders_CustomerID FOREIGN KEY (CustomerId) REFERENCES
Customer (CustomerId)
- B. ALTER TABLE Customer
ADD CONSTRAINT FK_Customer_CustomerID FOREIGN KEY (CustomerId) REFERENCES
Orders (CustomerId)
- C. ALTER TABLE Orders
ADD CONSTRAINT CK_Crders_CustomerID
CHECK (CustomerId IN (SELECT CustomerId FROM Customer))
- D. ALTER TABLE Customer
ADD OrderId INT NOT NULL;

ALTER TABLE Customer
ADD CONSTRAINT FK_Customer_OrderID FOREIGN KEY (CrderlD) REFERENCES Orders
(CrderlD);
- E. ALTER TABLE Orders
ADD CONSTRAINT PK Orders CustomerId PRIMARY KEY (CustomerID)

Correct Answer: A

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms189049.aspx>

QUESTION 7

You develop a Microsoft SQL Server 2012 database. You create a view from the Orders and OrderDetails

tables by using the following definition.

```
CREATE VIEW vOrders
WITH SCHEMABINDING
AS
SELECT o.ProductID,
       o.OrderDate,
       SUM(od.UnitPrice * od.OrderQty) AS Amount
FROM OrderDetails AS od INNER JOIN
     Orders AS o ON od.OrderID = o.OrderID
WHERE od.SalesOrderID = o.SalesOrderID
GROUP BY o.OrderDate, o.ProductID
GO
```

You need to ensure that users are able to modify data by using the view. What should you do?

- A. Create an AFTER trigger on the view.
- B. Modify the view to use the WITH VIEW_METADATA clause.
- C. Create an INSTEAD OF trigger on the view.
- D. Modify the view to an indexed view.

Correct Answer: C

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms187956.aspx>

QUESTION 8

You have a view that was created by using the following code:

```
CREATE VIEW Sales.OrdersByTerritory
AS
SELECT OrderID
       ,OrderDate
       ,SalesTerritoryID
       ,TotalDue
FROM Sales.Orders;
```

You need to create an inline table-valued function named Sales.fn_OrdersByTerritory, which must meet the following requirements:

- Accept the @T integer parameter.
- Use one-part names to reference columns.
- Filter the query results by SalesTerritoryID.
- Return the columns in the same order as the order used in OrdersByTerritoryView.

Which code segment should you use?

To answer, type the correct code in the answer area.

A.

Correct Answer:

Section: (none)

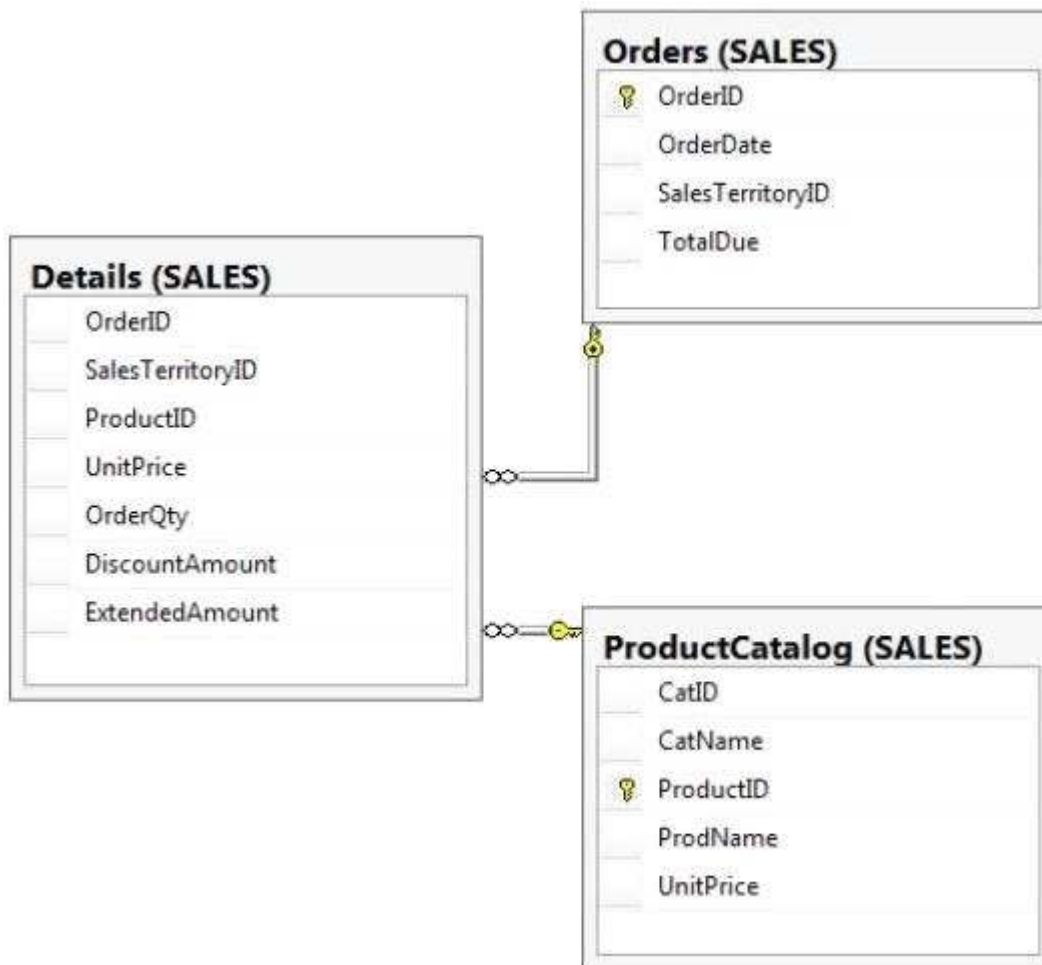
Explanation

Explanation/Reference:

```
CREATE FUNCTION Sales.fn_OrdersByTerritory (@T int)
RETURNS TABLE
AS
RETURN
(
    SELECT OrderID, OrderDate, SalesTerritoryID, TotalDue
    FROM Sales.OrdersByTerritory
    WHERE SalesTerritoryID = @T
)
```

QUESTION 9

You have a database that contains the tables shown in the exhibit. (Click the Exhibit button.)



You need to create a query that calculates the total sales of each OrderId from the Sales.Details table. The solution must meet the following requirements:

- Use one-part names to reference columns.
- Sort the order of the results from OrderId.
- NOT depend on the default schema of a user.
- Use an alias of TotalSales for the calculated ExtendedAmount.
- Display only the OrderId column and the calculated TotalSales column.

Which code segment should you use?
To answer, type the correct code in the answer area.

A.

Correct Answer:

Section: (none)

Explanation


Explanation/Reference:


```
SELECT OrderID, SUM(ExtendedAmount) AS TotalSales
FROM Sales.Details
GROUP BY OrderID
ORDER BY OrderID
```

QUESTION 10

You have a database that contains the tables shown in the exhibit. (Click the Exhibit button).

OrderDetails			
	Column Name	Data Type	Allow Nulls
	ListPrice	money	<input type="checkbox"/>
	Quantity	int	<input type="checkbox"/>
			<input type="checkbox"/>

Customers			
	Column Name	Data Type	Allow Nulls
	CustomerID	int	<input type="checkbox"/>
	FirstName	varchar(100)	<input type="checkbox"/>
	LastName	varchar(100)	<input type="checkbox"/>
			<input type="checkbox"/>

Orders			
	Column Name	Data Type	Allow Nulls
	OrderID	int	<input type="checkbox"/>
	OrderDate	datetime	<input type="checkbox"/>
	CustomerID	int	<input type="checkbox"/>
			<input type="checkbox"/>

You need to create a query for a report. The query must meet the following requirements:

- NOT use object delimiters.
- Return the most recent orders first.
- Use the first initial of the table as an alias.
- Return the most recent order date for each customer.
- Retrieve the last name of the person who placed the order.
- Return the order date in a column named MostRecentOrderDate that appears as the last column in the report.

The solution must support the ANSI SQL-99 standard.

Which code segment should you use?

To answer, type the correct code in the answer area.

A.

Correct Answer:

Section: (none)

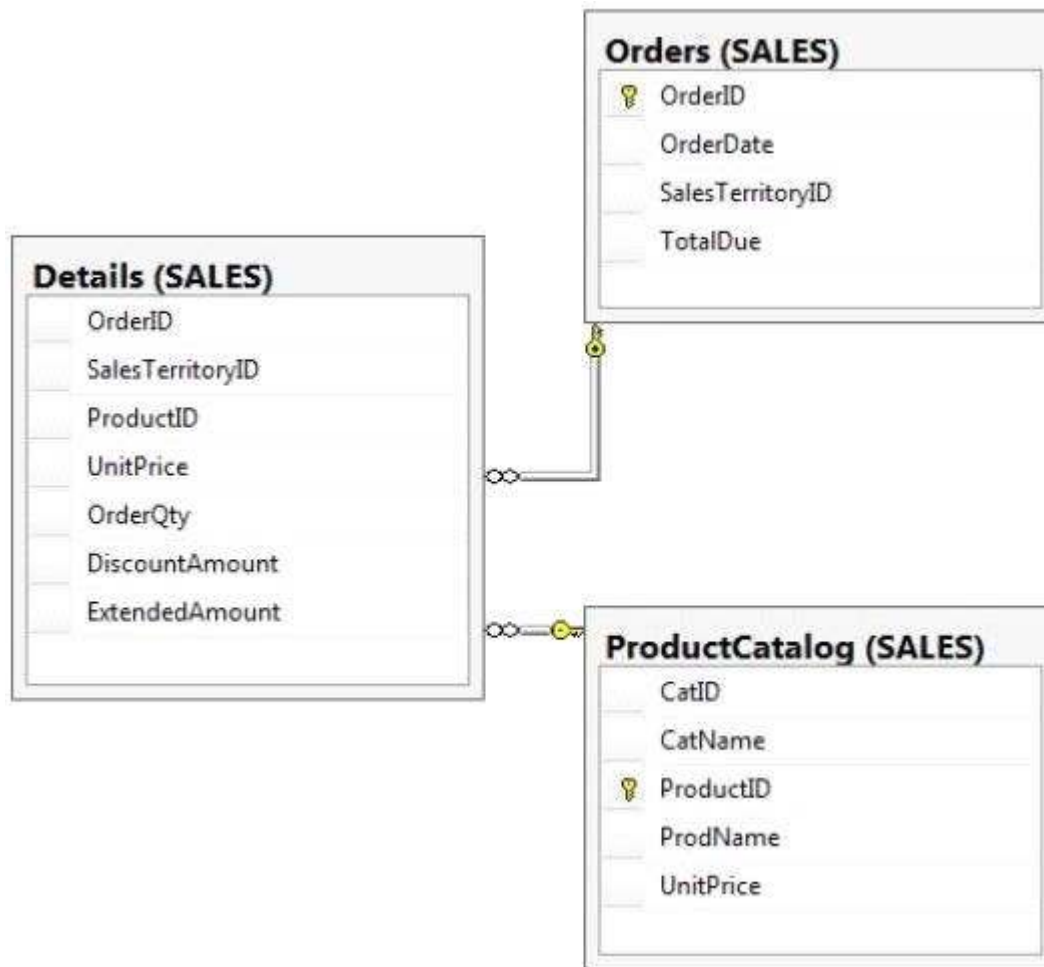
Explanation

Explanation/Reference:

```
SELECT C.LastName, MAX(O.OrderDate) AS MostRecentOrderDate
FROM Customers AS C INNER JOIN Orders AS O
ON C.CustomerID = O.CustomerID
GROUP BY C.LastName
ORDER BY O.OrderDate DESC
```

QUESTION 11

You have a database that contains the tables as shown in the exhibit. (Click the Exhibit button.)



You need to create a query that returns a list of products from Sales.ProductCatalog. The solution must meet the following requirements:

- UnitPrice must be returned in descending order.
- The query must use two-part names to reference the table.
- The query must use the RANK function to calculate the results.
- The query must return the ranking of rows in a column named PriceRank.
- The list must display the columns in the order that they are defined in the table.
- PriceRank must appear last.

Which code segment should you use?

To answer, type the correct code in the answer area.

A.

Correct Answer:

Section: (none)

Explanation


Explanation/Reference:


```
SELECT ProductCatalog.CatID, ProductCatalog.CatName, ProductCatalog.ProductID,
ProductCatalog.ProdName, ProductCatalog.UnitPrice,
RANK() OVER (PARTITION BY ProductCatalog.UnitPrice ORDER BY
ProductCatalog.UnitPrice DESC) AS PriceRank
FROM Sales.ProductCatalog
ORDER BY ProductCatalog.UnitPrice DESC
```

QUESTION 12

You have a database that contains the tables as shown below:

OrderDetails			
	Column Name	Data Type	Allow Nulls
	ListPrice	money	<input type="checkbox"/>
	Quantity	int	<input type="checkbox"/>
			<input type="checkbox"/>

Customers			
	Column Name	Data Type	Allow Nulls
	CustomerID	int	<input type="checkbox"/>
	FirstName	varchar(100)	<input type="checkbox"/>
	LastName	varchar(100)	<input type="checkbox"/>
			<input type="checkbox"/>

Orders			
	Column Name	Data Type	Allow Nulls
	OrderID	int	<input type="checkbox"/>
	OrderDate	datetime	<input type="checkbox"/>
	CustomerID	int	<input type="checkbox"/>
			<input type="checkbox"/>

You have a stored procedure named Procedure1. Procedure1 retrieves all order ids after a specific date. The rows for Procedure1 are not sorted. Procedure1 has a single parameter named Parameter1. Parameter1 uses the varchar type and is configured to pass the specific date to Procedure1. A database administrator discovers that OrderDate is not being compared correctly to Parameter1 after the data type of the column is changed to datetime. You need to update the SELECT statement to meet the following requirements:

- The code must NOT use aliases.
- The code must NOT use object delimiters.
- The objects called in Procedure1 must be able to be resolved by all users.
- OrderDate must be compared to Parameter1 after the data type of Parameter1 is changed to datetime.

Which SELECT statement should you use?

To answer, type the correct code in the answer area.

A.

Correct Answer:

Section: (none)

Explanation

Explanation/Reference:

```
SELECT Orders.OrderID
FROM Orders
WHERE Orders.OrderDate>CONVERT(datetime,@Parameter1)
```

QUESTION 13

You administer a Microsoft SQL Server database that supports a banking transaction management application. You need to retrieve a list of account holders who live in cities that do not have a branch location. Which Transact-SQL query or queries should you use? (Each correct answer presents a complete solution. Choose all that apply.)

- A.

```
SELECT AccountHolderID
FROM AccountHolder
WHERE CityID NOT IN (SELECT CityID FROM BranchMaster)
```
- B.

```
SELECT AccountHolderID
FROM AccountHolder
WHERE CityID <> ALL (SELECT CityID FROM BranchMaster)
```
- C.

```
SELECT AccountHolderID
FROM AccountHolder
WHERE CityID <> SOME (SELECT CityID FROM BranchMaster)
```
- D.

```
SELECT AccountHolderID
FROM AccountHolder
WHERE CityID <> ANY (SELECT CityID FROM BranchMaster)
```

Correct Answer: AB

Section: (none)

Explanation

Explanation/Reference:

Verified the answers as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms188047.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ms177682.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ms173545.aspx>

QUESTION 14

You administer a Microsoft SQL Server 2012 database. The database contains a table named Employee. Part of the Employee table is shown in the exhibit. (Click the Exhibit button.)

Employee (jek)

Column Name	Condensed Type
EmployeeID	int
EmployeeNum	char(10)
LastName	nvarchar(200)
FirstName	nvarchar(200)
MiddleName	nvarchar(200)
DateHired	date
DepartmentID	int
JobTitle	varchar(200)
ReportsToID	int

Column name	Description
EmployeeID(pk)	Uniquely identifies the employee record in the table Used throughout the database by all the other tables that reference the Employee table
EmployeeNum	An alphanumeric value calculated according to company requirements Has to be unique within the Employee table Exists only within the Employee table
DepartmentID	References another table named Department that contains data for each department in the company
ReportsToID	Contains the EmployeeID of the manager to whom an employee reports
ReportsToID	Contains the EmployeeID of the manager to whom an employee reports

Confidential information about the employees is stored in a separate table named EmployeeData. One record exists within EmployeeData for each record in the Employee table. You need to assign the appropriate constraints and table properties to ensure data integrity and visibility. On which column in the Employee table should you create a unique constraint?

- A. DateHired
- B. DepartmentID
- C. EmployeeID
- D. EmployeeNum
- E. FirstName
- F. JobTitle
- G. LastName
- H. MiddleName
- I. ReportsToID

Correct Answer: D

Section: (none)


Explanation

Explanation/Reference:

Verified answer as correct.

QUESTION 15

You administer a Microsoft SQL Server 2012 database. The database contains a table named Employee. Part of the Employee table is shown in the exhibit. (Click the Exhibit button.)

The image is a screenshot of a SQL Server Enterprise Manager window. The title bar of the window is blue and contains the text "Employee (jek)". Below the title bar is a table with two columns: "Column Name" and "Condensed Type". The table lists the following columns and their types: EmployeeID (int), EmployeeNum (char(10)), LastName (nvarchar(200)), FirstName (nvarchar(200)), MiddleName (nvarchar(200)), DateHired (date), DepartmentID (int), JobTitle (varchar(200)), and ReportsToID (int). The table is displayed in a grid format with a vertical scrollbar on the right side.

Column Name	Condensed Type
EmployeeID	int
EmployeeNum	char(10)
LastName	nvarchar(200)
FirstName	nvarchar(200)
MiddleName	nvarchar(200)
DateHired	date
DepartmentID	int
JobTitle	varchar(200)
ReportsToID	int

Column name	Description
EmployeeID	Uniquely identifies the employee record in the table Used throughout the database by all the other tables that reference the Employee table
EmployeeNum	An alphanumeric value calculated according to company requirements Has to be unique within the Employee table Exists only within the Employee table
DepartmentID	References another table named Department that contains data for each department in the company
ReportsToID	Contains the EmployeeID of the manager to whom an employee reports

Confidential information about the employees is stored in a separate table named EmployeeData. One record exists within EmployeeData for each record in the Employee table. You need to assign the appropriate constraints and table properties to ensure data integrity and visibility. On which column in the Employee table should you use an identity specification to include a seed of 1,000 and an increment of 1?

- A. DateHired
- B. DepartmentID
- C. EmployeeID
- D. EmployeeNum
- E. FirstName
- F. JobTitle
- G. LastName
- H. MiddleName
- I. ReportsToID

Correct Answer: C

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

QUESTION 16

You administer a Microsoft SQL Server 2012 database that includes a table named Products. The Products table has columns named ProductId, ProductName, and CreatedDateTime. The table contains a unique constraint on the combination of ProductName and CreatedDateTime. You need to modify the Products table to meet the following requirements:

- Remove all duplicates of the Products table based on the ProductName column.
- Retain only the newest Products row.

Which Transact-SQL query should you use?

- A. WITH CTEDupRecords
AS
(
 SELECT MAX(CreatedDateTime) AS CreatedDateTime, ProductName
 FROM Products
 GROUP BY ProductName
 HAVING COUNT(*) > 1
)
DELETE p
FROM Products p
JOIN CTEDupRecords cte ON
p.ProductName = cte.ProductName
AND p.CreatedDateTime > cte.CreatedDateTime
- B. WITH CTEDupRecords
AS
(
 SELECT MAX(CreatedDateTime) AS CreatedDateTime, ProductName
 FROM Products
 GROUP BY ProductName
 HAVING COUNT(*) > 1
)
DELETE p
FROM Products p
JOIN CTEDupRecords cte ON
cte.ProductName = p.ProductName
AND cte.CreatedDateTime > p.CreatedDateTime
- C. WITH CTEDupRecords
AS
(
 SELECT MIN(CreatedDateTime) AS CreatedDateTime, ProductName
 FROM Products
 GROUP BY ProductName
)
DELETE p
FROM Products p
JOIN CTEDupRecords cte ON
p.ProductName = cte.ProductName
- D. WITH CTEDupRecords
AS
(
 SELECT MAX(CreatedDateTime) AS CreatedDateTime, ProductName
 FROM Products
 GROUP BY ProductName
 HAVING COUNT(*) > 1
)
DELETE p
FROM Products p
JOIN CTEDupRecords cte ON
p.ProductName = cte.ProductName

Correct Answer: B

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

QUESTION 17

You develop three Microsoft SQL Server 2012 databases named Database1, Database2, and Database3. You have permissions on both Database1 and Database2. You plan to write and deploy a stored procedure named `dbo.usp_InsertEvent` in Database3. `dbo.usp_InsertEvent` must execute other stored procedures in the other databases. You need to ensure that callers that do not have permissions on Database1 or Database2 can execute the stored procedure. Which Transact-SQL statement should you use?

- A. `USE Database2`
- B. `EXECUTE AS OWNER`
- C. `USE Database1`
- D. `EXECUTE AS CALLER`

Correct Answer: B

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms188354.aspx>

Reference: <http://blog.sqlauthority.com/2007/10/06/sql-server-executing-remote-stored-procedure-calling-stored-procedure-on-linked-server/>

QUESTION 18

You develop a Microsoft SQL Server 2012 database. You need to create a batch process that meets the following requirements:

- Status information must be logged to a status table.
- If the status table does not exist at the beginning of the batch, it must be created.

Which object should you use?

- A. Scalar user-defined function
- B. Inline user-defined function
- C. Table-valued user-defined function
- D. Stored procedure

Correct Answer: D

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms186755.aspx>

QUESTION 19

You administer a database that includes a table named Customers that contains more than 750 rows. You create a new column named PartitionNumber of the int type in the table. You need to assign a PartitionNumber for each record in the Customers table. You also need to ensure that the PartitionNumber satisfies the following conditions:

- Always starts with 1.
- Starts again from 1 after it reaches 100.

Which Transact-SQL statement should you use?

- A.

```
CREATE SEQUENCE CustomerSequence AS int
START WITH 0
INCREMENT BY 1
```

- ```

MINVALUE 1
MAXVALUE 100
UPDATE Customers SET PartitionNumber = NEXT VALUE FOR CustomerSequence
DROP SEQUENCE CustomerSequence

```
- B. CREATE SEQUENCE CustomerSequence AS int  
 START WITH 1  
 INCREMENT BY 1  
 MINVALUE 1  
 MAXVALUE 100  
 CYCLE  
 UPDATE Customers SET PartitionNumber = NEXT VALUE FOR CustomerSequence  
 DROP SEQUENCE CustomerSequence
- C. CREATE SEQUENCE CustomerSequence AS int  
 START WITH 1  
 INCREMENT BY 1  
 MINVALUE 1  
 MAXVALUE 100  
 UPDATE Customers SET PartitionNumber = NEXT VALUE FOR CustomerSequence + 1  
 DROP SEQUENCE CustomerSequence
- D. CREATE SEQUENCE CustomerSequence AS int  
 START WITH 1  
 INCREMENT BY 1  
 MINVALUE 0  
 MAXVALUE 100  
 CYCLE  
 UPTATE Customers SET PartitionNumber = NEXT VALUE FOR CustomerSequence  
 DROP SEQUENCE CustomerSequence

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ff878091.aspx>

## QUESTION 20

You use Microsoft SQL Server 2012 to develop a database application. You create a stored procedure named DeleteJobCandidate. You need to ensure that if DeleteJobCandidate encounters an error, the execution of the stored procedure reports the error number. Which Transact-SQL statement should you use?

