Microsoft

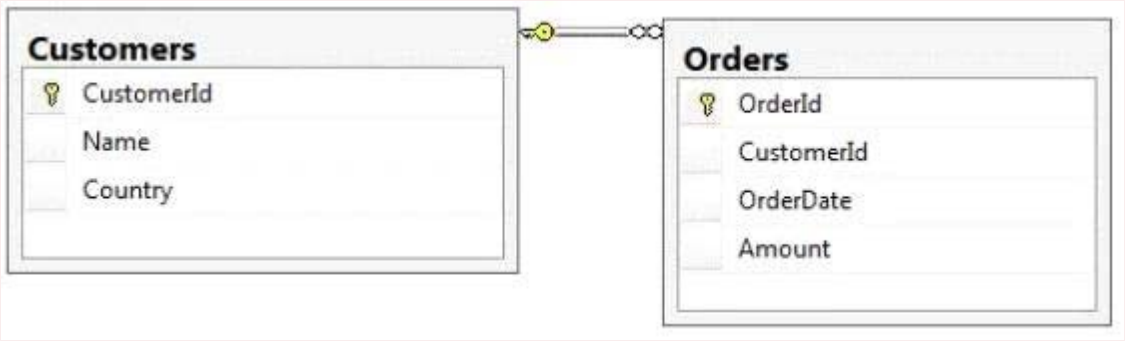
70-461

Querying Microsoft SQL Server 2012

http://www.test4certification.com/70-461.html

Question: 1

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



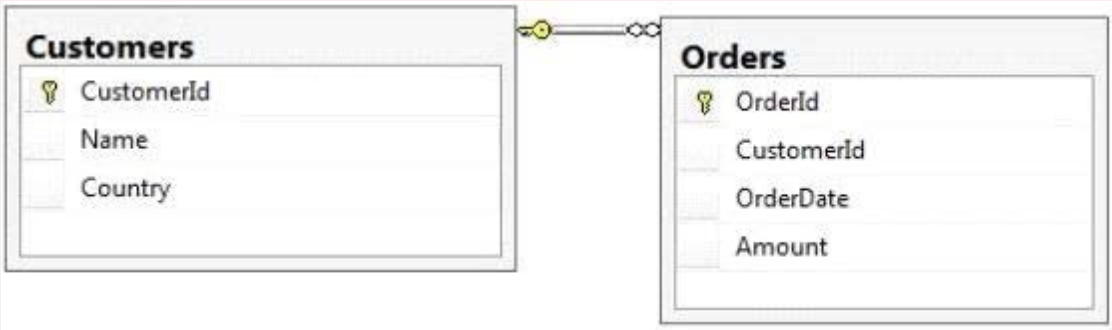
You need to display rows from the Orders table for the Customers row having the CustomerId value set to 1 in the following XML format. <row OrderId= "1" orderDate="2000-01-01T00:00:00", Amount="3400.00" Name="Customer A" Country="Australia" /> <row OrderId="2" OrderDate="2001-01-01T00:00:00" Amount="4300.00" 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, CrderId, 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')

Answer: A

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 CustomerIdvalue 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>
```

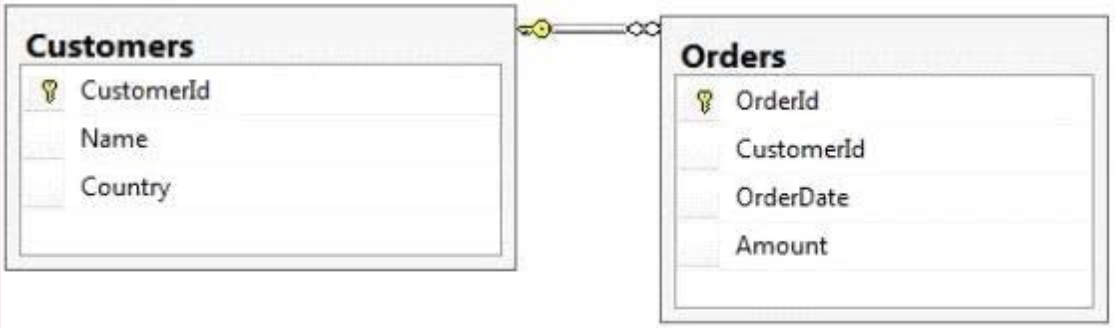
Which Transact-SQL query should you use?

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.CustomerXd - 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, CrderId, 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')

Answer: C

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="2000-01-01T00:00:00" Amount="3400.00" />
  <Orders OrderId="2" OrderDate="2001-01-01T00:00:00" Amount="4300.00" />
</Customers>
```

