

70-762.88q

Number: 70-762  
Passing Score: 800  
Time Limit: 120 min

70-762



**Website:** <https://vceplus.com>  
**VCE to PDF Converter:** <https://vceplus.com/vce-to-pdf/>  
**Facebook:** <https://www.facebook.com/VCE.For.All.VN/>  
**Twitter :** [https://twitter.com/VCE\\_Plus](https://twitter.com/VCE_Plus)

<https://www.vceplus.com/>

Developing SQL Databases

Exam A

QUESTION 1

<https://www.vceplus.com/>

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?



<https://www.vceplus.com/>

- 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.

**Correct Answer:** C

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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-](https://technet.microsoft.com/en-us/library/ms178104(v=sql.105).aspx)

[us/library/ms178104\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms178104(v=sql.105).aspx)

## QUESTION 2

You are developing an application that connects to a database.  
The application runs the following jobs:

<https://www.vceplus.com/>

Job	Transact-SQL statement	Description
JobA	Exec uspDeletePrevRecords	The stored procedure deletes all records from a table named tblBalanceTransactions that were created before the current month by using a single DELETE statement. Approximately 10 million records are deleted each time you run this stored procedure
JobB	Exec uspUpdateCurRecords	This stored procedure updates records in the tblBalanceTransaction table that were created in the current month. Only a few hundred records are updated each time you run this stored procedure.

The READ\_COMMITTED\_SNAPSHOT database option is set to OFF, and auto-content is set to ON. Within the stored procedures, no explicit transactions are defined.

If JobB starts before JobA, it can finish in seconds. If JobA starts first, JobB takes a long time to complete.

You need to use Microsoft SQL Server Profiler to determine whether the blocking that you observe in JobB is caused by locks acquired by JobA.

Which trace event class in the Locks event category should you use?

- A. LockAcquired
- B. LockCancel
- C. LockDeadlock
- D. LockEscalation

**Correct Answer:** A

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

The Lock:Acquiredevent class indicates that acquisition of a lock on a resource, such as a data page, has been achieved.

The Lock:Acquired and Lock:Released event classes can be used to monitor when objects are being locked, the type of locks taken, and for how long the locks were retained. Locks retained for long periods of time may cause contention issues and should be investigated.

**QUESTION 3**

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:

1. Allow users access to the CustomerName and CustomerNumber columns for active customers.
2. Allow changes to the columns that the view references. Modified data must be visible through the view.
3. Prevent the view from being published as part of Microsoft SQL Server replication. Sales.Female.Customers must meet the following requirements:
  1. Allow users access to the CustomerName, Address, City, State and PostalCode columns.
  2. Prevent changes to the columns that the view references.
  3. 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:

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.

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

```
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;
```

3. Users must be able to retrieve an account number by supplying customer information. You need to implement the design changes while minimizing data redundancy. What should you do?
- A. Split the table into three separate tables. Include the AccountNumber and CustomerID columns in the first table. Include the CustomerName and Gender columns in the second table. Include the AccountStatus column in the third table.
  - B. Split the table into two separate tables. Include AccountNumber, CustomerID, CustomerName and Gender columns in the first table. Include the AccountNumber and AccountStatus columns in the second table.
  - C. Split the table into two separate tables, Include the CustomerID and AccountNumber columns in the first table. Include the AccountNumber, AccountStatus, CustomerName and Gender columns in the second table.
  - D. Split the table into two separate tables, Include the CustomerID, CustomerName and Gender columns in the first table. Include AccountNumber, AccountStatus and CustomerID columns in the second table.

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

Two tables is enough. CustomerID must be in both tables.

#### QUESTION 4

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:

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:

1. Allow users access to the CustomerName and CustomerNumber columns for active customers.
  2. Allow changes to the columns that the view references. Modified data must be visible through the view.
  3. Prevent the view from being published as part of Microsoft SQL Server replication.
- Sales.FemaleCustomers must meet the following requirements:
1. Allow users access to the CustomerName, Address, City, State and PostalCode columns.
  2. Prevent changes to the columns that the view references.
  3. 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:

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

When you start uspUpdateCustomerSummary, there are no active transactions. The procedure fails at the second update statement due to a CHECK constraint violation on the TotalDepositAccountCount column.

What is the impact of the stored procedure on the CustomerDetails table?

- A. The value of the TotalAccountCount column decreased.
- B. The value of the TotalDepositAccountCount column is decreased.

- C. The statement that modifies TotalDepositAccountCount is excluded from the transaction.
- D. The value of the TotalAccountCount column is not changed.

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

### QUESTION 5

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 on 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 work on an OLTP database that has no memory-optimized file group defined.

You have a table names tblTransaction that is persisted on disk and contains the information described in the following table:

Item	Name	Data Type	Nullable	Notes
Column	TransactionDate	Date	No	For each transaction date, there are only about 100,000 records. The table contains over one billion records in total.
Column	SequenceNo	bigint	No	Uniquely identifies a transaction record within a date
Column	AccountId	int	No	
Column	ValueType	char(3)	No	
Column	Amount	decimal(20,2)	Yes	
	IX_ValueType			Nonclustered columnstore index on the ValueType column.

Users report that the following query takes a long time to complete.



```
SELECT TransactionDate, COUNT(*) AS TotalCount FROM tblTransaction
WHERE TransactionDate - DATEADD(D, -1, CONVERT (DATE, CONVERT (VARCHAR(8),
GETDATE(), 112) 112))
GROUP BY TransactionDate;
```

You need to create an index that:

- improves the query performance
- does not impact the existing index
- minimizes storage size of the table (inclusive of index pages).

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 hashindex on the table.

**Correct Answer: C**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

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.

## QUESTION 6

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 database named DB1. There is no memory-optimized filegroup in the database.

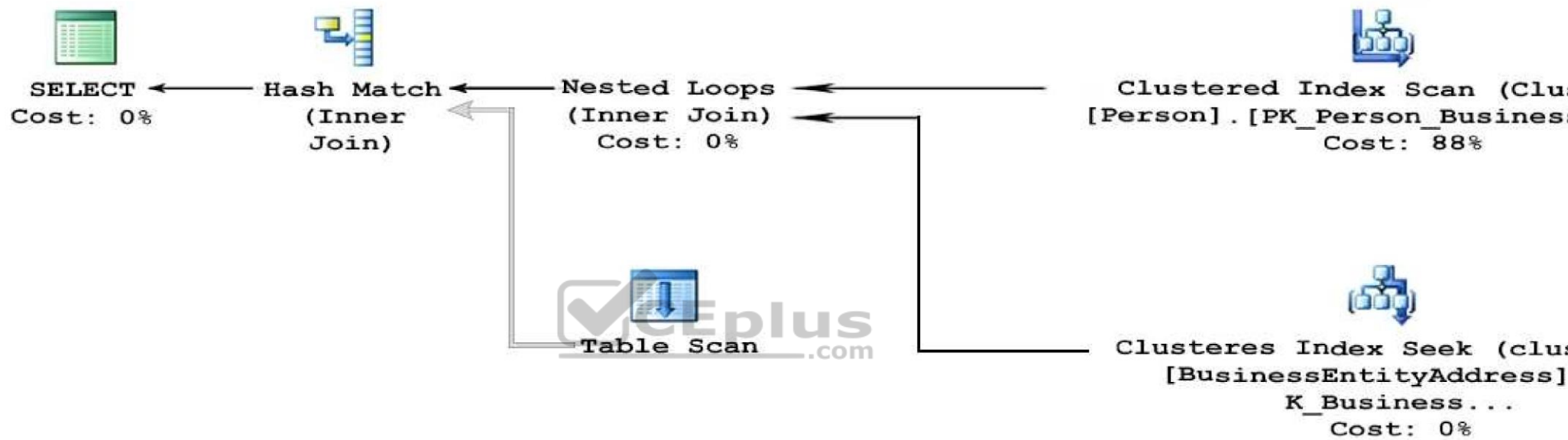
You run the following query:

```
DECLARE @LastName varchar(20) = "contoso";
SELECT P.FirstName, Addr.*
FROM Person.Person P
JOIN Person.BusinessEntityAddress ON P.BusinessEntityID =
BusinessEntityAddress.BusinessEntityID
JOIN Person.[Address] ADDR ON BusinessEntityAddress.AddressID =
Addr.AddressID
WHERE P.LastName = @LastName;
```

The following image displays the execution plan the query optimizer generates for this query:

Users frequently run the same query with different values for the local variable @lastName. The table named Person is persisted on disk. You need to create an index on the Person.Person table that meets the following requirements:





1. All users must be able to benefit from the index.
2. FirstName must be added to the index as an included column.

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.

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

By including nonkey columns, you can create nonclustered indexes that cover more queries. This is because the nonkeycolumns have the following benefits:

They can be data types not allowed as index key columns.

They are not considered by the Database Engine when calculating the number of index key columns or index key size.

#### QUESTION 7

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.

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

The columnstore index is the standard for storing and querying largedata 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.

### QUESTION 8

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 clustered columnstore index on the table.
- E. Create a nonclustered columnstore index on the table.
- F. Create a hash index on the table.

**Correct Answer:** C

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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)

### QUESTION 9

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 have a memory optimized filegroup.

You create a table by running the following Transact-SQL statement:

```
CREATE TABLE tblTransaction(  
    [TransactionID] [int] NOT NULL PRIMARY KEY,  
    [TransactionDate] [date] NOT NULL,  
    [AccountId] [int] NOT NULL,  
    [ValueType] [char](3) NOT NULL,  
    [Amount] [decimal](20,2) NULL  
);
```



The table is currently used for OLTP workloads. The analytics user group needs to perform real-time operational analytics that scan most of the records in the table to aggregate on a number of columns.

You need to add the most efficient index to support the analytics workload without changing the OLTP application.

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.

**Correct Answer: E**

**Section: (none)**

**Explanation**

### Explanation/Reference:

Explanation:

A nonclustered columnstore index enables real-time operational analytics in which the OLTP workload uses the underlying clustered index, while analytics run concurrently on the columnstore index.

Columnstore indexes can achieve up to 100xbetter performance on analytics and data warehousing workloads and up to 10x better data compression than traditional rowstore indexes. These recommendations will help your queries achieve the very fast query performance that columnstore indexes are designed to provide.

References:<https://msdn.microsoft.com/en-us/library/gg492088.aspx>

### QUESTION 10

#### DRAG DROP

You are analyzing the performance of a database environment.

You suspect there are several missing indexes in the current database.

You need to return a prioritized list of the missing indexes on the current database.

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 once or not at all. You may need to drag the split bar between panes or scroll to view content.

#### Select and Place:

Transact-SQL segments	Answer Area
sys.dm_db_missing_index_group_stats	USE DB_NAME( sys.dm_db_missing_index_group_stats
sys.dm_db_missing_index_details	AS state sys.dm_db_missing_index_details
sys.dm_db_missing_index_stats	FROM sys.dm_db_missing_index_group_stats AS g INNER JOIN sys.dm_db_missing_index_details AS d ON g.index_handle = d.index_handle
col_handle	ORDER BY (sys.dm_db_missing_index_group_stats.avg_user_seeks * sys.dm_db_missing_index_group_stats.avg_user_updates + sys.dm_db_missing_index_group_stats.avg_userDeletes)
plan_handle	WITH (NOLOCK) ORDER BY (sys.dm_db_missing_index_group_stats.avg_user_seeks * sys.dm_db_missing_index_group_stats.avg_user_updates + sys.dm_db_missing_index_group_stats.avg_userDeletes)
group_handle	

#### Correct Answer:

<https://www.vceplus.com/>

#### Transact-SQL segments

```
sys.dm_db_missing_index_group_stats
sys.dm_db_missing_index_details
sys.dm_db_missing_index_stats
col_handle
plan_handle
group_handle
```

#### Answer Option

```
USEDB1 GO
SELECT col_handle
FROM sys.dm_db_missing_index_stats AS stats
ORDER BY (avg_total_user_cost * avg_user_impact) * (user_seeks + user_scans) DESC

FROM sys.dm_db_missing_index_group_stats AS stats
INNER JOIN sys.dm_db_missing_index_details AS mid ON stats.group_handle =
mid.index_group_handle
INNER JOIN sys.dm_db_missing_index_stats AS mid_stats ON mid.index_handle =
mid_stats.index_handle
WHERE stats.group_handle = 24

SELECT col_handle
FROM sys.dm_db_missing_index_stats AS stats
ORDER BY (avg_total_user_cost * avg_user_impact) * (user_seeks +
user_scans) DESC
```

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Box 1: sys.dm\_db\_missing\_index\_group\_stats

Box 2: group\_handle

Example: The following query determines which missing indexes comprise a particular missing index group, and displays their column details. For the sake of this example, the missing index group handle is 24.

SELECT migs.group\_handle, mid.\*

FROM sys.dm\_db\_missing\_index\_group\_stats AS migs

INNER JOIN sys.dm\_db\_missing\_index\_groups AS mig

ON (migs.group\_handle = mig.index\_group\_handle)

INNER JOIN sys.dm\_db\_missing\_index\_details AS mid

ON (mig.index\_handle = mid.index\_handle)

WHERE migs.group\_handle = 24;

Box 3: sys.dm\_db\_missing\_index\_group\_stats

The sys.dm\_db\_missing\_index\_group\_stats table include the required columns for the subquery: avg\_total\_user\_cost and avg\_user\_impact.

Example: Find the 10 missing indexes with the highest anticipated improvement for user queries

The following query determines which 10 missing indexes would produce the highest anticipated cumulative improvement, in descending order, for user queries.

SELECT TOP 10 \*

FROM sys.dm\_db\_missing\_index\_group\_stats

ORDER BY avg\_total\_user\_cost \* avg\_user\_impact \* (user\_seeks + user\_scans) DESC;



**QUESTION 11**

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.

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

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

**QUESTION 12**

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 dynamic management view query
- D. an Activity Monitor session in Microsoft SQL Management Studio.

**Correct Answer:** C

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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.

### QUESTION 13

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 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.

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

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

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.

A CLR trigger is a type of DDL trigger. A CLR Trigger can be either an AFTER or INSTEAD OF trigger. A CLR trigger can also be a DDL trigger. Instead of executing a Transact-SQL stored procedure, a CLR trigger executes one or more methods written in managed code that are members of an assembly created in the .NET Framework and uploaded in SQL Server.

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

#### QUESTION 14

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 inserted successfully into the Customers table.
- Return a value of 1 if data is not inserted successfully into the Customers table.
- Support TRY...CATCH error handling- Be written by using Transact-SQL statements.

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



**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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>

#### QUESTION 15

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

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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>

### QUESTION 16

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 return the sum of orders that have been finalized, given a specified order identifier. This value will be used in other Transact-SQL statements.

You need to create a database object.

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

**Correct Answer: F**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

User-defined scalar functions return a single data value of the type defined in the RETURNS clause.

References: [https://technet.microsoft.com/en-us/library/ms177499\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms177499(v=sql.105).aspx)

### QUESTION 17

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 an object that meet the following requirements:

- Run managed code packaged in an assembly that was created in the Microsoft.NET Framework and uploaded in Microsoft SQL Server.
- Run within a transaction and roll back if a failure occurs.
- Run when a table is created or modified.

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

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

The common language runtime (CLR) is the heart of the Microsoft .NET Framework and provides the execution environment for all .NET Framework code. Code that runs within the CLR is referred to as managed code.

With the CLR hosted in Microsoft SQL Server (called CLR integration), you can author stored procedures, triggers, user-defined functions, user-defined types, and user-defined aggregates in managed code. Because managed code compiles to native code prior to execution, you can achieve significant performance increases in some scenarios.

#### **QUESTION 18**

You have a view that includes an aggregate.

You must be able to change the values of columns in the view. The changes must be reflected in the tables that the view uses.

You need to ensure that you can update the view.



<https://www.vceplus.com/>

What should you create?

- A. table-valued function
- B. a schema-bound view
- C. a partitioned view
- D. a DML trigger

<https://www.vceplus.com/>

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

When you use the SchemaBinding keyword while creating a view or function you bind the structure of any underlying tables or views. It means that as long as that schemabound object exists as a schemabound object (ie you don't remove schemabinding) you are limited in changes that can be made to the tables or views that it refers to.

References: <https://sqlstudies.com/2014/08/06/schemabinding-what-why/>

### QUESTION 19

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 and optimize the SQL Server to maximize throughput, response time, and overall SQL performance.

You need to identify previous situations where a modification has prevented queries from selecting data in tables.

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.

**Correct Answer:** G

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

<https://www.vceplus.com/>

sys.dm\_os\_wait\_stats returns information about all the waits encountered by threads that executed. You can use this aggregated view to diagnose performance issues with SQL Server and also with specific queries and batches.

#### QUESTION 20

**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 observe that many deadlocks appear to be happening during specific times of the day.

You need to monitor the SQL environment and capture the information about the processes that are causing the deadlocks. Captured information must be viewable as the queries are running.

What should you do?

- A. A. Create a sys.dm\_os\_waiting\_tasks query.
- B. Create a sys.dm\_exec\_sessions query.
- C. Create a PerformanceMonitor 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.



**Correct Answer:** F

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

To view deadlock information, the Database Engine provides monitoring tools in the form of two trace flags, and the deadlock graph event in SQL Server Profiler.

Trace Flag 1204 and Trace Flag 1222



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. It is possible to enable both trace flags to obtain two representations of the same deadlock event.

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)

### QUESTION 21

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.

What should you do?

- A. A. Create `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.



**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

`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. This information includes client version, client program name, client login time, login user, current session setting, and more. Use `sys.dm_exec_sessions` to first view the current system load and to identify a session of interest, and then learn more information about that session by using other dynamic management views or dynamic management functions.

Examples of use include finding long-running cursors, and finding idle sessions that have open transactions.

## 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 ensure that the performance of each instance is consistent for the same queried and query plans.

What should you do?

- A. 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.



**Correct Answer:** H

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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.

genemi4f\sql2016rc...xec\_ES\_2: Live Data SQLQuery2.sql - g...MOND\genemi (54))\*

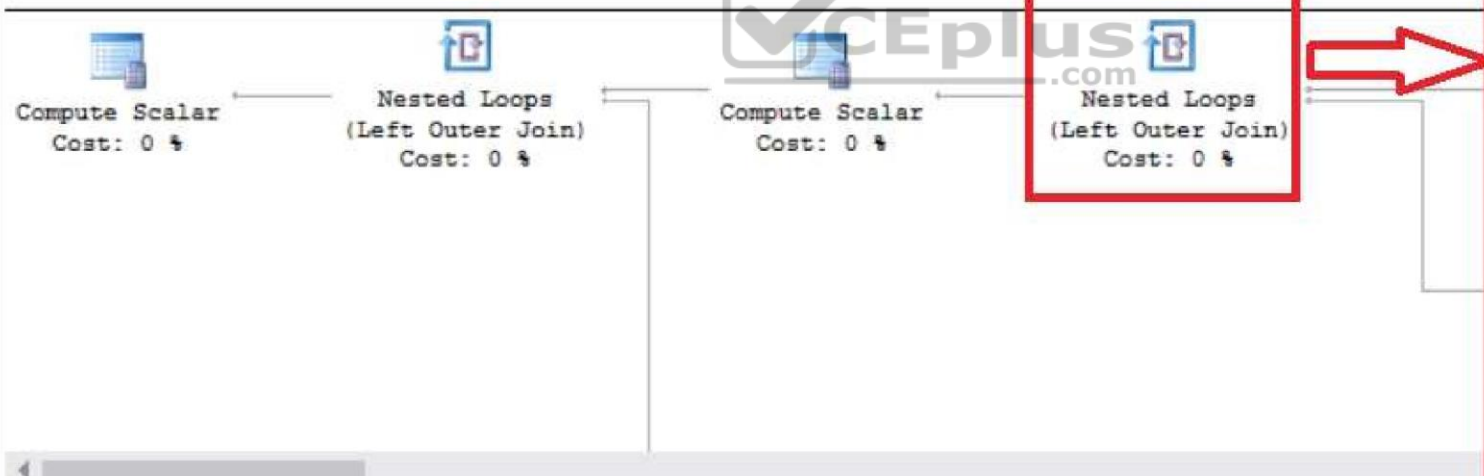
Displaying 30 Events

name	timestamp
query_post_execution_showpl...	2016-06-13 15:27:19.1959838
query_post_execution_showpl...	2016-06-13 15:28:05.8754644
query_post_execution_showpl...	2016-06-13 15:29:10.8823873

Event: query\_post\_execution\_showplan (2016-06-13 15:27:16.9781660)

Details **Query Plan**

Query 1: Query cost (relative to the batch): 100%



Compute Scalar  
Cost: 0 %

Nested Loops  
(Left Outer Join)  
Cost: 0 %

Compute Scalar  
Cost: 0 %

Nested Loops  
(Left Outer Join)  
Cost: 0 %

<b>Clustered Index Seek (Clustered Index)</b>	
Scanning a particular range of rows from a clustered index.	
Physical Operation	Clustered Index Seek
Logical Operation	Clustered Index Seek
Actual Execution Mode	
Estimated Execution Mode	
Storage	
Actual Number of Rows	
Actual Number of Batches	
Estimated Operator Cost	0.00000000
Estimated I/O Cost	
Estimated Subtree Cost	
Estimated CPU Cost	
Estimated Number of Executions	
Number of Executions	
Estimated Number of Rows	
Estimated Row Size	
Actual Rebinds	
Actual Rewinds	
Ordered	
Node ID	
<b>Object</b>	
[InMemTest2].[sys].[sysclsobjs].[clst] [s]	
<b>Output List</b>	
[InMemTest2].[sys].[sysclsobjs].name	
<b>Seek Predicates</b>	
Seek Keys[1]: Prefix: [InMemTest2].[sys].[sysclsobjs].id = Scalar Operator	
Scalar Operator([InMemTest2].[sys].[sysclsobjs].id)	
[nsid]	

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

### QUESTION 23

**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 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 collect query performance data while minimizing the performance impact on the SQL Server.

What should you do?

- A. 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.



**Correct Answer:** C

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

SQL Server Data Collector is a feature for performance monitoring and tuning available in SQL Server Management Studio.

Integration Services packages transform and load the collected data into the Microsoft Data Warehouse database.

Collection sets are defined and deployed on a server instance and can be run independently of each other. Each collection set can be applied to a target that matches the target types of all the collector types that are part of a collection set. The collection set is run by a SQL Server Agent job or jobs, and data is uploaded to the management data warehouse on a predefined schedule.

Predefined data collection sets include:

- The Query Statistics data collection set collects information about query statistics, activity, execution plans and text on the SQL Server instance. It does not store all executed statements, only 10 worst performing ones.
- Disk Usage data collection set collects information about disk space used by both data and log files for all databases on the SQL Server instance, growth trends, and average day growth. Etc.

References: <http://www.sqlshack.com/sql-server-performance-monitoring-data-collector/>

#### QUESTION 24

**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 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.

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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
```

## QUESTION 25

### HOTSPOT

You have a database named Sales.

You need to create a table named Customer that includes the columns described in the following table:

Column Name	Description	Masking requirement
CustomerID	An integer primary key for the customer record	None
FirstName	A string value that stored the first name of the customer	None
LastName	A string value that stored the last name of the customer	None
CreditLimit	A monetary value that stored the customer's credit limit	The entire field must be masked with a value between 1 and 100.
MobileNo	A ten-digit string value that stored the mobile number of the customer	The entire field must be masked with the exception of the first three digits.
Email	A string value that stored the email address of the customer	The entire field must be masked except the first letter of the user name, the @ symbol, and the top-level domain.

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

**Hot Area:**

### Answer Area

```
CREATE TABLE Customer
```

```
(
```

```
    CustomerID int IDENTITY PRIMARY KEY,
```

```
    FirstName varchar(100),
```

```
    LastName varchar(100) NOT NULL,
```

```
    CreditLimit money
```

MASKED WITH (FUNCTION = 'partial(1,"000",100)')

MASKED WITH (FUNCTION = 'default()')

MASKED WITH (FUNCTION = 'random(1,100)')

```
    MobileNO varchar(10)
```

MASKED WITH (FUNCTION = 'partial(3,"XXXXXXX",0)')

MASKED WITH (FUNCTION = 'default()')

MASKED WITH (FUNCTION = 'random(1,3)')

```
    Email varchar(100)
```

MASKED WITH (FUNCTION = 'email()')

MASKED WITH (FUNCTION = 'default()')

MASKED WITH (FUNCTION = 'partial(1,"XXXXXXX",4)')

MASKED WITH (FUNCTION = 'partial(1,"@",4)')

```
)
```

**Correct Answer:**



### Answer Area

```
CREATE TABLE Customer
(
    CustomerID int IDENTITY PRIMARY KEY,
    FirstName varchar(100),
    LastName varchar(100) NOT NULL,
    CreditLimit money
)
```

▼

MASKED WITH (FUNCTION = 'partial(1,"000",100)')

MASKED WITH (FUNCTION = 'default()')

MASKED WITH (FUNCTION = 'random(1,100)')

,

```
MobileNO varchar(10)
```

▼

MASKED WITH (FUNCTION = 'partial(3,"XXXXXXX",0)')

MASKED WITH (FUNCTION = 'default()')

MASKED WITH (FUNCTION = 'random(1,3)')

NULL,

```
Email varchar(100)
```

▼

MASKED WITH (FUNCTION = 'email()')

MASKED WITH (FUNCTION = 'default()')

MASKED WITH (FUNCTION = 'partial(1,"XXXXXXX",4)')

MASKED WITH (FUNCTION = 'partial(1,"@",4)')

NULL,

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Box 1: MASKED WITH (FUNCTION = 'default()')

TheDefault masking method provides full masking according to the data types of the designated fields.

Example column definition syntax: Phone# varchar(12) MASKED WITH (FUNCTION = 'default()') NULL

Box 2: MASKED WITH (FUNCTION = 'partial(3,"XXXXXX",0)')

The Custom String Masking method exposes the first and last letters and adds a custom padding string in the middle. prefix,[padding],suffix examples:

Phone Number varchar(10) MASKED WITH (FUNCTION = 'partial(5,"XXXXXXX",0)')

<https://www.vceplus.com/>



Box 3: MASKED WITH (FUNCTION = 'email()')

The Email masking method which exposes the first letter of an email address and the constant suffix ".com", in the form of an email address. .aXXX@XXXX.com.

Example definition syntax: Email varchar(100) MASKEDWITH (FUNCTION = 'email()') NULL References:<https://msdn.microsoft.com/en-us/library/mt130841.aspx>

#### QUESTION 26

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 stored information about the employees of your company.

You need to implement the following auditing rules for the Employees table:

- Record any changes that are made to the data in the Employees table.
- Customize the data recorded by the audit operations.

Solution: You implement a user-defined function on the Employees table.

Does the solution meet the goal?

- A. Yes
- B. No



**Correct Answer: A**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

SQL Server 2016 provides two features that track changes to data in a database: change data capture and change tracking. These features enable applications to determine the DML changes (insert, update, and delete operations) that were made to user tables in a database.

Change data is made available to change data capture consumers through table-valued functions (TVFs). References:<https://msdn.microsoft.com/en-us/library/cc645858.aspx>

#### QUESTION 27

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 stored information about the employees of your company.

You need to implement the following auditing rules for the Employees table:

- Record any changes that are made to the data in the Employees table.
- Customize the data recorded by the audit operations.

Solution: You implement a check constraint on the Employees table.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

Check constraints cannot be used to track changes in a table.



References: <https://msdn.microsoft.com/en-us/library/bb933994.aspx>

### QUESTION 28

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 stored information about the employees of your company.

You need to implement the following auditing rules for the Employees table:

- Record any changes that are made to the data in the Employees table.
- Customize the data recorded by the audit operations.

Solution: You implement a stored procedure on the Employees table.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

We should use table-valued functions, not procedures, to customize the recorded change data.

References: <https://msdn.microsoft.com/en-us/library/cc645858.aspx>

### **QUESTION 29**

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

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

<https://www.vceplus.com/>

Explanation:

Datetimeoffset, not datetimeofset, 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>

### QUESTION 30

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:

1. Employee ID
2. date and time employee checked in to work
3. 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,  
    DateCheckedIn datetime2 NOT NULL,  
    DateCheckedOut datetime2 NOT NULL)
```

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

Datetime2 stores fractional seconds.

Datetime2 defines a date that is combined with a time of day that is based on 24-hour clock. datetime2 can be considered as an extension of the existing datetime type that has a larger date range, a larger default fractional precision, and optional user-specified precision.

References: <https://docs.microsoft.com/en-us/sql/t-sql/data-types/datetime2-transact-sql?view=sql-server-2017> <https://msdn.microsoft.com/en-us/library/bb677335.aspx>

### QUESTION 31

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,  
    DateCheckedIn datetimeoffset(0) NOT NULL,  
    DateCheckedOut datetimeoffset(0) NOT NULL)
```

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: A**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

Datetimeoffset defines a date that is combined with a time of a day that has time zone awareness and is based on a 24-hour clock.

Syntax: datetimeoffset [ (fractional seconds precision) ]

For the use "datetimeoffset(0)", the Fractional seconds precision is 0, which is required here. References: [https://msdn.microsoft.com/en-](https://msdn.microsoft.com/en-us/library/bb630289.aspx)

[us/library/bb630289.aspx](https://msdn.microsoft.com/en-us/library/bb630289.aspx)

**QUESTION 32**

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 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 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 statement:

```
CREATE NONCLUSTERED INDEX PK_Account (AccountNumber) ;  
CREATE NONCLUSTERED INDEX IX_Account_productCode On Account (ProductCode)  
INCLUDE (Balance) ;
```

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

Create a clustered index on the AccountNumber column as it is unique, not a non nonclustered one. References:<https://msdn.microsoft.com/en-us/library/ms190457.aspx>

### QUESTION 33

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 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 CLUSTERED INDEX PK_Account ON Account(ProductCode);
```

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:



We need an index on the productCode column as well.

References: <https://msdn.microsoft.com/en-us/library/ms190457.aspx>

### QUESTION 34

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 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 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 statement:

```
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

**Correct Answer:** A

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

Create a clustered index on the AccountNumber column as it is unique.

Create a nonclustered index that includes the ProductCode column.

References: <https://msdn.microsoft.com/en-us/library/ms190457.aspx>

### QUESTION 35

**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 need to create a stored procedure that updates the Customer, CustomerInfo, OrderHeader, and OrderDetails tables in order.

You need to ensure that the stored procedure:

- Runs within a single transaction.
  - Commits updates to the Customer and CustomerInfo tables regardless of the status of updates to the OrderHeader and OrderDetail tables. ▪
- Commits changes to all four tables when updates to all four tables are successful.

Solution: You create a stored procedure that includes the following Transact-SQL segment:

```
BEGIN TRY
    BEGIN TRAN
        UPDATE Customer ...
        UPDATE CustomerInfo ...
        UPDATE OrderHeader ...
        UPDATE OrderDetail ...
    COMMIT TRAN
END TRAN
BEGIN CATCH
    IF XACT_STATE() = 1
        ROLLBACK TRAN
END CATCH
```

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

All four tables are updated in a single transaction.

Need to handle the case where the first two updates (OrderHeader, OrderDetail) are successful, but either the 3rd or the 4th (OrderHeader, OrderDetail) fail. Can add a variable in the BEGIN TRY block, and test the variable in the BEGIN CATCH block.

References: <https://docs.microsoft.com/en-us/sql/t-sql/language-elements/begin-transaction-transact-sql>

### QUESTION 36

**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 need to create a stored procedure that updates the Customer, CustomerInfo, OrderHeader, and OrderDetails tables in order.

You need to ensure that the stored procedure: ▪

Runs within a single transaction.

▪ Commits updates to the Customer and CustomerInfo tables regardless of the status of updates to the OrderHeader and OrderDetail tables. ▪

Commits changes to all four tables when updates to all four tables are successful.

Solution: You create a stored procedure that includes the following Transact-SQL segment:

```
DECLARE @CustomerComplete bit = 0
BEGIN TRY
    BEGIN TRAN
        UPDATE Customer ...
        UPDATE CustomerInfo ...

        SET @CustomerComplete = 1

        SAVE TRAN TR1

        UPDATE OrderHeader ...
        UPDATE OrderDetail ...
    COMMIT TRAN
END TRY
BEGIN CATCH
    IF (@CustomerComplete = 1) AND (XACT_STATE() = 1)
    BEGIN
        COMMIT TRAN
    END
    ELSE IF XACT_STATE() = 1
        ROLLBACK TRAN
END CATCH
```



Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: A**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

Need to handle the case where the first two updates (OrderHeader, OrderDetail) are successful, but either the 3rd or the 4th (OrderHeader, OrderDetail) fail. We add the @CustomerComplete variable in the BEGIN TRY block, and test it in the BEGIN CATCH block.

Note: XACT\_STATE indicates whether the request has an active user transaction, and whether the transaction is capable of being committed.

XACT\_STATE =1: the current request has an active user transaction. The request can perform any actions, including writing data and committing the transaction.

References:

<https://docs.microsoft.com/en-us/sql/t-sql/functions/xact-state-transact-sql>

**QUESTION 37**

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 a check constraint on the table.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: A**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

References: [https://en.wikipedia.org/wiki/Check\\_constraint](https://en.wikipedia.org/wiki/Check_constraint)

**QUESTION 38**

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 a FOR UPDATE trigger on the table.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** (none)

**Explanation**



**Explanation/Reference:**

Explanation:

References: <http://stackoverflow.com/questions/16081582/difference-between-for-update-of-and-for-update>

### QUESTION 39

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

**Correct Answer:** A

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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)

#### QUESTION 40

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 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 statement:

```
CREATE NONCLUSTERED INDEX IX_Account_ProductCode ON Account(ProductCode);
```

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: A**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

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

#### QUESTION 41

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.

You are developing a new application that uses a stored procedure. The stored procedure inserts thousands of records as a single batch into the Employees table.

Users report that the application response time has worsened since the stored procedure was updated. You examine disk-related performance counters for the Microsoft SQL Server instance and observe several high values that include a disk performance issue. You examine wait statistics and observe an unusually high WRITELOG value.

You need to improve the application response time.

Solution: You update the application to use implicit transactions when connecting to the database.



Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

References: <http://sqltouch.blogspot.co.za/2013/05/writelog-waittype-implicit-vs-explicit.html>

#### **QUESTION 42**

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.

You are developing a new application that uses a stored procedure. The stored procedure inserts thousands of records as a single batch into the Employees table.

Users report that the application response time has worsened since the stored procedure was updated. You examine disk-related performance counters for the Microsoft SQL Server instance and observe several high values that include a disk performance issue. You examine wait statistics and observe an unusually high WRITELOG value.

You need to improve the application response time.

Solution: You add a unique clustered index to the Employees table.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** A

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

References: <https://msdn.microsoft.com/en-us/library/ms190457.aspx>

#### QUESTION 43

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.

You are developing a new application that uses a stored procedure. The stored procedure inserts thousands of records as a single batch into the Employees table.

Users report that the application response time has worsened since the stored procedure was updated. You examine disk-related performance counters for the Microsoft SQL Server instance and observe several high values that include a disk performance issue. You examine wait statistics and observe an unusually high WRITELOG value.

You need to improve the application response time.

Solution: You replace the stored procedure with a user-defined function.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

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

#### QUESTION 44

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

**Correct Answer:** C

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

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

#### QUESTION 45

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.

<https://www.vceplus.com/>

What should you create?

- A. a System Monitor report
- B. a sys.dm\_tran\_database\_transaction dynamic management view query
- C. an Extended Events session that uses Query Editor
- D. an Activity Monitor session in Microsoft SQL Management Studio.

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

References: <https://msdn.microsoft.com/en-us/library/hh212951.aspx>

#### **QUESTION 46**

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 database named DB1. There is no memory-optimized filegroup in the database.

You have a table and a stored procedure that were created by running the following Transact-SQL statements:

```
CREATE TABLE Employee
(
    EmployeeId int NOT NULL PRIMARY KEY,
    FirstName varchar(20),
    LastName varchar(20),
    Status char(1),
    Address varchar(100),
    Department int NOT NULL
);

CREATE PROCEDURE uspSelectEmployeeDetails
(
    @LastName varchar(20)
)
AS
BEGIN
    SELECT e.FirstName, e.LastName, d.DepartmentName
    FROM Employee e
    JOIN Department d on e.DepartmentId = d.DepartmentId
    WHERE e.Status = 'T' AND e.LastName = @LastName;
END;
```

The Employee table is persisted on disk. You add 2,000 records to the Employee table.

You need to create an index that meets the following requirements:

- Optimizes the performance of the stored procedure.
- Covers all the columns required from the Employee table.
- Uses FirstName and LastName as included columns. ▪

Minimizes index storage size and index key size.

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.

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

References: [https://technet.microsoft.com/en-us/library/jj835095\(v=sql.110\).aspx](https://technet.microsoft.com/en-us/library/jj835095(v=sql.110).aspx)

#### **QUESTION 47**

Background

You have a database named HR1 that includes a table named Employee.

You have several read-only, historical reports that contain regularly changing totals. The reports use multiple queries to estimate payroll expenses. The queries run concurrently. Users report that the payroll estimate reports do not always run. You must monitor the database to identify issues that prevent the reports from running.

You plan to deploy the application to a database server that supports other applications. You must minimize the amount of storage that the database requires.

Employee Table

You use the following Transact-SQL statements to create, configure, and populate the Employee table:

```
CREATE TABLE dbo.Employee
(
    EmployeeId INT PRIMARY KEY,
    LastName varchar(50),
    FirstName varchar(50),
    DepartmentId int,
    HireDate datetime,
    TerminationDate datetime,
    SupervisorId int,
    CostCenterNumber int,
    EmployeeStatus int,
    EmployeePayRate int
)
GO

CREATE INDEX IX_1 on dbo.Employee (LastName, FirstName, DepartmentId) INCLUDE (HireDate)
CREATE INDEX IX_2 on dbo.Employee (LastName) INCLUDE (EmployeeId, FirstName, DepartmentId)
CREATE INDEX IX_3 on dbo.Employee (LastName, FirstName) INCLUDE (DepartmentId)
CREATE INDEX IX_4 on dbo.Employee (LastName, FirstName) INCLUDE (HireDate, DepartmentId)
GO

INSERT INTO Employee (EmployeeID, LastName, CostCenterNumber) VALUES(1001, 'Employee A', 3001001)
INSERT INTO Employee (EmployeeID, LastName, CostCenterNumber) VALUES(1002, 'Employee B', 3001001)
GO
```

#### Application

You have an application that updates the Employees table. The application calls the following stored procedures simultaneously and asynchronously: ▪

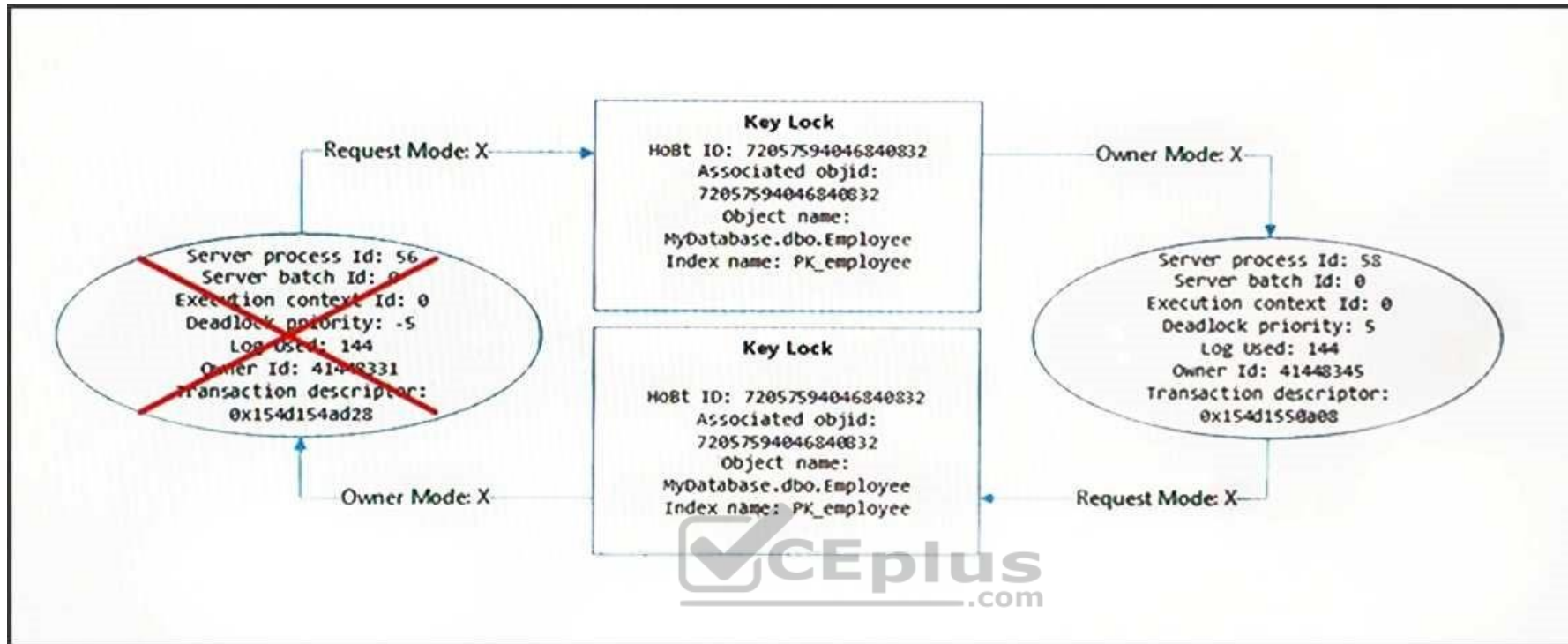
- UspA: This stored procedure updates only the EmployeeStatus column.
- UspB: This stored procedure updates only the EmployeePayRate column.

The application uses views to control access to data. Views must meet the following requirements:

- Allow user access to all columns in the tables that the view accesses. ▪

Restrict updates to only the rows that the view returns.

#### Exhibit



You are analyzing the performance of the database environment. You discover that locks that are held for a long period of time as the reports are generated.

You need to generate the reports more quickly. The database must not use additional resources.

What should you do?

- A. Update the transaction level of the report query session to READPAST.
- B. Modify the report queries to use the UNION statement to combine the results of two or more queries.
- C. Set the READ\_COMMITTED\_SNAPSHOT database option to ON.
- D. Update the transaction level of the report query session to READ UNCOMMITTED.

**Correct Answer: D**

**Section: (none)**

**Explanation**



**Explanation/Reference:**

Explanation:

Transactions running at the READ UNCOMMITTED level do not issue shared locks to prevent other transactions from modifying data read by the current transaction. This is the least restrictive of the isolation levels.

References: [https://technet.microsoft.com/en-us/library/ms173763\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms173763(v=sql.105).aspx)

**QUESTION 48**

**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 partition the largest tables.

Does this meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

**QUESTION 49**

**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 change the transaction log file size to expand dynamically in increments of 200 MB.

Does this meet the goal?

- A. Yes
- B. No

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**



#### QUESTION 50

**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

**Correct Answer:** A

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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].

#### QUESTION 51

**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 **Premium** service tier.

Does this meet the goal?

- A. Yes
- B. No

**Correct Answer:** A

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

Premium service is required for 3 TB of storage.

Single database DTU and storage limits

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

	Basic	Standard	P
Maximum storage size	2 GB	1 TB	
Maximum DTUs	5	3000	

#### QUESTION 52

**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

**Correct Answer:** B

**Section:** (none)

**Explanation**

<https://www.vceplus.com/>

**Explanation/Reference:**

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>

**QUESTION 53**

**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 Basic service tier.

Does this meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

<https://www.vceplus.com/>

Premium service is required for 3 TB of storage.

Single database DTU and storage limits

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

#### QUESTION 54

**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.

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

You have a table that has a clustered index and a nonclustered index. The indexes use different columns from the table. You have a query named Query1 that uses the nonclustered index.

Users report that Query1 takes a long time to report results. You run Query1 and review the following statistics for an index seek operation:

### Index Seek (NonClustered)

Scan a particular range of rows from a nonclustered index.

Physical Operation	Index Seek
Logical Operation	Index Seek
Actual Execution Mode	Row
Actual Number of Rows	3571454
Actual Number of Batches	0
Estimated I/O Cost	0.0093577
Estimated Operator Cost	0.0107304 (0%)
Estimated CPU Cost	0.0013727
Estimated Subtree Cost	0.0107304
Estimated Number of Executions	1
Number of Executions	8
Estimated Number of Rows	0
Estimated Row Size	19 B
Actual Rebinds	0
Actual Rewinds	0
Ordered	True
Node ID	100

You need to resolve the performance issue.

Solution: You drop the nonclustered index.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

We see Actual Number of Row is 3571454, while Estimated Number of Rows is 0.

This indicates that the statistics are old, and need to be updated.

#### **QUESTION 55**

**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 table that has a clustered index and a nonclustered index. The indexes use different columns from the table. You have a query named Query1 that uses the nonclustered index.

Users report that Query1 takes a long time to report results. You run Query1 and review the following statistics for an index seek operation:





### Index Seek (NonClustered)

Scan a particular range of rows from a nonclustered index.

Physical Operation	Index Seek
Logical Operation	Index Seek
Actual Execution Mode	Row
Actual Number of Rows	3571454
Actual Number of Batches	0
Estimated I/O Cost	0.0093577
Estimated Operator Cost	0.0107304 (0%)
Estimated CPU Cost	0.0013727
Estimated Subtree Cost	0.0107304
Estimated Number of Executions	1
Number of Executions	8
Estimated Number of Rows	0
Estimated Row Size	19 B
Actual Rebinds	0
Actual Rewinds	0
Ordered	True
Node ID	100



<https://www.vceplus.com/>

You need to resolve the performance issue.

<https://www.vceplus.com/>

Solution: You defragment both indexes.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

We see Actual Number of Row is 3571454, while Estimated Number of Rows is 0.

This indicates that the statistics are old, and need to be updated.

#### **QUESTION 56**

**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 table that has a clustered index and a nonclustered index. The indexes use different columns from the table. You have a query named Query1 that uses the nonclustered index.

Users report that Query1 takes a long time to report results. You run Query1 and review the following statistics for an index seek operation:

### Index Seek (NonClustered)

Scan a particular range of rows from a nonclustered index.

Physical Operation	Index Seek
Logical Operation	Index Seek
Actual Execution Mode	Row
Actual Number of Rows	3571454
Actual Number of Batches	0
Estimated I/O Cost	0.0093577
Estimated Operator Cost	0.0107304 (0%)
Estimated CPU Cost	0.0013727
Estimated Subtree Cost	0.0107304
Estimated Number of Executions	1
Number of Executions	8
Estimated Number of Rows	0
Estimated Row Size	19 B
Actual Rebinds	0
Actual Rewinds	0
Ordered	True
Node ID	100

You need to resolve the performance issue.

Solution: You update statistics for the nonclustered index.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** A

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

We see Actual Number of Row is 3571454, while Estimated Number of Rows is 0.

This indicates that the statistics are old, and need to be updated.

#### **QUESTION 57**

You have a reporting application that uses a table named Table1. You deploy a new batch update process to perform updates to Table1.

The environment is configured with the following properties:

- The database is configured with the default isolation setting.
- The application and process use the default transaction handling.

You observe the application cannot access any rows that are in use by the process.

You have the following requirements:

- Ensure the application is not blocked by the process. ▪
- Ensure the application has a consistent view of the data ▪
- Ensure the application does not read dirty data.

You need to resolve the issue and meet the requirements with the least amount of administrative effort.

What should you do?

- A. Enable the database for the `ALLOW_SNAPSHOT_ISOLATION` isolation level. Modify the application for the `SERIALIZABLE` isolation level.
- B. Enable the database for the `READ_COMMITTED_SNAPSHOT` isolation level.
- C. Enable the application for the `WITH (NOLOCK)` hint.
- D. Enable the database for the `ALLOW_SNAPSHOT_ISOLATION` isolation level. Modify the application and the update process for the `SNAPSHOT` isolation level.

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

Snapshot isolation must be enabled by setting the ALLOW\_SNAPSHOT\_ISOLATION ON database option before it is used in transactions. This activates the mechanism for storing row versions in the temporary database (tempdb).

READ COMMITTED is the default isolation level for SQL Server. It prevents dirty reads by specifying that statements cannot read data values that have been modified but not yet committed by other transactions. Other transactions can still modify, insert, or delete data between executions of individual statements within the current transaction, resulting in non-repeatable reads, or "phantom" data.

Incorrect Answers:

A: SERIALIZABLE is the most restrictive isolation level, because it locks entire ranges of keys and holds the locks until the transaction is complete. It encompasses REPEATABLE READ and adds the restriction that other transactions cannot insert new rows into ranges that have been read by the transaction until the transaction is complete.

References: <https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/sql/snapshot-isolation-in-sql-server>

### QUESTION 58

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 (ONLINE = OFF)
- B. CREATE UNIQUE INDEX IX\_CUSTOMER\_Code O 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)

<https://www.vceplus.com/>

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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.

Incorrect Answers:

ONLINE = { ON | OFF } specifies whether underlying tables and associated indexes are available for queries and data modification during the index operation. The default is OFF.

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

#### **QUESTION 59**

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

**Correct Answer:** AC

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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-2017>

**QUESTION 60**

You have the following stored procedure that is called by other stored procedures and applications:

```
CREATE PROCEDURE UpdateCustomer @CustomerId INT
AS
BEGIN
    EXEC ProcessCustomer_Internal @CustomerId
    DECLARE @Status INT
    SELECT @Status = Status FROM Customer WHERE CustomerId =
@CustomerId
END
```

You need to modify the stored procedure to meet the following requirements:

- Always return a value to the caller.
- Return 0 if @Status is NULL.
- Callers must be able to use @Status as a variable.

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

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

- A. Replace NULL values with 0. Add a PRINT statement to return @Status.
- B. Add a RETURN statement.
- C. Replace NULL values with 0. Add an output parameter to return @Status.
- D. Replace NULL values with 0. Add a SELECT statement to return @Status.
- E. Add a PRINT statement.
- F. Add a SELECT statement to return @Status.
- G. Add an output parameter to return @Status.

**Correct Answer:** BC

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

There are three ways of returning data from a procedure to a calling program: result sets, output parameters, and return codes.

References: <https://docs.microsoft.com/en-us/sql/relational-databases/stored-procedures/return-data-from-a-stored-procedure?view=sql-server-2017> **QUESTION 61 DRAG DROP**

You need to implement triggers to automate responses to the following events:

- SQL Server logons
- Database schema changes
- Database updates

Which trigger types should you use? To answer, drag the appropriate trigger types to the appropriate scenarios. Each trigger type 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.

**Select and Place:**

Trigger types	Answer Area	Trigger type
LOGON	<b>Scenario</b> Prevent a user from establishing a session if they have an existing session Insert into tables when data is inserted into a non-updatable view Insert the column name, table name, and user name when a column is added to a table	Trigger
INSTEAD OF INSERT		Trigger
AFTER INSERT		Trigger
DDL		

**Correct Answer:**

Trigger types	Answer Area	Trigger type
	<b>Scenario</b> Prevent a user from establishing a session if they have an existing session Insert into tables when data is inserted into a non-updatable view Insert the column name, table name, and user name when a column is added to a table	LOGON
AFTER INSERT		INSTEAD OF INSERT
		DDL



**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

Box 1: LOGON

Logon triggers fire stored procedures in response to a LOGON event. This event is raised when a user session is established with an instance of SQL Server.

Box 2: INSTEAD OF INSERT

An "INSTEAD of trigger" is executed instead of the original operation, and not combining with the operation. INSTEAD OF triggers override the standard actions of the triggering statement. It can be used to bypass the statement and execute a whole different statement, or just help us check and examine the data before the action is done.

Box 3: DDL

DDL statements (CREATE or ALTER primarily) issued either by a particular schema/user or by any schema/user in the database

Note:

You can write triggers that fire whenever one of the following operations occurs:

- DML statements (INSERT, UPDATE, DELETE) on a particular table or view, issued by any user
- DDL statements (CREATE or ALTER primarily) issued either by a particular schema/user or by any schema/user in the database
- Database events, such as logon/logoff, errors, or startup/shutdown, also issued either by a particular schema/user or by any schema/user in the database

References:

[https://docs.oracle.com/cd/B19306\\_01/server.102/b14220/triggers.htm](https://docs.oracle.com/cd/B19306_01/server.102/b14220/triggers.htm)

<https://social.technet.microsoft.com/wiki/contents/articles/28152.t-sql-instead-of-triggers.aspx> <https://docs.microsoft.com/en-us/sql/relational-databases/triggers/logon-triggers?view=sql-server-2017>

## QUESTION 62

**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 an object that meets to following requirements:

- Run managed code packaged in an assembly that was created in the Microsoft.NET Framework and uploaded in Microsoft SQL Server.
- Run written a transaction and roll back if a failure occurs. ▪

Run when a table is created or modified.

<https://www.vceplus.com/>

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

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

The common language runtime (CLR) is the heart of the Microsoft .NET Framework and provides the execution environment for all .NET Framework code. Code that runs within the CLR is referred to as managed code.

With the CLR hosted in Microsoft SQL Server (called CLR integration), you can author stored procedures, triggers, user-defined functions, user-defined types, and user-defined aggregates in managed code.

References: <https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/sql/introduction-to-sql-server-clr-integration>

### QUESTION 63

You are creating the following two stored procedures:

- A natively-compiled stored procedure
- An interpreted stored procedure that accesses both disk-based and memory-optimized tables

Both stored procedures run within transactions.

You need to ensure that cross-container transactions are possible.

Which setting or option should you use?

- A. the `SET TRANSACTION_READ_COMMITTED` isolation level for the connection
- B. the `SERIALIZABLE` table hint on disk-based tables

- C. the `SET MEMORY_OPTIMIZED_ELEVATE_TO_SNAPSHOT=ON` option for the database
- D. the `SET MEMORY_OPTIMIZED_ELEVATE_TO_SNAPSHOT=OFF` option for the database

**Correct Answer:** C

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

Provide a supported isolation level for the memory-optimized table using a table hint, such as `WITH (SNAPSHOT)`. The need for the `WITH (SNAPSHOT)` hint can be avoided through the use of the database option `MEMORY_OPTIMIZED_ELEVATE_TO_SNAPSHOT`. When this option is set to `ON`, access to a memory-optimized table under a lower isolation level is automatically elevated to `SNAPSHOT` isolation.

Incorrect Answers:

B: Accessing memory-optimized tables using the `READ COMMITTED` isolation level is supported only for autocommit transactions. It is not supported for explicit or implicit transactions.

References: <https://docs.microsoft.com/en-us/sql/relational-databases/in-memory-oltp/transactions-with-memory-optimized-tables?view=sql-server-2017>

#### **QUESTION 64**

You are developing a database reporting solution for a table that contains 900 million rows and is 103 GB.

The table is updated thousands of times a day, but data is not deleted.

The `SELECT` statements vary in the number of columns used and the amount of rows retrieved.

You need to reduce the amount of time it takes to retrieve data from the table. The must prevent data duplication.

Which indexing strategy should you use?

- A. a nonclustered index for each column in the table
- B. a clustered columnstore index for the table
- C. a hash index for the table
- D. a clustered index for the table and nonclustered indexes for nonkey columns

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

Columnstore indexes are 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.

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

Generally, you should define the clustered index key with as few columns as possible.

A nonclustered index contains the index key values and row locators that point to the storage location of the table data. You can create multiple nonclustered indexes on a table or indexed view. Generally, nonclustered indexes should be designed to improve the performance of frequently used queries that are not covered by the clustered index.

References: <https://docs.microsoft.com/en-us/sql/relational-databases/indexes/columnstore-indexes-overview?view=sql-server-2017>

**QUESTION 65**

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

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

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>

#### **QUESTION 66**

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 Performance statistics 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.error\_reported event added.

**Correct Answer: A**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

The Performance Statistics event class can be used to monitor the performance of queries, stored procedures, and triggers that are executing. Each of the six event subclasses indicates an event in the lifetime of queries, stored procedures, and triggers within the system. Using the combination of these event subclasses and the associated sys.dm\_exec\_query\_stats, sys.dm\_exec\_procedure\_stats and sys.dm\_exec\_trigger\_stats dynamic management views, you can reconstitute the performance history of any given query, stored procedure, or trigger.

References: <https://docs.microsoft.com/en-us/sql/relational-databases/event-classes/performance-statistics-event-class?view=sql-server-2017>

## **QUESTION 67**

### **Background**

You have a database named HR1 that includes a table named Employee.

You have several read-only, historical reports that contain regularly changing totals. The reports use multiple queries to estimate payroll expenses. The queries run concurrently. Users report that the payroll estimate reports do not always run. You must monitor the database to identify issues that prevent the reports from running.

You plan to deploy the application to a database server that supports other applications. You must minimize the amount of storage that the database requires.

### **Employee Table**

You use the following Transact-SQL statements to create, configure, and populate the Employee table:



```
CREATE TABLE dbo.Employee
  (EmployeeId INT PRIMARY KEY,
  LastName varchar(50),
  FirstName varchar(50),
  DepartmentId int,
  HireDate datetime,
  TerminationDate datetime,
  SupervisorId int,
  CostCenterNumber int,
  EmployeeStatus int,
  EmployeePayRate int)
) GO
CREATE INDEX IX_1 on dbo.Employee (LastName, FirstName, DepartmentId) INCLUDE (HireDate)
CREATE INDEX IX_2 on dbo.Employee (LastName) INCLUDE (EmployeeId, FirstName, DepartmentId)
CREATE INDEX IX_3 on dbo.Employee (LastName, FirstName) INCLUDE (DepartmentId)
CREATE INDEX IX_4 on dbo.Employee (LastName, FirstName) INCLUDE (HireDate, DepartmentId)
GO

INSERT INTO Employee (EmployeeID, LastName, CostCenterNumber) VALUES(1001, 'Employee A', 3001001)
INSERT INTO Employee (EmployeeID, LastName, CostCenterNumber) VALUES(1002, 'Employee B', 3001001)
GO
```

### Application

You have an application that updates the Employees table. The application calls the following stored procedures simultaneously and asynchronously: ▪

UspA: This stored procedure updates only the EmployeeStatus column. ▪

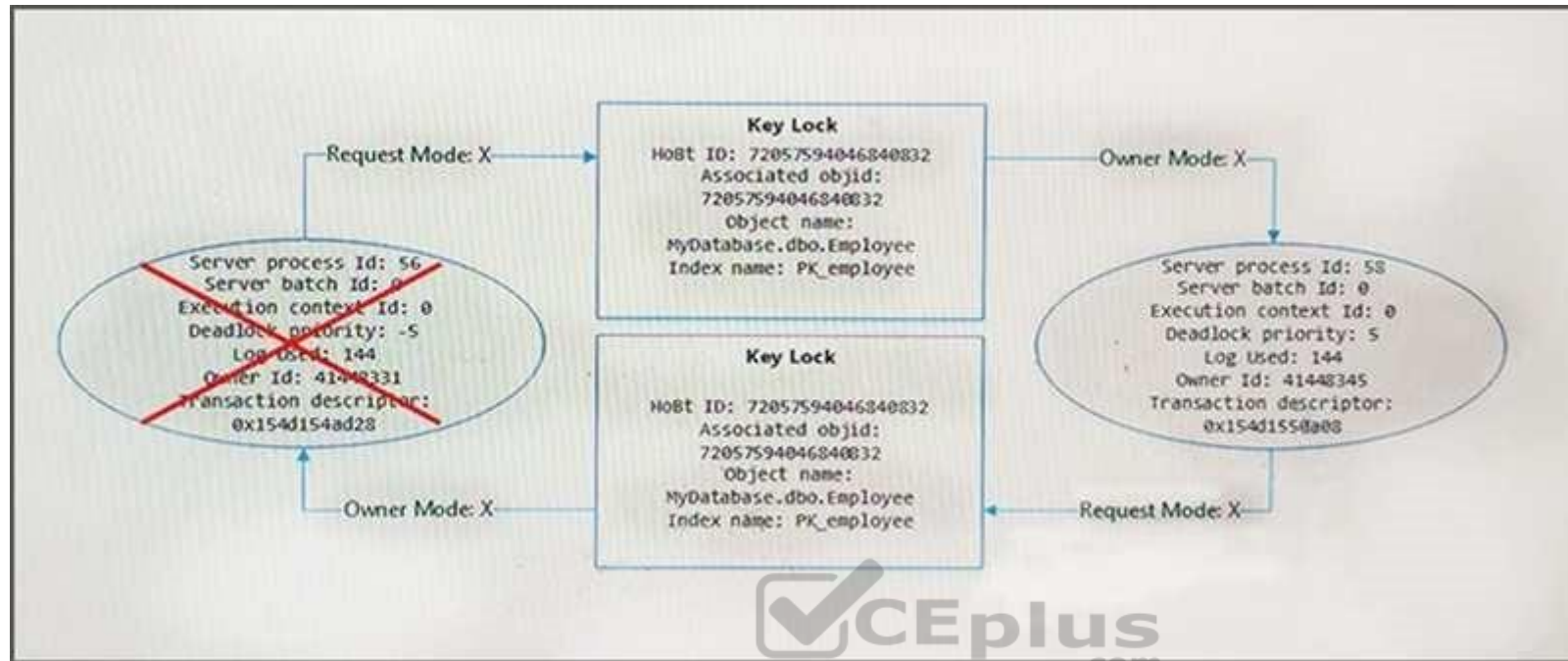
▪ UspB: This stored procedure updates only the EmployeePayRate column. ▪

The application uses views to control access to data. Views must meet the following requirements:

▪ Allow user access to all columns in the tables that the view accesses. ▪

Restrict updates to only the rows that the view returns.

Exhibit



You are analyzing the performance of the database environment. You discover that locks that are held for a long period of time as the reports are generated.

You need to generate the reports more quickly. The database must not use additional resources.

What should you do?

- A. Update all FROM clauses of the DML statements to use the IGNORE\_CONSTRAINTS table hint.
- B. Modify the report queries to use the UNION statement to combine the results of two or more queries.
- C. Apply a nonclustered index to all tables used in the report queries.
- D. Update the transaction level of the report query session to READ UNCOMMITTED.

**Correct Answer: D**

**Section: (none)**

**Explanation**

**Explanation/Reference:**



Explanation:

Transactions running at the READ UNCOMMITTED level do not issue shared locks to prevent other transactions from modifying data read by the current transaction. This is the least restrictive of the isolation levels.

References: [https://technet.microsoft.com/en-us/library/ms173763\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms173763(v=sql.105).aspx)

#### 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 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

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

Instead, create a file group for the indexes and a file group for the data files.

#### QUESTION 69

**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 modify the environment to meet the requirements for the **Orders** table.

What should you create?

- A. an AFTER UPDATE trigger
- B. a user-defined function
- C. a stored procedure with output parameters
- D. an INSTEAD OF INSERT trigger

**Correct Answer: D**

**Section: (none)**

## Explanation

### Explanation/Reference:

Explanation:

From Question: 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.

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

### QUESTION 70

**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 table that has a clustered index and a nonclustered index. The indexes use different columns from the table. You have a query named **Query1** that uses the nonclustered index.

Users report that **Query1** takes a long time to report results. You run **Query1** and review the following statistics for an index seek operation:



### Index Seek (NonClustered)

Scan a particular range of rows from a nonclustered index.

Physical Operation	Index Seek
Logical Operation	Index Seek
Actual Execution Mode	Row
Actual Number of Rows	3571454
Actual Number of Batches	0
Estimated I/O Cost	0.0093577
Estimated Operator Cost	0.0107304 (0%)
Estimated CPU Cost	0.0013727
Estimated Subtree Cost	0.0107304
Estimated Number of Executions	1
Number of Executions	8
Estimated Number of Rows	0
Estimated Row Size	19 B
Actual Rebinds	0
Actual Rewinds	0
Ordered	True
Node ID	100

You need to resolve the performance issue.

Solution: You rebuild the clustered index.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

The query uses the nonclustered index, so improving the clustered index will not help.

We should update statistics for the nonclustered index.

#### **QUESTION 71**

You have a nonpartitioned table that has a single dimension. The table is named **dim.Products.Projections**.

The table is queried frequently by several line-of-business applications. The data is updated frequently throughout the day by two processes.

Users report that when they query data from **dim.Products.Projections**, the responses are slower than expected. The issue occurs when a large number of rows are being updated.

You need to prevent the updates from slowing down the queries.

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.

**Correct Answer:** A

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

The `NOLOCK` hint allows SQL to read data from tables by ignoring any locks and therefore not being blocked by other processes.

This can improve query performance, but also introduces the possibility of dirty reads.

References: <https://www.mssqltips.com/sqlservertip/2470/understanding-the-sql-server-nolock-hint/>

#### **QUESTION 72**

<https://www.vceplus.com/>

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 `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.

**Correct Answer:** E

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

Snapshot isolation enhances concurrency for OLTP applications.

Once snapshot isolation is enabled, updated row versions for each transaction are maintained in tempdb. A unique transaction sequence number identifies each transaction, and these unique numbers are recorded for each row version. The transaction works with the most recent row versions having a sequence number before the sequence number of the transaction. Newer row versions created after the transaction has begun are ignored by the transaction.

References: <https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/sql/snapshot-isolation-in-sql-server>

### QUESTION 73

You have several real-time applications that constantly update data in a database. The applications run more than 400 transactions per second that insert and update new metrics from sensors.

A new web dashboard is released to present the data from the sensors. Engineers report that the applications take longer than expected to commit updates.

You need to change the dashboard queries to improve concurrency and to support reading uncommitted data.



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.

**Correct Answer:** A

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

The `NOLOCK` hint allows SQL to read data from tables by ignoring any locks and therefore not being blocked by other processes. This can improve query performance, but also introduces the possibility of dirty reads.

Incorrect Answers:

F: When `SET XACT_ABORT` is ON, if a Transact-SQL statement raises a run-time error, the entire transaction is terminated and rolled back.

G: `DISABLE`, not `AUTO`, would be better.

There are two more lock escalation modes: `AUTO` and `DISABLE`.

The `AUTO` mode enables lock escalation for partitioned tables only for the locked partition. For non-partitioned tables it works like `TABLE`.

The `DISABLE` mode removes the lock escalation capability for the table and that is important when concurrency issues are more important than memory needs for specific tables.

Note: SQL Server's locking mechanism uses memory resources to maintain locks. In situations where the number of row or page locks increases to a level that decreases the server's memory resources to a minimal level, SQL Server's locking strategy converts these locks to entire table locks, thus freeing memory held by the many single row or page locks to one table lock. This process is called lock escalation, which frees memory, but reduces table concurrency.

References: <https://www.mssqltips.com/sqlservertip/2470/understanding-the-sql-server-nolock-hint/>

#### **QUESTION 74**

You run the following Transact-SQL statements:

<https://www.vceplus.com/>

```
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.

**Correct Answer:** A

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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 75

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

Which Transact-SQL statement should you run?

- A. CREATE INDEX IX\_OrderLines\_1 ON OrderLines (OrderID, UnitPrice) INCLUDE (Description, Quantity)
- B. CREATE INDEX IX\_OrderLines\_1 ON OrderLines (OrderID, UnitPrice) INCLUDE (Quantity)
- C. CREATE INDEX IX\_OrderLines\_1 ON OrderLines (OrderID, UnitPrice, Quantity)
- D. CREATE INDEX IX\_OrderLines\_1 ON OrderLines (UnitPrice, OrderID) INCLUDE (Description, Quantity)

**Correct Answer: A**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

An index with nonkey columns can significantly improve query performance when all columns in the query are included in the index either as key or nonkey columns. Performance gains are achieved because the query optimizer can locate all the column values within the index; table or clustered index data is not accessed resulting in fewer disk I/O operations.

Note: All data types except text, ntext, and image can be used as nonkey columns.

Incorrect Answers:

C: Redesign nonclustered indexes with a large index key size so that only columns used for searching and lookups are key columns.

D: The most unique column should be the first in the index.

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

### QUESTION 76

You are designing a stored procedure for a database named **DB1**.

The following requirements must be met during the entire execution of the stored procedure:

- The stored procedure must only read changes that are persisted to the database.
- `SELECT` statements within the stored procedure should only show changes to the data that are made by the stored procedure.

You need to configure the transaction isolation level for the stored procedure.

Which Transact-SQL statement or statements should you run?

- A. `SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED`  
`ALTER DATABASE DB1 SET READ_COMMITTED_SNAPSHOT ON`
- B. `SET TRANSACTION ISOLATION LEVEL READ COMMITTED`  
`ALTER DATABASE DB1 SET READ_COMMITTED_SNAPSHOT OFF`
- C. `SET TRANSACTION ISOLATION LEVEL SERIALIZABLE`
- D. `SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED`  
`ALTER DATABASE SET READ_COMMITTED_SNAPSHOT OFF`

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

READ COMMITTED specifies that statements cannot read data that has been modified but not committed by other transactions. This prevents dirty reads. Data can be changed by other transactions between individual statements within the current transaction, resulting in nonrepeatable reads or phantom data. This option is the SQL Server default.

Incorrect Answers:

A, D: READ UNCOMMITTED specifies that statements can read rows that have been modified by other transactions but not yet committed.

References: <https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/sql/snapshot-isolation-in-sql-server>

### QUESTION 77

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.
- E. Create a table-valued user-defined function.

**Correct Answer:** A

**Section:** (none)

**Explanation**

### Explanation/Reference:

Explanation:

You can use UNIQUE constraints to make sure that no duplicate values are entered in specific columns that do not participate in a primary key. Although both a UNIQUE constraint and a PRIMARY KEY constraint enforce uniqueness, use a UNIQUE constraint instead of a PRIMARY KEY constraint when you want to enforce the uniqueness of a column, or combination of columns, that is not the primary key.

Incorrect Answers:

D: CHECK constraints enforce domain integrity by limiting the values that are accepted by one or more columns.

References: <https://docs.microsoft.com/en-us/sql/relational-databases/tables/unique-constraints-and-check-constraints?view=sql-server-2017>

#### QUESTION 78

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.

**Correct Answer:** CD

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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 @@TRANSCOUNT = 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 79

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.

**Correct Answer: G**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

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 80

**Note:** This question is part of a series of questions that use the same or similar answer choices. As 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"><li>• The table has 25 columns.</li><li>• The table will contain 10 million records</li><li>• Approximately 100,000 records will be inserted monthly.</li></ul>
TBL2	<ul style="list-style-type: none"><li>• The table has 25 columns.</li><li>• The table will contain 100,000 records.</li><li>• The frequency of inserting, updating, and deleting records is low</li></ul>

You frequently run the following queries:

```
SELECT *  
FROM TBL1  
WHERE Column1 BETWEEN '01/01/2016' AND '30/04/2016'
```

```
SELECT Column5, Column6  
FROM TBL2  
WHERE Column2 = 'ABC156XYZ'
```



There are no foreign key relationships between TBL1 and TBL2.

You need to minimize the amount of time required for the two queries to return records from the tables.

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 column 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. Create a nonclustered columnstore index on TBL1 . Create a nonclustered index on TBL2 .
- F. Drop existing indexes on TBL1 and then create a clustered columnstore index. Create a nonclustered columnstore index on TBL1 . Make no changes to TBL2 .
- G. Create CHECK constraints on both TBL1 and TBL2. Create a partitioned view that combines columns from TBL1 and TBL2.
- H. Create an indexed view that combines columns from TBL1 and TBL2.

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

#### QUESTION 81

**Note:** This question is part of a series of questions that use the same or similar answer choices. As 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"><li>• Column1 is configured as the primary key.</li><li>• The table will contain 20 million records.</li><li>• The table will contain historical data.</li><li>• Most queries of TBL1 return a high percentage of rows from the table with aggregates.</li></ul>
TBL2	<ul style="list-style-type: none"><li>• Column1 is configured as the primary key.</li><li>• The table will contain 25 million records.</li><li>• The frequency of updates and deleted to records in TBL2 is low.</li><li>• Most queries of TBL2 return a high percentage of rows from the table with aggregates.</li></ul>

There are no foreign key relationships between TBL1 and TBL2.

You need to minimize the amount of time required for queries that use data from TBL1 and TBL2 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 column 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. Create a nonclustered columnstore index on TBL1. Create a nonclustered index on TBL2.
- F. Drop existing indexes on TBL1 and then create a clustered columnstore index. Create a nonclustered columnstore index on TBL1. Make no changes to TBL2.
- G. Create CHECK constraints on both TBL1 and TBL2. Create a partitioned view that combines columns from TBL1 and TBL2.
- H. Create an indexed view that combines columns from TBL1 and TBL2.

**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

References: <http://www.sqlservergeeks.com/sql-server-indexing-for-aggregates-in-sql-server/>

## QUESTION 82

**Note: This question is part of a series of questions that use the same or similar answer choices. As 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"> <li>Column1 is configured as the primary key.</li> <li>Column2 will store the year.</li> <li>TBL1 only stores data for the year 2016.</li> <li>The table will contain 1 million records.</li> </ul>
TBL2	<ul style="list-style-type: none"> <li>Column1 is configured as the primary key.</li> <li>Column2 will store the year.</li> <li>TBL2 only stores data for the year 2015.</li> <li>The table will contain 1 million records.</li> </ul>

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.

- Create clustered indexes on TBL1 and TBL2.
- Create a clustered index on TBL1. Create a nonclustered index on TBL2 and add the most frequently queried column as included columns.
- Create a nonclustered index on TBL2 only.
- Create UNIQUE constraints on both TBL1 and TBL2. Create a partitioned view that combines columns from TBL1 and TBL2.
- Drop existing indexes on TBL1 and then create a clustered columnstore index. Create a nonclustered columnstore index on TBL1. Create a nonclustered index on TBL2.
- Drop existing indexes on TBL1 and then create a clustered columnstore index. Create a nonclustered columnstore index on TBL1. Make no changes to TBL2.
- Create CHECK constraints on both TBL1 and TBL2. Create a partitioned view that combines columns from TBL1 and TBL2.
- Create an indexed view that combines columns from TBL1 and TBL2.

**Correct Answer:** D

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

A partitioned view is a view defined by a UNION ALL of member tables structured in the same way, but stored separately as multiple tables in either the same instance of SQL Server or in a group of autonomous instances of SQL Server servers, called federated database servers.

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

### **QUESTION 83**

**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 need to create a stored procedure that updates the Customer, CustomerInfo, OrderHeader, and OrderDetail tables in order.

You need to ensure that the stored procedure:

- Runs within a single transaction.
  - Commits updates to the Customer and CustomerInfo tables regardless of the status of updates to the OrderHeader and OrderDetail tables. ▪
- Commits changes to all four tables when updates to all four tables are successful.

Solution: You create a stored procedure that includes the following Transact-SQL code:

```
BEGIN TRY
  BEGIN TRAN
    UPDATE Customer...
    Update CustomerInfo...

  SAVE TRAN TR1

  UPDATE OrderHeader...
  UPDATE OrderDetail...
  COMMIT TRAN
END TRAN
BEGIN CATCH
  IF XACT_STATE ( ) = 1
    COMMIT TRAN
END CATCH
```



Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

Need to handle the case where the first two updates (OrderHeader, OrderDetail) are successful, but either the 3rd or the 4th (OrderHeader, OrderDetail) fail. Can add a variable in the BEGIN TRY block, and test the variable in the BEGIN CATCH block.

Note: XACT\_STATE indicates whether the request has an active user transaction, and whether the transaction is capable of being committed.

XACT\_STATE =1: the current request has an active user transaction. The request can perform any actions, including writing data and committing the transaction.

References:

<https://docs.microsoft.com/en-us/sql/t-sql/functions/xact-state-transact-sql>

**QUESTION 84**

You have a view that includes an aggregate.

You must be able to change the values of columns in the view. The changes must be reflected in the tables that the view uses.

You need to ensure that you can update the view.

What should you create?

- A. a nonclustered index
- B. a schema-bound view
- C. a stored procedure
- D. an INSTEAD OF trigger



**Correct Answer: B**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

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.

Views or tables that participate in a view created with the SCHEMABINDING clause cannot be dropped unless that view is dropped or changed so that it no longer has schema binding.

References: <https://docs.microsoft.com/en-us/sql/t-sql/statements/create-view-transact-sql>

**QUESTION 85**

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 0  
)
```

```
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 0 FROM CreditValidation CV WHERE CV.CustomerID =  
I.CustomerID AND CV.ValidationDate >= I.InvoiceDate)  
AND I.IsCreditNote = 1  
AND I.IsCreditValidated = 0  
AND I.InvoiceDate >= 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.

**Correct Answer:** ABE

**Section:** (none)

**Explanation**

**Explanation/Reference:**

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 86

**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:

<https://www.vceplus.com/>



Table name	Description
TBL1	-Column1 is configured as the primary key. -The table will contain 20 million records. -Most queries of TBL1 return a high percentage of rows from the table with aggregates.
TBL2	-Column1 has been configured as the primary key. -The table will contain 25 million records. -The frequency of updates and deletes for TBL2 is low. -Most queries of TBL2 return a low percentage of rows and a high percentage of columns.

There are no foreign key relationships between TBL1 and TBL2.

You need to create a query that includes data from both tables and minimizes 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 column 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.  
Create a nonclustered columnstore index on TBL1. Create a nonclustered index on TBL2.
- F. Drop existing indexes on TBL1 and then create a clustered columnstore index. Create a nonclustered columnstore index on TBL1. Make no changes to TBL2.
- G. Create CHECK constraints on both TBL1 and TBL2.  
Create a partitioned view that combines columns from TBL1 and TBL2.
- H. Create an indexed view that combines columns from TBL1 and TBL2.

**Correct Answer:** G

**Section:** (none)

**Explanation**

**Explanation/Reference:**

<https://www.vceplus.com/>

Explanation:

A partitioned view is a view defined by a UNION ALL of member tables structured in the same way, but stored separately as multiple tables in either the same instance of SQL Server or in a group of autonomous instances of SQL Server servers, called federated database servers.

Conditions for Creating Partitioned Views Include:

The select list

- All columns in the member tables should be selected in the column list of the view definition.
- The columns in the same ordinal position of each select list should be of the same type, including collations. It is not sufficient for the columns to be implicitly convertible types, as is generally the case for UNION.

Also, at least one column (for example <col>) must appear in all the select lists in the same ordinal position. This <col> should be defined in a way that the member tables T1, ..., Tn have CHECK constraints C1, ..., Cn defined on <col>, respectively.

References: <https://docs.microsoft.com/en-us/sql/t-sql/statements/create-view-transact-sql>

#### QUESTION 87

You have multiple queries that take a long time to complete.

You need to identify the cause by using detailed information about the Transact-SQL statements in the queries. The Transact-SQL statements must not run as part of the analysis.

Which Transact-SQL statement should you run?

- A. SET STATISTICS PROFILE OFF
- B. SET SHOWPLAN\_TEXT OFF
- C. SET SHOWPLAN\_ALL ON
- D. SET STATISTICS PROFILE ON

**Correct Answer: C**

**Section: (none)**

**Explanation**

**Explanation/Reference:**

Explanation:

SET SHOWPLAN\_ALL ON causes Microsoft SQL Server not to execute Transact-SQL statements. Instead, SQL Server returns detailed information about how the statements are executed and provides estimates of the resource requirements for the statements.

Incorrect Answers:

D: When STATISTICS PROFILE is ON, each executed query returns its regular result set, followed by an additional result set that shows a profile of the query execution.

Displays the profile information for a statement. STATISTICS PROFILE works for ad hoc queries, views, and stored procedures.

References: <https://docs.microsoft.com/en-us/sql/t-sql/statements/set-showplan-all-transact-sql>

### QUESTION 88

You manage a database that supports an Internet of Things (IoT) solution. The database records metrics from over 100 million devices every minute. The database requires 99.995% uptime.

The database uses a table named Checkins that is 100 gigabytes (GB) in size. The Checkins table stores metrics from the devices. The database also has a table named Archive that stores four terabytes (TB) of data. You use stored procedures for all access to the tables.

You observe that the wait type PAGELATCH\_IO causes large amounts of blocking.

You need to resolve the blocking issues while minimizing downtime for the database.

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

- A. Convert all stored procedures that access the Checkins table to natively compiled procedures.
- B. Convert the Checkins table to an In-Memory OLTP table.
- C. Convert all tables to clustered columnstore indexes.
- D. Convert the Checkins table to a clustered columnstore index.

**Correct Answer:** AB

**Section:** (none)

**Explanation**

**Explanation/Reference:**

Explanation:

Natively compiled stored procedures are Transact-SQL stored procedures compiled to native code that access memory-optimized tables. Natively compiled stored procedures allow for efficient execution of the queries and business logic in the stored procedure.

SQL Server In-Memory OLTP helps improve performance of OLTP applications through efficient, memory-optimized data access, native compilation of business logic, and lock- and latch free algorithms. The In-Memory OLTP feature includes memory-optimized tables and table types, as well as native compilation of Transact-SQL stored procedures for efficient access to these tables.

References: <https://docs.microsoft.com/en-us/sql/relational-databases/in-memory-oltp/natively-compiled-stored-procedures> <https://docs.microsoft.com/en-us/sql/relational-databases/in-memory-oltp/memory-optimized-tables>



<https://www.vceplus.com/>



<https://www.vceplus.com/>