- A. DECLARE @ErrorVar INT;  
 DECLARE @RowCountVar INT;
- ```

EXEC DeleteJobCandidate

SELECT @ErrorVar = @@ERROR, @RowCountVar = @@ROWCOUNT;
IF (@ErrorVar <> 0)
    PRINT N'Error = ' + CAST(@@ErrorVar AS NVARCHAR(8)) +
        N', Rows Deleted = ' + CAST(@@RowCountVar AS NVARCHAR(8));
GO

```
- B. DECLARE @ErrorVar INT;
 DECLARE @RowCountVar INT;
- ```

EXEC DeleteJobCandidate

SELECT @ErrorVar = ERROR_STATE(), @RowCountVar = @@ROWCOUNT;
IF (@ErrorVar <> 0)
 PRINT N'Error = ' + CAST(ERRORSTATE() AS NVARCHAR(8)) +

```

```

 N', Rows Deleted = ' + CAST(@@RowCountVar AS NVARCHAR(8));
GO
C. EXEC DeleteJobCandidate

IF (ERROR_STATE() != 0)
 PRINT N'Error = ' + CAST(@@ERROR AS NVARCHAR(8)) +
 N', Rows Deleted = ' + CAST(@@ROWCOUNT AS NVARCHAR(8));
GO
D. EXEC DeleteJobCandidate

PRINT N'Error = ' + CAST(@@ERROR AS NVARCHAR(8)) +
 N', Rows Deleted = ' + CAST(@@ROWCOUNT AS NVARCHAR(8));
GO

```

**Correct Answer: A**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms190193.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ms188790.aspx>

#### QUESTION 21

You use Microsoft SQL Server 2012 to create a stored procedure as shown in the following code segment. (Line numbers are included for reference only.)

```

01 CREATE PROCEDURE DeleteCandidate
02 @InputCandidateID INT;
03 AS
04 BEGIN
05 BEGIN TRANSACTION;
06 BEGIN TRY
07 DELETE HumanResources.JobCandidate
08 WHERE JobCandidateID = @InputCandidateID;
09 INSERT INTO Audit.Log(Operation,OperationDate)
10 VALUES('Delete',SYSDATETIME());
11 COMMIT TRANSACTION;
12 END TRY
13 BEGIN CATCH
14
15 COMMIT TRANSACTION
16 ELSE
17 ROLLBACK TRANSACTION;
18 END CATCH
19 END;

```

The procedure can be called within other transactions. You need to ensure that when the DELETE statement from the HumanResourcesJobCandidate table succeeds, the modification is retained even if the insert into the Audit.Log table fails. Which code segment should you add to line 14?

A. IF @@TRANCOUNT = 0

- B. IF (XACT\_STATE ()) = 0
- C. IF (XACT\_STATE ()) = 1
- D. IF @@TRANCOUNT = 1

**Correct Answer:** C

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms189797.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ms187967.aspx>

## QUESTION 22

You use Microsoft SQL Server 2012 to develop a database application. Your application sends data to an NVARCHAR(MAX) variable named @var. You need to write a Transact-SQL statement that will find out the success of a cast to a decimal (36,9). Which code segment should you use?

- A. 

```
BEGIN TRY
 SELECT convert(decimal(36,9), @var) AS Value, 'True' AS BadCast
END TRY
BEGIN CATCH
 SELECT convert(decimal(36,9), @var) AS Value, 'False' AS BadCast
END CATCH
```
- B. 

```
TRY (
 SELECT convert(decimal(36,9), @var)
 SELECT 'True' AS BadCast
)
CATCH (
 SELECT 'False' AS BadCast
)
```
- C. 

```
SELECT
CASE
 WHEN convert(decimal(36,9), @var) IS NULL
 THEN 'True'
 ELSE 'False'
END
AS BadCast
```
- D. 

```
SELECT
 IIF(TRY_PARSE(@var AS decimal(36,9)) IS NULL, 'True', 'False')
AS BadCast
```

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/hh213126.aspx>

## QUESTION 23

You are writing a set of queries against a FILESTREAM-enabled database. You create a stored procedure that will update multiple tables within a transaction. You need to ensure that if the stored procedure raises a run-time error, the entire transaction is terminated and rolled back. Which Transact-SQL statement should you include at the beginning of the stored procedure?

- A. SET TRANSACTION ISOLATION LEVEL SERIALIZABLE

- B. SET XACT\_ABORT OFF
- C. SET TRANSACTION ISOLATION LEVEL SNAPSHOT
- D. SET IMPLICIT\_TRANSACTIONS ON
- E. SET XACT\_ABORT ON
- F. SET IMPLICIT\_TRANSACTIONS OFF

**Correct Answer:** E

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms188792.aspx>

#### QUESTION 24

You have a Microsoft SQL Server 2012 database that contains tables named Customers and Orders. The tables are related by a column named CustomerID. You need to create a query that meets the following requirements:

- Returns the CustomerName for all customers and the OrderDate for any orders that they have placed.
- Results must include customers who have not placed any orders.

Which Transact-SQL query should you use?

- A. 

```
SELECT CustomerName, OrderDate
FROM Customers
RIGHT OUTER JOIN Orders
ON Customers.CustomerID = Orders.CustomerID
```
- B. 

```
SELECT CustomerName, OrderDate
FROM Customers
JOIN Orders
ON Customers.CustomerID = Orders.CustomerID
```
- C. 

```
SELECT CustomerName, OrderDate
FROM Customers
CROSS JOIN Orders
ON Customers.CustomerID = Orders.CustomerID
```
- D. 

```
SELECT CustomerName, OrderDate
FROM Customers
LEFT OUTER JOIN Orders
ON Customers.CustomerID = Orders.CustomerID
```

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms177634.aspx>

#### QUESTION 25

You create a stored procedure that will update multiple tables within a transaction. You need to ensure that if the stored procedure raises a run-time error, the entire transaction is terminated and rolled back. Which Transact-SQL statement should you include at the beginning of the stored procedure?

- A. SET XACT\_ABORT ON
- B. SET ARITHABORT ON



- C. TRY
- D. BEGIN
- E. SET ARITHABORT OFF
- F. SET XACT\_ABORT OFF

**Correct Answer:** A

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms190306.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ms188792.aspx>

### QUESTION 26

Your database contains two tables named DomesticSalesOrders and InternationalSalesOrders. Both tables contain more than 100 million rows. Each table has a Primary Key column named SalesOrderId. The data in the two tables is distinct from one another. Business users want a report that includes aggregate information about the total number of global sales and total sales amounts. You need to ensure that your query executes in the minimum possible time. Which query should you use?

- A. 

```
SELECT COUNT(*) AS NumberOfSales, SUM(SalesAmount) AS TotalSalesAmount
FROM (
 SELECT SalesOrderId, SalesAmount
 FROM DomesticSalesOrders
 UNION ALL
 SELECT SalesOrderId, SalesAmount
 FROM InternationalSalesOrders
) AS p
```
- B. 

```
SELECT COUNT(*) AS NumberOfSales, SUM(SalesAmount) AS TotalSalesAmount
FROM (
 SELECT SalesOrderId, SalesAmount
 FROM DomesticSalesOrders
 UNION
 SELECT SalesOrderId, SalesAmount
 FROM InternationalSalesOrders
) AS p
```
- C. 

```
SELECT COUNT(*) AS NumberOfSales, SUM(SalesAmount) AS TotalSalesAmount
FROM DomesticSalesOrders
UNION
SELECT COUNT(*) AS NumberOfSales, SUM(SalesAmount) AS TotalSalesAmount
FROM InternationalSalesOrders
```
- D. 

```
SELECT COUNT(*) AS NumberOfSales, SUM(SalesAmount) AS TotalSalesAmount
FROM DomesticSalesOrders
UNION ALL
SELECT COUNT(*) AS NumberOfSales, SUM(SalesAmount) AS TotalSalesAmount
FROM InternationalSalesOrders
```

**Correct Answer:** A

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms180026.aspx>

Reference: <http://blog.sqlauthority.com/2009/03/11/sql-server-difference-between-union-vs-union-all-optimal-performance-comparison/>

### QUESTION 27

You are a database developer at an independent software vendor. You create stored procedures that contain proprietary code. You need to protect the code from being viewed by your customers. Which stored procedure option should you use?

- A. ENCRYPTBYKEY
- B. ENCRYPTION
- C. ENCRYPTBYPASSPHRASE
- D. ENCRYPTBYCERT

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://technet.microsoft.com/en-us/library/bb510663.aspx>

Reference: <http://technet.microsoft.com/en-us/library/ms174361.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ms187926.aspx>

Reference: <http://technet.microsoft.com/en-us/library/ms190357.aspx>

Reference: <http://technet.microsoft.com/en-us/library/ms188061.aspx>

### QUESTION 28

You use a Microsoft SQL Server 2012 database. You want to create a table to store Microsoft Word documents. You need to ensure that the documents must only be accessible via Transact-SQL queries. Which Transact-SQL statement should you use?

- A. 

```
CREATE TABLE DocumentStore
(
 [Id] INT NOT NULL PRIMARY KEY,
 [Document] VARBINARY(MAX) NULL
)
GO
```
- B. 

```
CREATE TABLE DocumentStore
(
 [Id] hierarchyid,
 [Document] NVARCHAR NOT NULL
)
GO
```
- C. 

```
CREATE TABLE DocumentStore AS FileTable
```
- D. 

```
CREATE TABLE DocumentStore
(
 [Id] [uniqueidentifier] ROWGUIDCOL NOT NULL UNIQUE,
 [Document] VARBINARY(MAX) FILESTREAM NULL
)
GO
```

**Correct Answer: A**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/gg471497.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ff929144.aspx>

## Exam B

### QUESTION 1

You develop a Microsoft SQL Server 2012 database that contains a heap named OrdersHistorical. You write the following Transact-SQL query:

```
INSERT INTO OrdersHistorical
SELECT * FROM CompletedOrders
```

You need to optimize transaction logging and locking for the statement. Which table hint should you use?

- A. HOLDLOCK
- B. ROWLOCK
- C. XLOCK
- D. UPDLOCK
- E. TABLOCK

**Correct Answer: E**

**Section: (none)**

**Explanation**

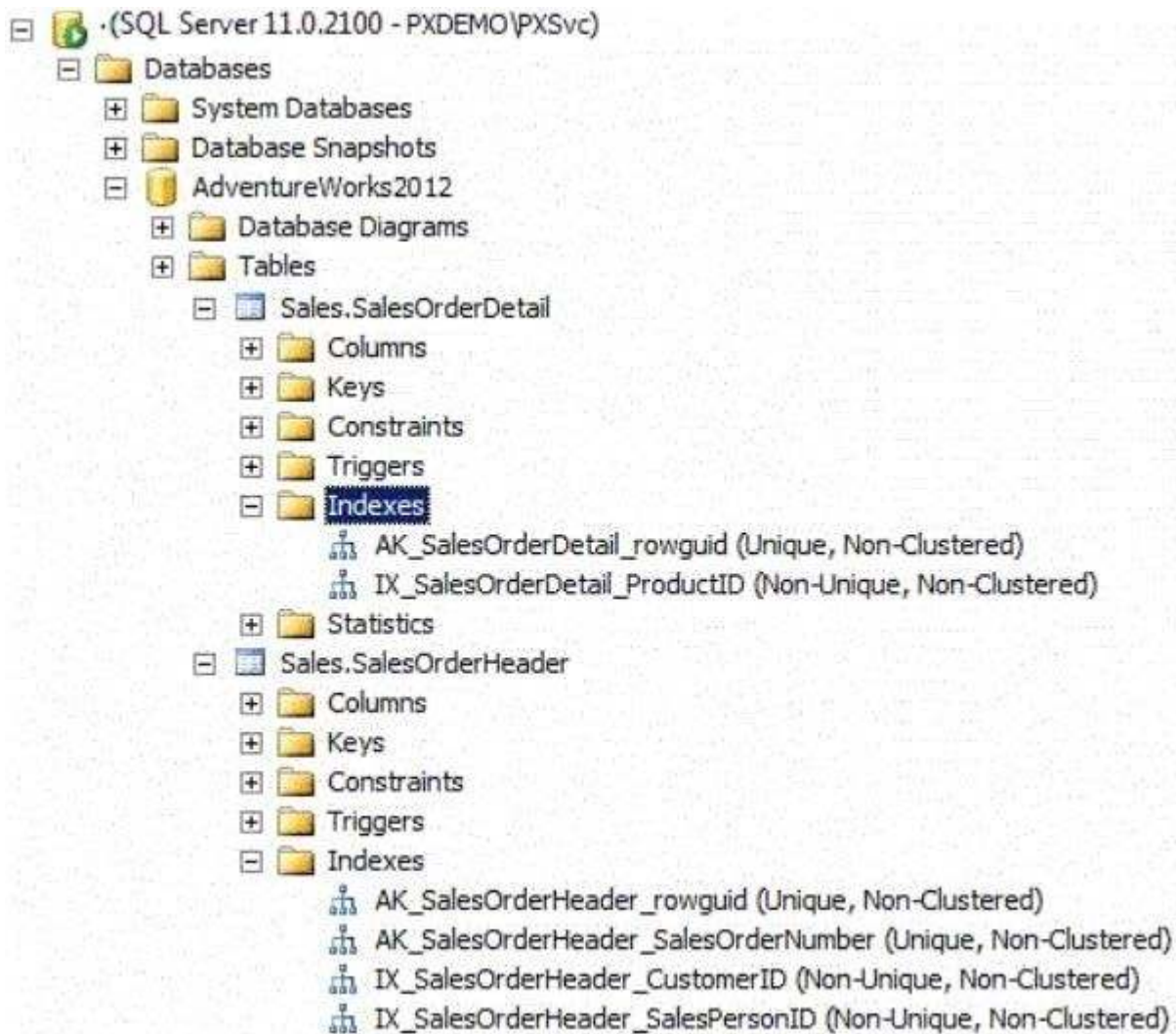
**Explanation/Reference:**

Reference: <http://technet.microsoft.com/en-us/library/ms189857.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ms187373.aspx>

### QUESTION 2

You use a Microsoft SQL Server 2012 database that contains two tables named SalesOrderHeader and SalesOrderDetail. The indexes on the tables are as shown in the exhibit. (Click the Exhibit button.)