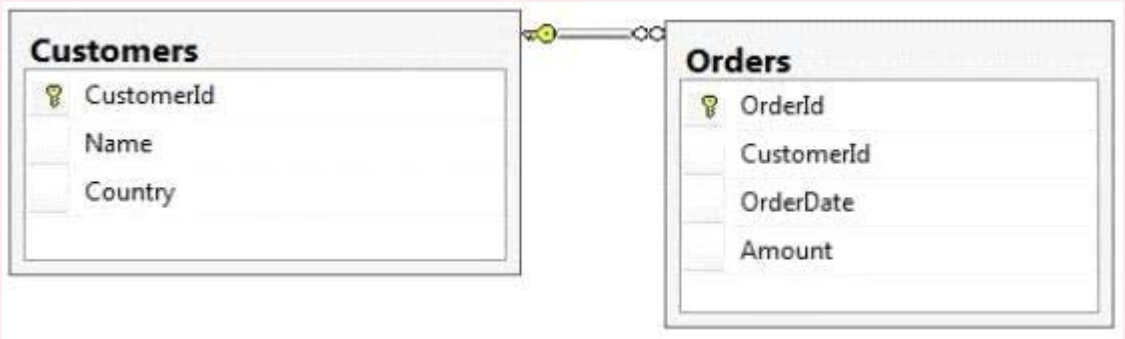
Which Transact-SQL query should you use?

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.CustomerXd - 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, CrderId, 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')

Answer: E

Question: 4

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</OrderId>
  <OrderDate>2000-01-01T00:00:00</OrderDate>
  <Amount>3400.00</Amount>
  <Customers>
    <Name>Customer A</Name>
    <Country>Australia</Country>
  </Customers>
</Orders>
<Orders>
  <OrderId>2</OrderId>
  <OrderDate>2001-01-01T00:00:00</OrderDate>
  <Amount>4300.00</Amount>
  <Customers>
    <Name>Customer A</Name>
    <Country>Australia</Country>
  </Customers>
</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.CustomerXd - 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, Crderld, 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')

Answer: D

Question: 5

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 top half of the students arranged by their average marks must be given a rank of 1 and the remaining students must be given a rank of 2. 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 FRCM (SELECT StudentCode, Marks AS Marks, RANK() OVER (PARTITION BY SubjectCode ORDER 3Y Marks DESC) AS Rank FRCM 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, RANXO OVER (PARTITION BY StudentCode ORDER BY Marks DESC) AS Rank FROM StudentMarks) tmp WHERE Rank = 1

Answer: D

Question: 6

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 FRCM (SELECT StudentCode, Marks AS Marks, RANK() OVER (PARTITION BY SubjectCode ORDER 3Y Marks DESC) AS Rank FRCM 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, RANXO OVER (PARTITION BY StudentCode ORDER BY Marks DESC) AS Rank FROM StudentMarks) tmp WHERE Rank = 1

