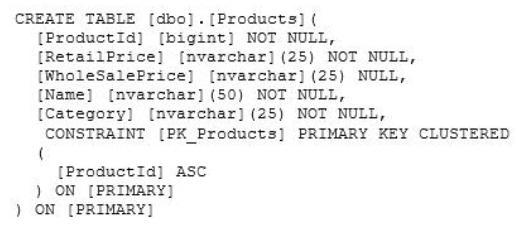
Question No : 1 -

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



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 CCLUMNS\_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

question\_answerVIEW ANSWER

SHOW COMMENTS9

Answer : D

Reference: http://msdn.microsoft.com/en-us/library/bb510663.aspx Reference: http://msdn.microsoft.com/en-us/library/ms186329.aspx

Question No : 2 -

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

question\_answerVIEW ANSWER

SHOW COMMENTS1

Answer : A

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

Question No : 3 -

You are developing a database that will contain price information.  
You need to store the prices that include a fixed precision and a scale of six digits.  
Which data type should you use?

* A. Real
* B. Small money
* C. Money
* D. Decimal

question\_answerVIEW ANSWER

SHOW COMMENTS9

Answer : D

Question No : 4 -

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

question\_answerVIEW ANSWER

SHOW COMMENTS2

Answer : C

Reference: http://msdn.microsoft.com/en-us/library/hh231256.aspx Reference: http://msdn.microsoft.com/en-us/library/hh213125.aspx

Question No : 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

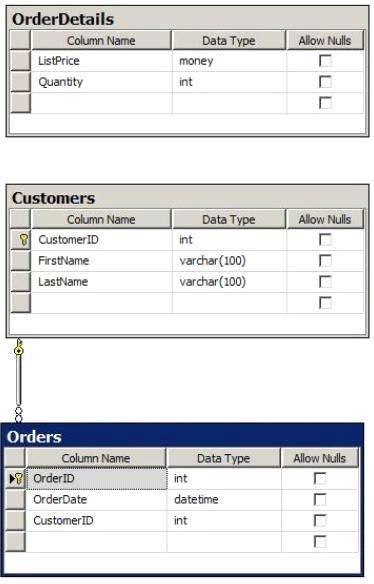
question\_answerVIEW ANSWER

SHOW COMMENTS3

Answer : A

Question No : 6 -

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



You have an application named Appl. You have a parameter named @Count that uses the  
int data type. App1 is configured to pass @Count to a stored procedure. You need to  
create a stored procedure named usp\_Customers for Appl. Usp\_Customers must meet the  
following requirements:  
✑ NOT use object delimiters.  
✑ Minimize sorting and counting.  
✑ Return only the last name of each customer in alphabetical order.  
✑ Return only the number of rows specified by the @Count parameter.  
✑ The solution must NOT use BEGIN and END statements.  
Which code segment should you use?  
To answer, type the correct code in the answer area.

question\_answerVIEW ANSWER

SHOW COMMENTS0

Answer : Please review the explanation part for this answer

Explanation: CREATE PROCEDURE usp\_Customers @Count int AS SELECT TOP(@Count) Customers.LastName FROM Customers ORDER BY Customers.LastName

Question No : 7 -

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

question\_answerVIEW ANSWER

SHOW COMMENTS7

Answer : A

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-optimalperformance-comparison/

Question No : 8 -

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

question\_answerVIEW ANSWER

SHOW COMMENTS0

Answer : E

Reference: http://technet.microsoft.com/en-us/library/ms189857.aspx Reference: http://msdn.microsoft.com/en-us/library/ms187373.aspx

Question No : 9 -

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
* B. ProductName = cte.ProductName AND p.CreatedDateTime > cte.CreatedDateTime
* C. 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
* D. WITH CTEDupRecords AS ( SELECT MIN(CreatedDateTime) AS CreatedDateTime, ProductName FROM Products GROUP BY ProductName ) DELETE p FROM Products p JOIN CTEDupRecords cte ON
* E. ProductName = cte.ProductName
* F. 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
* G. ProductName = cte.ProductName

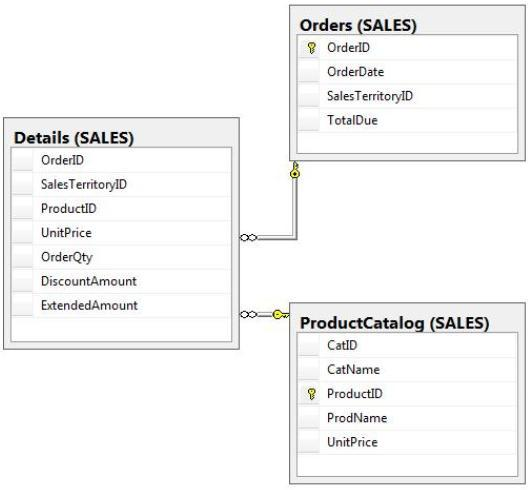
question\_answerVIEW ANSWER

SHOW COMMENTS7

Answer : B

Question No : 10 -

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.

question\_answerVIEW ANSWER

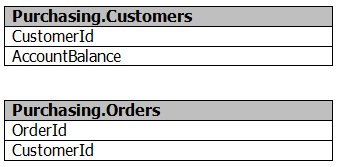
SHOW COMMENTS2

Answer : Please review the explanation part for this answer

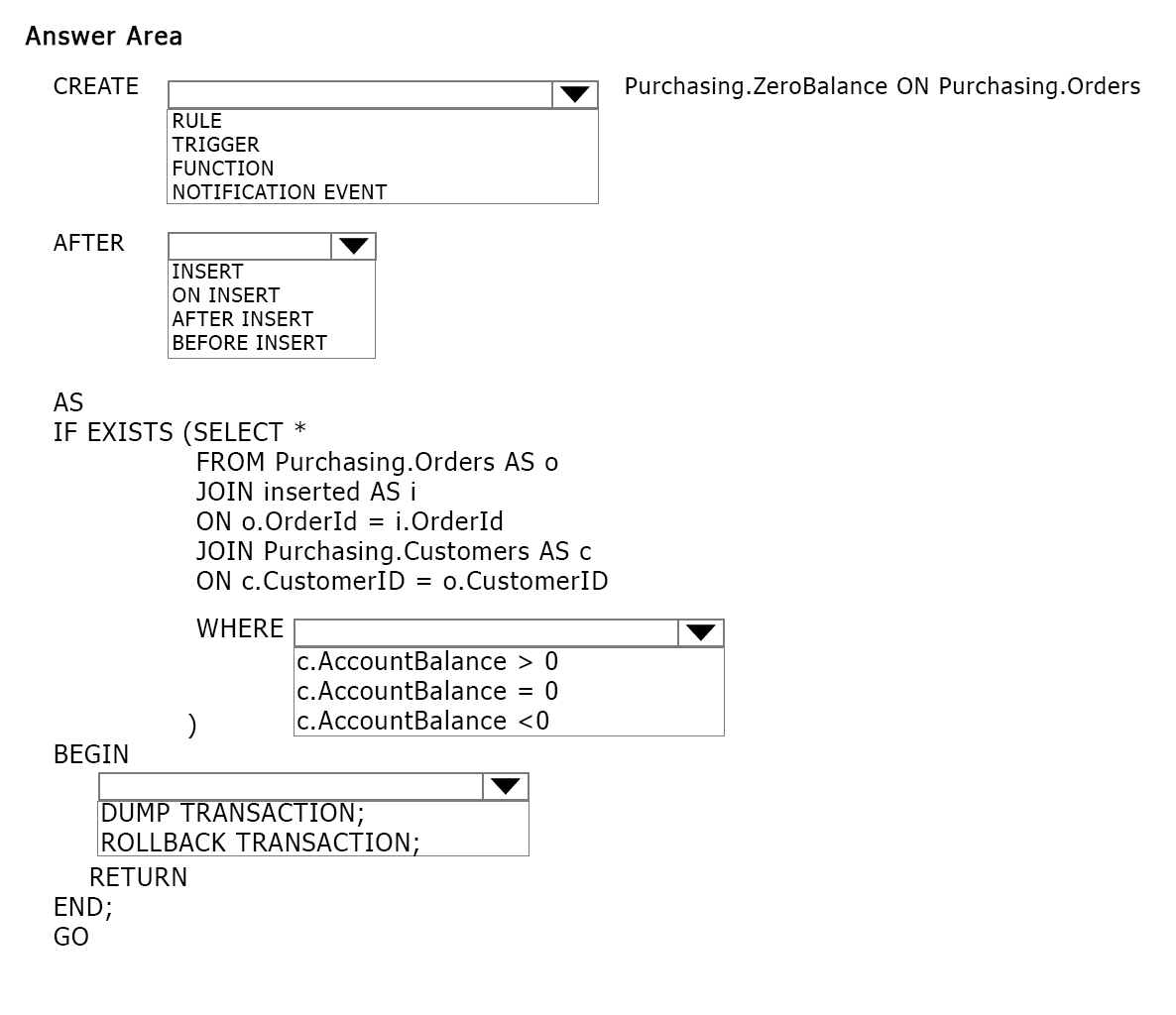
Explanation: SELECT OrderID, SUM(ExtendedAmount) AS TotalSales FROM Sales.Details GROUP BY OrderID ORDER BY OrderID

Question No : 11 -

You are designing an order entry system that uses an SQL Server database. The database  
includes the following tables:



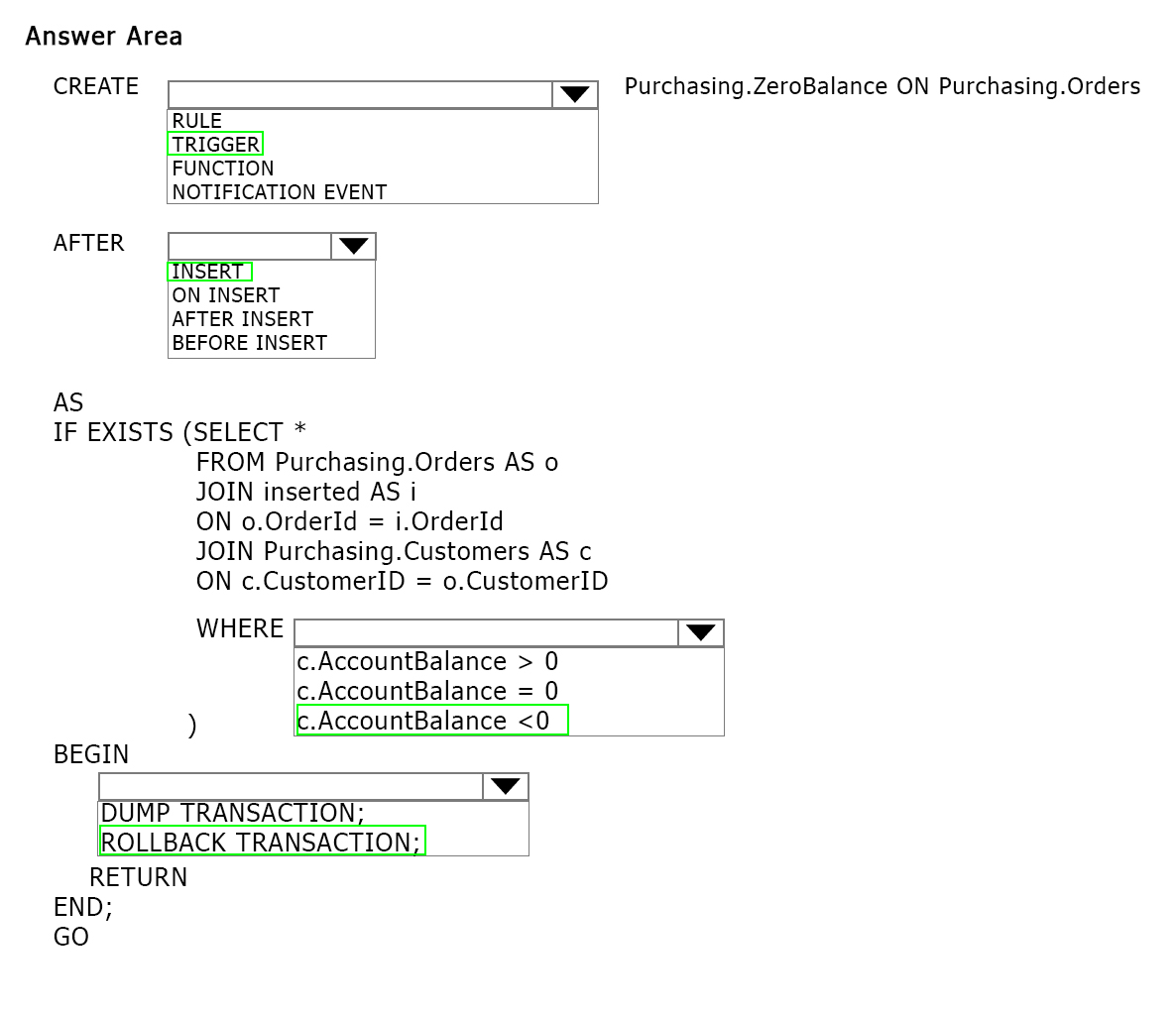
You need to ensure that Orders are added to the Orders table only for customers that have  
an account balance of zero.  
How should you complete the relevant Transact-SQL statement? To answer, select the  
correct Transact-SQL statement from each list in the answer area.



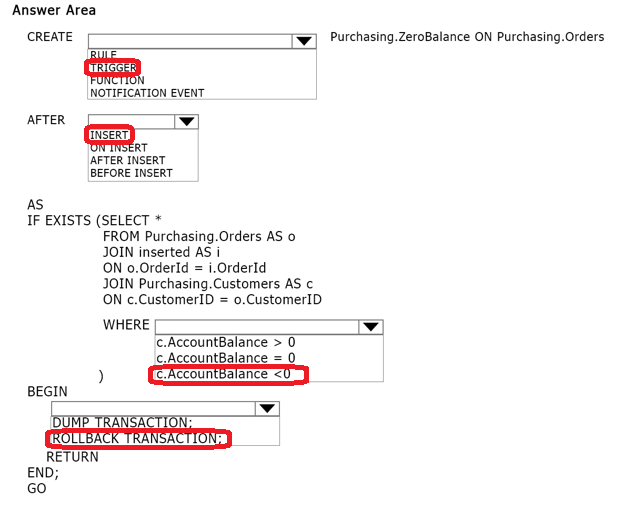
question\_answerVIEW ANSWER

SHOW COMMENTS15

Answer :



Explanation:



The Transact SQL CREATE TRIGGER command creates a DML, DDL, or logon trigger. A trigger is a special kind of stored procedure that automatically executes when an event occurs in the database server. DML triggers execute when a user tries to modify data through a data manipulation language (DML) event. DML events are INSERT, UPDATE, or DELETE statements on a table or view. These triggers fire when any valid event is fired, regardless of whether or not any table rows are affected. Partial syntax is: CREATE TRIGGER [ schema\_name . ]trigger\_name ON { table | view } [ WITH <dml\_trigger\_option> [ ,...n ] ] { FOR | AFTER | INSTEAD OF } { [ INSERT ] [ , ] [ UPDATE ] [ , ] [ DELETE ] }

Question No : 12 -

You use Microsoft SQL Server 2012 to develop a database application.  
You create a stored procedure named dbo.ModifyData that can modify rows.  
You need to ensure that when the transaction fails, dbo.ModifyData meets the following  
requirements:  
✑ Does not return an error  
✑ Closes all opened transactions  
Which Transact-SQL statement should you use?

* A. BEGIN TRANSACTION BEGIN TRY EXEC dbo.ModifyData COMMIT TRANSACTION END TRY BEGIN CATCH IF @@ TRANCOUNT = 0 ROLLBACK TRANSACTION; END CATCH
* B. BEGIN TRANSACTION BEGIN TRY EXEC dbo.ModifyData COMMIT TRANSACTION END TRY BEGIN CATCH IF @@ERROR != 0 ROLLBACK TRANSACTION; THROW; END CATCH
* C. BEGIN TRANSACTION BEGIN TRY EXEC dbo.ModifyData COMMIT TRANSACTION END TRY BEGIN CATCH IF @@TRANCOUNT = 0 ROLLBACK TRANSACTION; THROW; END CATCH
* D. BEGIN TRANSACTION BEGIN TRY EXEC dbo.ModifyData COMMIT TRANSACTION END TRY BEGIN CATCH IF @@ERROR != 0 ROLLBACK TRANSACTION; END CATCH

question\_answerVIEW ANSWER

SHOW COMMENTS0

Answer : D

Question No : 13 -

You are developing a database that will contain price information.  
You need to store the prices that include a fixed precision and a scale of six digits.  
Which data type should you use?

* A. Float
* B. Money
* C. Smallmoney
* D. Numeric

question\_answerVIEW ANSWER

SHOW COMMENTS0

Answer : D

Explanation: Numeric is the only one in the list that can give a fixed precision and scale. Reference: http://msdn.microsoft.com/en-us/library/ms179882.aspx

Question No : 14 -

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.

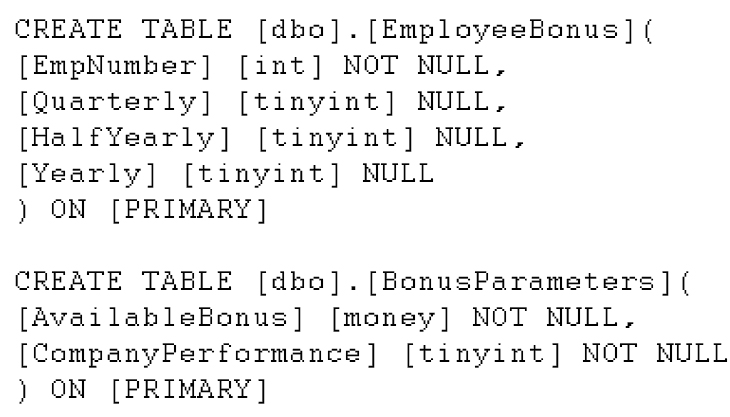
question\_answerVIEW ANSWER

SHOW COMMENTS3

Answer : K

Question No : 15 -

You have a Microsoft SQL Server database that includes two tables named  
EmployeeBonus and BonusParameters. The tables are defined by using the following  
Transact-SQL statements:



The tables are used to compute a bonus for each employee. The EmployeeBonus table  
has a non-null value in either the Quarterly, HalfYearly or Yearly column. This value  
indicates which type of bonus an employee receives. The BonusParameters table contains  
one row for each calendar year that stores the amount of bonus money available and a  
company performance indicator for that year.  
You need to calculate a bonus for each employee at the end of a calendar year.  
Which Transact-SQL statement should you use?

* A. SELECT CAST(CHOOSE((Quarterly \* AvailableBonus \* CompanyPerformance)/40, (HalfYearly \* AvailableBonus \* CompanyPerformance)/20, (Yearly \* AvailableBonus \* CompanyPerformance)/10) AS money) AS Bonus FROM EmployeeBonus, BonusParameters
* B. SELECT Bonus = CASE EmployeeBonus WHEN Quarterly=1 THEN (Quarterly \* AvailableBonus \* CompanyPerformance)/40 WHEN HalfYearly=1 THEN (HalfYearly \* AvailableBonus \* CompanyPerformance)/20 WHEN Yearly=1 THEN (Yearly \* AvailableBonus \* CompanyPerformance)/10 END FROM EmployeeBonus,BonusParameters
* C. SELECT CAST(COALESCE((Quarterly \* AvailableBonus \* CompanyPerformance)/40, (HalfYearly \* AvailableBonus \* CompanyPerformance)/20, (Yearly \* AvailableBonus \* CompanyPerformance)/10) AS money) AS Bonus FROM EmployeeBonus, BonusParameters
* D. SELECT NULLIF(NULLIF((Quarterly \* AvailableBonus \* CompanyPerformance)/40,(HalfYearly \* AvailableBonus \* CompanyPerformance)/20), (Yearly \* AvailableBonus \* CompanyPerformance)/10) AS Bonus FROM EmployeeBonus, BonusParameters

question\_answerVIEW ANSWER

SHOW COMMENTS22

Answer : B