You write the following Transact-SQL query:

```
SELECT h.SalesOrderID, h.TotalDue, d.OrderQty
FROM Sales.SalesOrderHeader AS h
 INNER JOIN Sales.SalesOrderDetail AS d
 ON h.SalesOrderID = d.SalesOrderID
WHERE h.TotalDue > 100
AND (d.OrderQty > 5 OR d.LineTotal < 1000.00);
```

You discover that the performance of the query is slow. Analysis of the query plan shows table scans where the estimated rows do not match the actual rows for SalesOrderHeader by using an unexpected index on SalesOrderDetail. You need to improve the performance of the query. What should you do?

- A. Use a FORCESCAN hint in the query.
- B. Add a clustered index on SalesOrderID in SalesOrderHeader.
- C. Use a FORCESEEK hint in the query.
- D. Update statistics on SalesOrderID on both tables.

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

References: <http://msdn.microsoft.com/en-us/library/ms187348.aspx>

### QUESTION 3

Your database contains a table named Purchases. The table includes a DATETIME column named PurchaseTime that stores the date and time each purchase is made. There is a non-clustered index on the PurchaseTime column. The business team wants a report that displays the total number of purchases made on the current day. You need to write a query that will return the correct results in the most efficient manner. Which Transact-SQL query should you use?

- A. 

```
SELECT COUNT(*)
FROM Purchases
WHERE PurchaseTime = CONVERT(DATE, GETDATE())
```
- B. 

```
SELECT COUNT(*)
FROM Purchases
WHERE PurchaseTime = GETDATE()
```
- C. 

```
SELECT COUNT(*)
FROM Purchases
WHERE CONVERT(VARCHAR, PurchaseTime, 112) = CONVERT(VARCHAR, GETDATE(), 112)
```
- D. 

```
SELECT COUNT(*)
FROM Purchases
WHERE PurchaseTime >= CONVERT(DATE, GETDATE())
AND PurchaseTime < DATEADD(DAY, 1, CONVERT(DATE, GETDATE()))
```

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Two answers will return the correct results (the "WHERE CONVERT..." and "WHERE ... AND ..." answers). The correct answer for Microsoft would be the answer that is most "efficient". Anybody have a clue as to which is most efficient? In the execution plan, the one that I've selected as the correct answer is the query with the shortest duration. Also, the query answer with "WHERE CONVERT..." threw warnings in the execution plan...something about affecting CardinalityEstimate and SeekPlan.

I also found this article, which leads me to believe that I have the correct answer:

<http://technet.microsoft.com/en-us/library/ms181034.aspx>

### QUESTION 4

You develop a database for a travel application. You need to design tables and other database objects. You need to store media files in several tables. Each media file is less than 1 MB in size. The media files will require fast access and will be retrieved frequently. What should you do?

- A. Use the CAST function.
- B. Use the DATE data type.
- C. Use the FORMAT function.
- D. Use an appropriate collation.
- E. Use a user-defined table type.
- F. Use the VARBINARY data type.
- G. Use the DATETIME data type.
- H. Use the DATETIME2 data type.
- I. Use the DATETIMEOFFSET data type.
- J. Use the TODATETIMEOFFSET function.

**Correct Answer:** F  
**Section:** (none)  
**Explanation**

**Explanation/Reference:**  
Reference: <http://msdn.microsoft.com/en-us/library/ms188362.aspx>

#### QUESTION 5

You are a database developer of a Microsoft SQL Server 2012 database. You are designing a table that will store Customer data from different sources. The table will include a column that contains the CustomerID from the source system and a column that contains the SourceID. A sample of this data is as shown in the following table.

| SourceID | CustomerID | Customer Name |
|----------|------------|---------------|
| 1        | 234        | John Smith    |
| 3        | 7345       | Jason Warren  |
| 3        | 4402       | Susan Burk    |
| 2        | 866        | Michael Allen |

You need to ensure that the table has no duplicate CustomerID within a SourceID. You also need to ensure that the data in the table is in the order of SourceID and then CustomerID. Which Transact- SQL statement should you use?

- A. 

```
CREATE TABLE Customer
(SourceID int NOT NULL IDENTITY,
CustomerID int NOT NULL IDENTITY,
CustomerName varchar(255) NOT NULL);
```
- B. 

```
CREATE TABLE Customer
(SourceID int NOT NULL,
CustomerID int NOT NULL PRIMARY KEY CLUSTERED,
CustomerName varchar(255) NOT NULL);
```
- C. 

```
CREATE TABLE Customer
(SourceID int NOT NULL PRIMARY KEY CLUSTERED,
CustomerID int NOT NULL UNIQUE,
CustomerName varchar(255) NOT NULL);
```
- D. 

```
CREATE TABLE Customer
(SourceID int NOT NULL,
CustomerID int NOT NULL,
CustomerName varchar(255) NOT NULL,
CONSTRAINT PK_Customer PRIMARY KEY CLUSTERED
(SourceID, CustomerID));
```

**Correct Answer:** D  
**Section:** (none)  
**Explanation**

**Explanation/Reference:**  
Verified the answer as correct.

#### QUESTION 6

You have three tables that contain data for vendors, customers, and agents. You create a view that is used to look up telephone numbers for these companies. The view has the following definition:

```
Create view apt.vwCompanyPhoneList
(Source, CompanyID, CompanyNumber,
 LastName, FirstName, BusinessName, Phone)
as

SELECT 'Customer' as Source
 , CustomerID
 , CustomerNumber
 , CustomerLastName
 , CustomerFirstName
 , CustomerBusinessName
 , Phone
FROM apt.Customer
UNION ALL
SELECT 'Agent' as Source
 , AgentID
 , AgentNumber
 , AgentLastName
 , AgentFirstName
 , AgentBusinessName
 , Phone
FROM apt.Agent
UNION ALL
SELECT 'Vendor' as Source
 , VendorID
 , VendorNumber
 , VendorLastName
 , VendorFirstName
 , VendorBusinessName
 , Phone
FROM apt.Vendor
GO
```

You need to ensure that users can update only the phone numbers by using this view. What should you do?

- A. Alter the view. Use the EXPAND VIEWS query hint along with each SELECT statement.
- B. Drop the view. Re-create the view by using the SCHEMABINDING clause, and then create an index on the view.
- C. Create an AFTER UPDATE trigger on the view.
- D. Create an INSTEAD OF UPDATE trigger on the view.

**Correct Answer: D**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Reference: <http://msdn.microsoft.com/en-us/library/ms187956.aspx>

**QUESTION 7**

You develop a Microsoft SQL Server 2012 database that contains a table named Products. The Products table has the following definition:

```
CREATE TABLE [dbo].[Products] (
 [ProductId] [bigint] NOT NULL,
 [RetailPrice] [nvarchar](25) NOT NULL,
 [WholeSalePrice] [nvarchar](25) NULL,
 [Name] [nvarchar](50) NOT NULL,
 [Category] [nvarchar](25) NOT NULL,
 CONSTRAINT [PK_Products] PRIMARY KEY CLUSTERED
 (
 [ProductId] ASC
) ON [PRIMARY]
) ON [PRIMARY]
```

You need to create an audit record only when either the RetailPrice or WholeSalePrice column is updated. Which Transact-SQL query should you use?

- A. CREATE TRIGGER TrgPriceChange ON Products FOR UPDATE AS  
IF COLUMNS\_CHANGED(RetailPrice, WholesalePrice)  
-- Create Audit Records
- B. CREATE TRIGGER TrgPriceChange ON Products FOR UPDATE AS  
IF EXISTS(SELECT RetailPrice from inserted) OR  
EXISTS (SELECT WholeSalePnce FROM inserted)  
-- Create Audit Records
- C. CREATE TRIGGER TrgPriceChange ON Products FOR UPDATE AS  
IF COLUMNS\_UPDATED(RetailPrice, WholesalePrice)  
-- Create Audit Records
- D. CREATE TRIGGER TrgPriceChange ON Products FOR UPDATE AS  
IF UPDATE(RetailPrice) OR UPDATE(WholeSalePrice)  
-- Create Audit Records

**Correct Answer: D**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/bb510663.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ms186329.aspx>

**QUESTION 8**

You develop a Microsoft SQL Server 2012 server database that supports an application. The application contains a table that has the following definition:

```
CREATE TABLE Inventory
(ItemID int NOT NULL PRIMARY KEY,
ItemsInStore int NOT NULL,
ItemsInWarehouse int NOT NULL)
```



You need to create a computed column that returns the sum total of the ItemsInStore and ItemsInWarehouse values for each row. Which Transact-SQL statement should you use?

- A. ALTER TABLE Inventory  
ADD TotalItems AS ItemsInStore + ItemsInWarehouse
- B. ALTER TABLE Inventory  
ADD ItemsInStore - ItemsInWarehouse = TotalItems
- C. ALTER TABLE Inventory  
ADD TotalItems = ItemsInStore + ItemsInWarehouse
- D. ALTER TABLE Inventory  
ADD TotalItems AS SUM(ItemsInStore, ItemsInWarehouse);

**Correct Answer: A**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://technet.microsoft.com/en-us/library/ms190273.aspx>

#### QUESTION 9

You develop a Microsoft SQL Server 2012 database. You create a view from the Orders and OrderDetails tables by using the following definition.

```
CREATE VIEW vOrders
WITH SCHEMABINDING
AS
SELECT o.ProductID,
 o.OrderDate,
 SUM(od.UnitPrice * od.OrderQty) AS Amount
FROM OrderDetails AS od INNER JOIN
 Orders AS o ON od.OrderID = o.OrderID
WHERE od.SalesOrderID = o.SalesOrderID
GROUP BY o.OrderDate, o.ProductID
GO
```

You need to improve the performance of the view by persisting data to disk. What should you do?

- A. Create an INSTEAD OF trigger on the view.
- B. Create an AFTER trigger on the view.
- C. Modify the view to use the WITH VIEW\_METADATA clause.
- D. Create a clustered index on the view.

**Correct Answer: D**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms188783.aspx>

#### QUESTION 10

Your database contains tables named Products and ProductsPriceLog. The Products table contains columns named ProductCode and Price. The ProductsPriceLog table contains columns named ProductCode, OldPrice, and NewPrice. The ProductsPriceLog table stores the previous price in the OldPrice column and the new price in the NewPrice column. You need to increase the values in the Price column of all products in the Products table by 5 percent. You also need to log the changes to the ProductsPriceLog table. Which Transact-SQL query should you use?

- A. 

```
UPDATE Products SET Price = Price * 1.05
OUTPUT inserted.ProductCode, deleted.Price, inserted.Price
INTO ProductsPriceLog(ProductCode, OldPrice, NewPrice)
```
- B. 

```
UPDATE Products SET Price = Price * 1.05
OUTPUT inserted.ProductCode, inserted.Price, deleted.Price
INTO ProductsPriceLog(ProductCode, OldPrice, NewPrice)
```
- C. 

```
UPDATE Products SET Price = Price * 1.05
OUTPUT inserted.ProductCode, deleted.Price, inserted.Price *
INTO ProductsPriceLog(ProductCode, OldPrice, NewPrice)
```
- D. 

```
UPDATE Products SET Price = Price * 1.05
INSERT INTO ProductsPriceLog (ProductCode, CldPnce, NewPrice;
SELECT ProductCode, Price, Price * 1.05 FROM Products
```

**Correct Answer: A**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms177564.aspx>

#### QUESTION 11

A table named Profits stores the total profit made each year within a territory. The Profits table has columns named Territory, Year, and Profit. You need to create a report that displays the profits made by each territory for each year and its previous year. Which Transact-SQL query should you use?