Answer: A

Question: 7

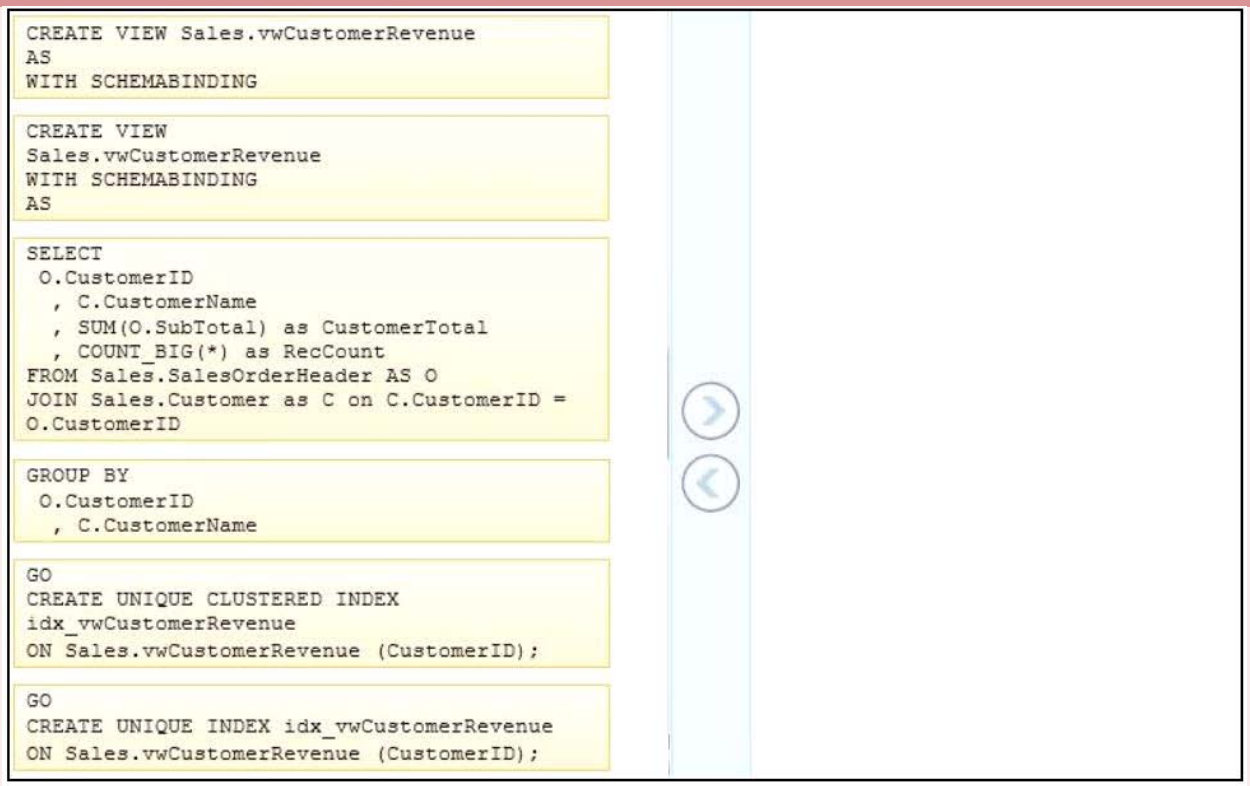
You create a table that has the StudentCode, SubjectCode, and Marks columns to record mid-year marks for students. The table has marks obtained by 50 students for various subjects. You need to retrieve the students who scored the highest marks for each subject along with the marks. Which Transact-SQL 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 FRCM (SELECT StudentCode, Marks AS Marks, RANK() OVER (PARTITION BY SubjectCode ORDER 3Y Marks DESC) AS Rank FRCM 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, RANXO OVER (PARTITION BY StudentCode ORDER BY Marks DESC) AS Rank FROM StudentMarks) tmp WHERE Rank = 1

Answer: F

Question: 8

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.)



```
CREATE VIEW Sales.vwCustomerRevenue
AS
WITH SCHEMABINDING
```

```
CREATE VIEW
Sales.vwCustomerRevenue
WITH SCHEMABINDING
AS
```

