70-433 Q&As

TS: Microsoft SQL Server 2008, Database Development

Pass Microsoft 70-433 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.pass4lead.com/70-433.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Microsoft Official Exam Center

- **Instant Download** After Purchase
- **100% Money Back** Guarantee
- **365 Days** Free Update
- **800,000+** Satisfied Customers

QUESTION 1

You have a table named Employees.

You want to identify the supervisor to which each employee reports. You write the following query.

SELECT e.EmloyeeName AS [EmployeeName],

EmployeeName AS [SuperVisorName]

FROM Employees e

You need to ensure that the query returns a list of all employees and their respective supervisor.

Which join clause should you use to complete the query?

A. LEFT JOIN Employees s ON e.ReportsTo = s.EmployeeId

B. RIGHT JOIN Employees s ON e.ReportsTo = s.EmployeeId

C. INNER JOIN Employees s ON e.EmployeeId = s.EmployeeId

D. LEFT JOIN Employees s ON e.EmployeeId = s.EmployeeId

Correct Answer: A

QUESTION 2

You are reviewing a trigger in the database which was deployed with the folowing script:

EXECUTE AS USER = \\'BuildUser\\' GO CREATE TRIGGER Inventory.TR_Stock ON Inventory.Stock FOR INSERT, UPDATE, DELETE EXECUTE AS SELF AS ...

A user \\'WebUser\\' insert rows into Inventory.Stock table. You need to identify under which security context the trigger will execute.

A. DBO

B. Inventory

C. WebUser

D. BuildUser

Correct Answer: D

QUESTION 3

You are creating a new table in a database. Your business requires you to store data in the table for only seven days.

You need to implement a partitioned table to meet this business requirement.

Which tasks should you complete?

A. Create the partition function Create the partition scheme Create the table

B. Create the partition function Create the table Create a filtered index

C. Add a secondary file to the primary filegroups Create the table Create the distributed partitioned view

D. Create the partition function Create the partition scheme Create the distributed partitioned view

Correct Answer: A

QUESTION 4

You are a developer for a Microsoft SQL Server 2008 R2 database instance. You create tables named order, customer, and product as follows:

```
CREATE TABLE [dbo].[order]
 ([OrderID] [int],
  [ProductID] [int],
  [CustomerID] [int],
  [OrderDate] [datetime]);

CREATE TABLE [dbo].[customer]
 ([CustomerID] [int],
  [CustomerName] [varchar](100),
  [Address] [varchar](200),
  [City] [varchar](100),
  [State] [varchar](50),
  [ZipCode] [varchar](5));

CREATE TABLE [dbo].[product]
 ([ProductID] [int],
  [ProductName] [varchar](100),
  [SalePrice] [money],
  [ManufacturerName] [varchar](100));
```


You need to write a query to return all customer names and total number of orders for customers who have placed more than 10 orders. Which SQL query should you use?

A.
```
SELECT
  c.CustomerName,
  p.ProductName,
  SUM(p.SalePrice) AS Sales
FROM
  product p INNER JOIN
  [order] o ON p.ProductID = o.ProductID INNER JOIN
  customer c ON o.CustomerID = c.CustomerID
GROUP BY GROUPING SETS ((c.CustomerName, p.ProductName), ());
```

B.
```
SELECT
  c.CustomerName,
  p.ProductName,
  SUM(p.SalePrice) AS Sales
FROM
  product p INNER JOIN
  [order] o ON p.ProductID = o.ProductID INNER JOIN
  customer c ON o.CustomerID = c.CustomerID
GROUP BY GROUPING SETS ((c.CustomerName), (p.ProductName), ());
```

C.
```
SELECT
  c.CustomerName,
  COUNT(o.OrderID) AS Orders
FROM
  customer c INNER JOIN
  [order] o ON c.CustomerID = o.CustomerID
WHERE
  COUNT(o.OrderID) > 10
GROUP BY
  c.CustomerName;
```

D.
```
SELECT
  c.CustomerName,
  COUNT(o.OrderID) AS Orders
FROM
  customer c INNER JOIN
  [order] o ON c.CustomerID = o.CustomerID
GROUP BY
  c.CustomerName
HAVING
  COUNT(o.OrderID) > 10;
```