- A. 

```
SELECT Territory, Year, Profit,
LEAD(Profit, 1, 0) OVER (PARTITION BY Territory ORDER BY Year) AS PrevProfit
FROM Profits
```
- B. 

```
SELECT Territory, Year, Profit,
LAG(Profit, 1, 0) OVER (PARTITION BY Year ORDER BY Territory) AS PrevProfit
FROM Profits
```
- C. 

```
SELECT Territory, Year, Profit,
LAG(Profit, 1, 0) OVER (PARTITION BY Territory ORDER BY Year) AS PrevProfit
FROM Profits
```
- D. 

```
SELECT Territory, Year, Profit,
LEAD(Profit, 1, 0) OVER (PARTITION BY Year ORDER BY Territory) AS PrevProfit
FROM Profits
```

**Correct Answer: C**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/hh231256.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/hh213125.aspx>

#### QUESTION 12

You administer a Microsoft SQL Server database that supports a shopping application. You need to retrieve a

list of customers who live in territories that do not have a sales person. Which Transact- SQL query or queries should you use? (Each correct answer presents a complete solution. Choose all that apply.)

- A. `SELECT CustomerID FROM Customer  
WHERE TerritoryID <> SOME (SELECT TerritoryID FROM Salesperson)`
- B. `SELECT CustomerID FROM Customer  
WHERE TerritoryID <> ALL (SELECT TerritoryID FROM Salesperson)`
- C. `SELECT CustomerID FROM Customer  
WHERE TerritoryID <> ANY (SELECT TerritoryID FROM Salesperson)`
- D. `SELECT CustomerID FROM Customer  
WHERE TerritoryID NOT IN (SELECT TerritoryID FROM Salesperson)`

**Correct Answer:** BD

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

### QUESTION 13

Your database contains a table named SalesOrders. The table includes a DATETIME column named OrderTime that stores the date and time each order is placed. There is a non-clustered index on the OrderTime column. The business team wants a report that displays the total number of orders placed on the current day. You need to write a query that will return the correct results in the most efficient manner. Which Transact-SQL query should you use?

- A. `SELECT COUNT(*) FROM SalesOrders  
WHERE OrderTime = CONVERT (DATE, GETDATE ())`
- B. `SELECT COUNT(*) FROM SalesOrders  
WHERE OrderTime = GETDATE ()`
- C. `SELECT COUNT(*) FROM SalesOrders  
WHERE CONVERT (VARCHAR, OrderTime, 112) = CONVERT (VARCHAR, GETDATE (I, 112))`
- D. `SELECT COUNT(*) FROM SalesOrders  
WHERE OrderTime >= CONVERT (DATE, GETDATE ())  
AND OrderTime < DATEADD (DAY, CONVERT (DATE, GETDATE ()))`

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Verified answer as correct.

### QUESTION 14

Your application contains a stored procedure for each country. Each stored procedure accepts an employee identification number through the @EmpID parameter. You plan to build a single process for each employee that will execute the stored procedure based on the country of residence. Which approach should you use?

- A. a recursive stored procedure
- B. Trigger
- C. An UPDATE statement that includes CASE
- D. Cursor
- E. The foreach SQLCLR statement

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

**QUESTION 15**

You are developing a database application by using Microsoft SQL Server 2012. An application that uses a database begins to run slowly. You discover that during reads, the transaction experiences blocking from concurrent updates. You need to ensure that throughout the transaction the data maintains the original version. What should you do?

- A. Add a HASH hint to the query.
- B. Add a LOOP hint to the query.
- C. Add a FORCESEEK hint to the query.
- D. Add an INCLUDE clause to the index.
- E. Add a FORCESCAN hint to the Attach query.
- F. Add a columnstore index to cover the query.
- G. Enable the optimize for ad hoc workloads option.
- H. Cover the unique clustered index with a columnstore index.
- I. Include a SET FORCEPLAN ON statement before you run the query.
- J. Include a SET STATISTICS PROFILE ON statement before you run the query.
- K. Include a SET STATISTICS SHOWPLAN\_XML ON statement before you run the query.
- L. Include a SET TRANSACTION ISOLATION LEVEL REPEATABLE READ statement before you run the query.
- M. Include a SET TRANSACTION ISOLATION LEVEL SNAPSHOT statement before you run the query.
- N. Include a SET TRANSACTION ISOLATION LEVEL SERIALIZABLE statement before you run the query.

**Correct Answer:** M

**Section:** (none)

**Explanation**

**Explanation/Reference:**

**QUESTION 16**

You are developing a database application by using Microsoft SQL Server 2012. You have a query that runs slower than expected. You need to capture execution plans that will include detailed information on missing indexes recommended by the query optimizer. What should you do?

- A. Add a HASH hint to the query.
- B. Add a LOOP hint to the query.
- C. Add a FORCESEEK hint to the query.
- D. Add an INCLUDE clause to the index.
- E. Add a FORCESCAN hint to the Attach query.
- F. Add a columnstore index to cover the query.
- G. Enable the optimize for ad hoc workloads option.
- H. Cover the unique clustered index with a columnstore index.
- I. Include a SET FORCEPLAN ON statement before you run the query.
- J. Include a SET STATISTICS PROFILE ON statement before you run the query.
- K. Include a SET STATISTICS SHOWPLAN\_XML ON statement before you run the query.
- L. Include a SET TRANSACTION ISOLATION LEVEL REPEATABLE READ statement before you run the query.
- M. Include a SET TRANSACTION ISOLATION LEVEL SNAPSHOT statement before you run the query.

N. Include a SET TRANSACTION ISOLATION LEVEL SERIALIZABLE statement before you run the query.

**Correct Answer:** K

**Section:** (none)

**Explanation**

**Explanation/Reference:**

#### QUESTION 17

You are developing a database application by using Microsoft SQL Server 2012. An application that uses a database begins to run slowly. You discover that a large amount of memory is consumed by single-use dynamic queries. You need to reduce procedure cache usage from these statements without creating any additional indexes. What should you do?

- A. Add a HASH hint to the query.
- B. Add a LOOP hint to the query.
- C. Add a FORCESEEK hint to the query.
- D. Add an INCLUDE clause to the index.
- E. Add a FORCESCAN hint to the Attach query.
- F. Add a columnstore index to cover the query.
- G. Enable the optimize for ad hoc workloads option.
- H. Cover the unique clustered index with a columnstore index.
- I. Include a SET FORCEPLAN ON statement before you run the query.
- J. Include a SET STATISTICS PROFILE ON statement before you run the query.
- K. Include a SET STATISTICS SHOWPLAN\_XML ON statement before you run the query.
- L. Include a SET TRANSACTION ISOLATION LEVEL REPEATABLE READ statement before you run the query.
- M. Include a SET TRANSACTION ISOLATION LEVEL SNAPSHOT statement before you run the query.
- N. Include a SET TRANSACTION ISOLATION LEVEL SERIALIZABLE statement before you run the query.

**Correct Answer:** G

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Reference: <http://msdn.microsoft.com/en-us/library/cc645587.aspx>

#### QUESTION 18

You are developing a database application by using Microsoft SQL Server 2012. An application that uses a database begins to run slowly. Your investigation shows the root cause is a query against a read-only table that has a clustered index. The query returns the following six columns:

- One column in its WHERE clause contained in a non-clustered index
  - Four additional columns
  - One COUNT (\*) column based on a grouping of the four additional columns
- You need to optimize the statement.

What should you do?

- A. Add a HASH hint to the query.
- B. Add a LOOP hint to the query.
- C. Add a FORCESEEK hint to the query.
- D. Add an INCLUDE clause to the index.
- E. Add a FORCESCAN hint to the Attach query.
- F. Add a columnstore index to cover the query.
- G. Enable the optimize for ad hoc workloads option.

- H. Cover the unique clustered index with a columnstore index.
- I. Include a SET FORCEPLAN ON statement before you run the query.
- J. Include a SET STATISTICS PROFILE ON statement before you run the query.
- K. Include a SET STATISTICS SHOWPLAN\_XML ON statement before you run the query.
- L. Include a SET TRANSACTION ISOLATION LEVEL REPEATABLE READ statement before you run the query.
- M. Include a SET TRANSACTION ISOLATION LEVEL SNAPSHOT statement before you run the query.
- N. Include a SET TRANSACTION ISOLATION LEVEL SERIALIZABLE statement before you run the query.

**Correct Answer:** F

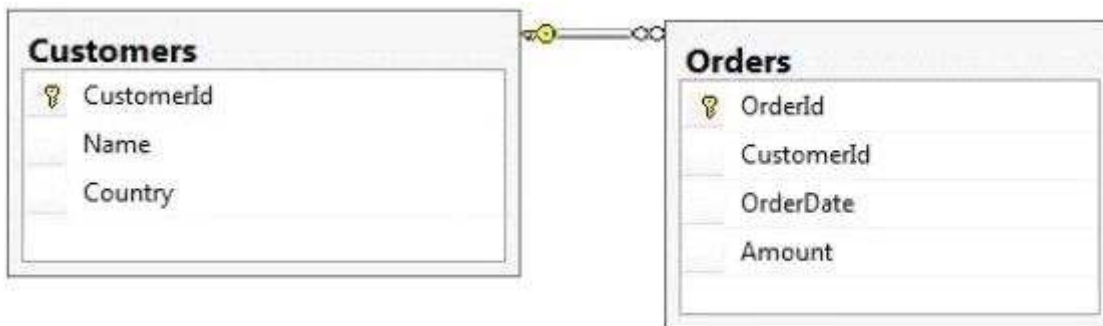
**Section:** (none)

**Explanation**

**Explanation/Reference:**

### QUESTION 19

You administer a Microsoft SQL Server 2012 database named ContosoDb. Tables are defined as shown in the exhibit. (Click the Exhibit button.)



You need to display rows from the Orders table for the Customers row having the CustomerId value set to 1 in the following XML format.

```

<Customers>
 <Name>Customer A</Name>
 <Country>Australia</Country>
 <Orders>
 <OrderId>1</OrderId>
 <OrderDate>2000-01-01T00:00:00</OrderDate>
 <Amount>3400.00</Amount>
 </Orders>
 <Orders>
 <OrderId>2</OrderId>
 <OrderDate>2001-01-01T00:00:00</OrderDate>
 <Amount>4300.00</Amount>
 </Orders>
</Customers>

```

Which Transact-SQL query should you use?

- A. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN

- ```
Customers ON Orders.CustomerId = Customers.CustomerId WHERE
Customers.CustomerId = 1
FOR XML RAW
```
- B. `SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML RAW, ELEMENTS`
- C. `SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML AUTO`
- D. `SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML AUTO, ELEMENTS`
- E. `SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML AUTO`
- F. `SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML AUTO, ELEMENTS`
- G. `SELECT Name AS '@Name', Country AS '@Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML PATH ('Customers')`
- H. `SELECT Name AS 'Customers/Name', Country AS 'Customers/Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML PATH ('Customers')`

Correct Answer: F

Section: (none)

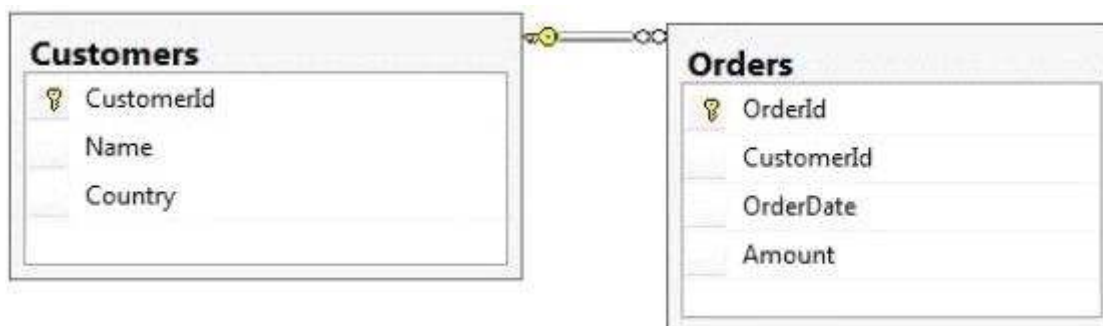
Explanation

Explanation/Reference:

Verified answer as correct.

QUESTION 20

You administer a Microsoft SQL Server 2012 database named ContosoDb. Tables are defined as shown in the exhibit. (Click the Exhibit button.)



You need to display rows from the Orders table for the Customers row having the CustomerId value set to 1 in the following XML format.

```

<Customers Name="Customer A" Country="Australia">
  <OrderId>1</OrderId>
  <OrderDate>2000-01-01T00:00:00</OrderDate>
  <Amount>3400.00</Amount>
</Customers>
<Customers Name="Customer A" Country="Australia">
  <OrderId>2</OrderId>
  <OrderDate>2001-01-01T00:00:00</OrderDate>
  <Amount>4300.00</Amount>
</Customers>

```

Which Transact-SQL query should you use?

- A. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1
FOR XML RAW
- B. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1
FOR XML RAW, ELEMENTS
- C. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1
FOR XML AUTO
- D. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId= 1
FOR XML AUTO, ELEMENTS
- E. SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1
FOR XML AUTO
- F. SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1
FOR XML AUTO, ELEMENTS
- G. SELECT Name AS '@Name', Country AS '@Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1
FOR XML PATH ('Customers')
- H. SELECT Name AS 'Customers/Name', Country AS 'Customers/Country', OrderId, OrderDate, Amount FROM Orders
INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1
FOR XML PATH ('Customers')

Correct Answer: G

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

QUESTION 21

You use Microsoft SQL Server 2012 to write code for a transaction that contains several statements. There is high contention between readers and writers on several tables used by your transaction. You need to minimize the use of the tempdb space. You also need to prevent reading queries from blocking writing queries. Which isolation level should you use?

- A. SERIALIZABLE
- B. SNAPSHOT
- C. READ COMMITTED SNAPSHOT
- D. REPEATABLE READ

Correct Answer: C

Section: (none)

Explanation

Explanation/Reference:

Reference: <http://msdn.microsoft.com/en-us/library/ms173763.aspx>

QUESTION 22

You create a table that has the StudentCode, SubjectCode, and Marks columns to record mid-year marks for students. The table has marks obtained by 50 students for various subjects. You need to ensure that the following requirements are met:

- Students must be ranked based on their average marks.
- If one or more students have the same average, the same rank must be given to these students.
- Consecutive ranks must be skipped when the same rank is assigned.

Which Transact-SQL query should you use?

- A.

```
SELECT StudentCode as Code,
       RANK() OVER(ORDER BY AVG (Marks) DESC) AS Value
FROM StudentMarks
GROUP BY StudentCode
```
- B.

```
SELECT Id, Name, Marks,
       DENSE_RANK() OVER(ORDER BY Marks DESC) AS Rank
FROM StudentMarks
```
- C.

```
SELECT StudentCode as Code,
       DENSE_RANK() OVER(ORDER BY AVG (Marks) DESC) AS Value
FROM StudentMarks
GROUP BY StudentCode
```
- D.

```
SELECT StudentCode as Code,
       NTILE(2) OVER(ORDER BY AVG (Marks) DESC) AS Value
FROM StudentMarks
GROUP BY StudentCode
```
- E.

```
SELECT StudentCode AS Code,Marks AS Value FROM (
    SELECT StudentCode, Marks AS Marks,
           RANK() OVER(PARTITION BY SubjectCode ORDER BY Marks ASC) AS Rank
    FROM StudentMarks) tmp
WHERE Rank = 1
```
- F.

```
SELECT StudentCode AS Code,Marks AS Value FROM (
    SELECT StudentCode, Marks AS Marks,
           RANK() OVER(PARTITION BY SubjectCode ORDER BY Marks DESC) AS Rank
    FROM StudentMarks) tmp
WHERE Rank = 1
```
- G.

```
SELECT StudentCode AS Code,Marks AS Value FROM (
    SELECT StudentCode, Marks AS Marks,
           RANK() OVER(PARTITION BY StudentCode ORDER BY Marks ASC) AS Rank
    FROM StudentMarks) tmp
WHERE Rank = 1
```
- H.

```
SELECT StudentCode AS Code,Marks AS Value FROM (
```

```

SELECT StudentCode, Marks AS Marks,
       RANK() OVER(PARTITION BY StudentCode ORDER BY Marks DESC) AS Rank
FROM StudentMarks) tmp
WHERE Rank = 1

```

Correct Answer: A

Section: (none)

Explanation

Explanation/Reference:

Reference: <http://msdn.microsoft.com/en-us/library/ms189798.aspx>

QUESTION 23

You create a table that has the StudentCode, SubjectCode, and Marks columns to record mid-year marks for students. The table has marks obtained by 50 students for various subjects. You need to retrieve the students who scored the highest marks for each subject along with the marks. Which Transact-SQL query should you use?

- A.

```
SELECT StudentCode as Code, RANK() OVER(ORDER BY AVG(Marks) DESC) AS Value
FROM StudentMarks
GROUP BY StudentCode
```
- B.

```
SELECT Id, Name, Marks, DENSE_RANK() OVER(ORDER BY Marks DESC) AS Rank
FROM StudentMarks
```
- C.

```
SELECT StudentCode as Code, DENSE_RANK() OVER(ORDER BY AVG(Marks) DESC) AS Value
FROM StudentMarks
GROUP BY StudentCode
```
- D.

```
SELECT StudentCode as Code, NTILE(2) OVER(ORDER BY AVG(Marks) DESC) AS Value
FROM StudentMarks
GROUP BY StudentCode
```
- E.

```
SELECT StudentCode AS Code, Marks AS Value FROM (
    SELECT StudentCode, Marks AS Marks,
           RANK() OVER(PARTITION BY SubjectCode ORDER BY Marks ASC) AS Rank
    FROM StudentMarks) tmp
WHERE Rank = 1
```
- F.

```
SELECT StudentCode AS Code, Marks AS Value FROM (
    SELECT StudentCode, Marks AS Marks,
           RANK() OVER(PARTITION BY SubjectCode ORDER BY Marks DESC) AS Rank
    FROM StudentMarks) tmp
WHERE Rank = 1
```
- G.

```
SELECT StudentCode AS Code, Marks AS Value FROM (
    SELECT StudentCode, Marks AS Marks,
           RANK() OVER(PARTITION BY StudentCode ORDER BY Marks ASC) AS Rank
    FROM StudentMarks) tmp
WHERE Rank = 1
```
- H.

```
SELECT StudentCode AS Code, Marks AS Value FROM (
    SELECT StudentCode, Marks AS Marks,
           RANK() OVER(PARTITION BY StudentCode ORDER BY Marks DESC) AS Rank
    FROM StudentMarks) tmp
WHERE Rank = 1
```

Correct Answer: F

Section: (none)

Explanation

Explanation/Reference:

QUESTION 24

You develop a database for a travel application. You need to design tables and other database objects. You create the Airline_Schedules table. You need to store the departure and arrival dates and times of flights along with time zone information. What should you do?

- A. Use the CAST function.
- B. Use the DATE data type.
- C. Use the FORMAT function.
- D. Use an appropriate collation.
- E. Use a user-defined table type.
- F. Use the VARBINARY data type.
- G. Use the DATETIME data type.
- H. Use the DATETIME2 data type.
- I. Use the DATETIMEOFFSET data type.
- J. Use the TODATETIMEOFFSET function.

Correct Answer: I

Section: (none)

Explanation

Explanation/Reference:

Reference: <http://msdn.microsoft.com/en-us/library/ff848733.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/bb630289.aspx>

QUESTION 25

You develop a database for a travel application. You need to design tables and other database objects. You create a stored procedure. You need to supply the stored procedure with multiple event names and their dates as parameters. What should you do?

- A. Use the CAST function.
- B. Use the DATE data type.
- C. Use the FORMAT function.
- D. Use an appropriate collation.
- E. Use a user-defined table type.
- F. Use the VARBINARY data type.
- G. Use the DATETIME data type.
- H. Use the DATETIME2 data type.
- I. Use the DATETIMEOFFSET data type.
- J. Use the TODATETIMEOFFSET function.

Correct Answer: E

Section: (none)

Explanation

Explanation/Reference:

QUESTION 26

You develop a Microsoft SQL Server 2012 database. The database is used by two web applications that access a table named Products. You want to create an object that will prevent the applications from accessing the table directly while still providing access to the required data. You need to ensure that the following requirements are met:

- Future modifications to the table definition will not affect the applications' ability to access data.
- The new object can accommodate data retrieval and data modification.
- You need to achieve this goal by using the minimum amount of changes to the applications.

What should you create for each application?

- A. Synonyms
- B. Common table expressions
- C. Views
- D. Temporary tables

Correct Answer: C

Section: (none)

Explanation

Explanation/Reference:

Reference: <http://msdn.microsoft.com/en-us/library/ms190174.aspx>

QUESTION 27

You use Microsoft SQL Server 2012 to develop a database application. You need to create an object that meets the following requirements:

- Takes an input variable
- Returns a table of values
- Cannot be referenced within a view

Which object should you use?

- A. Scalar-valued function
- B. Inline function
- C. User-defined data type
- D. Stored procedure

Correct Answer: D

Section: (none)

Explanation

Explanation/Reference:

QUESTION 28

You are a database developer for an application hosted on a Microsoft SQL Server 2012 server. The database contains two tables that have the following definitions:

```
CREATE TABLE Customer
(CustomerID int NOT NULL PRIMARY KEY,
 CustomerName varchar(50) NOT NULL)

CREATE TABLE Orders
(OrderID int NOT NULL PRIMARY KEY,
 CustomerID int NOT NULL FOREIGN KEY REFERENCES Customer (CustomerID),
 OrderAmount money NOT NULL,
 ShippingCountry varchar(50) NOT NULL)
```

Global customers place orders from several countries. You need to view the country from which each customer has placed the most orders. Which Transact-SQL query do you use?

- A. `SELECT c.CustomerID, c.CustomerName, o.ShippingCountry`

```

FROM Customer c
INNER JOIN
  (SELECT CustomerID, ShippingCountry,
   RANK() OVER (PARTITION BY CustomerID
    ORDER BY COUNT(OrderAmount) DESC) AS Rnk
   FROM Orders
   GROUP BY CustomerID, ShippingCountry) AS o
ON c.CustomerID = o.CustomerID
WHERE o.Rnk = 1

```

- B. SELECT c.CustomerID, c.CustomerName, o.ShippingCountry
FROM
 (SELECT c.CustomerID, c.CustomerName, o.ShippingCountry,
 RANK() OVER (PARTITION BY CustomerID
 ORDER BY COUNT(o.OrderAmount) ASC) AS Rnk
 FROM Customer c
 INNER JOIN Orders o
 ON c.CustomerID = o.CustomerID
 GROUP BY c.CustomerID, c.CustomerName, o.ShippingCountry) cs
WHERE Rnk = 1
- C. SELECT c.CustomerID, c.CustomerName, o.ShippingCountry
FROM Customer c
INNER JOIN
 (SELECT CustomerID, ShippingCountry,
 RANK() OVER (PARTITION BY CustomerID
 ORDER BY OrderAmount DESC) AS Rnk
 FROM Orders
 GROUP BY CustomerID, ShippingCountry) AS o
ON c.CustomerID = o.CustomerID
WHERE o.Rnk = 1
- D. SELECT c.CustomerID, c.CustomerName, o.ShippingCountry
FROM Customer c
INNER JOIN
 (SELECT CustomerID, ShippingCountry,
 COUNT(OrderAmount) DESC) AS OrderAmount
 FROM Orders
 GROUP BY CustomerID, ShippingCountry) AS o
ON c.CustomerID = o.CustomerID
ORDER BY OrderAmount DESC

Correct Answer: A

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

QUESTION 29

You want to add a new GUID column named BookGUID to a table named dbo.Book that already contains data. BookGUID will have a constraint to ensure that it always has a value when new rows are inserted into dbo.Book. You need to ensure that the new column is assigned a GUID for existing rows. Which four Transact-SQL statements should you use? (To answer, move the appropriate SQL statements from the list of statements to the answer area and arrange them in the correct order.)