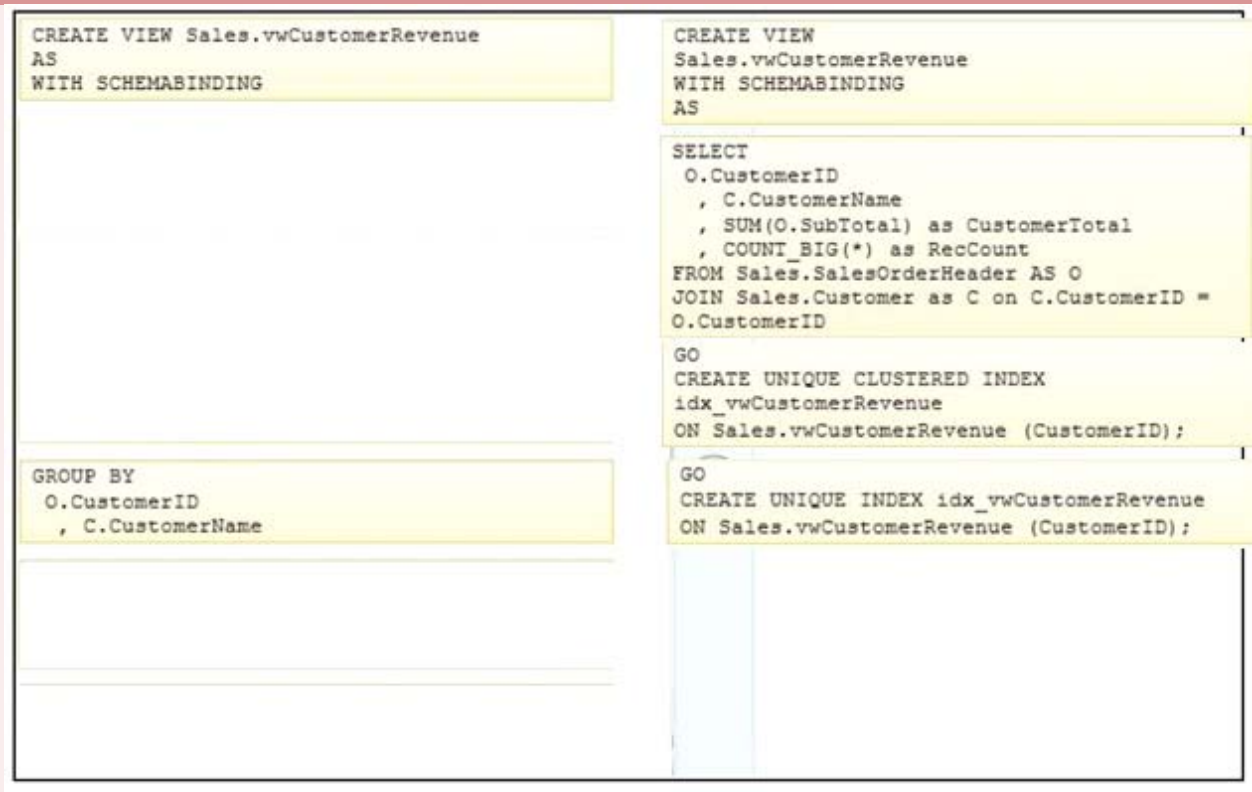
```
SELECT
 O.CustomerID
  , C.CustomerName
  , SUM(O.SubTotal) as CustomerTotal
  , COUNT_BIG(*) as RecCount
FROM Sales.SalesOrderHeader AS O
JOIN Sales.Customer as C on C.CustomerID =
O.CustomerID
```

```
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);
```

Answer:



Question: 9

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 All TotalItems AS Item3lnStore + 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 All TotalItems AS SUM (ItemsInStore, ItemsInWarehouse)

Answer: B

Explanation: Ref: http://www.kodyaz.com/articles/sql-server-computed-column-calculated-column-sample.aspx

Question: 10

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

```
CREATE TABLE [dbo].[Customers](
   [CustomerId] [bigint] NOT NULL,
   [MobileNumber] [nvarchar](25) NOT NULL,
   [HomeNumber] [nvarchar](25) NULL,
   [Name] [nvarchar](50) NOT NULL,
   [Country] [nvarchar](25) NOT NULL,
    CONSTRAINT [PK_Customers] PRIMARY KEY CLUSTERED
   (
     [CustomerId] ASC
   ) ON [PRIMARY]
) ON [PRIMARY]
```

