

Microsoft

Exam Questions 70-762

Developing SQL Databases (beta)



NEW QUESTION 1

You use Microsoft SQL Server Profiler to evaluate a query named Query1. The Profiler report indicates the following issues:

- At each level of the query plan, a low total number of rows are processed.
- The query uses many operations. This results in a high overall cost for the query. You need to identify the information that will be useful for the optimizer. What should you do?

- A. Start a SQL Server Profiler trace for the event class Auto Stats in the Performance event category.
- B. Create one Extended Events session with the sqlserver.missing_column_statistics event added.
- C. Start a SQL Server Profiler trace for the event class Soft Warnings in the Errors and Warnings event category.
- D. Create one Extended Events session with the sqlserver.missing_join_predicate event added.

Answer: D

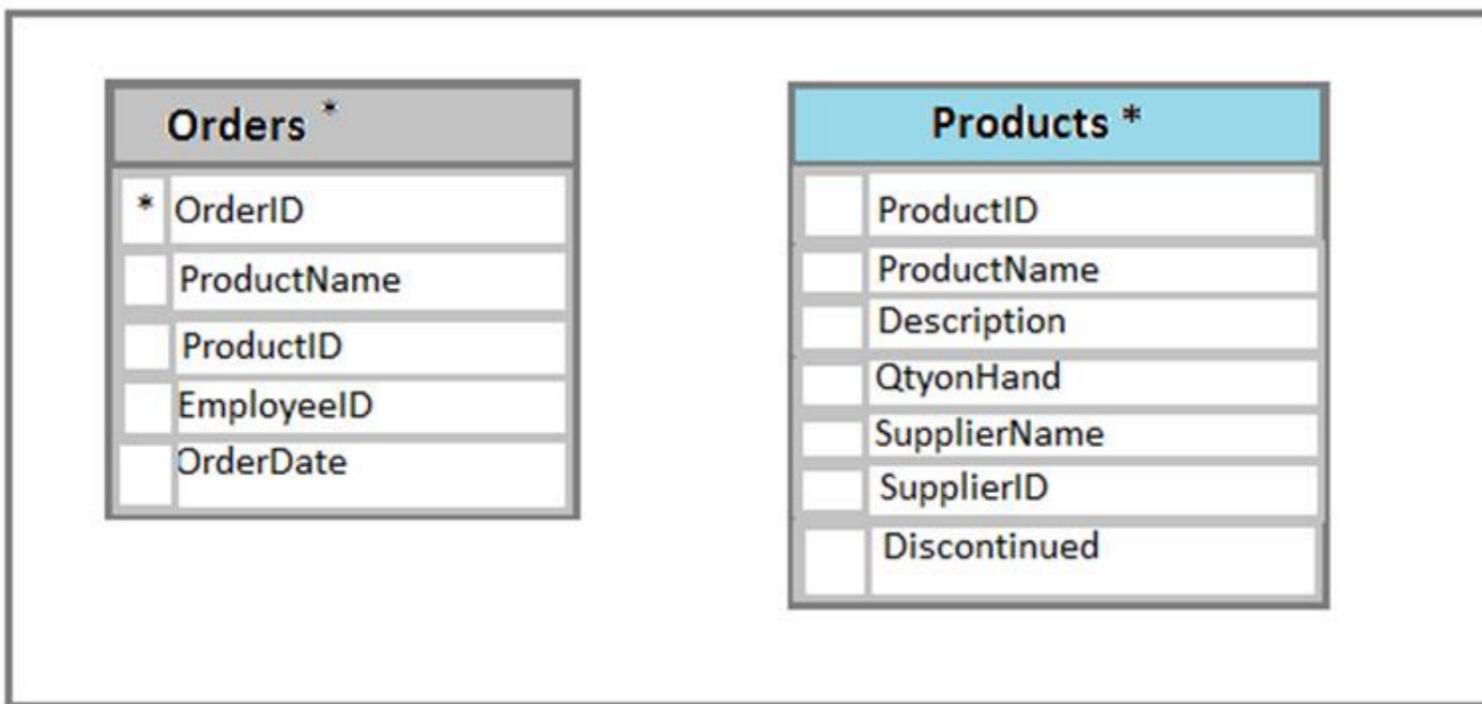
Explanation:

The Missing JoinPredicate event class indicates that a query is being executed that has no join predicate. This could result in a long-running query.

NEW QUESTION 2

Note: This question is part of a series of questions that use the same scenario. For your convenience, the scenario is repeated in each question. Each question presents a different goal and answer choices, but the text of the scenario is exactly the same in each question in this series.

You have a database named Sales that contains the following database tables: Customer, Order, and Products. The Products table and the Order table are shown in the following diagram.



The customer table includes a column that stores the data for the last order that the customer placed.

You plan to create a table named Leads. The Leads table is expected to contain approximately 20,000 records. Storage requirements for the Leads table must be minimized.

The Leads table must include the columns described in the following table.

Column name	Description
LeadID	This column stores a unique value that is automatically assigned for each lead.
IsCustomer	This column indicates whether the lead is for a current customer.

The data types chosen must consume the least amount of storage possible. You need to select the appropriate data types for the Leads table. In the table below, identify the data type that must be used for each table column. NOTE: Make only one selection in each column.

Answer Area

Data type	LeadID	IsCustomer
smallint	<input type="radio"/>	<input type="radio"/>
int	<input type="radio"/>	<input type="radio"/>
binary	<input type="radio"/>	<input type="radio"/>
numeric	<input type="radio"/>	<input type="radio"/>
bit	<input type="radio"/>	<input type="radio"/>

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Bit is a Transact-SQL integer data type that can take a value of 1, 0, or NULL.

Smallint is a Transact-SQL integer data type that can take a value in the range from -32,768 to 32,767. int, bigint, smallint, and tinyint (Transact-SQL)

Exact-number data types that use integer data.

Data type	Range	Storage
bigint	-2^{63} (-9,223,372,036,854,775,808) to $2^{63}-1$ (9,223,372,036,854,775,807)	8 Bytes
int	-2^{31} (-2,147,483,648) to $2^{31}-1$ (2,147,483,647)	4 Bytes
smallint	-2^{15} (-32,768) to $2^{15}-1$ (32,767)	2 Bytes
tinyint	0 to 255	1 Byte

References: <https://msdn.microsoft.com/en-us/library/ms187745.aspx> <https://msdn.microsoft.com/en-us/library/ms177603.aspx>

NEW QUESTION 3

Note: The question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other question in the series. Information and details provided in a question apply only to that question. You have a reporting database that includes a non-partitioned fact table named Fact_Sales. The table is persisted on disk. Users report that their queries take a long time to complete. The system administrator reports that the table takes too much space in the database. You observe that there are no indexes defined on the table, and many columns have repeating values. You need to create the most efficient index on the table, minimize disk storage and improve reporting query performance. What should you do?

- A. Create a clustered index on the table.
- B. Create a nonclustered index on the table.
- C. Create a nonclustered filtered index on the table.
- D. Create a clustered columnstore index on the table.
- E. Create a nonclustered columnstore index on the table.
- F. Create a hash index on the table.

Answer: D

Explanation:

The columnstore index is the standard for storing and querying large data warehousing fact tables. It uses column-based data storage and query processing to achieve up to 10x query performance gains in your data warehouse over traditional row-oriented storage, and up to 10x data compression over the uncompressed data size.

A clustered columnstore index is the physical storage for the entire table.

NEW QUESTION 4

Your company runs end-of-the-month accounting reports. While the reports run, other financial records are updated in the database. Users report that the reports take longer than expected to run. You need to reduce the amount of time it takes for the reports to run. The reports must show committed data only. What should you do?

- A. Use the NOLOCK option.
- B. Execute the BDCC UPDATEUSAGE statement.
- C. Use the max worker threads Option.
- D. Use a table-valued parameter.
- E. Set SET ALLOW_SNAPSHOT_ISOLATION to ON.
- F. Set SET XACT ABORT to ON.
- G. Execute the alter table ti set (Lock_ESCALATION = AUTO); statement.
- H. Use the output parameters.

Answer: G

NEW QUESTION 5

You use Query Store to optimize a query in a database. The query has two execution plans:

Plan 2 is shown in the Plan 2 Execution Plan exhibit.

Plan 10 is shown in the Plan 10 Execution Plan exhibit.

You create an index at 22:24 based on the missing index suggestion in Plan 2.

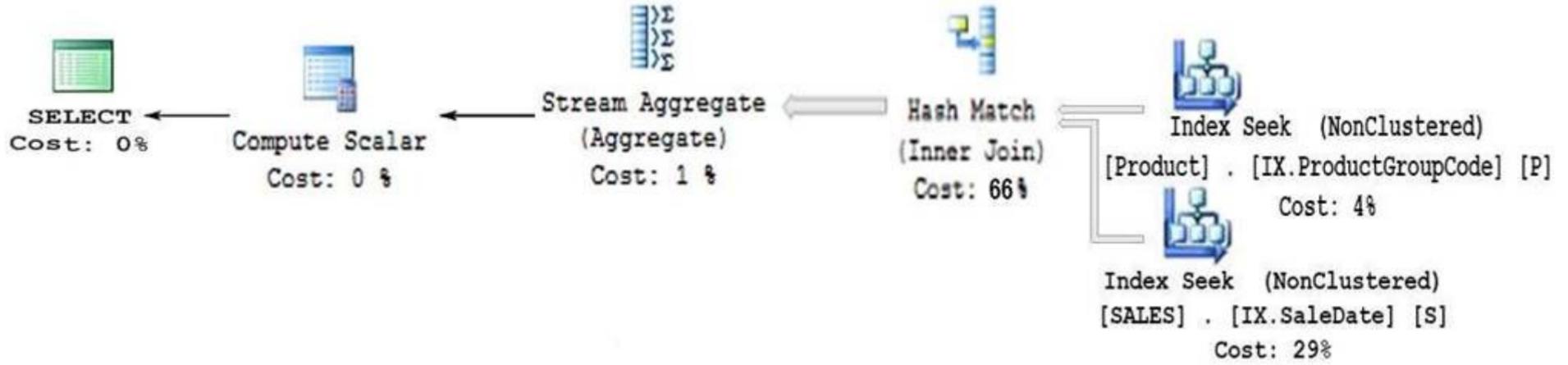
The average duration statistics for the query is shown in the Tracked Queries exhibit. (Click the Exhibit button.)

