70-461:

Querying Microsoft

SQL Server 2012

Version:

Demo

About Exambible

Your Partner of IT Exam

Found in 1998

Exambible is a company specialized on providing high quality IT exam practice study materials, especially Cisco CCNA, CCDA, CCNP, CCIE, Checkpoint CCSE, CompTIA A+, Network+ certification practice exams and so on. We guarantee that the candidates will not only pass any IT exam at the first attempt but also get profound understanding about the certificates they have got.

There are so many alike companies in this industry, however, Exambible has its unique advantages that other companies could not achieve.

Our Advances

* 99.9% Uptime

All examinations will be up to date.

* 24/7 Quality Support

We will provide service round the clock.

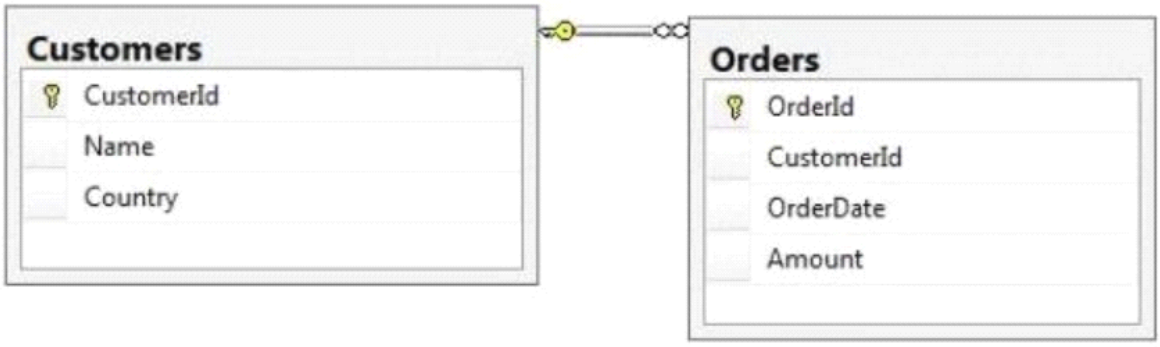
* 100% Pass Rate

Our guarantee that you will pass the exam.

* Unique Gurantee

If you do not pass the exam at the first time, we will not only arrange FULL REFUND for you, but also provide you another exam of your claim, ABSOLUTELY FREE!

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

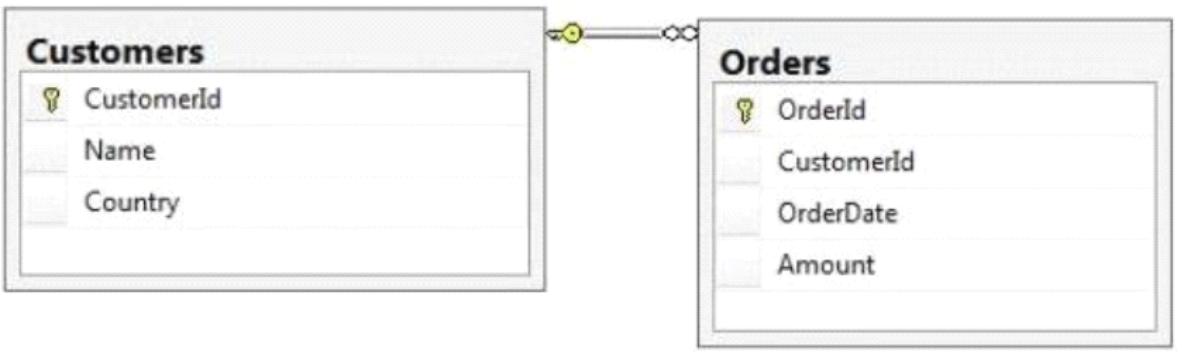
INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId

WHERE Customers.CustomerId= 1

FOR XML PATH ('Customers')

Answer: A

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

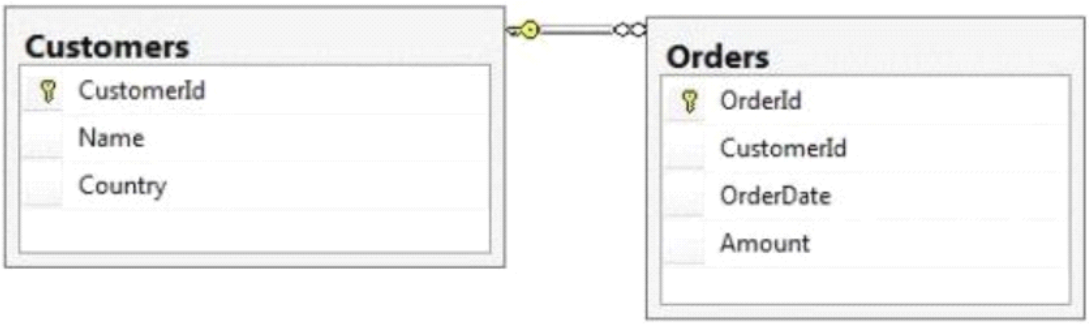
INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId

WHERE Customers.CustomerId= 1

FOR XML PATH ('Customers')

Answer: C

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

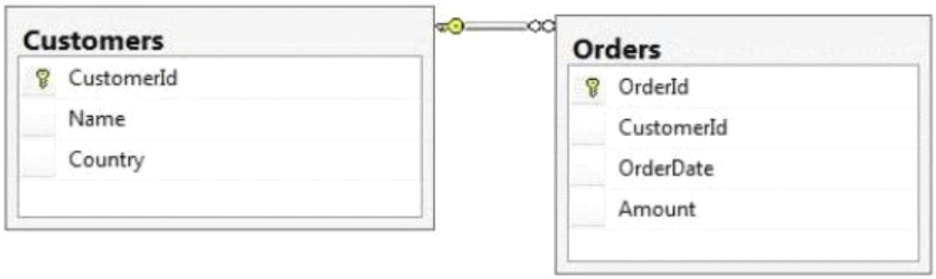
INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId

WHERE Customers.CustomerId= 1

FOR XML PATH ('Customers')

Answer: E

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, 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: D

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

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

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

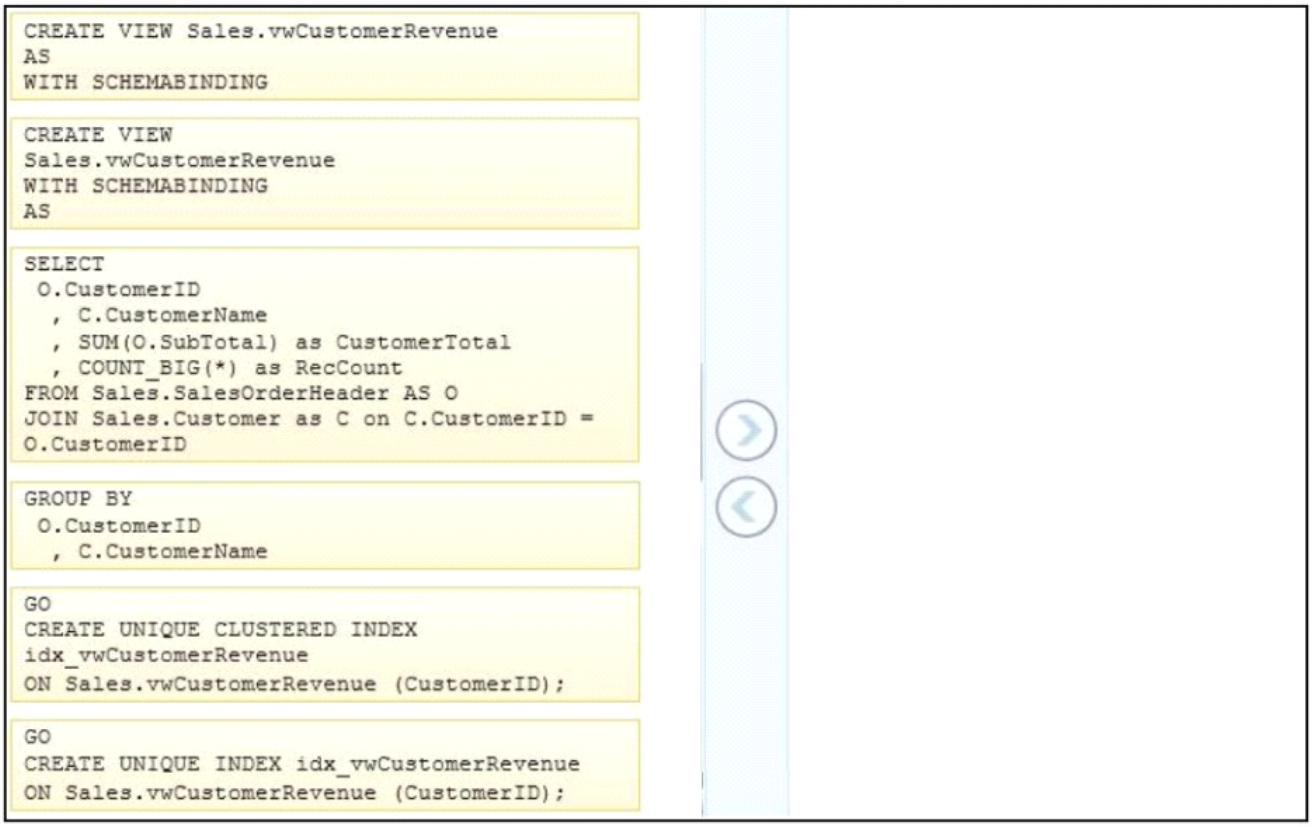
Answer: F

8. DRAG DROP

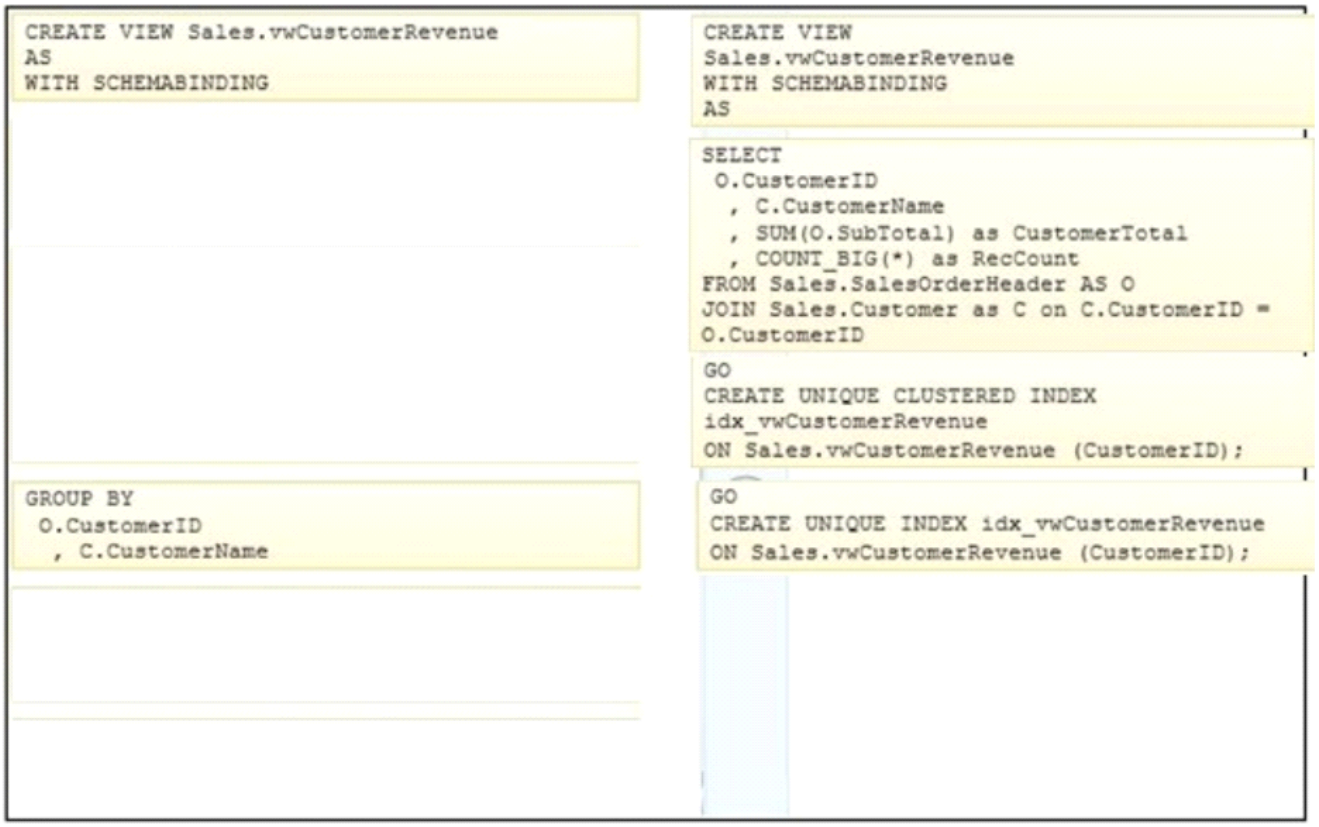
You use a Microsoft SQL Server 2012 database.

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.)



Answer:



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 Item3InStore + 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: C

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

Contact us

We are proud of our high-quality customer service, which serves you around the clock 24/7.

Contact us by emails

Please allow up to 24 hours for us to respond

problems about sales and so on **Sales Email** sales@exambible.com	problems about payment,installation, acitvation and so on **Support Email** support@exambible.com

Please Note: Please accept mail from an official, do not believe any of the other E-mail