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

A. CREATE TRIGGER TrgPhoneNumberChange ON Customers FOR UPDATE AS IF COLUMNS_UPDATED (HomeNumber, MobiieNumber) - - Create Audit Records
B. CREATE TRIGGER TrgPhoneNumberChange ON Customers FOR UPDATE AS IF EXISTS(SELECT HomeNumber from inserted) OR EXISTS (SELECT MobileNumber FROM inserted) - - Create Audit Records
C. CREATE TRIGGER TrgPhoneNumberChange ON Customers FOR UPDATE AS IF COLUMNS_CHANGED (HomeNumber, MobileNumber) - - Create Audit Records
D. CREATE TRIGGER TrgPhoneNumberChange ON Customers FOR UPDATE AS IF UPDATE (HomeNumber) OR UPDATE (MobileNumber) - - Create Audit Records

Answer: D

Question: 11

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
```

A. Option A
B. Option B

C. Option C
D. Option D

Answer: A

Question: 12

You develop a Microsoft SQL Server 2012 database. You create a view that performs the following tasks: Joins 8 tables that contain up to 500,000 records each. Performs aggregations on 5 fields. The view is frequently used in several reports. You need to improve the performance of the reports. What should you do?

A. Convert the view into a table-valued function.
B. Convert the view into a Common Table Expression (CTE).
C. Convert the view into an indexed view.
D. Convert the view into a stored procedure and retrieve the result from the stored procedure into a temporary table.

Answer: C

Question: 13

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 {Cu3tomerID; 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)

Answer: A

Question: 14

You have three tables that contain data for dentists, psychiatrists, and physicians. You create a view that is used to look up their email addresses and phone numbers. The view has the following definition:

```
Create view apt.vwProviderList
(Specialty, CompanyID, CompanyNumber, LastName,
 FirstName, BusinessName, Email, Phone)

as

SELECT 'Dentist' as Specialty
   , DentistID
   , DentistNumber
   , DentistLastName
   , DentistFirstName
   , DentistBusinessName
   , Email
   , Phone
FROM apt.Dentist
UNION ALL
SELECT 'Psychiatrist' as Specialty
   , PsychiatristID
   , PsychiatristNumber
   , PsychiatristLastName
   , PsychiatristFirstName
   , PsychiatristBusinessName
   , Email
   , Phone
SELECT 'Physician' as Specialty
   , PhysicianID
   , PhysicianNumber
   , PhysicianLastName
   , PhysicianFirstName
   , PhysicianBusinessName
   , Email
   , Phone
FROM apt.Physician
GO
```

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

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

Answer: B

Question: 15

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.

Answer: C

Thank You for Using Our Product

Microsoft

70-461

Querying Microsoft SQL Server 2012

http://www.test4certification.com/70-461.html

Test4certification provide latest certification material only.

Microsoft	Cisco	IBM	HP	Apple
MCSE 70-281 70-282	CCNA 640-760 640-816	IBM Mastery 00M-646 00M-245	HP APC HP0-J47 HP0-S29	ACMT 9L0-063 9L0-010
MCTS 70-664 70-672	CCNP 642-812 642-825	IBM Lotus 000-M64 000-M48	HP ASP HP3-C15 HP2-B44	Mac OS X 9L0-407 9L0-510
MCITP 70-667 70-642	CCDP 642-902 642-874	IBM Tivoli software 000-560 000-033	HP ASC HP2-E34 HP2-K22	ACTC 9L0-062 9L0-624
MBS MB6-872 MB2-876	CCSP 642-627 642-545	IBM Cloud Computing 000-032 000-280	HP Specialist HP3-X01 HP2-E44	ACSP 9L0-402 9L0-410
MCSA 2003 70-620 70-680	CCVP 642-427 642-642	IBM Storage 000-203 000-957	Master ASE HP0-J44 HP0-M24	ACMA 9L0-827 9L0-619
MCAS 77-884 77-602	CCDA 640-863 640-864	IBM WebSphere 000-172 000-216	HP Sales HP2-E43 HP2-Z14	Logic Pro 9 9L0-837 9L0-839