You need to analyze the operators in the two execution plans.

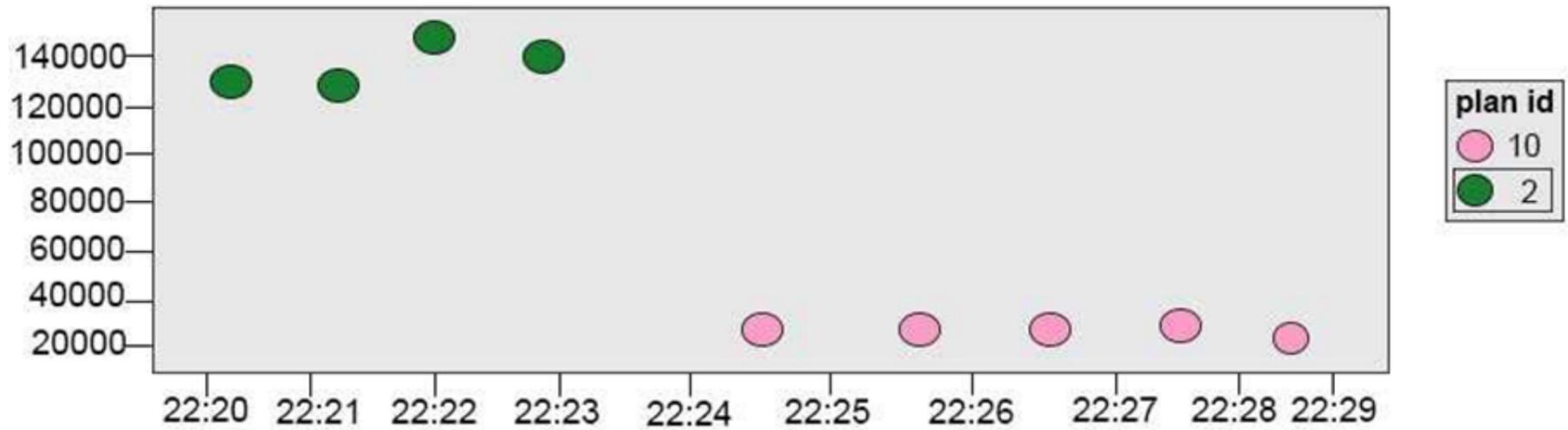
For each of the following statements, select Yes if the statement is true. Otherwise, select No. Plan 2 Execution Plan



Plan 10 Execution Plan



Tracked Queries



Answer Area

Yes

No

The Hash Match operator in Plan 10 processed more data than the operator in Plan 2

The Clustered Index Scan operator in Plan 2 was less efficient than the Nonclustered Index Seek operator in Plan 10

Query throughput will increase by using Plan 10 instead of Plan 2

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Answer Area

	Yes	No
The Hash Match operator in Plan 10 processed more data than the operator in Plan 2	<input checked="" type="radio"/>	<input type="radio"/>
The Clustered Index Scan operator in Plan 2 was less efficient than the Nonclustered Index Seek operator in Plan 10	<input checked="" type="radio"/>	<input type="radio"/>
Query throughput will increase by using Plan 10 instead of Plan 2	<input type="radio"/>	<input checked="" type="radio"/>

NEW QUESTION 6

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You have a database that is 130 GB and contains 500 million rows of data.

Granular transactions and mass batch data imports change the database frequently throughout the day. Microsoft SQL Server Reporting Services (SSRS) uses the database to generate various reports by using several filters.

You discover that some reports time out before they complete. You need to reduce the likelihood that the reports will time out.

Solution: You increase the number of log files for the database. You store the log files across multiple disks. Does this meet the goal?

- A. Yes
- B. No

Answer: A

NEW QUESTION 7

Note: This question is part of a series of questions that use the same or similar answer choices. An Answer choice may be correct for more than one question in the series. Each question independent of the other questions in this series. Information and details provided in a question apply only to that question.

You are a database developer for a company. The company has a server that has multiple physical disks. The disks are not part of a RAID array. The server hosts three Microsoft SQL Server instances. There are many SQL jobs that run during off-peak hours.

You must monitor the SQL Server instances in real time and optimize the server to maximize throughput, response time, and overall SQL performance.

You need to ensure that the performance of each instance is consistent for the same queried and query plans. What should you do?

- A. Create a sys.dm_os_waiting_tasks query.
- B. Create a sys.dm_exec_sessions query.
- C. Create a Performance Monitor Data Collector Set.
- D. Create a sys.dm_os_memory_objects query.
- E. Create a sp_configure 'max server memory' query.
- F. Create a SQL Profiler trace.
- G. Create a sys.dm_os_wait_stats query.
- H. Create an Extended Event.

Answer: H

Explanation:

Advanced Viewing of Target Data from Extended Events in SQL Server

When your event session is currently active, you might want to watch the event data in real time, as it is received by the target.

Management > Extended Events > Sessions > [your-session] > Watch Live Data.

The query_post_execution_showplan extended event enables you to see the actual query plan in the SQL Server Management Studio (SSMS) UI. When the Details pane is visible, you can see a graph of the query plan on the Query Plan tab. By hovering over a node on the query plan, you can see a list of property names and their values for the node.

Clustered Index Seek (Clustered)
 Scanning a particular range of rows from a clustered index.

Physical Operation	Clustered Index Seek
Logical Operation	Clustered Index Seek
Actual Execution Mode	Row
Estimated Execution Mode	Row
Storage	RowStore
Actual Number of Rows	0
Actual Number of Batches	0
Estimated Operator Cost	0.0033263 (4%)
Estimated I/O Cost	0.003125
Estimated Subtree Cost	0.0033263
Estimated CPU Cost	0.0001581
Estimated Number of Executions	1.272901
Number of Executions	1
Estimated Number of Rows	1
Estimated Row Size	139 B
Actual Rebinds	0
Actual Rewinds	0
Ordered	True
Node ID	228

Object
 [InMemTest2].[sys].[sysclsobjs].[clst] [s]

Output List
 [InMemTest2].[sys].[sysclsobjs].name

Seek Predicates
 Seek Keys[1]: Prefix: [InMemTest2].[sys].[sysclsobjs].class,
 [InMemTest2].[sys].[sysclsobjs].id = Scalar Operator((50)),
 Scalar Operator([InMemTest2].[sys].[sysnsobjs].[nsid] as [s],
 [nsid])

References: <https://msdn.microsoft.com/en-us/library/mt752502.aspx>

NEW QUESTION 8

You have a database that contains three encrypted store procedures named dbo.Proc1, dbo.Proc2 and dbo.Proc3. The stored procedures include INSERT, UPDATE, DELETE and BACKUP DATABASE statements.

You have the following requirements:

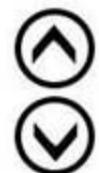
- * You must run all the stored procedures within the same transaction.
- * You must automatically start a transaction when stored procedures include DML statements.
- * You must not automatically start a transaction when stored procedures include DDL statements. You need to run all three stored procedures.

Which four Transact-SQL segments should you use to develop the solution? To answer, move the appropriate Transact-SQL segments to the answer area and arrange them in the correct order.

Transact-SQL segments

- BEGIN CATCH
IF (XACT_STATE() != 0)
ROLLBACK TRANSACTION
END CATCH
- IF (@TRANCOUNT > 0)
ROLLBACK TRANSACTION
- BEGIN TRAN
- EXEC dbo.Proc1
EXEC dbo.Proc2
EXEC dbo.Proc3
- SET IMPLICIT_TRANSACTIONS OFF
- SET IMPLICIT_TRANSACTIONS ON
- COMMIT TRANSACTION
- BEGIN TRY
EXEC dbo.Proc1
EXEC dbo.Proc2
EXEC dbo.Proc3
IF (XACT_STATE() = 1)
COMMIT TRANSACTION;
END TRY

Answer Area



- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Note:

Implicit transaction mode remains in effect until the connection executes a SET IMPLICIT_TRANSACTIONS OFF statement, which returns the connection to autocommit mode. In autocommit mode, all individual statements are committed if they complete successfully.

When a connection is in implicit transaction mode and the connection is not currently in a transaction, executing any of the following statements starts a transaction:

Note 2: XACT_STATE returns the following values.

1 The current request has an active user transaction. The request can perform any actions, including writing data and committing the transaction. The transaction is committable.

-1 The current request has an active user transaction, but an error has occurred that has caused the transaction to be classified as an uncommittable transaction. The transaction is uncommittable and should be rolled back.

0 There is no active user transaction for the current request. A commit or rollback operation would generate an error.

References:

[https://technet.microsoft.com/en-us/library/ms187807\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms187807(v=sql.105).aspx) [https://technet.microsoft.com/en-us/library/ms189797\(v=sql.110\).aspx](https://technet.microsoft.com/en-us/library/ms189797(v=sql.110).aspx)

NEW QUESTION 9

Note: This question is part of a series of questions that use the same scenario. For your convenience, the scenario is repeated in each question. Each question presents a different goal and answer choices, but the text of the scenario is exactly the same in each question in this series.

You have a database named DB1 that contains the following tables: Customer, CustomerToAccountBridge, and CustomerDetails. The three tables are part of the Sales schema. The database also contains a schema named Website. You create the Customer table by running the following Transact-SQL statement:

```
CREATE TABLE Customer
(
    CustomerNumber int NOT NULL,
    CustomerName varchar(50) NOT NULL,
    CreateDate date NOT NULL,
    Gender bit,
    Address varchar(50)
    City varchar(50)
    State char(2),
    CustomerStatus bit NOT NULL,
    MaritalStatus bit,
    Segment varchar(5),
    CountryCode char(2),
    Birthday date,
    PostalCode char(5),
    PhoneNumber varchar(20),
    Account1 char(7),
    Account1Status bit,
    Account2 char(7),
    Account2Status bit,
    CONSTRAINT PK_Customer PRIMARY KEY CLUSTERED (CustomerNumber)
);
```

The value of the CustomerStatus column is equal to one for active customers. The value of the Account1Status and Account2Status columns are equal to one for active accounts. The following table displays selected columns and rows from the Customer table.

Customer ID	CustomerName	Gender	Account1	Account1Status	Account2	Account2Status
101	Name A	0	0001001	0	0001002	1
102	Name B	1	0002001	1	0002002	0
103	Name C	0	0003001	1	0003002	1

You plan to create a view named Website.Customer and a view named Sales.FemaleCustomers. Website.Customer must meet the following requirements:

- * Allow users access to the CustomerName and CustomerNumber columns for active customers.
- * Allow changes to the columns that the view references. Modified data must be visible through the view.
- * Prevent the view from being published as part of Microsoft SQL Server replication. Sales.FemaleCustomers must meet the following requirements:
- * Allow users access to the CustomerName, Address, City, State and PostalCode columns.
- * Prevent changes to the columns that the view references.
- * Only allow updates through the views that adhere to the view filter.

You have the following stored procedures: spDeleteCustAcctRelationship and spUpdateCustomerSummary. The spUpdateCustomerSummary stored procedure was created by running the following Transact-SQL statement:

```
CREATE PROCEDURE uspUpdateCustomerSummary
@CustomerId INT
AS
BEGIN
SET NOCOUNT on;
UPDATE CustomerDetails SET TotalDepositAccountCount = TotalDepositAccountCount + 1 WHERE CustomerID = @CustomerId;
BEGIN TRAN;
BEGIN TRY
UPDATE CustomerDetails SET TotalAccountCount = TotalAccountCount + 1 WHERE CustomerID = @CustomerId;
END TRY
BEGIN CATCH
IF @@TRANCOUNT > 0
ROLLBACK TRAN;
END CATCH
IF @@TRANCOUNT > 0
COMMIT TRAN;
```

You run the spUpdateCustomerSummary stored procedure to make changes to customer account summaries. Other stored procedures call the spDeleteCustAcctRelationship to delete records from the CustomerToAccountBridge table. You need to create Sales.FemaleCustomers.

How should you complete the view definition? To answer, drag the appropriate Transact-SQL segments to the correct locations. Each Transact_SQL segment may be used once, more than once or not at all. You may need to drag the split bar between panes or scroll to view content.

Transact-SQL segments

- WITH SCHEMABINDING
- WITH ENCRYPTION
- WITH CHECK OPTION
- WITH VIEW_METADATA

Answer Area

```
CREATE VIEW Website.Customer
AS SELECT s.CustomerNumber, s.CustomerName
FROM Sales.Customers as s
WHERE s.CustomerStatus=1
```

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Box 1: WITH SCHEMABINDING:

SCHEMABINDING binds the view to the schema of the underlying table or tables. When SCHEMABINDING is specified, the base table or tables cannot be modified in a way that would affect the view definition.

Box 2: WITH CHECK OPTION

CHECK OPTION forces all data modification statements executed against the view to follow the criteria set within select_statement. When a row is modified through a view, the WITH CHECK OPTION makes sure the data remains visible through the view after the modification is committed.

Note: Sales.Female.Customers must meet the following requirements: References: <https://msdn.microsoft.com/en-us/library/ms187956.aspx>

NEW QUESTION 10

Note: This question is part of a series of questions that present the same scenario. Each question in this series contains a unique solution. Determine whether the solution meets the stated goals.

The Account table was created by using the following Transact-SQL statement:

```
CREATE TABLE Account
(
AccountNumber int NOT NULL,
ProductCode char(2) NOT NULL,
Status tinyint NOT NULL,
OpenDate date NOT NULL,
CloseDate date,
Balance decimal(15,2),
AvailableBalance decimal(15,2)
);
```

There are more than 1 billion records in the Account table. The Account Number column uniquely identifies each account. The ProductCode column has 100 different values. The values are evenly distributed in the table. Table statistics are refreshed and up to date.

You frequently run the following Transact-SQL SELECT statements:

```
SELECT ProductCode, SUM(Balance) AS TotalSUM FROM Account WHERE ProductCode
<> 'CD' GROUP BY ProductCode;
SELECT AccountNumber, Balance FROM Account WHERE Production = 'CD'
```

You must avoid table scans when you run the queries. You need to create one or more indexes for the table. Solution: You run the following Transact-SQL statement:

```
CREATE NONCLUSTERED INDEX IX_Account_ProductCode ON Account(ProductCode); Does the solution meet the goal?
```

- A. Yes
- B. No

Answer: A

Explanation:

References: <https://msdn.microsoft.com/en-za/library/ms189280.aspx>

NEW QUESTION 10

Note: This question is part of a series of questions that use the same scenario. For your convenience, the scenario is repeated in each question. Each question presents a different goal and answer choices, but the text of the scenario is exactly the same in each question in this series.

You have a database named DB1 that contains the following tables: Customer, CustomerToAccountBridge, and CustomerDetails. The three tables are part of the Sales schema. The database also contains a schema named Website. You create the Customer table by running the following Transact-SQL statement:

```
CREATE TABLE Customer
(
    CustomerNumber int NOT NULL,
    CustomerName varchar(50) NOT NULL,
    CreateDate date NOT NULL,
    Gender bit,
    Address varchar(50)
    City varchar(50)
    State char(2),
    CustomerStatus bit NOT NULL,
    MaritalStatus bit,
    Segment varchar(5),
    CountryCode char(2),
    Birthday date,
    PostalCode char(5),
    PhoneNumber varchar(20),
    Account1 char(7),
    Account1Status bit,
    Account2 char(7),
    Account2Status bit,
    CONSTRAINT PK_Customer PRIMARY KEY CLUSTERED (CustomerNumber)
);
```

The value of the CustomerStatus column is equal to one for active customers. The value of the Account1Status and Account2Status columns are equal to one for active accounts. The following table displays selected columns and rows from the Customer table.

Customer ID	CustomerName	Gender	Account1	Account1Status	Account2	Account2Status
101	Name A	0	0001001	0	0001002	1
102	Name B	1	0002001	1	0002002	0
103	Name C	0	0003001	1	0003002	1

You plan to create a view named Website.Customer and a view named Sales.FemaleCustomers. Website.Customer must meet the following requirements:

- * Allow users access to the CustomerName and CustomerNumber columns for active customers.
- * Allow changes to the columns that the view references. Modified data must be visible through the view.
- * Prevent the view from being published as part of Microsoft SQL Server replication. Sales.FemaleCustomers must meet the following requirements:
- * Allow users access to the CustomerName, Address, City, State and PostalCode columns.
- * Prevent changes to the columns that the view references.
- * Only allow updates through the views that adhere to the view filter.

You have the following stored procedures: spDeleteCustAcctRelationship and spUpdateCustomerSummary. The spUpdateCustomerSummary stored procedure was created by running the following Transact-SQL statement:

```
CREATE PROCEDURE uspUpdateCustomerSummary
@CustomerId INT
AS
BEGIN
    SET NOCOUNT on;
    UPDATE CustomerDetails SET TotalDepositAccountCount = TotalDepositAccountCount + 1 WHERE CustomerID = @CustomerId;
    BEGIN TRAN;
        BEGIN TRY
            UPDATE CustomerDetails SET TotalAccountCount = TotalAccountCount + 1 WHERE CustomerID = @CustomerId;
        END TRY
        BEGIN CATCH
            IF @@TRANCOUNT > 0
                ROLLBACK TRAN;
        END CATCH
    IF @@TRANCOUNT > 0
        COMMIT TRAN;
```

You run the spUpdateCustomerSummary stored procedure to make changes to customer account summaries. Other stored procedures call the spDeleteCustAcctRelationship to delete records from the CustomerToAccountBridge table.

You need to create Website Customer.

How should you complete the view definition? To answer, drag the appropriate Transact-SQL segments to the correct locations. Each Transact-SQL segment may be used once, more than once or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point. NOTE: Each correct selection is worth one point.

Transact-SQL segments

- WITH SCHEMABINDING
- WITH ENCRYPTION
- WITH CHECK OPTION
- WITH VIEW_METADATA

Answer area

```
CREATE VIEW Website.Customer
    Transact-SQL statement
AS SELECT s.CustomerID, s.CustomerName
FROM Sales.Customer as s
WHERE s.CustomerStatus=1
    Transact-SQL statement
```

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Box 1: WITH ENCRYPTION

Using WITH ENCRYPTION prevents the view from being published as part of SQL Server replication. Box 2: WITH CHECK OPTION
 CHECK OPTION forces all data modification statements executed against the view to follow the criteria set within select_statement. When a row is modified through a view, the WITH CHECK OPTION makes sure the data remains visible through the view after the modification is committed.

Note: Website.Customer must meet the following requirements:

NEW QUESTION 13

Note: This question is part of a series of questions that present the same scenario. Each question in this series contains a unique solution. Determine whether the solution meets the stated goals.

Your company has employees in different regions around the world.

You need to create a database table that stores the following employee attendance information:

- Employee ID
- date and time employee checked in to work
- date and time employee checked out of work

Date and time information must be time zone aware and must not store fractional seconds. Solution: You run the following Transact-SQL statement:

```
CREATE TABLE [dbo].[EmployeeAttendance] (
    EmployeeID int NOT NULL,
    DateChekedIn datetimeoffset NOT NULL,
    DateCheclOut datetimeoffset NOT NULL)
```

Does the solution meet the goal?

- A. Yes
- B. No

Answer: B

Explanation:

Datetimeoffset, not datetimeoffset, defines a date that is combined with a time of a day that has time zone awareness and is based on a 24-hourclock.

Syntax: datetimeoffset [(fractional seconds precision)]

For the use "datetimeoffset", the Fractional seconds precision is 7. References: <https://msdn.microsoft.com/en-us/library/bb630289.aspx>

NEW QUESTION 14

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution. Determine whether the solution meets the stated goals.

You have a database that contains a table named Employees. The table stores information about the employees of your company.

You need to implement and enforce the following business rules:

- Limit the values that are accepted by the Salary column.
- Prevent salaries less than \$15,000 and greater than \$300,000 from being entered.
- Determine valid values by using logical expressions.
- Do not validate data integrity when running DELETE statements.

Solution: You implement cascading referential integrity constraints on the table. Does the solution meet the goal?

- A. Yes
- B. No

Answer: A

Explanation:

References: [https://technet.microsoft.com/en-us/library/ms186973\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms186973(v=sql.105).aspx)

NEW QUESTION 19

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution. Determine whether the solution meets the stated goals.

The Account table was created by using the following Transact-SQL statement:

```
CREATE TABLE Account
(
    AccountNumber int NOT NULL,
    ProductCode char(2) NOT NULL,
    Status tinyint NOT NULL,
    OpenDate date NOT NULL,
    CloseDate date,
    Balance decimal(15,2),
```

There are more than 1 billion records in the Account table. The AccountNuwber column uniquely identifies each account. The productcode column has 100 different values. The values are evenly distributed in the table. Table statistics are refreshed and up to date.

You frequently run the following Transact-SQL select statements:

```
SELECT ProductCode, SUM(Balance) AS TotalSUH FROM Account WHERE ProductCode <> 'CD' GROUP BY ProductCode;
SELECT AccountNupber, Balance FROM Account WHERE ProductCode = 'CD';
```

You must avoid table scans when you run the queries. You need to create one or more indexes for the table. Solution: You run the following Transact-SQL statements:

```
CREATE CLUSTERED INDEX PK_Account ON Account(AccountNumber);
CREATE NONCLUSTERED INDEX IX_Account_ProductCode ON Account(ProductCode) INCLUDE (Balance);
```

Does the solution meet the goal?

- A. Yes
- B. No

Answer: A

NEW QUESTION 21

Note: This question is part of a series of questions that use the same scenario. For your convenience, the scenario is repeated in each question. Each question presents a different goal and answer choices, but the text of the scenario is exactly the same in each question in this series.
 You have a database that contains the following tables: BlogCategory, BlogEntry, ProductReview, Product, and SalesPerson. The tables were created using the following Transact SQL statements:

```
CREATE TABLE BlogCategory
(
    CategoryID int NOT NULL PRIMARY KEY,
    CategoryName nvarchar (20)
);

CREATE TABLE BlogEntry
(
    Entry int NOT PRIMARY KEY,
    Entrytitle nvarchar (50),
    Category int NOT NULL FOREIGN KEY REFERENCES BlogCategory
(CategoryID)
);

CREATE TABLE dbo.ProductReview
(
    ProductReviewID IDENTITY(1,1) PRIMARY KEY,
    Product int NOT NULL,
    Review varchar (1000) NOT NULL
);

CREATE TABLE dbo.Product
(
    ProductID int Identity(1,1) PRIMARY KEY,
    Name varchar(1000) NOT NULL
);

CREATE TABLE dbo.SalesPerson
(
    SalesPersonID int IDENTITY(1,1) PRIMARY KEY,
    Name varchar (1000) NOT NULL,
    SalesID Money
)
```

You must modify the ProductReview Table to meet the following requirements:

- * The table must reference the ProductID column in the Product table
- * Existing records in the ProductReview table must not be validated with the Product table.
- * Deleting records in the Product table must not be allowed if records are referenced by the ProductReview table.
- * Changes to records in the Product table must propagate to the ProductReview table.

You also have the following database tables: Order, ProductTypes, and SalesHistory, The transact-SQL statements for these tables are not available.

You must modify the Orders table to meet the following requirements:

- * Create new rows in the table without granting INSERT permissions to the table.
- * Notify the sales person who places an order whether or not the order was completed.

You must add the following constraints to the SalesHistory table:

- * a constraint on the SaleID column that allows the field to be used as a record identifier
- * a constant that uses the ProductID column to reference the Product column of the ProductTypes table
- * a constraint on the CategoryID column that allows one row with a null value in the column
- * a constraint that limits the SalePrice column to values greater than four

Finance department users must be able to retrieve data from the SalesHistory table for sales persons where the value of the SalesYTD column is above a certain threshold.

You plan to create a memory-optimized table named SalesOrder. The table must meet the following requirements:

- * The table must hold 10 million unique sales orders.
- * The table must use checkpoints to minimize I/O operations and must not use transaction logging.
- * Data loss is acceptable.

Performance for queries against the SalesOrder table that use Where clauses with exact equality operations must be optimized.

You need to update the SalesHistory table

How should you complete the Transact_SQL statement? To answer? select the appropriate Transact-SQL, segments in the answer area.

Answer Area

```
IF OBJECT_id(*SalesHistory*)>0 DROP TABLE SalesHistory
GO
IF OBJECT_ID(*ProductTypes*)>0 DROP TABLE ProductTypes
GO
CREATE TABLE ProductTypes
(
    ProductID SMALLINT,
    ProductDescription VARCHAR(255),
    CONSTRAINT pk_ProductID PRIMARY KEY (ProductID)
)
GO
CREATE TABLE [dbp].[SalesHistoryK]
[SalesID] [int]
```

▼

IDENTITY(1,4)
 IDENTITY(1,4) NOT NULL PRIMARY KEY
 UNIQUE

```
[ProductID] SMALLINT NULL ,
[SalesDate] [datetime] NULL
```

[SalesPrice] [money]

▼

NOT NULL
 NULL CHECK (SalesPrice > 4)
 UNIQUE

[CategoryID] [smallint]

▼

NOT NULL
 NULL CHECK (SalesPrice > 4)
 UNIQUE

▼

CONSTRAINT fk_SalesHistoryProductID FOREIGN KEY (ProductID) REFERENCES SalesHistory(CategoryID)
 CONSTRAINT fk_SalesHistoryProductID FOREIGN KEY (ProductID) REFERENCES ProductTypes(ProductID)

```
)
GO
```

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Box 1:
 SaleID must be the primary key, as a constraint on the SaleID column that allows the field to be used as a record identifier is required.

Box2:
 A constraint that limits the SalePrice column to values greater than four. Box 3: UNIQUE

Box 4:
 A constraint on the CategoryID column that allows one row with a null value in the column.

A foreign key constraint must be put on the productID referencing the ProductTypes table, as a constraint that uses the ProductID column to reference the Product column of the ProductTypes table is required.

Note: Requirements are:
 You must add the following constraints to the SalesHistory table:

NEW QUESTION 22

Note: This question is part of a series of questions that use the same or similar answer choices. An Answer choice may be correct for more than one question in the series. Each question independent of the other questions in this series. Information and details provided in a question apply only to that question.

You are a database developer for a company. The company has a server that has multiple physical disks. The disks are not part of a RAID array. The server hosts three Microsoft SQL Server instances. There are many SQL jobs that run during off-peak hours.

You must monitor the SQL Server instances in real time and optimize the server to maximize throughput, response time, and overall SQL performance. You need to create a baseline set of metrics to report how the computer running SQL Server operates under normal load. The baseline must include the resource usage associated with the server processes. What should you do?

- A. Create a sys.dm_os_waiting_tasks query.
- B. Create a sys.dm_exec_sessions query.
- C. Create a Performance Monitor Data Collector Set.
- D. Create a sys.dm_os_memory_objects query.
- E. Create a sp_configure 'max server memory' query.
- F. Create a SQL Profiler trace.
- G. Create a sys.dm_os_wait_stats query.
- H. Create an Extended Event.

Answer: D

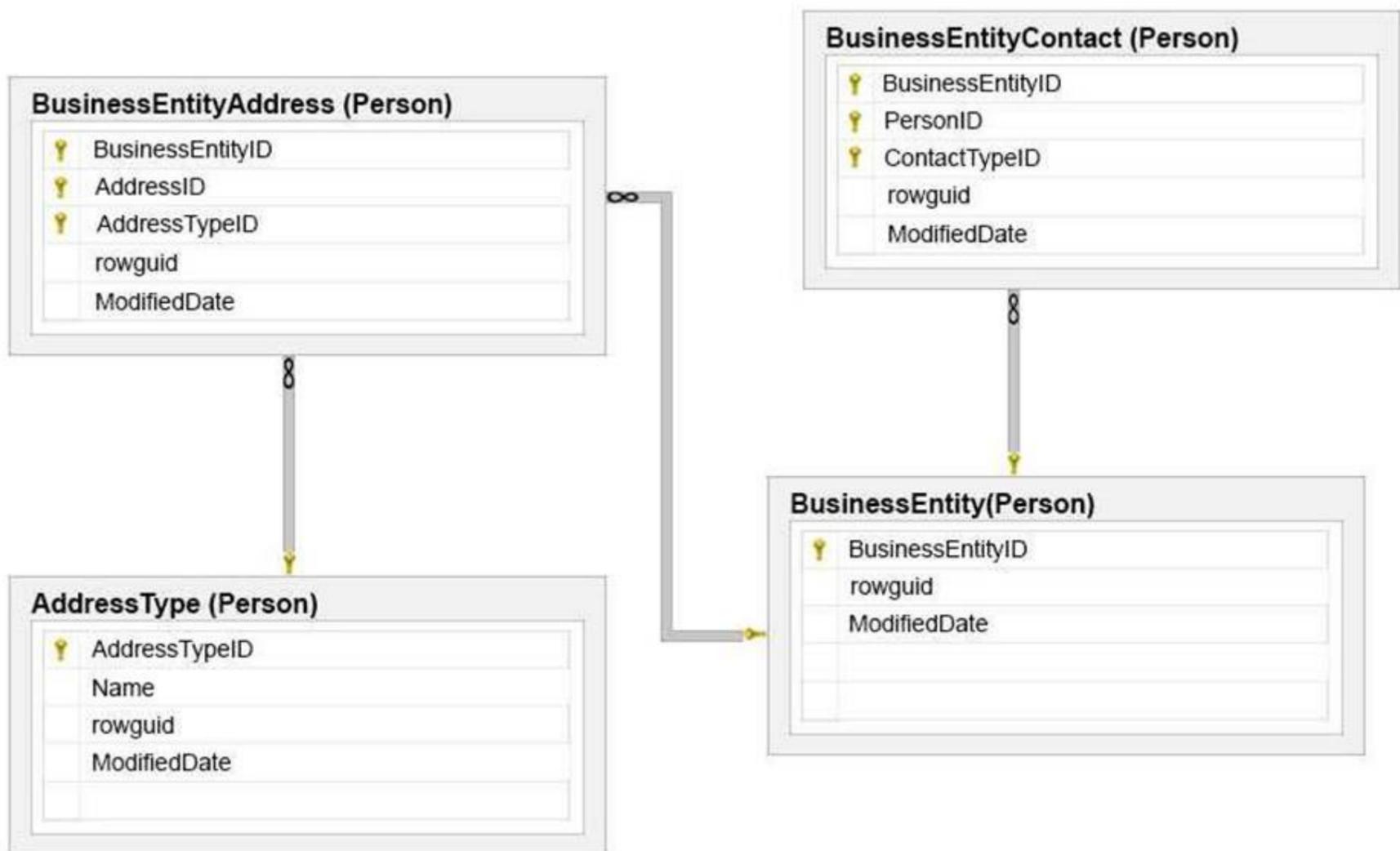
Explanation:

sys.dm_os_memory_objects returns memory objects that are currently allocated by SQL Server. You can use sys.dm_os_memory_objects to analyze memory use and to identify possible memory leaks.

Example: The following example returns the amount of memory allocated by each memory object type. `SELECT SUM (pages_in_bytes) as 'Bytes Used', type FROM sys.dm_os_memory_objects GROUP BY type ORDER BY 'Bytes Used' DESC; GO`

NEW QUESTION 25

You are creating a stored procedure which will insert data into the table shown in the Database schema exhibit. (Click the exhibit button.)



You need to insert a new customer record into the tables as a single unit of work.

In which order should you use the Transact-SQL segments to develop the solution? To answer, move the appropriate Transact-SQL segments to the answer area and arrange them, in the correct order.

NOTE: More than one order of answer choices is correct. You will receive credit for any of the correct orders you select.

Transact-SQL segments

- COMMIT TRANSACTION
- INSERT INTO Person.AddressType
- INSERT INFO Person.BusinessEntityAddress
- INSERT INTO Person. BusinessEntity
- BEGIN TRANSACTION
- INSERT INTO Person.BusinessEntityContact

Answer Area



- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

The entities on the many side, of the 1-many relations, must be added before we add the entities on the 1-side. We must insert new rows into BusinessEntityContact and BusinessEntityAddress tables, before we insert the corresponding rows into the BusinessEntity and AddressType tables.

NEW QUESTION 29

You create a database table named FactSales by running the following Transact-SQL statements:

```
CREATE TABLE FactSales (
DateKey [int] NOT NULL,
Productkey [int] NOT NULL,
CustomerKey [int] NOT NULL,
EmployeeKey [int] NOT NULL,
SalesAmount money,
SalesCost money
INDEX ix_FactSales CLUSTERED (DateKey));

CREATE INDEX ix_FactSales_ProductKey ON FactSales (ProductKey);
CREATE INDEX ix_FactSales_CustomerKey ON FactSales (CustomerKey);
CREATE INDEX ix_FactSales_EmployeeKey ON FactSales (EmployeeKey);
```

You must optimize the indexes without making changes to the ix_FactSales_EmployeeKey index. You need to implement a columnstore index for the table. How should you complete the Transact-SQL code? To answer, select the appropriate Transact-SQL segments in the answer area.
 NOTE: Each correct selection is worth one point.

Answer Area

```
DROP INDEX ix_FactSales_ProductKey ON FactSales
DROP INDEX ix_FactSales_EmployeeKey ON FactSales
DROP INDEX ix_FactSales ON FactSales
```

```
CREATE CLUSTERED COLUMNSTORE INDEX ix FactSales ON FactSales
DROP INDEX ix_FactSales_CustomerKey ON FactSales
DROP INDEX ix_FactSales_EmployeeKey ON FactSales
DROP INDEX ix_FactSales ON FactSales
```

```
CREATE CLUSTERED COLUMNSTORE INDEX ix FactSales ON FactSales
CREATE CLUSTERED COLUMNSTORE INDEX ix_FactSales ON FactSales WITH (DROP_EXISTING = ON)
ALTER INDEX ix_FactSales_EmployeeKey ON FactSales REBUILD
CREATE INDEX ix_FactSales_EmployeeKey ON FactSales (EmployeeKey);
```

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Answer Area

```
DROP INDEX ix_FactSales_ProductKey ON FactSales
DROP INDEX ix_FactSales_EmployeeKey ON FactSales
DROP INDEX ix_FactSales ON FactSales
```

```
CREATE CLUSTERED COLUMNSTORE INDEX ix FactSales ON FactSales
DROP INDEX ix_FactSales_CustomerKey ON FactSales
DROP INDEX ix_FactSales_EmployeeKey ON FactSales
DROP INDEX ix_FactSales ON FactSales
```

```
CREATE CLUSTERED COLUMNSTORE INDEX ix FactSales ON FactSales
CREATE CLUSTERED COLUMNSTORE INDEX ix_FactSales ON FactSales WITH (DROP_EXISTING = ON)
ALTER INDEX ix_FactSales_EmployeeKey ON FactSales REBUILD
CREATE INDEX ix_FactSales_EmployeeKey ON FactSales (EmployeeKey);
```

NEW QUESTION 30

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some questions sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You have a 3-TB database. The database server has 64 CPU cores. You plan to migrate the database to Microsoft Azure SQL Database.

You need to select the service tier for the Azure SQL database. The solution must meet or exceed the current processing capacity.

Solution: You select the Standard service tier. Does this meet the goal?

- A. Yes
- B. No

Answer: B

Explanation:

Premium service is required for 3 TB of storage. Single database DTU and storage limits

	Basic	Standard	Premium
Maximum storage size	2 GB	1 TB	4 TB
Maximum DTUs	5	3000	4000

References: <https://docs.microsoft.com/en-us/azure/sql-database/sql-database-service-tiers-dtu>

NEW QUESTION 34

You suspect deadlocks on a database.

Which two trace flags in the Microsoft SQL Server error log should you locate? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. 1204
- B. 1211
- C. 1222
- D. 2528
- E. 3205

Answer: AC

Explanation:

Trace flag 1204 returns the resources and types of locks participating in a deadlock and also the current command affected.

Trace flag 1222 returns the resources and types of locks that are participating in a deadlock and also the current command affected, in an XML format that does not comply with any XSD schema.

References:

<https://docs.microsoft.com/en-us/sql/t-sql/database-console-commands/dbcc-traceon-trace-flags-transact-sql?view=sql-server-11>

NEW QUESTION 36

Note: This question is part of a series of questions that use the same answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other questions in this series. Information and details provided in a question apply only to that question.

You are developing an application to track customer sales. You create tables to support the application. You need to create a database object that meets the following data entry requirements:

Table name	Data entry requirements
Customers	Inserts must be grouped in a batch of Transact-SQL statements.
CustomerTransactions	Inserts must be grouped in a batch of Transact-SQL statements. Inserts must be grouped in a batch of Transact-SQL statements. All of the data modifications made must become a permanent part of the database if data entry is successful.
Invoices	All data entry modifications must be erased if an error occurs. Inserts must be grouped in a batch of Transact-SQL statements. All of the data modifications made must become a permanent part of the database if data entry is successful.
Orders	All data entry modifications must be erased if an error occurs.

What should you create?

- A. extended procedure
- B. CLR procedure
- C. user-defined procedure
- D. DML trigger
- E. DDL trigger
- F. scalar-valued function
- G. table-valued function

Answer: C

Explanation:

References: <https://msdn.microsoft.com/en-us/library/ms345075.aspx>

NEW QUESTION 38

Note: This question is part of a series of questions that use the same scenario. For your convenience, the scenario is repeated in each question. Each question presents a different goal and answer choices, but the text of the scenario is exactly the same in each question in this series.

You have a database named DB1 that contains the following tables: Customer, CustomerToAccountBridge, and CustomerDetails. The three tables are part of the Sales schema. The database also contains a schema named Website. You create the Customer table by running the following Transact-SQL statement:

```
CREATE TABLE Customer
(
    CustomerNumber int NOT NULL,
    CustomerName varchar(50) NOT NULL,
    CreateDate date NOT NULL,
    Gender bit,
    Address varchar(50)
    City varchar(50)
    State char(2),
    CustomerStatus bit NOT NULL,
    MaritalStatus bit,
    Segment varchar(5),
    CountryCode char(2),
    Birthday date,
    PostalCode char(5),
    PhoneNumber varchar(20),
    Account1 char(7),
    Account1Status bit,
    Account2 char(7),
    Account2Status bit,
    CONSTRAINT PK_Customer PRIMARY KEY CLUSTERED (CustomerNumber)
);
```

The value of the CustomerStatus column is equal to one for active customers. The value of the Account1Status and Account2Status columns are equal to one for active accounts. The following table displays selected columns and rows from the Customer table.

Customer ID	CustomerName	Gender	Account1	Account1Status	Account2	Account2Status
101	Name A	0	0001001	0	0001002	1
102	Name B	1	0002001	1	0002002	0
103	Name C	0	0003001	1	0003002	1

You plan to create a view named Website.Customer and a view named Sales.FemaleCustomers. Website.Customer must meet the following requirements:

- * Allow users access to the CustomerName and CustomerNumber columns for active customers.
- * Allow changes to the columns that the view references. Modified data must be visible through the view.
- * Prevent the view from being published as part of Microsoft SQL Server replication. Sales.FemaleCustomers must meet the following requirements:
- * Allow users access to the CustomerName, Address, City, State and PostalCode columns.
- * Prevent changes to the columns that the view references.
- * Only allow updates through the views that adhere to the view filter.

You have the following stored procedures: spDeleteCustAcctRelationship and spUpdateCustomerSummary. The spUpdateCustomerSummary stored procedure was created by running the following Transact-SQL statement:

```
CREATE PROCEDURE uspUpdateCustomerSummary
@CustomerId INT
AS
BEGIN
    SET NOCOUNT on;
    UPDATE CustomerDetails SET TotalDepositAccountCount = TotalDepositAccountCount + 1 WHERE CustomerID = @CustomerId;
    BEGIN TRAN;
    BEGIN TRY
        UPDATE CustomerDetails SET TotalAccountCount = TotalAccountCount + 1 WHERE CustomerID = @CustomerId;
    END TRY
    BEGIN CATCH
        IF @@TRANCOUNT > 0
            ROLLBACK TRAN;
    END CATCH
    IF @@TRANCOUNT > 0
        COMMIT TRAN;
```

You run the spUpdateCustomerSummary stored procedure to make changes to customer account summaries. Other stored procedures call the spDeleteCustAcctRelationship to delete records from the CustomerToAccountBridge table.

You must update the design of the Customer table to meet the following requirements.

- * You must be able to store up to 50 accounts for each customer.
- * Users must be able to retrieve customer information by supplying an account number.
- * Users must be able to retrieve an account number by supplying customer information.

Which three Transact-SQL segments should you use to develop the solution? To answer, move the appropriate code blocks from the list of code blocks to the answer area and arrange them in the correct order.

Transact-SQL segments

Answer Area

```
WHERE CustomerStatus <> 1;

ON Customer(CreateDate, CustomerStatus) INCLUDE
(Segment, ContryCode, PhoneNumber)

WHERE CustomerID = 1;

CREATE NONCLUSTERED INDEX IX_Customer_CreatedDate

ON Customer(CreateDate) INCLUDE (Segment, Country-
Code, PhoneNumber)

CREATE CLUSTERED INDEX IX_Customer_CreatedDate
```



- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Box 1: Clustered Index

With the same size of keys, the nonclustered indexes need more space than clustered indexes. Box 2, Box 3: Include the CustomerStatus column in the index, and only when CustomerStatus not equal to 1 (the active customers).

References:

<http://www.sqlserverlogexplorer.com/overview-of-cluster-and-noncluster-index/>

NEW QUESTION 42

You have a table named Person.Address that includes the following columns:

- AddressID
- AddressLine1
- AddressLine2
- City
- StateProvinceID
- PostalCode
- RowGuid
- ModifiedDate

You need to create a nonclustered index on PostalCode named IX_Address_PostalCode that uses the following included columns:

- AddressLine1
- AddressLine2
- City
- StateProvinceID

How should you complete the Transact-SQL statement? To answer, select the appropriate Transact_SQL segments in the answer are.

Answer Area

```
CREATE NONCLUSTERED [ ] IX_Address_PostalCode
[ ]
ON [ ]
INCLUDE [ ]
INDEX [ ]
WHERE [ ]
(Person.Address (PostalCode
(AddressLine1, AddressLine2, City, StateProvinceID)
```

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Box 1: INDEX

Box 2: ON

Box 3: INCLUDE

INCLUDE (column [,... n]) specifies the non-key columns to be added to the leaf level of the nonclustered index. The nonclustered index can be unique or non-unique.

References:

<https://docs.microsoft.com/en-us/sql/t-sql/statements/create-index-transact-sql?view=sql-server-2017>

NEW QUESTION 43

Note: The question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other question in the series. Information and details provided in a question apply only to that question.

You have a database named DB1. The database does not use a memory-optimized filegroup. The database contains a table named Table1. The table must support the following workloads:

Workload	Type	Description
Reporting	Existing	The reporting workload must scan most of the records in the table to aggregate on a number of columns. A clustered columnstore index is already created on the table to support this workload.
OLTP	New	The OLTP workload must support 3,000 transactions per second. Rows are identified by using two columns. The filter is variant on one of the two columns while constant on the other. Only a small number of records with a few columns are returned by the query.

You need to add the most efficient index to support the new OLTP workload, while not deteriorating the existing Reporting query performance. What should you do?

- A. Create a clustered index on the table.
- B. Create a nonclustered index on the table.
- C. Create a nonclustered filtered index on the table.
- D. Create a clusteredcolumnstore index on the table.
- E. Create a nonclustered columnstore index on the table.
- F. Create a hash index on the table.

Answer: C

Explanation:

A filtered index is an optimized nonclustered index, especially suited to cover queries that select from a well-defined subset of data. It uses a filter predicate to index a portion of rows in the table. A well-designed filtered index can improve query performance, reduce index maintenance costs, and reduce index storage costs compared with full-table indexes.

References: [https://technet.microsoft.com/en-us/library/cc280372\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/cc280372(v=sql.105).aspx)

NEW QUESTION 46

You run the following Transact-SQL statement:

```
CREATE TABLE OrderLines(
    OrderLineID INT NOT NULL IDENTITY PRIMARY KEY CLUSTERED,
    OrderID INT NOT NULL,
    StockItemID INT NOT NULL,
    Description NVARCHAR(100) NOT NULL,
    Quantity INT NOT NULL,
    UnitPrice DECIMAL(18, 2) NULL
)
```

There are multiple unique OrderID values. Most of the UnitPrice values for the same OrderID are different. You need to create a single index seek query that does not use the following operators:

- *Nested loop
- *Sort
- *Key lookup

- A. CREATE INDEX IX_OrderLines_I ON OrderLines (OrderID, UnitPrice) INCLUDE(Description, Quantity)
- B. CREATE INDEX IX_OrderLines_I ON OrderLines (OrderID, UnitPrice) INCLUOE(Quantity)
- C. CREATE INDEX IX_OrderLines_I ON OrderLines (OrderID, UnitPrice, Quantity)
- D. CREATE INDEX IX_OrderLines_I ON OrderLines (UnitPrice, OrderID) INCLUDE(Description, Quantity)

Answer: A

NEW QUESTION 47

Note: this question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other questions in the series. Information and details provided in a question apply only to that question.

You are developing an application to track customer sales.

You need to create a database object that meets the following requirements:

- Return a value of 0 if data is inserted successfully into the Customers table.
- Return a value of 1 if data is not inserted successfully into the Customers table.
- Support logic that is written by using managed code.
- Support TRY...CATCH error handling. What should you create?

- A. extended procedure
- B. CLR procedure
- C. user-defined procedure
- D. DML trigger
- E. scalar-valued function
- F. table-valued function

Answer: D

Explanation:

DML triggers is a special type of stored procedure that automatically takes effect when a data manipulation language (DML) event takes place that affects the table or view defined in the trigger. DML events include INSERT, UPDATE, or DELETE statements. DML triggers can be used to enforce business rules and data integrity, query other tables, and include complex Transact-SQL statements.

References: <https://msdn.microsoft.com/en-us/library/ms178110.aspx>

NEW QUESTION 50

Note: this question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other questions in the series. Information and details provided in a question apply only to that question.

You are developing an application to track customer sales.

You need to create a database object that meets the following requirements:

- Launch when table data is modified.
- Evaluate the state a table before and after a data modification and take action based on the difference.
- Prevent malicious or incorrect table data operations.
- Prevent changes that violate referential integrity by cancelling the attempted data modification.
- Run managed code packaged in an assembly that is created in the Microsoft.NET Framework and located into Microsoft SQL Server.

What should you create?

- A. extended procedure
- B. CLR procedure
- C. user-defined procedure
- D. DML trigger
- E. scalar-valued function
- F. table-valued function

Answer: B

Explanation:

You can create a database object inside SQL Server that is programmed in an assembly created in the Microsoft .NET Framework common language runtime (CLR). Database objects that can leverage the rich programming model provided by the CLR include DML triggers, DDL triggers, stored procedures, functions, aggregate functions, and types.

Creating a CLR trigger (DML or DDL) in SQL Server involves the following steps:

Define the trigger as a class in a .NETFramework-supported language. For more information about how to program triggers in the CLR, see CLR Triggers. Then, compile the class to build an assembly in the .NET Framework using the appropriate language compiler.

Register the assembly in SQL Server using the CREATE ASSEMBLY statement. For more information about assemblies in SQL Server, see Assemblies (Database Engine). Create the trigger that references the registered assembly.

References: <https://msdn.microsoft.com/en-us/library/ms179562.aspx>

NEW QUESTION 55

You have the following stored procedure:

```
CREATE PROCEDURE AddNextNumber @Number INT
AS
BEGIN
    SET ANSI_DEFAULTS ON
    INSERT INTO Numbers (Number) VALUES (@Number)
END
```

The Numbers table becomes unavailable when you run the stored procedure. The stored procedure obtains an exclusive lock on the table and does not release the lock.

What are two possible ways to resolve the issue? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

- A. Remove the implicit transaction and the SET ANSI_DEFAULTS ON statement.
- B. Set the ANSI_DEFAULT statement to OFF and add a COMMIT TRANSACTION statement after the INSERT statement.
- C. Add a COMMIT TRANSACTION statement after the INSERT statement.
- D. Remove the SET ANSI DEFAULTS ON statement.

Answer: B

NEW QUESTION 56

You are experiencing performance issues with the database server.

You need to evaluate schema locking issues, plan cache memory pressure points, and backup I/O problems. What should you create?

- A. a System Monitor report
- B. a sys.dm_exec_query_stats dynamic management view query
- C. a sys.dm_exec_session_wait_stats dynamicmanagement view query
- D. an Activity Monitor session in Microsoft SQL Management Studio.

Answer: C

Explanation:

sys.dm_exec_session_wait_stats returns information about all the waits encountered by threads that executed for each session. You can use this view to diagnose performance issues with the SQL Server session and also with specific queries and batches.

Note: SQL Server wait stats are, at their highest conceptual level, grouped into two broad categories: signal waits and resource waits. A signal wait is accumulated by processes running on SQL Server which are waiting for a CPU to become available (so called because the process has "signaled" that it is ready for processing). A resource wait is accumulated by processes running on SQL Server which are waiting for a specific resource to become available, such as waiting for the release of a lock on a specific record.

NEW QUESTION 57

You are designing a solution for a company that operates retail stores. Each store has a database that tracks sales transactions. You create a summary table in the database at the corporate office. You plan to use the table to record the quantity of each product sold at each store on each day. Managers will use this data to identify reorder levels for products.

Every evening, stores must transmit sales data to the corporate office. The data must be inserted into the summary table that includes the StoreID, ProductID, Qtysold, Totprodsales, and Datesold Columns.

You need to prevent duplicate rows in the summary table. Each row must uniquely identify the store that sold the product and the total amount sold for that store on a specific date.

What should you include in your solution?

- A. Create a unique constraint.
- B. Create a foreign key constraint to the storeID column in each of the store tables.
- C. Create a rule and bind it to the storeID column.
- D. Create a check constraint.

Answer: B

NEW QUESTION 61

Note: This question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other questions in this series. Information and details provided in a question apply only to that question.

You have a Microsoft SQL Server database named DB1 that contains the following tables:

Table name	Description
TBL1	<ul style="list-style-type: none"> • Column1 is configured as the primary key. • Column2 will store the year. • TBL1 only stores data for the year 2016. • The table will contain 1 million records.
TBL2	<ul style="list-style-type: none"> • Column1 is configured as the primary key. • Column2 will store the year. • TBL2 only stores data for the year 2015. • The table will contain 1 million records.

Users frequently run the following query. The users report that the query takes a long time to return results.

```
SELECT Column1, Column2, Column3
FROM (
    SELECT Column1, Column2, Column3
    FROM TBL1
    UNION ALL
    SELECT Column1, Column2, Column3
    FROM TBL2)
WHERE Column2 = <year> AND Column3 = 1
```

You need to minimize the amount of time required for the query to return data. What should you do?

- A. Create clustered indexes on TBL1 and TBL2.
- B. Create a clustered index on TBL1. Create a nonclustered index on TBL2 and add the most frequently queried columns as included columns.
- C. Create a nonclustered index on TBL2 only.
- D. Create unique constraints on both TBL1 and TBL2. Create a partitioned view that combines columns from TBL1 and TBL2.
- E. Drop existing indexes on TBL1 and then create a clustered columnstore index.
- F. Create a nonclustered columnstore index on TBL1. Create a nonclustered index on TBL2.
- G. Drop existing indexes on TBL1 and then create a clustered columnstore index.
- H. Create a nonclustered columnstore index on TBL1.
- I. Make no changes to TBL2.
- J. Create CHECK constraints on both TBL1 and TBL2. Create a partitioned view that combines columns from TBL1 and TBL2.
- K. Create an indexed view that combines columns from TBL1 and TBL2.

Answer: H

NEW QUESTION 66

Note: this question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other questions in the series. Information and details provided in a question apply only to that question.

You are developing an application to track customer sales.

You need to create a database object that meets the following requirements:

Launch when table data is modified.

Evaluate the state of a table before and after a data modification and take action based on the difference. Prevent malicious or incorrect table data operations.

Prevent changes that violate referential integrity by cancelling the attempted data modification.

Run managed code packaged in an assembly that is created in the Microsoft.NET Framework and located into Microsoft SQL Server.

What should you create?

- A. extended procedure
- B. CLR procedure
- C. user-defined procedure

- D. DDL trigger
- E. scalar-valued function
- F. table-valued function

Answer: B

Explanation:

You can create a database object inside SQL Server that is programmed in an assembly created in the Microsoft .NET Framework common language runtime (CLR). Database objects that can leverage the rich programming model provided by the CLR include DML triggers, DDL triggers, stored procedures, functions, aggregate functions, and types.

Creating a CLR trigger (DML or DDL) in SQL Server involves the following steps:

Define the trigger as a class in a .NETFramework-supported language. For more information about how to program triggers in the CLR, see CLR Triggers. Then, compile the class to build an assembly in the .NET Framework using the appropriate language compiler.

Register the assembly in SQL Server using the CREATE ASSEMBLY statement. For more information about assemblies in SQL Server, see Assemblies (Database Engine). Create the trigger that references the registered assembly.

References: <https://msdn.microsoft.com/en-us/library/ms179562.aspx>

NEW QUESTION 68

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some questions sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You have a database that is 130 GB and contains 500 million rows of data.

Granular transactions and mass batch data imports change the database frequently throughout the day. Microsoft SQL Server Reporting Services (SSRS) uses the database to generate various reports by using several filters.

You discover that some reports time out before they complete. You need to reduce the likelihood that the reports will time out.

Solution: You create a file group for the indexes and a file group for the data files. You store the files for each file group on separate disks.

Does this meet the goal?

- A. Yes
- B. No

Answer: A

Explanation:

Consider creating two additional File Groups: Tables and Indexes. It is best not to put your stuff in PRIMARY as that is where SQL SERVER stores all of its data and meta-data about your objects. You create your Table and Clustered Index (as that is the data for the table) on [Tables] and all Non-Clustered indexes on [Indexes].

NEW QUESTION 71

You are analyzing the memory usage of a Microsoft SQL Server instance. You need to obtain the information described on the following table.

Requirement	Details
Requirement 1	Total amount of memory currently used by SQL Server
Requirement 2	Total amount of memory required by SQL Server for running processers efficiently
Requirement 3	Total amount of memory used by a process

Which performance counter should you use for each requirement? To answer, drag the appropriate performance counters to the correct requirements. Each performance counter may be used once, more than once or not at all. You may need to drag the split bat between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

Transact-SQL segments

- Memory: Available Bytes
- SQL Server: Memory Manager: SQL Cache Memory (KB)
- SQL Server: Buffer Manager: Page reads/sec
- SQL Server: Memory Manager: Total Server Memory (KB)
- SQL Server: Memory Manager: Target Server Memory (KB)
- SQL Server: Memory Manager: Granted Workspace Memory (KB)
- SQL Server: Memory Manager: Maximum Workspace Memory (KB)
- Process: working Set

Answer Area

Requirement	Performancr counter
Requirement 1	Performancr counter
Requirement 2	Performancr counter
Requirement 3	Performancr counter

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Requirement1: SQL Server: Memory Manager: Total Server Memory (KB)

This counter specifies the amount of memory the server has committed using the memory manager. Requirement2: SQL Server: Memory Manager: Granted Workspace Memory (KB)

Specifies the total amount of memory currently granted to executing processes, such as hash, sort, bulk copy, and index creation operations.

Requirement3: Process: working Set

Each time a process is created, it reserves the minimum working set size for the process. The virtual memory manager attempts to keep enough memory for the minimum working set resident when the process is active, but keeps no more than the maximum size.

References:

<https://msdn.microsoft.com/en-us/library/ms190924.aspx> <https://blogs.technet.microsoft.com/askperf/2007/05/18/sql-and-the-working-set/>

NEW QUESTION 72

Note: This question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other questions in this series. Information and details provided in a question apply only to that question.

You have a Microsoft SQL Server database named DB1 that contains the following tables:

Table Name	Description
TBL1	<ul style="list-style-type: none"> The table will contain 10 million records. The frequency of inserting, updating, and deleting records is low.
TBL2	<ul style="list-style-type: none"> The table will contain 1 million records.

Users frequently run the following query:

```
SELECT TBL1.Column2, TBL2.Column2, SUM(TBL1.Column3), SUM(TBL1.Column4 * TBL1.Column5 * (TBL2.Column4 - TBL2.Column3))
FROM TBL1
INNER JOIN TBL2 TBL1.Column1 = TBL2.Column1
GROUP BY TBL1.Column2, TBL2.Column2
```

Users report that the query takes a long time to return results.

You need to minimize the amount of time required for the query to return data. What should you do?

- A. Create clustered indexes on TBL1 and TBL2.
- B. Create a clustered index on TBL1. Create a nonclustered index on TBL2 and add the most frequently queried columns as included columns.
- C. Create a nonclustered index on TBL2 only.
- D. Create unique constraints on both TBL1 and TBL2. Create a partitioned view that combines columns from TBL1 and TBL2.
- E. Drop existing indexes on TBL1 and then create a clustered columnstore index on TBL1.
- F. Create a nonclustered columnstore index on TBL1. Create a nonclustered index on TBL2.
- G. Drop existing indexes on TBL1 and then create a clustered columnstore index on TBL1.
- H. Create a nonclustered columnstore index on TBL1.
- I. Make no changes to TBL2.
- J. Create CHECK constraints on both TBL1 and TBL2. Create a partitioned view that combines columns from TBL1 and TBL2.
- K. Create an indexed view that combines columns from TBL1 and TBL2.

Answer: D

NEW QUESTION 73

You maintain a Microsoft Azure SQL Database instance.

You grant User1 the SELECT and EXECUTE permissions for all objects in the dbo schema. You must create a stored procedure that allows User1 to view the following information:

details for each connection to the database

a list of all active user connections and internal tasks

You need to create the stored procedure for User1 and ensure that User1 can run the stored procedure without any error.

How should you complete the Transact-SQL statements? To answer, drag the appropriate Transact-SQL segments to the correct locations. Each Transact-SQL segment may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: More than one combination of answer choices is correct. You will receive credit for any of the correct combinations you select.

Transact-SQL segments

- GRANT SELECT ON SCHEMA: :sys TO User1
- GRANT VIEW DATABASE STATE TO User1
- GRANT VIEW DEFINITION TO User1
- SELECT = FROM sys.dm_exec_connections
- SELECT = FROM sys.dm_exec_requests
- SELECT = FROM sys.dm_exec_sessions

Answer Area

CREATE PROCEDURE ViewConnections

AS [Transact-SQL segment]

[Transact-SQL segment]

GO

[Transact-SQL segment]

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Box 1: Sys.dm_exec_connections

sys.dm_exec_connections returns information about the connections established to this instance of SQL Server and the details of each connection. Returns server wide connection information for SQL Server. Returns current database connection information for SQL Database.

Box 2: sys.dm_exec_sessions

sys.dm_exec_sessions returns one row per authenticated session on SQL Server. sys.dm_exec_sessions is a server-scope view that shows information about all active user connections and internal tasks.

Box 3: GRANT VIEW DATABASE STATE To User1

SQL Database: Requires VIEW DATABASE STATE to see all connections to the current database. VIEW DATABASE STATE cannot be granted in the master database.

NEW QUESTION 75

You have a database that is experiencing deadlock issues when users run queries. You need to ensure that all deadlocks are recorded in XML format. What should you do?

- A. Create a Microsoft SQL Server Integration Services package that uses sys.dm_tran_locks.
- B. Enable trace flag 1224 by using the Database Consistency Checker(BDCC).
- C. Enable trace flag 1222 in the startup options for Microsoft SQL Server.
- D. Use the Microsoft SQL Server Profiler Lock:Deadlock event class.

Answer: C

Explanation:

When deadlocks occur, trace flag 1204 and trace flag 1222 return information that is captured in the SQL Server error log. Trace flag 1204 reports deadlock information formatted by each node involved in the deadlock. Trace flag 1222 formats deadlock information, first by processes and then by resources.

The output format for Trace Flag 1222 only returns information in an XML-like format. References: [https://technet.microsoft.com/en-us/library/ms178104\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms178104(v=sql.105).aspx)

NEW QUESTION 77

You run the following Transact-SQL following statement:

```
CREATE TABLE Customer(  
    CustomerId INT IDENTITY (1, 1) PRIMARY KEY,  
    Code CHAR(5) NOT NULL,  
    FirstName VARCHAR (50) NOT NULL,  
    LastName VARCHAR (50) NOT NULL  
)
```

Customer records may be inserted individually or in bulk from an application. You observe that the application attempts to insert duplicate records.

You must ensure that duplicate records are not inserted and bulk insert operations continue without notifications.

Which Transact-SQL statement should you run?

- A. CREATE UNIQUE NONCLUSTERED INDEX IX_Customer_Code ON Customer (Code) WITH(OFFLINE = OFF)
- B. CREATE UNIQUE INDEX IX_CUSTOMER_Code ON Customer (Code) WITH (IGNORE_DUP_KEY= ON)
- C. CREATE UNIQUE INDEX IX_Customer_Code ON Customer (Code) WITH (IGNORE_DUP_KEY=OFF)
- D. CREATE UNIQUE NONCLUSTERED INDEX IX_Customer_Code ON Customer (Code)
- E. CREATE UNIQUE NONCLUSTERED INDEX IX_Customer_Code ON Customer (Code) WITH (ONLINE = ON)

Answer: B

Explanation:

IGNORE_DUP_KEY = { ON | OFF } specifies the error response when an insert operation attempts to insert duplicate key values into a unique index. The IGNORE_DUP_KEY option applies only to insert operations after the index is created or rebuilt. The option has no effect when executing CREATE INDEX, ALTER INDEX, or UPDATE. The default is OFF.

NEW QUESTION 79

You are maintaining statistics for a database table named tblTransaction. The table contains more than 10 million records.

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

- On weekdays, update statistics for a sample of the total number of records in the table.
- On weekends, update statistics by sampling all rows in the table. A maintenance task will call this stored procedure daily.

How should you complete the stored procedure? To answer, select the appropriate Transact-SQL segments in the answer area.

NOTE: Each correct selection is worth one point.

```

CREATE PROCEDURE uspUpdateTxnStats
AS
BEGIN
    SET NOCOUNT ON;
    SET DATEFIRST 1;
    DECLARE @isWeekDay bit;
    SELECT @isWeekday = CASE WHEN DATEPART (dw, GETDATE()) <=5 THEN 1 ELSE 0 END;
    IF @isWEEKday = 1
        BEGIN
            UPDATE STATISTICS
            SET STATISTICS
            UPDATE TOP(20) STATISTICS
            UPDATE #STATISTICS

            tblTransaction
            WITH FULLSCAN
            WITH RESAMPLE
            WITH SAMPLE 20 PERCENT
            WITH SAMPLE 200000 ROWS
            SELECT TOP(20) PERCENT

        END
    ELSE
        BEGIN
            UPDATE STATISTICS
            SET STATISTICS
            UPDATE TOP(20) STATISTICS
            UPDATE #STATISTICS

            tblTransaction
            WITH FULLSCAN
            WITH RESAMPLE
            WITH SAMPLE 20 PERCENT
            WITH SAMPLE 200000 ROWS
            SELECT TOP(20) PERCENT

        END
    END;
END;

```

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Box 1: UPDATE STATISTICS Box 2: SAMPLE 20 PERCENT

UPDATE STATISTICS tablenameSAMPLE number { PERCENT | ROWS }

Specifies the approximate percentage or number of rows in the table or indexed view for the query optimizer to use when it updates statistics. For PERCENT, number can be from 0 through 100 and for ROWS, number can be from 0 to the total number of rows.

Box 3: UPDATE STATISTICS Box 4: WITH FULLSCAN

FULLSCAN computes statistics by scanning all rows in the table or indexed view. FULLSCAN and SAMPLE 100 PERCENT have the same results. FULLSCAN cannot be used with the SAMPLE option.

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

NEW QUESTION 80

.....

Thank You for Trying Our Product

We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questions and Answers in PDF Format

70-762 Practice Exam Features:

- * 70-762 Questions and Answers Updated Frequently
- * 70-762 Practice Questions Verified by Expert Senior Certified Staff
- * 70-762 Most Realistic Questions that Guarantee you a Pass on Your First Try
- * 70-762 Practice Test Questions in Multiple Choice Formats and Updates for 1 Year

100% Actual & Verified — Instant Download, Please Click
[Order The 70-762 Practice Test Here](#)