E.
```
SELECT
  c.CustomerName,
  AVG(p.SalePrice) AS Sales
FROM
  product p INNER JOIN
  [order] o ON p.ProductID = o.ProductID INNER JOIN
  customer c ON o.CustomerID = c.CustomerID
WHERE
  o.OrderDate > '09/01/2011'
GROUP BY
  c.CustomerName
HAVING
  AVG(p.SalePrice) >= 500
```

F.
```
SELECT
  c.CustomerName,
  AVG(p.SalePrice) AS Sales
FROM
  product p INNER JOIN
  [order] o ON p.ProductID = o.ProductID INNER JOIN
  customer c ON o.CustomerID = c.CustomerID
WHERE
  o.OrderDate > '09/01/2011' AND
  AVG(p.SalePrice) >= 500
```

G.
```
SELECT
  p.ProductName,
  DATEPART(mm, o.OrderDate) OrderMonth,
  SUM(p.SalePrice) AS Sales
FROM
  product p INNER JOIN
  [order] o ON p.ProductID = o.ProductID
GROUP BY CUBE(p.ProductName, DATEPART(mm, o.OrderDate));
```

H.
```
SELECT
  p.ProductName,
  DATEPART(mm, o.OrderDate) OrderMonth,
  SUM(p.SalePrice) AS Sales
FROM
  product p INNER JOIN
  [order] o ON p.ProductID = o.ProductID
GROUP BY CUBE;
```

A. B. C. D. E. F. G. H.

I.
```
SELECT
  p.ProductName,
  DATEPART(mm, o.OrderDate) OrderMonth,
  SUM(p.SalePrice) AS Sales
FROM
  product p INNER JOIN
  [order] o ON p.ProductID = o.ProductID
GROUP BY p.ProductName, OrderMonth;
```

J.
```
SELECT
  p.ProductName,
  DATEPART(mm, o.OrderDate) OrderMonth,
  SUM(p.SalePrice) AS Sales
FROM
  product p INNER JOIN
  [order] o ON p.ProductID = o.ProductID
GROUP BY p.ProductName, DATEPART(mm, o.OrderDate);
```


I. J.

Correct Answer: D

QUESTION 5

You are the database developer for an order-processing application. The database has the following tables:

```
CREATE TABLE dbo.Product
  (ProdID INT NOT NULL PRIMARY KEY,
   ProdName VARCHAR(100) NOT NULL,
   SalePrice MONEY NOT NULL,
   ManufacturerName VARCHAR(100) NOT NULL);

CREATE TABLE dbo.Customer
  (CustID INT NOT NULL PRIMARY KEY,
   CustName VARCHAR(100) NOT NULL,
   CustAddress VARCHAR(200) NOT NULL,
   CustCity VARCHAR(100) NOT NULL,
   CustState VARCHAR(50) NOT NULL,
   CustPostalCode VARCHAR(5) NOT NULL);

CREATE TABLE dbo.[Order]
(OrderID INT NOT NULL PRIMARY KEY,
   ProdID INT NOT NULL
     REFERENCES dbo.Product(ProdId),
   CustID INT NOT NULL
     REFERENCES dbo.Customer(CustId),
   OrderDate DATETIME NOT NULL);
```


You need to ensure that the following requirements are met:

Data is loaded into the tables.

Data that has been inserted will be removed if any statement fails. No open transactions are performed after the batch has executed.

Which Transact-SQL statements should you use?

A.

```
BEGIN TRY

BEGIN TRANSACTION

INSERT INTO dbo.Product VALUES
  (1, 'Chair', 146.58, 'Contoso'),
  (2, 'Table', 458.36, 'Contoso'),
  (3, 'Cabinet', 398.17, 'Northwind Traders'),
  (4, 'Desk', 1483.25, 'Northwind Traders');

INSERT INTO dbo.Customer VALUES
  (1, 'John Smith', '200 West 2nd St', 'Seattle', 'WA', '98060'),
  (2, 'Bob Jones', '300 Main St', 'Portland', 'OR', '97211'),
  (3, 'Fred Thomson', '100 Park Ave', 'San Francisco', 'CA', '94172');

INSERT INTO dbo.[Order] VALUES
  (1, 1, 2, '09/15/2011'),
  (2, 4, 2, '09/15/2011'),
  (3, 2, 1, '08/17/2011'),
  (4, 2, 3, '07/01/2011'),
  (5, 3, 3, '10/02/2011');

END TRY
BEGIN CATCH

IF @@TRANCOUNT > 0 BEGIN
  ROLLBACK TRANSACTION;
END;

END CATCH;
```

B.

```
BEGIN TRANSACTION

INSERT INTO dbo.Product VALUES
  (1, 'Chair', 146.58, 'Contoso'),
  (2, 'Table', 458.36, 'Contoso'),
  (3, 'Cabinet', 398.17, 'Northwind Traders'),
  (4, 'Desk', 1483.25, 'Northwind Traders');

IF @@ERROR > 0 ROLLBACK TRANSACTION;

INSERT INTO dbo.Customer VALUES
  (1, 'John Smith', '200 West 2nd St', 'Seattle', 'WA', '98060'),
  (2, 'Bob Jones', '300 Main St', 'Portland', 'OR', '97211'),
  (3, 'Fred Thomson', '100 Park Ave', 'San Francisco', 'CA', '94172');

IF @@ERROR > 0 ROLLBACK TRANSACTION;

INSERT INTO dbo.[Order] VALUES
  (1, 1, 2, '09/15/2011'),
  (2, 4, 2, '09/15/2011'),
  (3, 2, 1, '08/17/2011'),
  (4, 2, 3, '07/01/2011'),
  (5, 3, 3, '10/02/2011');

IF @@ERROR > 0 ROLLBACK TRANSACTION;

COMMIT TRANSACTION;
```

A. B.

C.

```
BEGIN TRY

SAVE TRANSACTION DataLoad

INSERT INTO dbo.Product VALUES
  (1, 'Chair', 146.58, 'Contoso'),
  (2, 'Table', 458.36, 'Contoso'),
  (3, 'Cabinet', 398.17, 'Northwind Traders'),
  (4, 'Desk', 1483.25, 'Northwind Traders');

INSERT INTO dbo.Customer VALUES
  (1, 'John Smith', '200 West 2nd St', 'Seattle', 'WA', '98060'),
  (2, 'Bob Jones', '300 Main St', 'Portland', 'OR', '97211'),
  (3, 'Fred Thomson', '100 Park Ave', 'San Francisco', 'CA', '94172');

INSERT INTO dbo.[Order] VALUES
  (1, 1, 2, '09/15/2011'),
  (2, 4, 2, '09/15/2011'),
  (3, 2, 1, '08/17/2011'),
  (4, 2, 3, '07/01/2011'),
  (5, 3, 3, '10/02/2011');

COMMIT TRANSACTION DataLoad;

END TRY
BEGIN CATCH

IF @@TRANCOUNT > 0 BEGIN
  ROLLBACK TRANSACTION DataLoad;
END;

END CATCH;
```

D.

```
SET XACT_ABORT ON
BEGIN TRANSACTION

INSERT INTO dbo.Product VALUES
  (1, 'Chair', 146.58, 'Contoso'),
  (2, 'Table', 458.36, 'Contoso'),
  (3, 'Cabinet', 398.17, 'Northwind Traders'),
  (4, 'Desk', 1483.25, 'Northwind Traders');

INSERT INTO dbo.Customer VALUES
  (1, 'John Smith', '200 West 2nd St', 'Seattle', 'WA', '98060'),
  (2, 'Bob Jones', '300 Main St', 'Portland', 'OR', '97211'),
  (3, 'Fred Thomson', '100 Park Ave', 'San Francisco', 'CA', '94172');

INSERT INTO dbo.[Order] VALUES
  (1, 1, 2, '09/15/2011'),
  (2, 4, 2, '09/15/2011'),
  (3, 2, 1, '08/17/2011'),
  (4, 2, 3, '07/01/2011'),
  (5, 3, 3, '10/02/2011');

COMMIT TRANSACTION;
```

C. D.

Correct Answer: B

To Read the Whole Q&As, please purchase the Complete Version from Our website.

Try our product !

100% Guaranteed Success
100% Money Back Guarantee
365 Days Free Update
Instant Download After Purchase
24x7 Customer Support
Average 99.9% Success Rate
More than 800,000 Satisfied Customers Worldwide
Multi-Platform capabilities - Windows, Mac, Android, iPhone, iPod, iPad, Kindle

We provide exam PDF and VCE of Cisco, Microsoft, IBM, CompTIA, Oracle and other IT Certifications.
You can view Vendor list of All Certification Exams offered:

https://www.pass4lead.com/allproducts

Need Help

Please provide as much detail as possible so we can best assist you.
To update a previously submitted ticket: