DUMPSARENA

Developing SQL Databases

Microsoft 70-762

Version Demo

Total Demo Questions: 10

Total Premium Questions: 181

Buy Premium PDF

https://dumpsarena.co

sales@dumpsarena.co

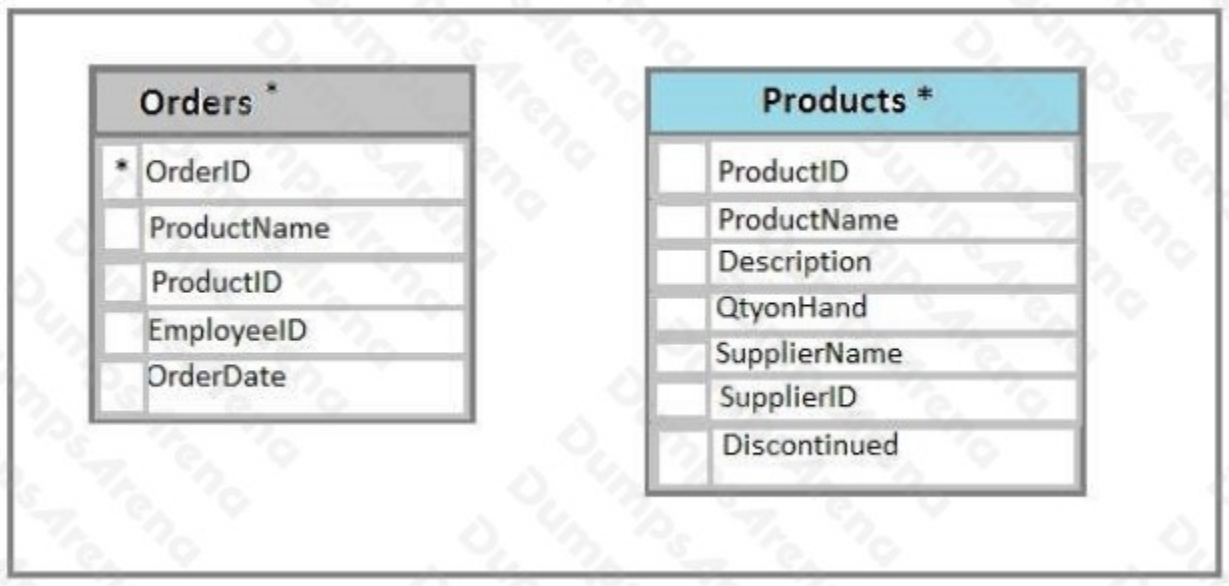
sales@dumpsarena.co
dumpsarena.co

QUESTION NO: 1 - (HOTSPOT)

HOTSPOT

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.

You need to modify the database design to meet the following requirements:

- Rows in the Orders table must always have a valid value for the ProductID column.

- Rows in the Products table must not be deleted if they arepart of any rows in the Orders table. - All rows in both tables must be unique.

In the table below, identify the constraint that must be configured for each table.

NOTE: Make only one selection in each column.

Hot Area:

ANSWER:

Explanation:

A FOREIGN KEY in one table points to a PRIMARY KEY in another table. Here the foreign key constraint is put on the ProductID in the Orders, and points to the ProductID of the Products table.

With a check constraint on the ProductID we can ensure that the Products table contains only unique rows. References: http://www.w3schools.com/sql/sql_foreignkey.asp

QUESTION NO: 2

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: C D

Explanation:

SET ANSI_DEFAULTS is a server-side setting that the client does not modify. When enabled (ON), this option enables SET IMPLICIT_TRANSACTIONS (and some other options).

The SET IMPLICIT_TRANSACTIONS, when ON, the system is in implicit transaction mode.

This means that if @@TRANCOUNT = 0, any of the following Transact-SQL statements begins a new transaction. It is equivalent to an unseen BEGIN TRANSACTION being executed first: ALTER TABLE, FETCH, REVOKE, BEGIN TRANSACTION, GRANT, SELECT, CREATE, INSERT, TRUNCATE TABLE, DELETE, OPEN, UPDATE, DROP.

References: https://docs.microsoft.com/en-us/sql/t-sql/statements/set-implicit-transactions-transact-sql?view=sql-server-2017

QUESTION NO: 3 - (DRAG DROP)

DRAG DROP

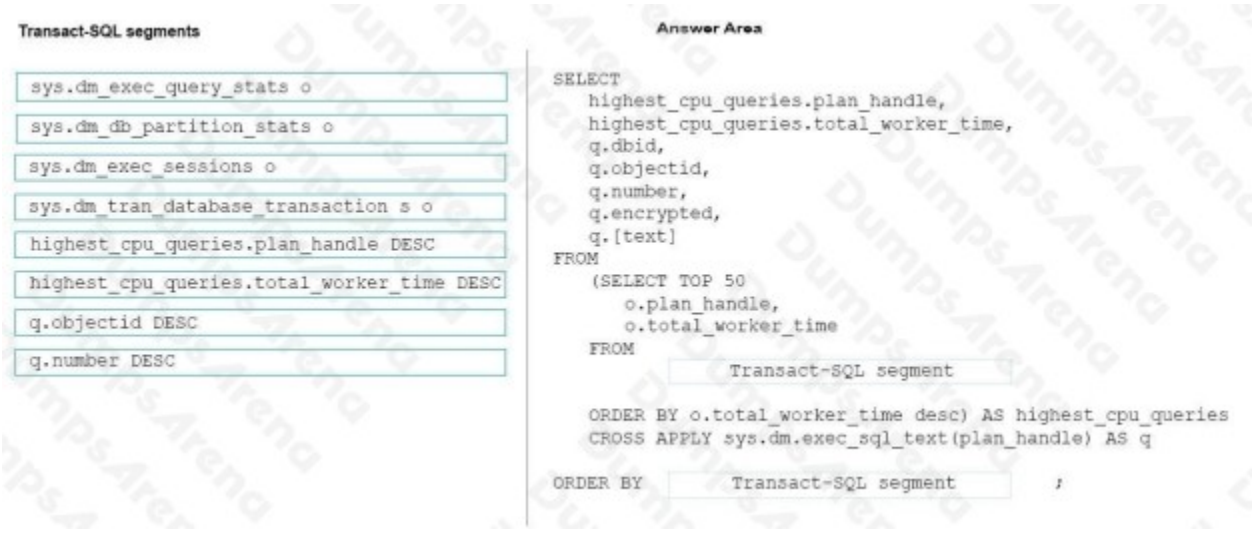
You are monitoring a Microsoft Azure SQL Database.

The database is experiencing high CPU consumption.

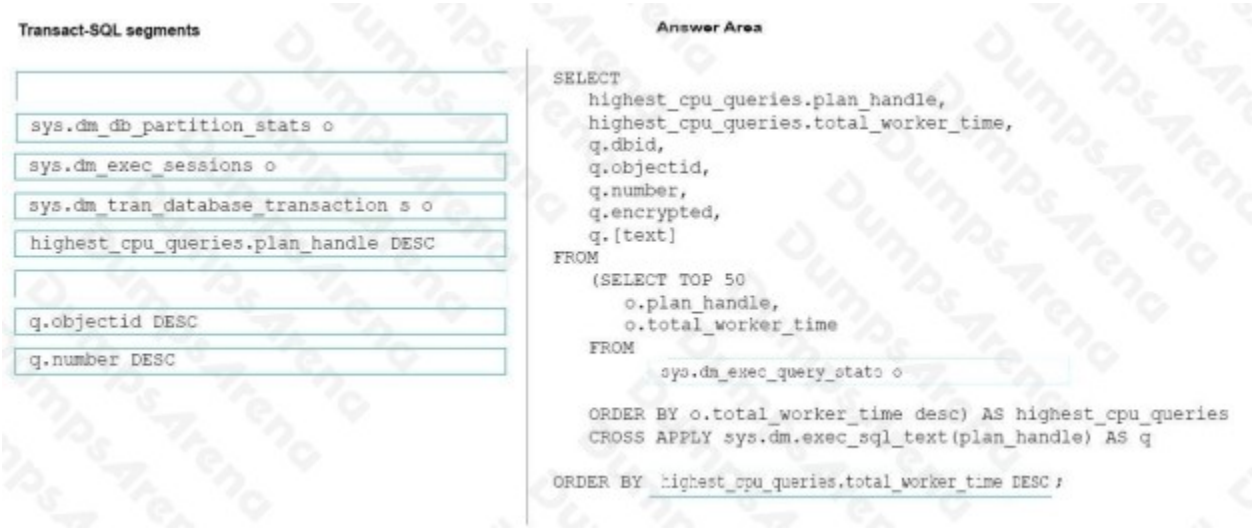
You need to determine which query uses the most cumulative CPU.

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

Select and Place:



ANSWER:



Explanation:

Box 1: sys.dm_exec_query_stats sys.dm_exec_query_stats returns aggregateperformance statistics for cached query plans in SQL Server.

Box 2: highest_cpu_queries.total_worker_time DESC Sort on total_worker_time column

Example: The following example returns information about the top five queries ranked by average CPU time.

Thisexample aggregates the queries according to their query hash so that logically equivalentqueries are grouped by their cumulative resource consumption.

USE AdventureWorks2012;

GO

SELECT TOP 5 query_stats.query_hash AS "Query Hash",

SUM(query_stats.total_worker_time) / SUM(query_stats.execution_count) AS "Avg CPU Time",
MIN(query_stats.statement_text) AS "Statement Text"

FROM

(SELECT QS.*,

SUBSTRING(ST.text, (QS.statement_start_offset/2) + 1,

((CASE statement_end_offset

WHEN -1 THEN DATALENGTH(ST.text)

ELSE QS.statement_end_offset END

- QS.statement_start_offset)/2) + 1) AS statement_text

FROM sys.dm_exec_query_stats AS QS

CROSS APPLY sys.dm_exec_sql_text(QS.sql_handle)as ST) as query_stats

GROUP BY query_stats.query_hash

ORDER BY 2 DESC;

References: https://msdn.microsoft.com/en-us/library/ms189741.aspx

QUESTION NO: 4

You run the following Transact-SQL statements:

```
CREATE TABLE Customers (
 CustomerID INT NOT NULL IDENTITY PRIMARY KEY CLUSTERED,
 CustomerName NVARCHAR (100) UNIQUE NOT NULL
)

CREATE TABLE Orders (
 OrderID INT NOT NULL IDENTITY PRIMARY KEY CLUSTERED,
 CustomerID INT NOT NULL REFERENCES Customers (CustomerID),
 OrderDate DATE NOT NULL
)
CREATE VIEW v_CustomerOrder
AS SELECT
 b.CustomerName, a.OrderID, a.OrderDate,
 (SELECT COUNT(*) FROM Orders c WHERE c.CustomerID = a.CustomerID) AS CustomerOrderCount
FROM Orders a
INNER JOIN Customers b ON a.CustomerID = b.CustomerID
```

Records must only be added to the Orders table by using the view. If a customer name does not exist, then a new customer name must be created.

You need to ensure that you can insert rows into the Orders table by using the view.

A. Add the CustomerID column from the Orders table and the WITH CHECK OPTION statement to the view.

B. Create an INSTEAD OF trigger on the view.

C. Add the WITH SCHEMABINDING statement to the view statement and create a clustered index on the view.

D. Remove the subquery from the view, add the WITH SCHEMABINDING statement, and add a trigger to the Orders table to perform the required logic.

ANSWER: A

Explanation:

The WITH CHECK OPTION clause forces all data-modification statements executed against the view to adhere to the criteria set within the WHERE clause of the SELECT statement defining the view. Rows cannot be modified in a way that causes them to vanish from the view. References: http://www.informit.com/articles/article.aspx?p=130855&seqNum=4

QUESTION NO: 5

You are optimizing the performance of a batch update process. You have tables and indexes that were created by running the following Transact-SQL statements:

```
CREATE TABLE Invoices (
    InvoiceID INT NOT NULL IDENTITY PRIMARY KEY CLUSTERED,
    CustomerID INT NOT NULL,
    OrderID INT NULL,
    IsCreditNote BIT NOT NULL,
    IsCreditValidated BIT NOT NULL DEFAULT Ø
)

CREATE INDEX IX_invoices_CustomerID_Filter_IsCreditValidated ON Invoices
(CustomerID) WHERE IsCreditValidated = 1

CREATE TABLE CreditValidation (
    CreditValidationID INT NOT NULL IDENTITY PRIMARY KEY CLUSTERED,
    CustomerID INT NOT NULL,
    ValidationDate DATETIME NOT NULL
)
```

The following query runs nightly to update the isCreditValidated field:

```
UPDATE I
SET IsCreditValidated = 1
FROM Invoices I
WHERE EXISTS (SELECT Ø FROM CreditValidation CV WHERE CV.CustomerID =
I.CustomerID AND CV.ValidationDate >= I.InvoiceDate)
AND I.IsCreditNote = 1
AND I.IsCreditValidated = Ø
AND I.InvocieDate >= DATEADD (DD, -7, GETDATE () )
```

You review the database and make the following observations:

- Most of the IsCreditValidated values in the Invoices table are set to a value of 1.
- There are many unique InvoiceDate values.
- The CreditValidation table does not have an index.
- Statistics for the index IX_invoices_CustomerID_Filter_IsCreditValidated indicate there are no individual seeks but multiple individual updates.

You need to ensure that any indexes added can be used by the update query. If the IX_invoices_CustomerId_Filter_IsCreditValidated index cannot be used by the query, it must be removed. Otherwise, the query must be modified to use with the index.

Which three actions should you perform? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

A. Add a filtered nonclustered index to Invoices on InvoiceDate that selects where IsCreditNote= 1 and IsCreditValidated = 0.

B. Rewrite the update query so that the condition for IsCreditValidated = 0 precedes the condition for IsCreditNote = 1.

C. Create a nonclustered index for invoices in IsCreditValidated, InvoiceDate with an include statement using IsCreditNote and CustomerID.

D. Add a nonclustered index for CreditValidation on CustomerID.

E. Drop the IX_invoices_CustomerId_Filter_IsCreditValidatedIndex.

ANSWER: A B E

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 as well as reduce index maintenance and storage costs compared with full-table indexes.

References: https://docs.microsoft.com/en-us/sql/relational-databases/indexes/create-filtered-indexes

QUESTION NO: 6

You have a Microsoft Azure SQL Database. You enable Query Store for the database and configure the store to use the following settings:

- SIZE_BASED_CLEANUP_MODE = OFF
- STALE_QUERY_THRESHOLD_DAYS = 60
- MAX_STORAGE_SIZE_MB = 100
- QUERY_CAPTURE_MODE = ALL

You use Azure Query Performance Insight to review queries. You observe that new queries are not displayed after 15 days and that the Query Store is set to read-only mode.

If the Query Store runs low on data space, the store must prioritize queries that run regularly or queries that consume applicant resources.

You must set the Query Store to read_write mode and determine the performance of queries from the past 60 days.

Which three actions should you perform? Each correct step presents part of the solution.

NOTE: Each correct selection is worth one point.

A. Set the value of the CLEANUP_POLICY setting to (STALE_QUERY_THRESHOLD_DAYS = 75)

B. Set the value of the QUERY_CAPTURE_MODE setting to AUTO

C. Increase the value for the MAX_STORAGE_SIZE_MB setting

D. Set the value of the SIZE_BASED_CLEANUP_MODE setting to AUTO

E. In the Azure portal, navigate to Query Performance Insight. Use the Custom tab to select a period of 2 months.

ANSWER: B C D

Explanation:

B: Capture mode:

All – Captures all queries. This is the default option.

Auto – Infrequent queries and queries with insignificant cost are ignored. (Ad hoc recommended) None – Query Store stops capturing new queries.

C: Max Size (MB): Specifies the limit for the data space that Query Store can consume within the database. This is the most important setting that directly affects operation mode of the Query Store.

While Query Store collects queries, execution plans and statistics, its size in the database grows until this limit is reached. When that happens, Query Store automatically changes the operation mode to read-only and stops collecting new data. You should monitor this closely to make sure you have sized the store appropriately to contain the full history you'd like to retain.

D: Size Based Cleanup Mode: Specifies whether automatic data cleanup will take place when Query Store data size approaches the limit.

It is strongly recommended to activate size-based cleanup to makes sure that Query Store always runs in read-write mode and collects the latest data.

References:

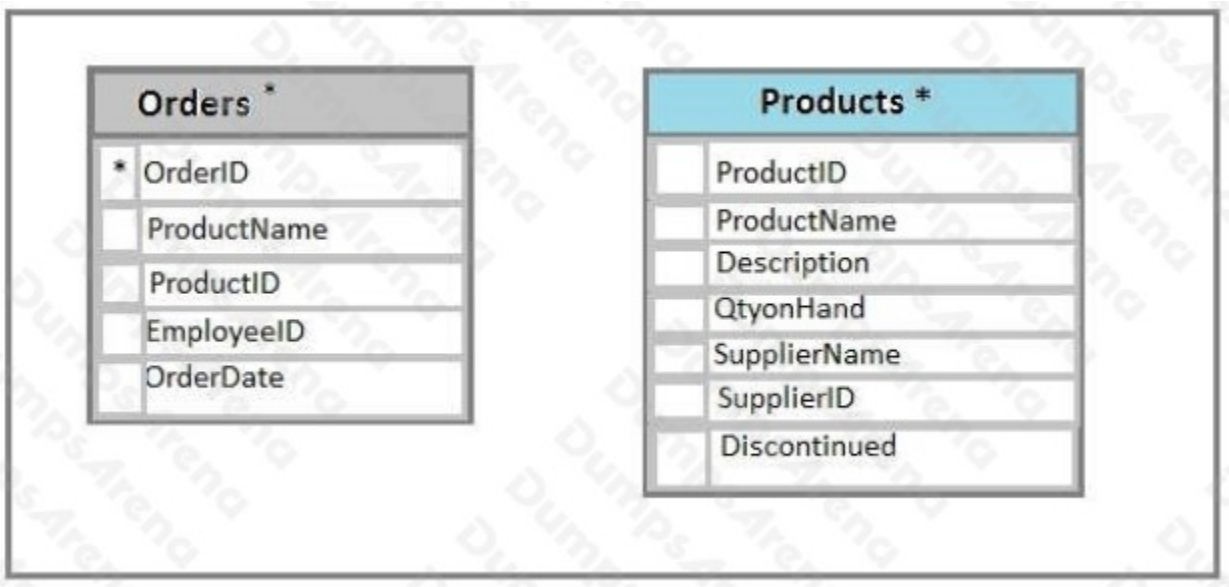
https://docs.microsoft.com/en-us/sql/relational-databases/performance/best-practice-with-the-query-store

QUESTION NO: 7 - (HOTSPOT)

HOTSPOT

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.

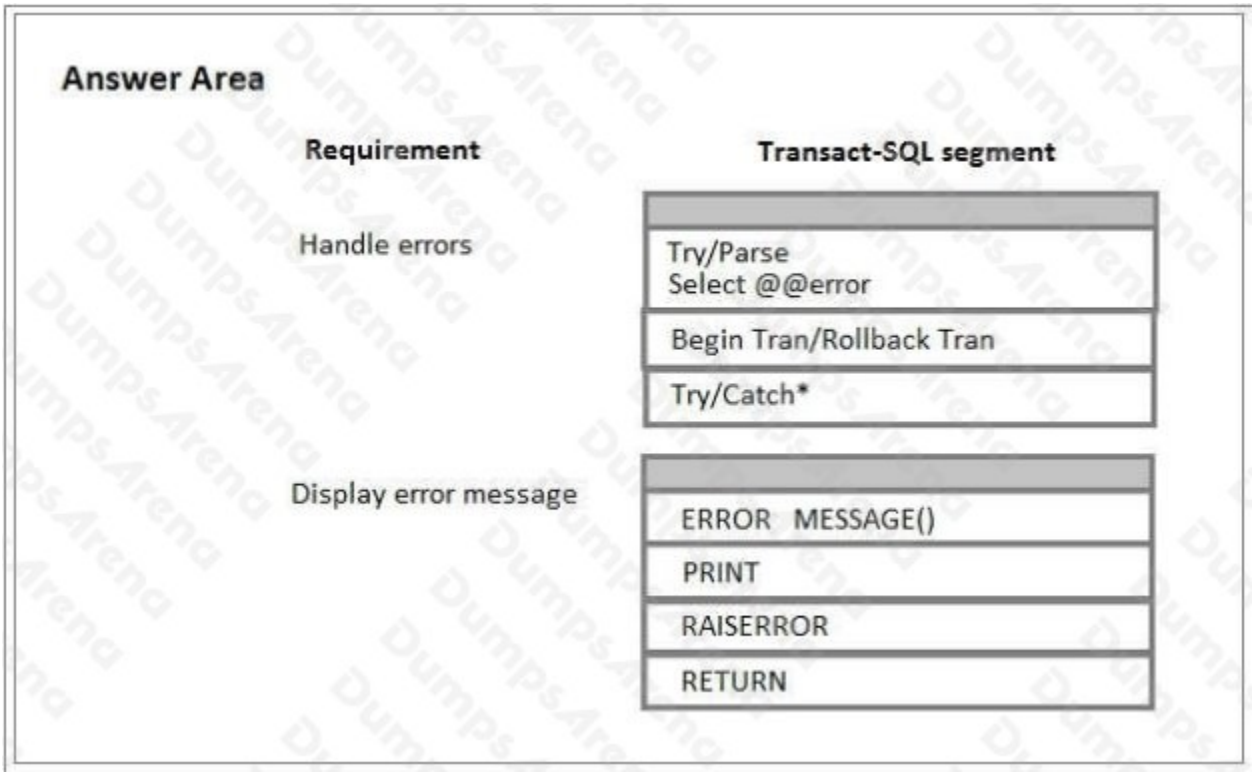
You plan to create a table named Leads. The Leads table is expected to contain approximately 20,000 records. Storage requirements for the Leads table must be minimized.

You need to implement a stored procedure that deletes a discontinued product from the Products table. You identify the following requirements:

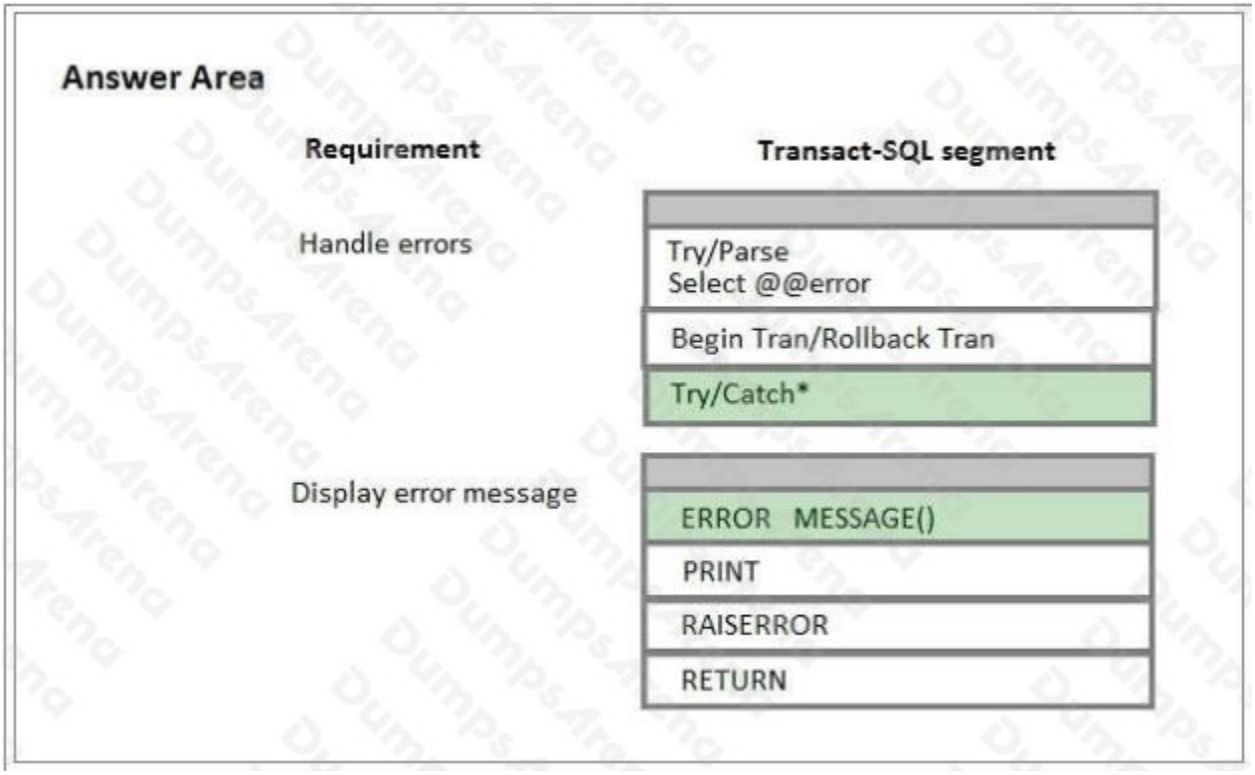
- If an open order includes a discontinued product, the records for the product must not be deleted.
- The stored procedure must return a custom error message if a product record cannot be deleted. The message must identify the OrderID for the open order.

What should you do? To answer, select the appropriate Transact-SQL segments in the answer area.

Hot Area:



ANSWER:



Explanation:

Using TRY...CATCH in Transact-SQL

Errors in Transact-SQL code can be processed by using a TRY…CATCH construct.

TRY…CATCH can use the following error function to capture error information:

ERROR_MESSAGE() returns the complete text of the error message. The text includes the values supplied for any substitutable parameters such as lengths, object names, or times.

References: https://technet.microsoft.com/en-us/library/ms179296(v=sql.105).aspx

QUESTION NO: 8

You have a relational data warehouse that contains 1 TB of data.

You have a stored procedure named usp_sp1 that generated an HTML fragment. The HTML fragment contains color and font style.

You need to return the HTML fragment.

What should you do?

A. Use the NOLOCK option.

B. Execute the DBCC 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 T1 SET (LOCK_ESCALATION = AUTO); statement.

H. Use the OUTPUT parameters.

ANSWER: G

Explanation:

A SQL Server stored procedure that you can call is one that returns one or more OUT parameters, which are parameters that the stored procedure uses to return data back to the calling application.

References: https://docs.microsoft.com/en-us/sql/connect/jdbc/using-a-stored-procedure-with-output-parameters?view=sql-server-2017

QUESTION NO: 9 - (DRAG DROP)

DRAG DROP

You are tuning a database named MyDatabase.

You need to create an Extended Events session to capture execution plans for queries that run for at least 10 minutes. The following requirements must be met:

- The target must write complete buffers to disk asynchronously.
- The system must retain a maximum of 10 files
- Each session must allocate no more than 10 megabytes (MB) of memory for event buffering.

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

Select and Place:

ANSWER:

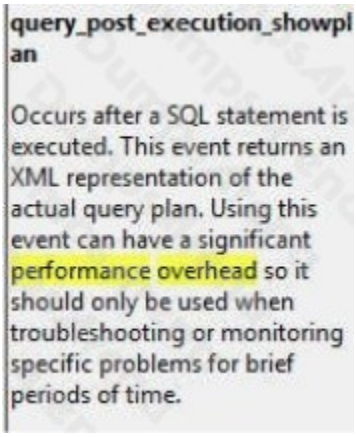
Explanation:

Step 1: CREATE EVENT SESSION SubOptimalExecPlans ON SERVER

Step 2: ADD EVENT sqlserver.query.query_pre_execution_showplan

Incorrect Answers:

Query_post_execution_showplan



Step 3: ADD TARGET package0.asynchronous_file_target .. max_rollower_files=10) The target must write complete buffers to disk asynchronously.

The system must retain a maximum of 10 files

Step 4: WITH (MAX_MEMORY = 10 MB) ;

Each session must allocate no more than 10 megabytes (MB) of memory for event buffering.

Example:

CREATE EVENT SESSION test_session

ON SERVER

ADD EVENT sqlos.async_io_requested,

ADD EVENT sqlserver.lock_acquired

ADD TARGET package0.etw_classic_sync_target

(SET default_etw_session_logfile_path = N'C:\demo\traces\sqletw.etl')

WITH (MAX_MEMORY=4MB, MAX_EVENT_SIZE=4MB);

References:

https://www.sqlservercentral.com/steps/stairway-to-sql-server-extended-events-level-4-extended-events-engine-essential-concepts

QUESTION NO: 10

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 asys.dm_os_wait_stats query.

H. Create an Extended Event.

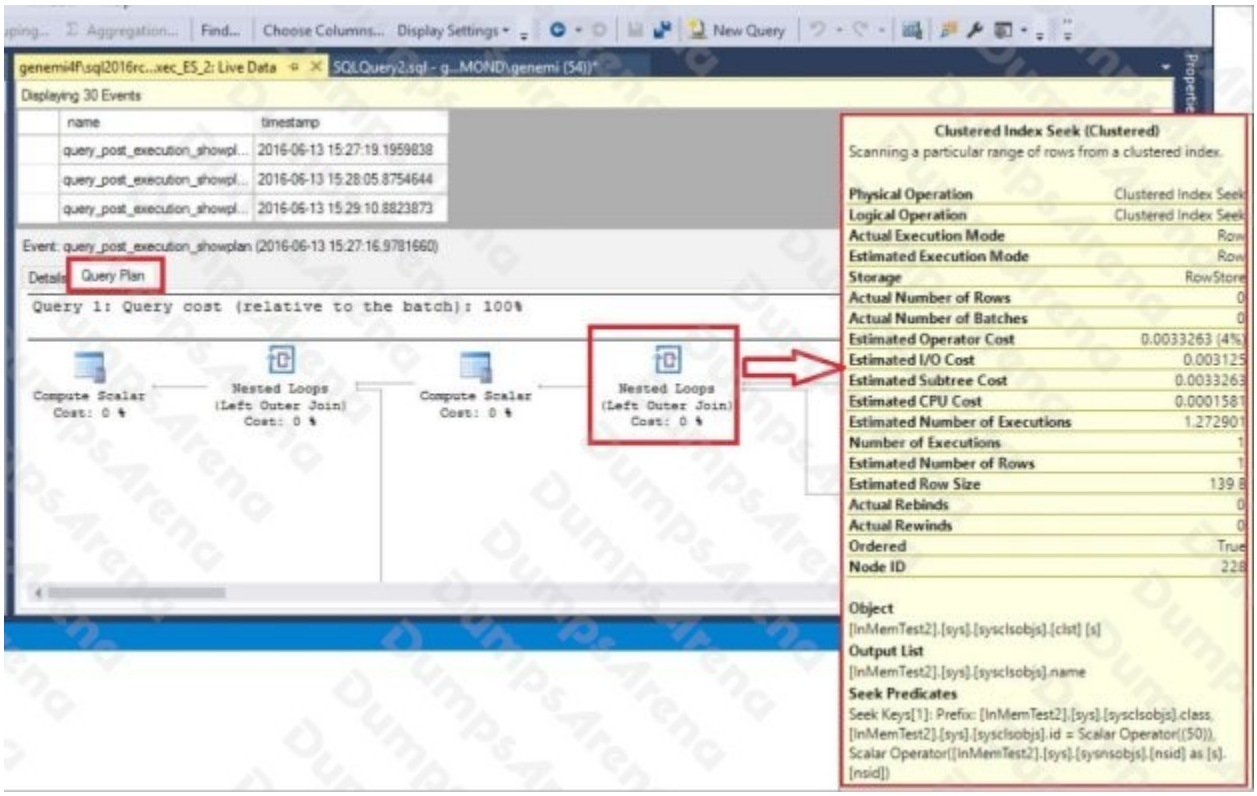
ANSWER: H

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 cansee a list of property names and their values for the node.



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