Build List and Reorder:

Ordered List Title	Answer Choices Title
<div> <div>▲</div> <div>▼</div> </div> <div></div>	<pre> newid() newguid() WITH VALUES WITH EXISTING CONSTRAINT CK_BookGuid CHECK CONSTRAINT DF_BookGuid DEFAULT ALTER TABLE dbo.Book ADD BookGuid VARCHAR(10) NOT NULL ALTER TABLE dbo.Book ADD BookGuid Uniqueidentifier NOT NULL </pre>
	<div><< Move</div> <div>Remove >></div>

Correct Answer:

```

ALTER TABLE dbo.Book
ADD BookGuid Uniqueidentifier NOT NULL
CONSTRAINT DF_BookGuid DEFAULT
newid()
WITH VALUES

```

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct. Actually, in the real world, you don't have to use WITH VALUES at the end of the statement and it works just as well. But because the question specifically states which FOUR TSQL statements to use, we have to include it.

Reference: <http://msdn.microsoft.com/en-us/library/ms190273.aspx>

QUESTION 30

You create a view based on the following statement:

```

CREATE VIEW dbo.vwItemList
AS
SELECT
    b.BatchID
    , b.MailItemID
    , c.ContractNum
    , c.FirstName + ' ' + c.LastName as ContractName
    , a.Address1
    , a.City + ', ' + a.State + ' ' + a.Zip
FROM BatchLog b
join Contract c on b.MailItemID = c.ContractID
join Address a on a.ContractID = c.ContractID
WHERE
    b.ProcessDate >= dateadd(d, 1, EOMONTH(GETDATE(), -2));

```

You grant the Select permission to User1 for this view. You need to change the view so that it displays only the records that were processed in the month prior to the current month. You need to ensure that after the changes, the view functions correctly for User1. Which four Transact-SQL statements should you use? (To answer, move the appropriate SQL statements from the list of statements to the answer area and arrange them in the correct order.)

Build List and Reorder:

Ordered List Title	Answer Choices Title
<div><div>▲</div><div>▼</div></div> <div></div>	<pre>DROP VIEW dbo.wvltemList; GO CREATE VIEW dbo.wvltemList AS ALTER VIEW dbo.wvltemList AS WHERE b.ProcessDate >= dateadd(d, 1, EOMONTH (GETDATE(), -2)) AND b.ProcessDate <= EOMONTH(GETDATE(), -1); WHERE b.ProcessDate >= dateadd(d, 1, EOMONTH (GETDATE(), -2)) AND b.ProcessDate < dateadd(d, 1, EOMONTH (GETDATE(), -1)) SELECT b.BatchID , b.MailltemID , c.ContractNum , c.FirstName + ' ' + c.LastName as ContractName , a.Address1 , a.City + ', ' + a.State + ' ' + a.Zip FROM BacthLog b JOIN Contract c ON b.MailltemID = c.ContractID JOIN Address a ON a.ContractID = c.ContractID GO GRANT SELECT ON SCHEMA::wvltemList TO User1;</pre>
	<div><< Move</div> <div>Remove >></div>

Correct Answer:


```
ALTER VIEW dbo.wvltemList
AS
SELECT
    b.BatchID
    , b.MailltemID
    , c.ContractNum
    , c.FirstName + ' ' + c.LastName as
ContractName
    , a.Address1
    , a.City + ' ' + a.State + ' ' + a.Zip
FROM BacthLog b
JOIN Contract c ON b.MailltemID =
c.ContractID
JOIN Address a ON a.ContractID =
c.ContractID
WHERE b.ProcessDate >= dateadd(d, 1,
EOMONTH (GETDATE(), -2))
AND b.ProcessDate < dateadd(d, 1,
EOMONTH (GETDATE(), -1))
```

Section: (none)

Explanation

Explanation/Reference:

Reference: <http://msdn.microsoft.com/en-us/library/hh213020.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ms186819.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ms173846.aspx>

QUESTION 31

You use a Microsoft SQL Server 2012 database. You need to create an indexed view within the database for a report that displays Customer Name and the total revenue for that customer. Which four T-SQL statements should you use? (To answer, move the appropriate SQL statements from the list of statements to the answer area and arrange them in the correct order.)

Build List and Reorder:

Ordered List Title	Answer Choices Title
<div><div>▲</div><div>▼</div></div> <div></div>	<div>CREATE VIEW Sales.vwCustomerRevenue AS WITH SCHEMABINDING CREATE VIEW Sales.vwCustomerRevenue WITH SCHEMABINDING AS SELECT O.CustomerID , C.CustomerName , SUM(O.SubTotal) AS CustomerTotal , COUNT_BIG(*) AS RecCount FROM Sales.SalesOrderHeader AS O JOIN Sales.Customer AS C ON C.CustomerID = O.CustomerID GROUP BY O.CustomerID , C.CustomerName GO CREATE UNIQUE CLUSTERED INDEX idx_vwCustomerRevenue ON Sales.vwCustomerRevenue (CustomerID); GO CREATE UNIQUE INDEX idx_vwCustomerRevenue ON Sales.vwCustomerRevenue (CustomerID);</div>

<< Move

Remove >>

Correct Answer:

```

CREATE VIEW Sales.wwCustomerRevenue
WITH SCHEMABINDING
AS
SELECT
O.CustomerID
, C.CustomerName
, SUM(O.SubTotal) AS CustomerTotal
, COUNT_BIG(*) AS RecCount
FROM Sales.SalesOrderHeader AS O
JOIN Sales.Customer AS C ON C.CustomerID
= O.CustomerID
GROUP BY
O.CustomerID
, C.CustomerName
GO
CREATE UNIQUE CLUSTERED INDEX
idx_wwCustomerRevenue
ON Sales.wwCustomerRevenue (CustomerID);

```

Section: (none)

Explanation

Explanation/Reference:

Reference: <http://msdn.microsoft.com/en-us/library/ms191432.aspx>

QUESTION 32

You administer a Microsoft SQL Server 2012 database. You use an OrderDetail table that has the following definition:

```

CREATE TABLE [dbo].[OrderDetail]
(
    [SalesOrderID] [int] NOT NULL,
    [SalesOrderDetailID] [int] IDENTITY(1,1) NOT NULL,
    [CarrierTrackingNumber] [nvarchar](25) NULL,
    [OrderQty] [smallint] NOT NULL,
    [ProductID] [int] NOT NULL,
    [SpecialOfferID] [int] NULL,
    [UnitPrice] [money] NOT NULL);

```

You need to create a non-clustered index on the SalesOrderID column in the OrderDetail table to include only rows that contain a value in the SpecialOfferID column. Which four Transact-SQL statements should you use? (To answer, move the appropriate statements from the list of statements to the answer area and arrange them in the correct order.)

Build List and Reorder:

Ordered List Title	Answer Choices Title
<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> <div style="border: 1px solid black; width: 20px; height: 20px; margin-bottom: 5px;"></div> <div style="border: 1px solid black; width: 20px; height: 20px;"></div> </div> <div style="border: 1px solid black; width: 280px; height: 180px;"></div> </div>	<div style="border: 1px solid black; padding: 5px;"> WHERE FILTER ON SpecialOfferID IS NOT NULL; ON dbo.OrderDetail(SalesOrderID) ON dbo.OrderDetail(SalesOrderID) AS FILTERED_INDEX CREATE NONCLUSTERED INDEX FIdx_SpecialOfferID CREATE NONCLUSTERED FILTERED INDEX FIdx_SpecialOfferID </div>
<div style="display: flex; justify-content: center; gap: 20px;"> <div style="border: 1px solid black; padding: 2px 10px;"><< Move</div> <div style="border: 1px solid black; padding: 2px 10px;">Remove >></div> </div>	

Correct Answer:

```
CREATE NONCLUSTERED INDEX
FIdx_SpecialOfferID
ON dbo.OrderDetail(SalesOrderID)
WHERE
SpecialOfferID IS NOT NULL;
```

Section: (none)

Explanation

Explanation/Reference:

QUESTION 33

You use Microsoft SQL Server 2012 to develop a database application. You need to implement a computed column that references a lookup table by using an INNER JOIN against another table. What should you do?

- A. Reference a user-defined function within the computed column.
- B. Create a BEFORE trigger that maintains the state of the computed column.
- C. Add a default constraint to the computed column that implements hard-coded values.
- D. Add a default constraint to the computed column that implements hard-coded CASE statements.

Correct Answer: A

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

Exam C

QUESTION 1

You administer a Microsoft SQL Server 2012 database named ContosoDb. The database contains a table named Suppliers and a column named IsActive in the Purchases schema. You create a new user named ContosoUser in ContosoDb. ContosoUser has no permissions to the Suppliers table. You need to ensure that ContosoUser can delete rows that are not active from Suppliers. You also need to grant ContosoUser only the minimum required permissions. Which Transact-SQL statement should you use?

- A. `GRANT DELETE ON Purchases.Suppliers TO ContosoUser`
- B. `CREATE PROCEDURE Purchases.PurgeInactiveSuppliers
WITH EXECUTE AS USER = 'dbo'
AS
DELETE FROM Purchases.Suppliers WHERE IsActive = 0
GO
GRANT EXECUTE ON Purchases.PurgeInactiveSuppliers TO ContosoUser`
- C. `GRANT SELECT ON Purchases.Suppliers TO ContosoUser`
- D. `CREATE PROCEDURE Purchases.PurgeInactiveSuppliers
AS
DELETE FROM Purchases.Suppliers WHERE IsActive = 0
GO
GRANT EXECUTE ON Purchases.PurgeInactiveSuppliers TO ContosoUser`

Correct Answer: B

Section: (none)

Explanation

Explanation/Reference:

Reference: <http://msdn.microsoft.com/en-us/library/ms188354.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ms187926.aspx>

QUESTION 2

You administer a Microsoft SQL Server 2012 database that has multiple tables in the Sales schema. Some users must be prevented from deleting records in any of the tables in the Sales schema. You need to manage users who are prevented from deleting records in the Sales schema. You need to achieve this goal by using the minimum amount of administrative effort. What should you do?

- A. Create a custom database role that includes the users. Deny Delete permissions on the Sales schema for the custom database role.
- B. Include the Sales schema as an owned schema for the db_denydatawriter role. Add the users to the db_denydatawriter role.
- C. Deny Delete permissions on each table in the Sales schema for each user.
- D. Create a custom database role that includes the users. Deny Delete permissions on each table in the Sales schema for the custom database role.

Correct Answer: A

Section: (none)

Explanation

Explanation/Reference:

Looks good.

QUESTION 3

You generate a daily report according to the following query:

```

SELECT c.CustomerName
FROM Sales.Customer c
WHERE Sales.ufnGetLastOrderDate(c.CustomerID) <
    DATEADD(DAY, -90, GETDATE())

```

The Sales.ufnGetLastOrderDate user-defined function (UDF) is defined as follows:

```

CREATE FUNCTION Sales.ufnGetLastOrderDate(@CustomerID int)
RETURNS datetime
AS
BEGIN
    DECLARE @lastOrderDate datetime
    SELECT @lastOrderDate = MAX(OrderDate)
    FROM Sales.SalesOrder
    WHERE CustomerID = @CustomerID
    RETURN @lastOrderDate
END

```

You need to improve the performance of the query. What should you do?

A. Drop the UDF and rewrite the report query as follows:

```

WITH cte(CustomerID, LastOrderDate) AS (
    SELECT CustomerID, MAX(OrderDate) AS [LastOrderDate]
    FROM Sales.SalesOrder
    GROUP BY CustomerID
)
SELECT c.CustomerName
FROM cte
INNER JOIN Sales.Customer c
ON cte.CustomerID = c.CustomerID
WHERE cte.LastOrderDate < DATEADD(DAY, -90, GETDATE())

```

B. Drop the UDF and rewrite the report query as follows:

```

SELECT c.CustomerName
FROM Sales.Customer c
WHERE NOT EXISTS (
    SELECT s.OrderDate
    FROM Sales.SalesOrder
    WHERE s.OrderDate > DATEADD(DAY, -90, GETDATE())
    AND s.CustomerID = c.CustomerID)

```

C. Drop the UDF and rewrite the report query as follows:

```

SELECT DISTINCT c.CustomerName
FROM Sales.Customer c
INNER JOIN Sales.SalesOrder s
ON c.CustomerID = s.CustomerID
WHERE s.OrderDate < DATEADD(DAY, -90, GETDATE())

```

D. Rewrite the report query as follows:

```

SELECT c.CustomerName
FROM Sales.Customer c
WHERE NOT EXISTS (SELECT OrderDate FROM Sales.ufnGetRecentOrders(c.CustomerID,
90))

```

Rewrite the UDF as follows:

```
CREATE FUNCTION Sales.ufnGetRecentOrders(@CustomerID int, @MaxAge datetime)
RETURNS TABLE AS RETURN (
    SELECT OrderDate
    FROM Sales.SalesOrder
    WHERE s.CustomerID = @CustomerID
    AND s.OrderDate > DATEADD(DAY, -@MaxAge, GETDATE())
```

Correct Answer: A

Section: (none)

Explanation

Explanation/Reference:

QUESTION 4

You develop a database application for a university. You need to create a view that will be indexed that meets the following requirements:

- Displays the details of only students from Canada.
- Allows insertion of details of only students from Canada.

Which four Transact-SQL statements should you use? (To answer, move the appropriate SQL statements from the list of statements to the answer area and arrange them in the correct order.)

Build List and Reorder:

Ordered List Title	Answer Choices Title
<div><div>▲▼</div><div></div></div>	<div>WITH ENCRYPTION</div> <div>WITH CHECK OPTION</div> <div>WITH SCHEMABINDING</div> <div>WITH VIEW_METADATA</div> <div>CREATE VIEW</div> <div>dbo.CanadianStudents</div> <div>CREATE INDEXED VIEW</div> <div>dbo.CanadianStudents</div> <div>AS</div> <div>SELECT s.LastName, s.FirstName,</div> <div>s.JobTitle, a.Country,</div> <div>e.LastQualification</div> <div>FROM Student s</div> <div>INNER JOIN NativeAddress a ON</div> <div>a.AddressID = s.AddressID</div> <div>INNER JOIN EducationHistory e ON</div> <div>s.StudentID = e.StudentID</div> <div>WHERE a.Country = 'Canada'</div>
	<div><< Move</div> <div>Remove >></div>

Correct Answer:

```
CREATE VIEW dbo.CanadianStudents
WITH SCHEMABINDING
AS
SELECT s.LastName, s.FirstName, s.JobTitle,
a.Country, e.LastQualification
FROM Student s
INNER JOIN NativeAddress a ON a.AddressID =
s.AddressID
INNER JOIN EducationHistory e ON
s.StudentID = e.StudentID
WHERE a.Country = 'Canada'
WITH CHECK OPTION
```

Section: (none)

Explanation

Explanation/Reference:

Verified answer as correct.

Reference: <http://msdn.microsoft.com/en-us/library/ms187956.aspx>

QUESTION 5

You use Microsoft SQL Server 2012 to develop a database application. You create two tables by using the following table definitions.



```
CREATE TABLE Employees
(
    empid int NOT NULL
    , mgrid int NULL
    , empname varchar(25) NOT NULL
    , salary money NOT NULL
    CONSTRAINT PK_Employees PRIMARY KEY(empid)
);
CREATE TABLE Departments
(
    deptid INT NOT NULL PRIMARY KEY
    , deptname VARCHAR(25) NOT NULL
    , deptmgrid INT NULL REFERENCES Employees(empid)
);
```

You need to write a Transact-SQL statement that will support the following query:

```
SELECT D.deptid, D.deptname, D.deptmgrid
    , ST.empid, ST.empname, ST.mgrid
FROM Departments AS D
    CROSS APPLY getsubtree(D.deptmgrid) AS ST;
```

Which six Transact-SQL statements should you use? (To answer, move the appropriate SQL statements from the list of statements to the answer area and arrange them in the correct order.)

Build List and Reorder:

Ordered List Title	Answer Choices Title
<div data-bbox="232 279 269 352"> </div>	<pre>CREATE FUNCTION dbo.getsubtree(@empid AS INT) RETURNS @Tree TABLE (empid INT NOT NULL, empname VARCHAR(25) NOT NULL, mgrid INT NULL, lv INT NOT NULL) AS BEGIN (SELECT empid, empname, mgrid, 0 FROM Employees WHERE empid = @empid UNION ALL SELECT e.empid, e.empname, e.mgrid, es.lv+1 FROM Employees AS e JOIN Employees_Subtree AS es ON e.mgrid = es.empid) SELECT * FROM Employees_Subtree; CREATE PROCEDURE ebo.getsubtree(@empid AS INT) AS BEGIN RETURN END INSERT INTO @TREE SELECT empid, empname, mgrid, 0 FROM Employees WHERE empid = @empid UNION ALL SELECT e.empid, e.empname, e.mgrid, es.lv+1 FROM Employees AS e JOIN Employees_Subtree AS es ON e.mgrid = es.empid WITH Employees_Subtree(empid, empname, mgrid, lv) AS</pre>

<< Move

Remove >>

Correct Answer:

```

CREATE FUNCTION dbo.getsubtree(@empid
AS INT)
RETURNS @Tree TABLE (
empid INT NOT NULL,
empname VARCHAR(25) NOT NULL,
mgrid INT NULL,
lv INT NOT NULL)
AS
BEGIN
WITH Employees_Subtree(empid, empname,
mgrid, lv)
AS
(SELECT empid, empname, mgrid, 0
FROM Employees WHERE empid = @empid
UNION ALL
SELECT e.empid, e.empname, e.mgrid,
es.lv+1
FROM Employees AS e JOIN
Employees_Subtree AS es ON e.mgrid =
es.empid)
INSERT INTO @Tree
SELECT * FROM Employees_Subtree;
RETURN
END

```

Section: (none)

Explanation

Explanation/Reference:

QUESTION 6

You use Microsoft SQL Server 2012 to develop a database application. You create a table by using the following definition:

```

CREATE TABLE Prices (
    PriceId int IDENTITY(1,1) PRIMARY KEY,
    ActualPrice NUMERIC(16,9),
    PredictedPrice NUMERIC(16,9)
)

```

You need to create a computed column based on a user-defined function named `udf_price_index`. You also need to ensure that the column supports an index. Which three Transact-SQL statements should you use? (To answer, move the appropriate SQL statements from the list of statements to the answer area and arrange them in the correct order.)

Build List and Reorder:

Ordered List Title	Answer Choices Title
<div data-bbox="232 247 267 321"> <input type="button" value="▲"/> <input type="button" value="▼"/> </div>	<pre> CREATE FUNCTION udf_price_index (@actualprice FLOAT, @predictedprice FLOAT) RETURNS FLOAT ALTER TABLE Prices ADD [PriceIndex] AS dbo.udf_price_index([ActualPrice], [PredictedPrice]) PERSISTED ALTER TABLE Prices ADD [PriceIndex] AS dbo.udf_price_index([ActualPrice], [PredictedPrice]) AS BEGIN SELECT @priceindex = CASE WHEN @predictedprice = 0 THEN 0 ELSE @acualprice/@predictedprice END END GO CREATE FUNCTION udf_price_index (@actualprice NUMERIC(16,9), @predictedprice NUMERIC(16,9)) RETURNS NUMERIC(16,9) WITH SCHEMABINDING AS BEGIN DECLARE @priceindex NUMERIC(16,9) SELECT @priceindex = CASE WHEN @predictedprice = 0 THEN 0 ELSE @acualprice/@predictedprice END RETURN @priceindex END GO </pre>
	<div data-bbox="756 915 899 953"><< Move</div> <div data-bbox="756 970 899 1008">Remove >></div>

Correct Answer:

```

CREATE FUNCTION udf_price_index
(@actualprice NUMERIC(16,9),
@predictedprice NUMERIC(16,9))
RETURNS NUMERIC(16,9)
WITH SCHEMABINDING
AS
BEGIN
    DECLARE @priceindex NUMERIC(16,9)
    SELECT @priceindex = CASE
        WHEN @predictedprice = 0 THEN 0
        ELSE @actualprice/@predictedprice
    END
    RETURN @priceindex
END
GO

ALTER TABLE Prices ADD [PriceIndex]
AS dbo.udf_price_index([ActualPrice],
[PredictedPrice]) PERSISTED

```

Section: (none)

Explanation

Explanation/Reference:

QUESTION 7

You use Microsoft SQL Server 2012 to develop a database that has two tables named Div1Cust and Div2Cust. Each table has columns named DivisionID and CustomerId. None of the rows in Div1Cust exist in Div2Cust. You need to write a query that meets the following requirements:

- The rows in Div1Cust must be combined with the rows in Div2Cust.
- The result set must have columns named Division and Customer.
- Duplicates must be retained.

Which three Transact-SQL statements should you use? (To answer, move the appropriate statements from the list of statements to the answer area and arrange them in the correct order.)

Build List and Reorder:

Ordered List Title	Answer Choices Title
<div> <div>▲</div> <div>▼</div> </div> <div></div>	<div>EXCEPT</div> <div>SELECT DivisionID, CustomerID FROM Div2Cust</div> <div>SELECT DISTINCT DivisionID, CustomerID FROM Div1Cust, Div2Cust</div> <div>INTERSECT</div> <div>SELECT DivisionID AS Division, CustomerID AS Customer FROM Div1Cust</div> <div>UNION ALL</div> <div>INNER JOIN</div> <div>UNION</div> <div>SELECT DivisionID, CustomerID FROM Div1Cust, Div2Cust ON Div1Cust.CustID = Div2Cust.CustID</div> <div>SELECT DivisionID, CustomerID FROM Div1Cust</div>
	<div><< Move</div> <div>Remove >></div>

Correct Answer:

```
SELECT DivisionID AS Division, CustomerID AS
Customer
FROM Div1Cust
UNION ALL
SELECT DivisionID, CustomerID
FROM Div2Cust
```

Section: (none)
Explanation

Explanation/Reference:

Reference: <http://msdn.microsoft.com/en-us/library/ms180026.aspx>

Reference: <http://msdn.microsoft.com/en-us/library/ms188055.aspx>

Exam D

QUESTION 1

You administer a Microsoft SQL Server 2012 database that contains a table named OrderDetail. You discover that the NCI_OrderDetail_CustomerID non-clustered index is fragmented. You need to reduce fragmentation. You need to achieve this goal without taking the index offline. Which Transact-SQL batch should you use?

- A.

```
CREATE INDEX NCI_OrderDetail_CustomerID ON OrderDetail.CustomerID WITH DROP EXISTING
```
- B.

```
ALTER INDEX NCI_OrderDetail_CustomerID ON OrderDetail.CustomerID REORGANIZE
```
- C.

```
ALTER INDEX ALL ON OrderDetail REBUILD
```
- D.

```
ALTER INDEX NCI_OrderDetail_CustomerID ON OrderDetail.CustomerID REBUILD
```

Correct Answer: B

Section: (none)

Explanation

Explanation/Reference:

Reference: <http://msdn.microsoft.com/en-us/library/ms188388.aspx>

QUESTION 2

You develop a Microsoft SQL Server 2012 database.

You need to create and call a stored procedure that meets the following requirements:

Accepts a single input parameter for CustomerID.

Returns a single integer to the calling application.

Which Transact-SQL statement or statements should you use? (Each correct answer presents part of the solution. Choose all that apply.)

- A.

```
CREATE PROCEDURE dbo.GetCustomerRating @Customer INT, @CustomerRating INT OUTPUT  
AS  
SET NOCOUNT ON  
SELECT @CustomerRating = CustomerOrders/CustomerValue  
FROM Customers WHERE CustomerID = @CustomerID  
RETURN  
GO
```
- B.

```
EXECUTE dbo.GetCustomerRating 1745
```
- C.

```
DECLARE @customerRatingByCustomer INT  
DECLARE @Result INT  
EXECUTE @Result = dbo.GetCustomerRating  
1745  
, @CustomerRatingByCustomer
```
- D.

```
CREATE PROCEDURE dbo.GetCustomerRating @CustomerID INT, @CustomerRating INT OUTPUT  
AS  
  
SET NOCOUNT ON  
SELECT @Result = CustomerOrders/CustomerValue  
FROM Customers WHERE CustomerID = @CustomerID  
RETURN @Result  
GO
```
- E.

```
DECLARE @CustomerRatingByCustomer INT  
EXECUTE dbo.GetCustomerRating @CustomerID = 1745,  
@CustomerRating = @CustomerRatingByCustomer OUTPUT
```
- F.

```
CREATE PROCEDURE dbo.GetCustomerRating  
@CustomerID INT  
AS  
DECLARE @Result INT  
SET NOCOUNT ON  
SELECT @Result = CustomerOrders/CustomerValue
```

```
FROM Customers
WHERE Customer= = @CustomerID
RETURNS @Result
```

Correct Answer: CF

Section: (none)

Explanation

Explanation/Reference: