

PASS4SURES.COM

A Composite Solution With Just One Click

Microsoft

70-464 PRACTICE EXAM

Developing Microsoft SQL Server 2012 Databases

Product Questions: 189

Version: 17.0

Case Study: 1

Scenario 1

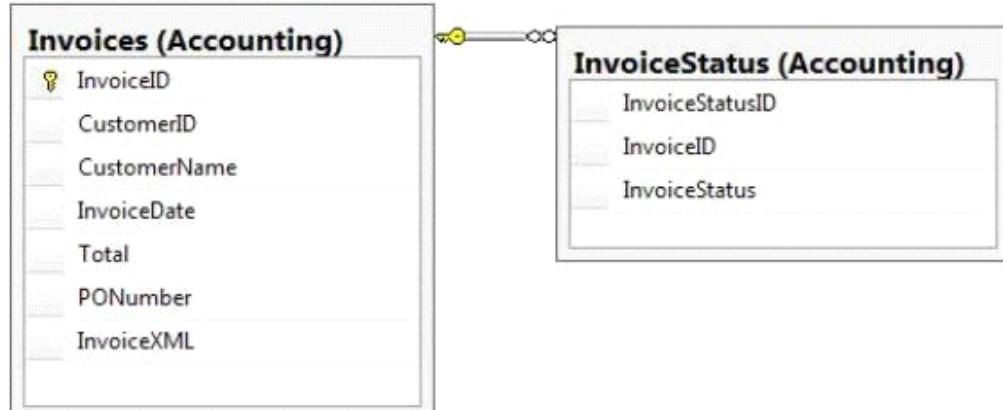
Application Information

Your company receives invoices in XML format from customers. Currently, the invoices are stored as files and processed by a desktop application. The application has several performance and security issues. The application is being migrated to a SQL Server-based solution. A schema named InvoiceSchema has been created for the invoices xml.

The data in the invoices is sometimes incomplete. The incomplete data must be stored and processed as-is. Users cannot filter the data provided through views.

You are designing a SQL Server database named DB1 that will be used to receive, process, and securely store the invoice data. A third-party Microsoft .NET Framework component will be purchased to perform tax calculations. The third-party tax component will be provided as a DLL file named Treytax.dll and a source code file named Amortize.cs. The component will expose a class named TreyResearch and a method named Amortize(). The files are located in c:\temp\.

The following graphic shows the planned tables:



You have a sequence named Accounting.InvoiceID_Seq.

You plan to create two certificates named CERT1 and CERT2. You will create CERT1 in master. You will create CERT2 in DB1.

You have a legacy application that requires the ability to generate dynamic T-SQL statements against DB1. A sample of the queries generated by the legacy application appears in Legacy.sql.

Application Requirements

The planned database has the following requirements:

- All stored procedures must be signed.
- The original XML invoices must be stored in the database.
- An XML schema must be used to validate the invoice data.
- Dynamic T-SQL statements must be converted to stored procedures.

- Access to the .NET Framework tax components must be available to T-SQL objects.
- Columns must be defined by using data types that minimize the amount of space used by each table.
- Invoices stored in the InvoiceStatus table must refer to an invoice by the same identifier used by the Invoice table.
- To protect against the theft of backup disks, invoice data must be protected by using the highest level of encryption.
- The solution must provide a table-valued function that provides users with the ability to filter invoices by customer.
- Indexes must be optimized periodically based on their fragmentation by using the minimum amount of administrative effort.

Usp_InsertInvoices.sql

```
01 CREATE PROCEDURE InsertInvoice @XML nvarchar(1000)
02 AS
03 DECLARE @XmlDocumentHandle INT;
04 DECLARE @XmlDocument nvarchar(1000);
05 SET @XmlDocument = @XML;
06
07 EXEC sp_xml_preparedocument @XmlDocumentHandle OUTPUT, @XmlDocument;
08
09 INSERT INTO DB1.Accounting.Invoices (
10     InvoiceID,
11     InvoiceXML,
12     CustomerID,
13     CustomerName,
14     InvoiceDate,
15     Total,
16     PONumber
17 )
18 SELECT (NEXT VALUE FOR Accounting.InvoiceID_Seq),
19     @XML, * FROM OPENXML (@XmlDocumentHandle, '/Invoice',2)
20 WITH (
21     CustomerID nvarchar(11) 'Customer/@ID',
22     CustomerName nvarchar(50) 'Customer/@Name',
23     InvoiceDate date 'InvoiceDate',
24     Total decimal(8, 2) 'Total',
25     PONumber bigint 'PONumber'
26 );
27
28 EXEC sp_xml_removedocument @XmlDocumentHandle;
```

Invoices.xml

All customer IDs are 11 digits. The first three digits of a customer ID represent the customer's country. The remaining eight digits are the customer's account number.

The following is a sample of a customer invoice in XML format:

```
01 <?xml version="1.0"?>
02 <Invoice InvoiceDate="2012-02-20">
03     <Customer ID="00156590099" Name="Litware" />
04     <Total>125</Total>
05     <PONumber>1666</PONumber>
06 </Invoice>
```

InvoicesByCustomer.sql

```
01 (SELECT CustomerID,
02   CustomerName,
03   InvoiceID,
04   InvoiceDate,
05   Total,
06   PONumber
07   FROM Accounting.Invoices
08 WHERE CustomerID=@CustID);
```

Legacy.sql

```
01 DECLARE @sqlstring AS nvarchar(1000);
02 DECLARE @CustomerID AS varchar(11), @Total AS decimal(8,2);
03
04 SET @sqlstring=N'SELECT CustomerID, InvoiceID, Total
05   FROM Accounting.Invoices
06 WHERE CustomerID=@CustomerID AND Total > @Total;';
07
08 EXEC sys.sp_executesql
09   @statement=@sqlstring,
10   @params=N'@CustomerID AS varchar(11), @Total AS decimal(8,2)',
11   @CustomerID=999, @Total=500;
```

CountryFromID.sql

```
01 CREATE FUNCTION CountryFromID (@CustomerID varchar(11)) RETURNS varchar(20)
02 AS
03 BEGIN
04   DECLARE @Country varchar(20);
05   SET @CustomerID = LEFT(@CustomerID,3);
06   SELECT @Country = CASE @CustomerID
07     WHEN '001'
08       THEN 'United States'
09     WHEN '002'
10       THEN 'Spain'
11     WHEN '003'
12       THEN 'Japan'
13     WHEN '004'
14       THEN 'China'
15     WHEN '005'
16       THEN 'Brazil'
17     ELSE 'Other'
18   END;
19   RETURN @Country;
20 END;
```

IndexManagement.sql

```

01 DECLARE @IndexTable TABLE (
02     TableName varchar(100), IndexName varchar(100), Fragmentation int, RowNumber int
03 );
04 DECLARE @TableName sysname, @IndexName sysname, @Fragmentation int,
05     @RowNumber int, @sqlcommand varchar(1000);
06
07 INSERT INTO @IndexTable (TableName, IndexName, Fragmentation, Rownumber)
08     SELECT OBJECT_NAME(i.Object_id),
09         i.name AS IndexName,
10         indexstats.avg_fragmentation_in_percent,
11         ROW_NUMBER() OVER(ORDER BY i.name DESC) AS 'RowNumber'
12     FROM sys.dm_db_index_physical_stats(DB_ID(), NULL, NULL, NULL, 'DETAILED')
13     AS indexstats INNER JOIN sys.indexes AS i
14     ON i.OBJECT_ID = indexstats.OBJECT_ID AND i.index_id = indexstats.index_id;
15
16 DECLARE @counter int = 0;
17
18 WHILE @counter < (SELECT RowNumber FROM @indextable)
19     BEGIN
20         SET @counter = @counter + 1;
21         WITH t AS (
22             SELECT TableName, IndexName, Fragmentation
23             FROM @IndexTable WHERE RowNumber = @counter
24         )
25         SELECT
26             @TableName= TableName,
27             @IndexName = IndexName,
28             @Fragmentation = Fragmentation
29         FROM t;
30
31         IF @Fragmentation <= 30
32             BEGIN
33                 SET @sqlCommand =
34                     N'ALTER INDEX '+@indexName+N' ON '+@TableName+N' REORGANIZE';
35                 EXEC sp_executesql @sqlCommand;
36             END;
37         ELSE
38             BEGIN
39                 SET @sqlCommand=N'ALTER INDEX '+@indexName+N' ON '+@TableName+N' REBUILD';
40                 EXEC sp_executesql @sqlCommand;
41             END;
42     END;

```

Question: 1

You are testing disaster recovery procedures.

You attempt to restore DB1 to a different server and you receive the following error message: "Msg 33111.

Level 16, State 3, Line 1

Cannot find server certificate with thumbprint

,0xA694FBEA88C9354E5E2567C30A2A69E8FB4C44A9\

Msg 3013, Level 16, State 1, Line 1

RESTORE DATABASE is terminating abnormally."

You need to ensure that you can restore DB1 to a different server.

Which code segment should you execute?

- A. RESTORE CERTIFICATE CERT2
FROM FILE='CERT2.CER'
WITH PRIVATE KEY (FILE = 'CERT2.KEY',
DECRYPTION BY PASSWORD='p@ssw0rd1');
- B. CREATE CERTIFICATE CERT1
FROM FILE='CERT1.CER'
WITH PRIVATE KEY (FILE = 'CERT1.KEY',
DECRYPTION BY PASSWORD='p@ssw0rd1');
- C. CREATE CERTIFICATE CERT2
ENCRYPTION BY PASSWORD='p@ssw0rd1'
WITH SUBJECT = 'EncryptionCertificate';
- D. CREATE CERTIFICATE CERT1
ENCRYPTION BY PASSWORD='p@ssw0rd1'
WITH SUBJECT = 'EncryptionCertificate';

- A. Option A
B. Option B
C. Option C
D. Option D

Answer: B

Question: 2

You need to create the InvoiceStatus table in DB1.
How should you define the InvoiceID column in the CREATE TABLE statement?

- A. InvoiceID bigint
DEFAULT (NEXT VALUE FOR Accounting.InvoiceID_Seq) NOT NULL,
- B. InvoiceID bigint DEFAULT ((NEXT VALUE
FOR Accounting.InvoiceID_Seq OVER
(ORDER BY InvoiceStatusID))) NOT NULL FOREIGN
KEY REFERENCES Accounting.Invoices(InvoiceID),
- C. InvoiceID bigint FOREIGN KEY REFERENCES
Accounting.Invoices(InvoiceID) NOT NULL,
- D. InvoiceID bigint DEFAULT ((NEXT VALUE
FOR Accounting.InvoiceID_Seq
OVER (ORDER BY InvoiceStatusID))) NOT NULL,

- A. Option A
B. Option B
C. Option C
D. Option D

Answer: C

Question: 3

Which data type should you use for CustomerID?

- A. varchar(11)
- B. bigint
- C. nvarchar(11)
- D. char(11)

Answer: D

Explanation:

Invoices.xml

All customer IDs are 11 digits. The first three digits of a customer ID represent the customer's country. The remaining eight digits are the customer's account number.

int: -2^31 (-2,147,483,648) to 2^31-1 (2,147,483,647) (just 10 digits max)

bigint: -2^63 (-9,223,372,036,854,775,808) to 2^63-1 (9,223,372,036,854,775,807)

<http://msdn.microsoft.com/en-us/library/ms176089.aspx>

<http://msdn.microsoft.com/en-us/library/ms187745.aspx>

Question: 4

You need to modify InsertInvoice to comply with the application requirements.

Which code segment should you execute?

A. OPEN CERT1;
ALTER PROCEDURE Accounting.usp_InsertInvoice
WITH ENCRYPTION;
CLOSE CERT1;

B. OPEN CERT2;
ALTER PROCEDURE Accounting.usp_InsertInvoice
WITH ENCRYPTION;
CLOSE CERT2;

C. ADD SIGNATURE TO Accounting.usp_InsertInvoice
BY CERTIFICATE CERT1;

D. ADD SIGNATURE TO Accounting.usp_InsertInvoice
BY CERTIFICATE CERT2;

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: D

Question: 5

You attempt to process an invoice by using usp_InsertInvoice.sql and you receive the following error message: "Msg 515, Level 16, State 2, Procedure usp_InsertInvoice, Line 10
Cannot insert the value NULL into column 'InvoiceDate', table 'DB1.Accounting.Invoices'; column does not allow nulls.
INSERT fails."

You need to modify usp_InsertInvoice.sql to resolve the error.
How should you modify the INSERT statement?

- A. InvoiceDate varchar(100) 'InvoiceDate',
- B. InvoiceDate varchar(100) 'Customer/InvoiceDate',
- C. InvoiceDate date '@InvoiceDate',
- D. InvoiceDate date 'Customer/@InvoiceDate',

Answer: C

Question: 6

You need to modify the function in CountryFromID.sql to ensure that the country name is returned instead of the country ID.

Which line of code should you modify in CountryFromID.sql?

- A. 04
- B. 05
- C. 06
- D. 19

Answer: D

Explanation:

<http://msdn.microsoft.com/en-us/library/ms186755.aspx>
<http://msdn.microsoft.com/en-us/library/ms191320.aspx>

Question: 7

You execute IndexManagement.sql and you receive the following error message:

"Msg 512, Level 16, State 1, Line 12

Subquery returned more than 1 value. This is not permitted when the subquery follows =,! =, <, <= ,>, > = or when the subquery is used as an expression."

You need to ensure that IndexManagement.sql executes properly.

Which WHILE statement should you use at line 18?

- A.WHILE SUM(@RowNumber) < (SELECT @counter FROM @indextable)
- B.WHILE @counter < (SELECT COUNT(RowNumber) FROM @indextable)
- C.WHILE COUNT(@RowNumber) < (SELECT @counter FROM @indextable)
- D.WHILE @counter < (SELECT SUM(RowNumber) FROM @indextable)

Answer: B

Question: 8

You need to convert the functionality of Legacy.sql to use a stored procedure.
Which code segment should the stored procedure contain?

- A. CREATE PROC usp_InvoicesByCustomerAboveTotal(
 @sqlstring AS nvarchar(1000),
 @CustomerID AS char(11),
 @Total AS decimal(8,2))
 AS
 ...
- B. CREATE PROC usp_InvoicesByCustomerAboveTotal(
 @sqlstring AS nvarchar(1000))
 AS
 ...
- C. CREATE PROC usp_InvoicesByCustomerAboveTotal(
 @sqlstring AS nvarchar(1000),
 OUTPUT @CustomerID AS char(11),
 OUTPUT @Total AS decimal(8,2))
 AS
 ...
- D. CREATE PROC usp_InvoicesByCustomerAboveTotal (
 @CustomerID AS char(11), @Total AS decimal(8,2))
 AS
 ...

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: D

Explanation:

- http://msdn.microsoft.com/en-us/library/ms187926.aspx
- http://msdn.microsoft.com/en-us/library/ms190782.aspx
- http://msdn.microsoft.com/en-us/library/bb669091.aspx
- http://msdn.microsoft.com/en-us/library/windows/desktop/ms709342.aspx
- http://msdn.microsoft.com/en-us/library/ms188001.aspx

Question: 9

You need to create a function that filters invoices by CustomerID. The SELECT statement for the function is contained in InvoicesByCustomer.sql.

Which code segment should you use to complete the function?

- A. CREATE FUNCTION Accounting.fnInvoicesByCustomerTest (@CustID varchar(11))
RETURNS @TblInvoices TABLE (CustomerID bigint, CustomerName NVARCHAR(255),
InvoiceID bigint, InvoiceDate date, Total decimal(8,2), PONumber bigint)
AS
- B. CREATE FUNCTION Accounting.fnInvoicesByCustomer (@CustID varchar(11))
RETURNS @tblInvoices TABLE (CustomerID bigint, CustomerName NVARCHAR(255),
InvoiceID bigint, InvoiceDate date, Total decimal(8,2), PONumber bigint)
AS
INSERT INTO @tblInvoices
- C. CREATE FUNCTION Accounting.fnInvoicesByCustomer (@CustID varchar(11))
RETURNS xml
AS
RETURN
- D. CREATE FUNCTION Accounting.fnInvoicesByCustomerTest (@CustID varchar(11))
RETURNS @TblInvoices TABLE (CustomerID bigint, CustomerName NVARCHAR(255),
InvoiceID bigint, InvoiceDate date, Total decimal(8,2), PONumber bigint)
AS

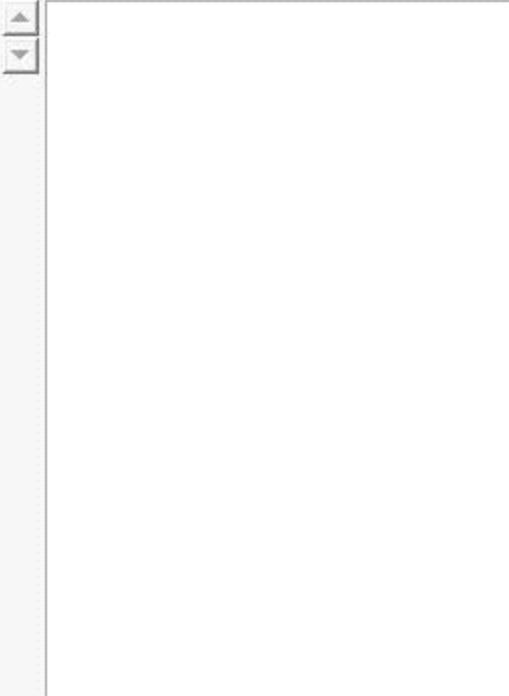
- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: A

Question: 10

DRAG DROP

You need to build a stored procedure that amortizes the invoice amount. Which code segment should you use to create the stored procedure? To answer, move the appropriate code segments from the list of code segments to the answer area and arrange them in the correct order.

Ordered List Title	Answer Choices Title
 <p style="text-align: right;"> << Move Remove >> </p>	<pre> RECONFIGURE; EXEC sp_configure 'clr enabled', '1'; EXEC sp_recomplie @objname = 'TaxCalc' CREATE PROCEDURE Accounting.Amortize(@total decimal(8,2), @period int) RETURNS decimal(8,2) AS EXTERNAL NAME TaxCalc.TreyResearch.Amortize; CREATE ASSEMBLY TaxCalc FROM 'C:\temp\TreyTax.DLL' CREATE ASSEMBLY TaxCalc FROM 'C:\temp\Amortize.cs' </pre>

Answer:

```

EXEC sp_configure 'clr enabled', '1';
RECONFIGURE;
CREATE ASSEMBLY TaxCalc FROM
'C:\temp\TreyTax.DLL'
CREATE PROCEDURE
Accounting.Amortize(@total
decimal(8,2), @period int)
RETURNS decimal(8,2)
AS EXTERNAL NAME
TaxCalc.TreyResearch.Amortize;

```

Explanation:

<http://msdn.microsoft.com/en-us/library/ms131089.aspx>
<http://msdn.microsoft.com/en-us/library/ms131048.aspx>
<http://msdn.microsoft.com/en-us/library/ms187926.aspx>

Case Study: 2

Scenario 2

Application Information

You have two servers named SQL1 and SQL2 that have SQL Server 2012 installed.

You have an application that is used to schedule and manage conferences.

Users report that the application has many errors and is very slow.

You are updating the application to resolve the issues.

You plan to create a new database on SQL1 to support the application. A junior database administrator has created all the scripts that will be used to create the database. The script that you plan to use to create the

tables for the new database is shown in Tables.sql. The script that you plan to use to create the stored procedures for the new database is shown in StoredProcedures.sql. The script that you plan to use to create the indexes for the new database is shown in Indexes.sql. (Line numbers are included for reference only.) A database named DB2 resides on SQL2. DB2 has a table named SpeakerAudit that will audit changes to a table named Speakers.

A stored procedure named usp_UpdateSpeakersName will be executed only by other stored procedures. The stored procedures executing usp_UpdateSpeakersName will always handle transactions.

A stored procedure named usp_SelectSpeakersByName will be used to retrieve the names of speakers. Usp_SelectSpeakersByName can read uncommitted data.

A stored procedure named usp_GetFutureSessions will be used to retrieve sessions that will occur in the future.

Procedures.sql

```
01 CREATE PROCEDURE usp_UpdateSpeakerName
02   @SpeakerID int,
03   @LastName nvarchar(100)
04 AS
05
06 BEGIN TRY
07
08 UPDATE Speakers
09 SET LastName = @LastName
10 WHERE SpeakerID = @SpeakerID;
11
12 INSERT INTO SQL2.DB2.dbo.SpeakerAudit(SpeakerID, LastName)
13 VALUES (@SpeakerID, @LastName);
14
15 END TRY
16 BEGIN CATCH
17
18 END CATCH;
19
20 GO
21
22 CREATE PROCEDURE usp_SelectSpeakersByName
23   @LastName nvarchar(100)
24 AS
25 SELECT SpeakerID,
26   FirstName,
27   LastName
28 FROM Speakers
29 WHERE LastName LIKE @LastName + '%'
30
31 GO
32
33 CREATE PROCEDURE usp_InsertSessions
34   @SessionData SessionDataTable READONLY
35 AS
36 INSERT INTO Sessions
37   (SpeakerID, Title, Absract, DeliveryTime, TitleAndSpeaker)
38 SELECT SpeakerID, Title, Absract, DeliveryTime, TitleAndSpeaker
39 FROM @SessionData;
40 GO
41
42 CREATE PROCEDURE usp_UpdateSessionRoom
43   @RoomID int,
44   @SpeakerID int
45 AS
```

```
46 SET TRANSACTION ISOLATION LEVEL SNAPSHOT
47 BEGIN TRANSACTION;
48
49 SELECT SessionID,
50     Title
51 FROM Sessions
52 WHERE SpeakerID = @SpeakerID;
53
54 UPDATE Sessions
55 SET RoomID = @RoomID
56 WHERE SpeakerID = @SpeakerID;
57
58 COMMIT TRANSACTION;
59
60 CREATE PROCEDURE usp_AttendeesReport
61     @LastName varchar(100)
62 AS
63 SELECT FirstName + ' ' + LastName AS FullName
64 FROM Attendees
65 WHERE LastName = @LastName;
66 GO
67
68 CREATE PROCEDURE usp_GetFutureSessions
69 AS
70 SELECT SpeakerID,
71     RoomID,
72     DeliveryTime
73 FROM Sessions
74
75 GO
76
77 CREATE PROCEDURE usp_TestSpeakers
78 AS
79 EXECUTE usp_SelectSpeakersByName 'a';
80 EXECUTE usp_SelectSpeakersByName 'an';
81 EXECUTE usp_SelectSpeakersByName 'and';
82 EXECUTE usp_SelectSpeakersByName 'ander';
83 EXECUTE usp_SelectSpeakersByName 'anderson';
84 EXECUTE usp_SelectSpeakersByName 'b';
85 EXECUTE usp_SelectSpeakersByName 'bi';
86 ...
87 EXECUTE usp_SelectSpeakersByName 'zzz';
88 GO
```

Indexes.sql

```
01 CREATE INDEX IX_Sessions ON Sessions
02 (SessionID, DeliveryTime)
03 INCLUDE (RoomID)
04
05 GO
06
07 CREATE INDEX IX_Speakers ON Speakers
08 (LastName);
09 GO
10
11 CREATE INDEX IX_Attendees_Name ON Attendees
12 (FirstName, LastName);
13
14 GO
15
16 CREATE INDEX IX_Attendees_Confirmed ON Attendees
17 (Confirmed);
18 GO
```

Tables.sql

```
01 CREATE DATABASE Conference;
02 GO
03
04 ALTER DATABASE Conference
05 SET READ_COMMITTED_SNAPSHOT ON;
06 GO
07
08 CREATE TABLE Attendees
09 (
10     AttendeeID int IDENTITY (1,1) NOT NULL,
11     FirstName nvarchar(100) NOT NULL,
12     LastName nvarchar(100) NOT NULL,
13     EmailAddress nvarchar(100) NOT NULL,
14
15     CONSTRAINT PK_Attendees_AttendeeID PRIMARY KEY (AttendeeID)
16 );
17 GO
18
19 CREATE TABLE Speakers
20 (
21     SpeakerID int IDENTITY(1,1) NOT NULL,
22     FirstName nvarchar(100) NOT NULL,
23     LastName nvarchar(100) NOT NULL,
24     Photo varbinary(max),
25     CONSTRAINT PK_Speakers_SpeakerID PRIMARY KEY (SpeakerID)
26 );
27 GO
28
29 CREATE TABLE Sessions
30 (
31     SessionID uniqueidentifier NOT NULL
32     CONSTRAINT DF_SessionID DEFAULT (NEWID()),
33     SpeakerID int NOT NULL,
34     Title nvarchar(100) NOT NULL,
35     Abstract nvarchar(max) NOT NULL,
36     DeliveryTime datetime NOT NULL,
37     TitleAndSpeaker nvarchar(200)
38
39 );
40 GO
41
42 CREATE TABLE Rooms
43 (
44     RoomID uniqueidentifier NOT NULL CONSTRAINT DF_RoomID DEFAULT (NEWID()),
45     Location varchar(100) NOT NULL
46 );
```

Question: 1

You need to provide referential integrity between the Sessions table and Speakers table.
Which code segment should you add at line 47 of Tables.sql?

- A.

```
ALTER TABLE dbo.Sessions ADD CONSTRAINT
FK_Sessions_Speakers FOREIGN KEY (SessionID)
REFERENCES dbo.Speakers (SpeakerID);
```
- B.

```
ALTER TABLE dbo.Sessions ADD CONSTRAINT
FK_Sessions_Speakers FOREIGN KEY (SpeakerID)
REFERENCES dbo.Speakers (SpeakerID);
```
- C.

```
ALTER TABLE dbo.Speakers ADD CONSTRAINT
FK_Speakers_Sessions FOREIGN KEY (SpeakerID)
REFERENCES dbo.Sessions (SessionID);
```
- D.

```
ALTER TABLE dbo.Speakers ADD CONSTRAINT
FK_Speakers_Sessions FOREIGN KEY (SessionID)
REFERENCES dbo.Sessions (SessionID);
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: B

Explanation:

<http://msdn.microsoft.com/en-us/library/ms189049.aspx>
<http://msdn.microsoft.com/en-us/library/ms179610.aspx>
<http://msdn.microsoft.com/en-us/library/ff878370.aspx>

Question: 2

You execute `usp_TestSpeakers`.
You discover that `usp_SelectSpeakersByName` uses inefficient execution plans.
You need to update `usp_SelectSpeakersByName` to ensure that the most efficient execution plan is used.
What should you add at line 30 of `Procedures.sql`?

- A. OPTION (FORCESCAN)
- B. OPTION (FORCESEEK)
- C. OPTION (OPTIMIZE FOR UNKNOWN)
- D. OPTION (OPTIMIZE FOR (@LastName= 'Anderson'))

Answer: C

Explanation:

<http://msdn.microsoft.com/en-us/library/ms181714.aspx>

Question: 3

You need to recommend a solution to ensure that SQL1 supports the auditing requirements of `usp_UpdateSpeakerName`.

What should you include in the recommendation?

- A. The Distributed Transaction Coordinator (DTC)
- B. Transactional replication
- C. Change data capture
- D. Change tracking

Answer: A

Question: 4

You are evaluating the table design.

You need to recommend a change to Tables.sql that reduces the amount of time it takes for usp_AttendeesReport to execute.

What should you add at line 14 of Tables.sql?

- A. FullName nvarchar(100) NOT NULL CONSTRAINT DF_FullName DEFAULT (dbo.CreateFullName (FirstName, LastName)),
- B. FullName AS (FirstName +''+ LastName),
- C. FullName nvarchar(100) NOT NULL DEFAULT (dbo.CreateFullName (FirstName, LastName)).
- D. FullName AS (FirstName +''+ LastName) PERSISTED,

Answer: D

Explanation:

<http://msdn.microsoft.com/en-us/library/ms188300.aspx>

<http://msdn.microsoft.com/en-us/library/ms191250.aspx>

Question: 5

You need to modify usp_SelectSpeakersByName to support server-side paging. The solution must minimize the amount of development effort required.

What should you add to usp_SelectSpeakersByName?

- A. A table variable
- B. An OFFSET-FETCH clause
- C. The ROWNUMBER keyword
- D. A recursive common table expression

Answer: B

Explanation:

<http://www.mssqltips.com/sqlservertip/2696/comparing-performance-for-different-sql-server-paging-methods/>

<http://msdn.microsoft.com/en-us/library/ms188385.aspx>

<http://msdn.microsoft.com/en-us/library/ms180152.aspx>

<http://msdn.microsoft.com/en-us/library/ms186243.aspx>

<http://msdn.microsoft.com/en-us/library/ms186734.aspx>

<http://www.sqlserver-training.com/how-to-use-offset-fetch-option-in-sql-server-order-by-clause/>

http://www.sqlservercentral.com/blogs/juggling_with_sql/2011/11/30/using-offset-and-fetch/

Question: 6

You need to add a new column named Confirmed to the Attendees table.

The solution must meet the following requirements:

Have a default value of false.

Minimize the amount of disk space used.

Which code block should you use?

A. ALTER TABLE Attendees

ADD Confirmed bit DEFAULT 0;

B. ALTER TABLE Attendees

ADD Confirmed char(1) DEFAULT '1';

C. ALTER TABLE Attendees

ADD Confirmed bit DEFAULT 1;

D. ALTER TABLE Attendees

ADD Confirmed char(1) DEFAULT '1';

Answer: A

Explanation:

<http://msdn.microsoft.com/en-us/library/ms177603.aspx>

Question: 7

You need to create the object used by the parameter of usp_InsertSessions.

Which statement should you use?

A. CREATE XML SCHEMA COLLECTION SessionDataTable

B. CREATE TYPE SessionDataTable AS Table

C. CREATE SCHEMA SessionDataTable

D. CREATE TABLE SessionDataTable

Answer: B

Question: 8

Developers report that usp_UpdateSessionRoom periodically returns error 3960.

You need to prevent the error from occurring. The solution must ensure that the stored procedure returns the original values to all of the updated rows.

What should you configure in Procedures.sql?

A. Replace line 46 with the following code:

SET TRANSACTION ISOLATION LEVEL SERIALIZABLE

B. Replace line 46 with the following code:

SET TRANSACTION ISOLATION LEVEL REPEATABLE READ

C. Move the SELECT statement at line 49 to line 57.

D. Move the SET statement at line 46 to line 53.

Answer: A

Question: 9

You discover that usp.SelectSpeakersByName executes slowly if usp_UpdateSpeakerName executes simultaneously. You need to minimize the execution time of usp.SelectSpeakersByName. The solution must not affect the performance of the other stored procedures.

What should you update?

- A. Usp_UpdateSpeakerName to use the NOLOCK query hint
- B. Usp_UpdateSpeakerName to use snapshot isolation
- C. Usp_SelectSpeakersByName to use the NOLOCK query hint
- D. Usp_SelectSpeakersByName to use snapshot isolation

Answer: C

NOLOCK

Is equivalent to READUNCOMMITTED.

READUNCOMMITTED

Specifies that dirty reads are allowed.

Question: 10

You are evaluating the index design.

You need to recommend a change to Indexes.sql that will minimize the amount of time it takes for usp_AttendeesReport to execute. The solution must minimize the amount of database fragmentation.

Which line of code should you use to replace line 12 of Indexes.sql?

- A. (LastName);
- B. (FirstName) INCLUDE (LastName);
- C. (LastName, FirstName);
- D. (LastName) INCLUDE (FirstName);

Answer: C

Question: 11

While testing usp.GetFutureSessions, you discover that IX_Sessions is accessed by a scan rather than a seek.

You need to minimize the amount of time it takes to execute usp_GetFutureSessions.

What should you do? (Each correct answer presents part of the solution. Choose all that apply.)

- A. Change line 02 of Indexes.sql to:

(DeliveryTime, SessionID)

- B. At line 04 of Indexes.sql, add:

WHERE GETDATE() < DeliveryTime;

- C. Change line 02 of Indexes.sql to:

(SpeakerID, RoomID, DeliveryTime)

- D. Change line 74 of Procedures.sql to:

WHERE GETDATE() > DeliveryTime;

- E. Change line 74 of Procedures.sql to:

WHERE GETDATE() < DeliveryTime;

- F. At line 04 of Indexes.sql, add:

WHERE GETDATE() > DeliveryTime;

A. Option A

B. Option B

C. Option C

D. Option D

E. Option E

F. Option F

Answer: BE

Future delivery dates.

Question: 12

You need to ensure that if any of the statements in usp_UpdateSpeakerName return an error message, all of the changes executed by usp_UpdateSpeakerName are not committed to the database.

What should you do in Procedures.sql? (Each correct answer presents part of the solution. Choose all that apply.)

- A. Add the following at line 17:

ROLLBACK TRANSACTION

- B. Add the following at line 05:

BEGIN TRANSACTION SpeakerUpdate

- C. Add the following at line 05:

SAVE TRANSACTION SpeakerUpdate

- D. Add the following at line 17:

ROLLBACK TRANSACTION SpeakerUpdate

- E. Add the following at line 07:

BEGIN TRANSACTION

- A. Option A
- B. Option B
- C. Option C
- D. Option D
- E. Option E

Answer: BD

Question: 13

You need to create the object used by the parameter of usp_InsertSessions.
Which statement should you use?

- A. CREATE SCHEMA SessionDataTable
- B. CREATE TYPE SessionDataTable AS Table
- C. CREATE TABLE SessionDataTable
- D. CREATE XML SCHEMA COLLECTION SessionDataTable

Answer: A

Case Study: 3

Scenario 3

Application Information

You have two servers named SQL1 and SQL2. SQL1 has SQL Server 2012 Enterprise installed. SQL2 has SQL Server 2008 Standard installed.

You have an application that is used to manage employees and office space.

Users report that the application has many errors and is very slow.

You are updating the application to resolve the issues.

You plan to create a new database on SQL1 to support the application. The script that you plan to use to create the tables for the new database is shown in Tables.sql. The script that you plan to use to create the stored procedures for the new database is shown in StoredProcedures.sql. The script that you plan to use to create the indexes for the new database is shown in Indexes.sql.

A database named DB2 resides on SQL2. DB2 has a table named EmployeeAudit that will audit changes to a table named Employees.

A stored procedure named usp_UpdateEmployeeName will be executed only by other stored procedures. The stored procedures executing usp_UpdateEmployeeName will always handle transactions.

A stored procedure named usp_SelectEmployeesByName will be used to retrieve the names of employees. Usp_SelectEmployeesByName can read uncommitted data.

A stored procedure named usp_GetFutureOfficeAssignments will be used to retrieve office assignments that will occur in the future.

StoredProcedures.sql

```

01 CREATE PROCEDURE usp_UpdateEmployeeName
02   @EmployeesInfo EmployeesInfo READONLY
03 AS
04
05 BEGIN TRY
06
07 UPDATE Employees
08 SET LastName = ei.LastName
09 FROM Employees e
10   INNER JOIN @EmployeesInfo ei ON e.EmployeeID = ei.EmployeeID;
11
12 INSERT INTO SQL2.DB2.dbo.EmployeeAudit(EmployeeID, LastName)
13 SELECT EmployeeID, LastName
14 FROM @EmployeesInfo;
15
16 END TRY
17 BEGIN CATCH
18
19 END CATCH;
20
21 GO
22
23 CREATE PROCEDURE usp_SelectEmployeesByName
24   @LastName nvarchar(100)
25 AS
26 SELECT EmployeeID,
27   FirstName,
28   LastName
29 FROM Employees
30 WHERE LastName LIKE @LastName + '%'
31
32 GO
33
34 CREATE PROCEDURE usp_UpdateOffice
35   @OfficeID int,
36   @EmployeeID int
37 AS
38 SET TRANSACTION ISOLATION LEVEL SNAPSHOT
39 BEGIN TRANSACTION;
40
41 SELECT OfficeID,
42   OfficeName
43 FROM Offices
44 WHERE EmployeeID = @EmployeeID;
45
46 UPDATE Offices
47 SET EmployeeID = @EmployeeID,
48   StartDate = GETDATE()
49 WHERE OfficeID = @OfficeID;
50
51 COMMIT TRANSACTION;
52
53 CREATE PROCEDURE usp_GetFutureOfficeAssignments
54 AS
55 SELECT EmployeeID,
56   OfficeID,
57   StartDate
58 FROM Offices
59 WHERE StartDate > GETDATE();
60 GO
61

```

Indexes.sql

```

01 CREATE INDEX IX_Offices ON Offices
02 (EmployeeID, StartDate)
03 INCLUDE (OfficeID)
04
05 GO
06
07 CREATE INDEX IX_Employees ON Employees
08 (LastName);
09 GO
10

```

Tables.sql

```

01 CREATE DATABASE HumanResources;
02 GO
03
04 ALTER DATABASE HumanResources
05 SET ALLOW_SNAPSHOT_ISOLATION ON;
06 GO
07
08 USE HumanResources
09 GO
10
11 CREATE TABLE Employees
12 (
13     EmployeeID int IDENTITY(1,1) NOT NULL,
14     FirstName nvarchar(100) NOT NULL,
15     LastName nvarchar(100) NOT NULL,
16
17 );
18 GO
19
20 CREATE TABLE Offices
21 (
22     OfficeID int IDENTITY(1,1) NOT NULL,
23     EmployeeID int NOT NULL,
24     OfficeName nvarchar(100) NOT NULL,
25     StartDate datetime NOT NULL
26 );
27 GO

```

Question: 1

You execute `usp_SelectEmployeesByName` multiple times, passing strings of varying lengths to `@LastName`. You discover that `usp_SelectEmployeesByName` uses inefficient execution plans.

You need to update `usp_SelectEmployeesByName` to ensure that the most efficient execution plan is used. What should you add at line 31 of `StoredProcedures.sql`?

- A. OPTION (ROBUST PLAN)
- B. OPTION (OPTIMIZE FOR UNKNOWN)
- C. OPTION (KEEP PLAN)
- D. OPTION (KEEPFIXED PLAN)

Answer: B

Explanation:

<http://msdn.microsoft.com/en-us/library/ms181714.aspx>

Question: 2

You need to recommend a solution to ensure that SQL1 supports the auditing requirements of usp_UpdateEmployeeName.

What should you include in the recommendation?

- A. Change data capture
- B. Change tracking
- C. Transactional replication
- D. The Distributed Transaction Coordinator (DTC)

Answer: D

Question: 3

You need to add a new column named Confirmed to the Employees table. The solution must meet the following requirements:

Have a default value of TRUE.

Minimize the amount of disk space used.

Which code segment should you use?

- A.

```
ALTER TABLE Employees
ADD Confirmed char(1) DEFAULT '1';
```
- B.

```
ALTER TABLE Employees
ADD Confirmed char(1) DEFAULT '0';
```
- C.

```
ALTER TABLE Employees
ADD Confirmed bit DEFAULT 0;
```
- D.

```
ALTER TABLE Employees
ADD Confirmed bit DEFAULT 1;
```

- A.Option A
- B.Option B
- C.Option C
- D.Option D

Answer: D

Question: 4

You need to create the object used by the parameter of usp_UpdateEmployeeName.

Which code segment should you use?

- A. CREATE XML SCHEMA COLLECTION EmployeesInfo
- B. CREATE TYPE EmployeesInfo AS Table
- C. CREATE SCHEMA EmployeesInfo

D. CREATE TABLE EmployeesInfo

Answer: B

Explanation:

Example Usage of Table-Valued Parameters (Database Engine)

<http://msdn.microsoft.com/en-us/library/bb510489.aspx> (Benefits of using Table-Valued Parameters)

```
/* Create a table type. */
CREATE TYPE LocationTableType AS TABLE
( LocationName VARCHAR(50)
, CostRate INT );
GO
/* Create a procedure to receive data for the table-valued parameter. */
CREATE PROCEDURE dbo. usp_InsertProductionLocation
@TVP LocationTableType READONLY
AS
SET NOCOUNT ON
INSERT INTO AdventureWorks2012.Production.Location
(Name
,CostRate
,Availability
,ModifiedDate)
SELECT *, 0, GETDATE()
FROM @TVP;
GO
```

Also:

<http://msdn.microsoft.com/en-us/library/ms175007.aspx>(CREATE TYPE *tabltypename* AS TABLE)

<http://msdn.microsoft.com/en-us/library/ms175010.aspx>(table data types)

Wrong Answers:

<http://msdn.microsoft.com/en-us/library/ms174979.aspx>(CREATE TABLE)

<http://msdn.microsoft.com/en-us/library/ms189462.aspx>(CREATE SCHEMA)

<http://msdn.microsoft.com/en-us/library/ms176009.aspx>(CREATE XML SCHEMA COLLECTION)

Question: 5

You need to provide referential integrity between the Offices table and Employees table.

Which code segment or segments should you add at line 27 of Tables.sql? (Each correct answer presents part of the solution. Choose all that apply.)

- A. `ALTER TABLE dbo.Offices ADD CONSTRAINT
PK_Offices_EmployeeID PRIMARY KEY (EmployeeID);`
- B. `ALTER TABLE dbo.Employees ADD CONSTRAINT
FK_Employees_Offices FOREIGN KEY (OfficeID)
REFERENCES dbo.Offices (OfficeID);`
- C. `ALTER TABLE dbo.Employees ADD CONSTRAINT
PK_Employees_EmployeeID PRIMARY KEY (EmployeeID);`
- D. `ALTER TABLE dbo.Offices ADD CONSTRAINT
FK_Offices_Employees FOREIGN KEY (EmployeeID)
REFERENCES dbo.Employees (EmployeeID);`

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: C, D

Explanation:
<http://msdn.microsoft.com/en-us/library/ms189049.aspx>

Question: 6

You need to modify `usp_SelectEmployeesByName` to support server-side paging. The solution must minimize the amount of development effort required.

What should you add to `usp_SelectEmployeesByName`?

- A. A table variable
- B. The ROWNUMBER keyword
- C. An OFFSET-FETCH clause
- D. A recursive common table expression

Answer: C

Explanation:
<http://www.mssqltips.com/sqlservertip/2696/comparing-performance-for-different-sql-serverpaging-methods/>
<http://msdn.microsoft.com/en-us/library/ms188385.aspx>
<http://msdn.microsoft.com/en-us/library/ms180152.aspx>
<http://msdn.microsoft.com/en-us/library/ms186243.aspx>
<http://msdn.microsoft.com/en-us/library/ms186734.aspx>
<http://www.sqlserver-training.com/how-to-use-offset-fetch-option-in-sql-server-order-byclause/>
http://www.sqlservercentral.com/blogs/juggling_with_sql/2011/11/30/using-offset-and-fetch/

Case Study: 4

Scenario 4

Application Information

You are a database administrator for a manufacturing company.

You have an application that stores product data. The data will be converted to technical diagrams for the manufacturing process.

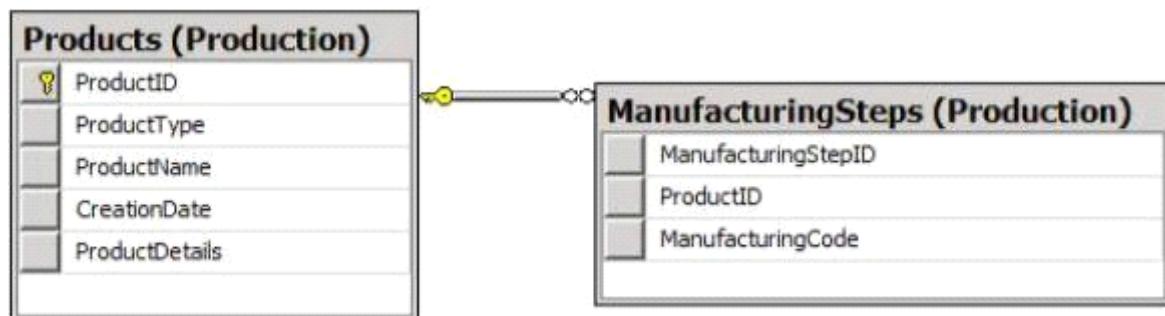
The product details are stored in XML format. Each XML must contain only one product that has a root element named Product. A schema named Production.ProductSchema has been created for the products xml. You develop a Microsoft .NET Framework assembly named ProcessProducts.dll that will be used to convert the XML files to diagrams. The diagrams will be stored in the database as images. ProcessProducts.dll contains one class named ProcessProduct that has a method name of Convert(). ProcessProducts.dll was created by using a source code file named ProcessProduct.cs.

All of the files are located in C:\Products\.

The application has several performance and security issues.

You will create a new database named ProductsDB on a new server that has SQL Server 2012 installed. ProductsDB will support the application.

The following graphic shows the planned tables for ProductsDB:



You will also add a sequence named Production.ProductID_Seq.

You plan to create two certificates named DBCert and ProductsCert. You will create ProductsCert in master. You will create DBCert in ProductsDB.

You have an application that executes dynamic T-SQL statements against ProductsDB. A sample of the queries generated by the application appears in Dynamic.sql.

Application Requirements

The planned database has the following requirements:

- All stored procedures must be signed.
- The amount of disk space must be minimized.
- Administrative effort must be minimized at all times.
- The original product details must be stored in the database.
- An XML schema must be used to validate the product details.
- The assembly must be accessible by using T-SQL commands.
- A table-valued function will be created to search products by type.
- Backups must be protected by using the highest level of encryption.
- Dynamic T-SQL statements must be converted to stored procedures.
- Indexes must be optimized periodically based on their fragmentation.
- Manufacturing steps stored in the ManufacturingSteps table must refer to a product by the same identifier used by the Products table.

ProductDetails_Insert.sql

```

01 CREATE PROCEDURE Production.ProductDetails_Insert @XML nvarchar(1000)
02 AS
03 DECLARE @handle INT;
04 DECLARE @document nvarchar(1000);
05 SET @document = @XML;
06
07 EXEC sp_xml_preparedocument @handle OUTPUT, @document;
08
09 INSERT INTO PRODUCTSDDB.Production.Invoices (
10     ProductID,
11     ProductDetails,
12     ProductType,
13     ProductName,
14     CreationDate
15 )
16 SELECT (NEXT VALUE FOR Production.ProductID_Seq),
17     @XML, * FROM OPENXML (@handle, '/Invoice',2)
18     WITH (
19         ProductType nvarchar(11) 'ProductType/ID',
20         ProductName nvarchar(50) '@ProductName',
21         CreationDate date 'CreationDate'
22     );
23
24 EXEC sp_xml_removedocument @handle;

```

Product, xml

All product types are 11 digits. The first five digits of the product id reference the category of the product and the remaining six digits are the subcategory of the product.

The following is a sample customer invoice in XML format:

```

01 <?xml version="1.0"?>
02 <Product ProductName="Widget">
03     <ProductType ID="00156590099" />
04     <CreationDate>2011-08-05</CreationDate>
05 </Invoice>

```

ProductsByProductType.sql

```

01 (SELECT ProductID,
02     ProductType,
03     CreationDate
04     FROM Production.Products
05     WHERE ProductType=@ProductType);

```

Dynamic.sql

```

01 DECLARE @tsql AS nvarchar(500);
02 DECLARE @ProductType AS varchar(11), @CreationDate AS date;
03
04 SET @sqlstring=N'SELECT ProductID, ProductType, CreationDate
05     FROM Production.Product
06     WHERE ProductID=@ProductID AND CreationDate > @CreationDate;';
07
08 EXEC sys.sp_executesql
09     @statement=@sqlstring,
10     @params=N'@ ProductType AS varchar(11), @CreationDate AS date',
11     @ProductType=00125061246, @Total='2012-05-10';

```

Category FromType.sql

```

01 CREATE FUNCTION CategoryFromType (@Type varchar(11)) RETURNS nvarchar(20)
02 AS
03 BEGIN
04     DECLARE @Category AS varchar(20);
05     SET @Category = LEFT(@Category,5);
06     SELECT @Category = CASE @Type
07         WHEN '00001'
08             THEN 'Bikes'
09         WHEN '00002'
10             THEN 'Wheels'
11         ...
12     ELSE 'Other'
13 END;
14 RETURN @Category;
15 END;

```

IndexManagement.sql

```

01 DECLARE @IndexTable TABLE (
02     TableName varchar(100), IndexName varchar(100), Fragmentation int, RowNumber int
03 );
04 DECLARE @TableName sysname, @IndexName sysname, @Fragmentation int,
05     @RowNumber int, @sqlcommand varchar(1000);
06
07 INSERT INTO @IndexTable (TableName, IndexName, Fragmentation, Rownumber)
08     SELECT OBJECT_NAME(i.Object_id),
09         i.name AS IndexName,
10         indexstats.avg_fragmentation_in_percent,
11         ROW_NUMBER() OVER(ORDER BY i.name DESC) AS 'RowNumber'
12     FROM sys.dm_db_index_physical_stats(DB_ID(), NULL, NULL, NULL, 'DETAILED')
13     AS indexstats INNER JOIN sys.indexes AS i
14     ON i.OBJECT_ID = indexstats.OBJECT_ID AND i.index_id = indexstats.index_id;
15
16 DECLARE @counter int = 0;
17
18 WHILE @counter < (SELECT RowNumber FROM @indextable)
19     BEGIN
20         SET @counter = @counter + 1;
21         WITH t AS (
22             SELECT TableName, IndexName, Fragmentation
23             FROM @IndexTable WHERE RowNumber = @counter
24         )
25         SELECT
26             @TableName= TableName,
27             @IndexName = IndexName,
28             @Fragmentation = Fragmentation
29         FROM t;
30
31     IF @Fragmentation <= 30
32         BEGIN
33             SET @sqlCommand =
34                 N'ALTER INDEX '+@indexName+N' ON '+@TableName+N' REORGANIZE';
35             EXEC sp_executesql @sqlCommand;
36         END;
37     ELSE
38         BEGIN
39             SET @sqlCommand=N'ALTER INDEX '+@indexName+N' ON '+@TableName+N' REBUILD';
40             EXEC sp_executesql @sqlCommand;
41         END;
42     END;

```

Question: 1

Which code segment should you use to define the ProductDetails column?

- A. ProductDetails xml (DOCUMENT Production.ProductDetailsSchema) NULL
- B. ProductDetails xml NULL
- C. ProductDetails xml (CONTENT Production.ProductDetailsSchema) NULL
- D. ProductDetails varchar(MAX) NULL

Answer: D

Question: 2

You need to modify Production.ProductDetails_Insert to comply with the application requirements.
Which code segment should you execute?

- A. OPEN PRODUCTSCERT;
ALTER PROCEDURE Production.ProductDetails_Insert
WITH ENCRYPTION;
CLOSE PRODUCTSCERT;
- B. OPEN DBCERT;
ALTER PROCEDURE Production.ProductDetails_Insert
WITH ENCRYPTION;
CLOSE DBCERT;
- C. ADD SIGNATURE TO Production.ProductDetails_Insert
BY CERTIFICATE DBCERT;
- D. ADD SIGNATURE TO Production.ProductDetails_Insert
BY CERTIFICATE PRODUCTSCERT;

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: C

Explanation:
<http://msdn.microsoft.com/en-us/library/bb669102.aspx>

Question: 3

You need to create a function that will use a SELECT statement in ProductsByProductType.sql.
Which code segment should you use to complete the function?

- C A. CREATE FUNCTION Production.fnProductsByProductType (@ProductType varchar(11))
RETURNS @tblInvoices TABLE (ProductID bigint, ProductType varchar(11), CreationDate
date)
AS
INSERT INTO @tblInvoices
- C B. CREATE FUNCTION Production.fnProductsByProductType (@ProductType varchar(11))
RETURNS TABLE
AS
RETURN
- C C. CREATE FUNCTION Production.fnProductsByProductType (@ProductType varchar(11))
RETURNS @Tb1Invoices TABLE (ProductID bigint, ProductType varchar(11), CreationDate
date)
AS
- C D. CREATE FUNCTION Production.fnProductsByProductType (@ProductType varchar(11))
RETURNS xml
AS
RETURN
- A. Option A
B. Option B
C. Option C
D. Option D

Answer: B

Explanation:

<http://msdn.microsoft.com/en-us/library/ms191320.aspx>
<http://msdn.microsoft.com/en-us/library/ms186755.aspx>

Question: 4

An administrator provides a digital certificate named ServerCert.
You need to implement Transparent Data Encryption (TDE) on ProductsDB.
Which code segment should you use?

- A. USE PRODUCTSDB;
GO
CREATE DATABASE ENCRYPTION KEY WITH ALGORITHM = TRIPLE_DES_3KEY
ENCRYPTION BY SERVER CERTIFICATE DBCERT;
GO
ALTER DATABASE PRODUCTSDB SET ENCRYPTION ON;
GO
- B. USE PRODUCTSDB;
GO
CREATE DATABASE ENCRYPTION KEY WITH ALGORITHM = TRIPLE_DES_3KEY
ENCRYPTION BY SERVER CERTIFICATE PRODUCTSCERT;
GO
ALTER DATABASE PRODUCTSDB SET ENCRYPTION ON;
GO
- C. USE PRODUCTSDB;
GO

```
CREATE DATABASE ENCRYPTION KEY WITH ALGORITHM = AES_256  
ENCRYPTION BY SERVER CERTIFICATE PRODUCTSCERT;  
GO  
ALTER DATABASE PRODUCTSDB SET ENCRYPTION ON;  
GO  
D. USE PRODUCTSDB;  
GO  
CREATE DATABASE ENCRYPTION KEY WITH ALGORITHM = AES_256  
ENCRYPTION BY SERVER CERTIFICATE DBCERT;  
GO  
ALTER DATABASE PRODUCTSDB SET ENCRYPTION ON;  
GO
```

Answer: C

Explanation:

<http://msdn.microsoft.com/en-us/library/bb934049.aspx>

Question: 5

You execute IndexManagement.sql and you receive the following error message: "Msg 512, Level 16, State 1, Line 12 Subquery returned more than 1 value. This is not permitted when the subquery follows =, !=, <, >, <=, >= or when the subquery is used as an expression."

You need to ensure that IndexManagement.sql executes properly.

Which WHILE statement should you use at line 18?

- A. WHILE SUM(@RowNumber) < (SELECT @counter FROM @indextable)
- B. WHILE @counter < (SELECT SUM(RowNumber) FROM @indextable)
- C. WHILE COUNT(@RowNumber) < (SELECT @counter FROM @indextable)
- D. WHILE @counter < (SELECT COUNT(RowNumber) FROM @indextable)

Answer: D

Question: 6

You are planning the ManufacturingSteps table.

You need to define the ProductID column in the CREATE TABLE statement.

Which code segment should you use?

- A. ProductID bigint
DEFAULT (NEXT VALUE FOR Production.ProductID_Seq) NOT NULL,
- B. ProductID bigint FOREIGN KEY REFERENCES
Production.Product(ProductID) NOT NULL,
- C. ProductID bigint DEFAULT
((NEXT VALUE FOR Production.ProductID_Seq OVER
(ORDER BY ManufacturingStepID))) NOT NULL,
- D. ProductID bigint DEFAULT
((NEXT VALUE FOR Production.ProductID_Seq OVER
(ORDER BY ManufacturingStepID)))
NOT NULL FOREIGN KEY REFERENCES
Production.Product(ProductID),

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: B

Explanation:

<http://msdn.microsoft.com/en-us/library/ms189049.aspx>
<http://msdn.microsoft.com/en-us/library/ms179610.aspx>
<http://msdn.microsoft.com/en-us/library/ff878370.aspx>

Question: 7

You are testing disaster recovery procedures.

When you attempt to restore ProductsDB to another server, you receive the following error message: "Msg 33111, Level 16, State 3, Line 5

Cannot find server certificate with thumbprint '0x9D876A3468B911EIBA4CFCBF4724019B\

Msg 3013, Level 16, State 1, Line 5

RESTORE DATABASE is terminating abnormally."

You need to ensure that you can restore ProductsDB to another server.

Which code segment should you execute on the other server?

- A. RESTORE CERTIFICATE DBCERT
FROM FILE='DBCERT.CER'
WITH PRIVATE KEY (FILE = 'c:\DBCERT.KEY',
DECRYPTION BY PASSWORD = 'SecretP@ss');
- B. CREATE CERTIFICATE PRODUCTSCERT
ENCRYPTION BY PASSWORD = 'SecretP@ss'
WITH SUBJECT = 'SecurityCertificate';
- C. CREATE CERTIFICATE DBCERT
ENCRYPTION BY PASSWORD = 'SecretP@ss'
WITH SUBJECT = 'SecurityCertificate';
- D. CREATE CERTIFICATE PRODUCTSCERT
FROM FILE='PRODUCTSCERT.CER'
WITH PRIVATE KEY (FILE = 'c:\PRODUCTSCERT.KEY',
DECRYPTION BY PASSWORD = 'SecretP@ss');

- A. Option A
B. Option B
C. Option C
D. Option D

Answer: D

Question: 8

You need to prepare the database to use the .NET Framework ProcessProducts component.

Which code segments should you execute? (Each correct answer presents part of the solution. Choose all that apply.)

- A. CREATE PROCEDURE Production.ProcessProduct(
 @ProductId int, @ProductType varchar(11)
)
AS EXTERNAL NAME ProductionAssembly.ProcessProducts.Process;
- B. EXEC sp_recompile @objname = 'Production.ProcessProduct';
- C. RECONFIGURE;
- D. Exec SP_CONFIGURE 'clr enabled', '1';
- E. CREATE ASSEMBLY ProductionAssembly FROM 'C:\Products\ProcessProducts.DLL'
- F. CREATE ASSEMBLY ProductionAssembly FROM 'C:\Products\ProcessProducts.cs';
- G. CREATE TYPE Production.ProcessProduct
EXTERNAL NAME ProductionAssembly.ProcessProductss.Process;

- A. Option A

- B. Option B
- C. Option C
- D. Option D
- E. Option E
- F. Option F
- G. Option G

Answer: A, C, D, E

Explanation:

<http://msdn.microsoft.com/en-us/library/ms131048.aspx>
<http://msdn.microsoft.com/en-us/library/ms131052.aspx>
<http://msdn.microsoft.com/en-us/library/ms189524.aspx>
<http://msdn.microsoft.com/en-us/library/ms345106.aspx>
<http://msdn.microsoft.com/en-us/library/ms131107.aspx>

Question: 9

While testing the CategoryFromType function, you discover that the function is returning 'Other'.

You need to update CategoryFromType to return the category name.

Which line of code should you modify in CategoryFromType.sql?

- A.04
- B.05
- C.12
- D.14

Answer: B

Question: 10

Which data type should you use for ProductType?

- A. varchar(11)
- B. nvarchar(11)
- C. char(11)
- D. bigint

Answer: C

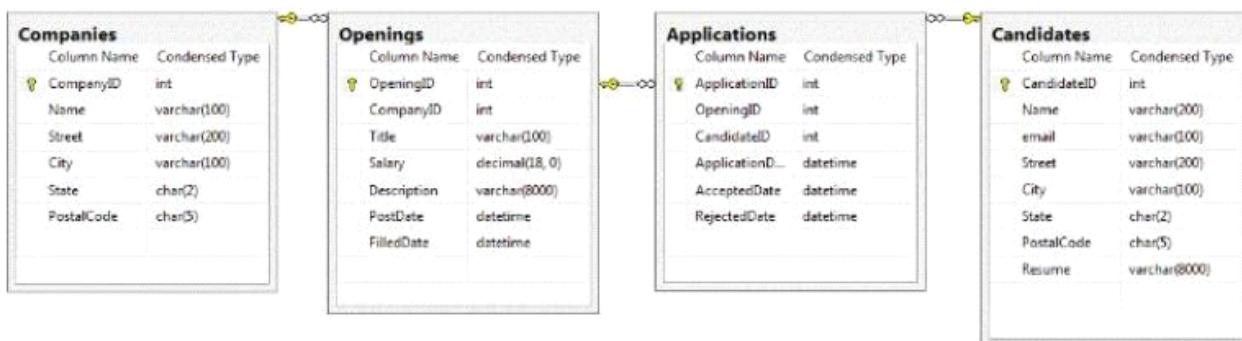
Case Study: 5

Litware, Inc

Overview

General Overview

You are a database developer for a company named Litware, Inc. Litware has a main office in Miami. Litware has a job posting web application named WebApp1. WebApp1 uses a database named DB1. DB1 is hosted on a server named Server1. The database design of DB1 is shown in the exhibit. (Click the Exhibit button.)



WebApp1 allows a user to log on as a job poster or a job seeker. Candidates can search for job openings based on keywords, apply to an opening, view their application, and load their resume in Microsoft Word format. Companies can add a job opening, view the list of candidates who applied to an opening, and mark an application as denied.

Users and Roles

DB1 has five database users named Company, CompanyWeb, Candidate, CandidateWeb, and Administrator. DB1 has three user-defined database roles. The roles are configured as shown in the following table.

Role name	Role member
Companies	Company Administrator CompanyWeb
Candidates	Candidate Administrator CandidateWeb
Administrators	Administrator

Keyword Search

The keyword searches for the job openings are performed by using the following stored procedure named usp_GetOpenings:

```

01 CREATE PROCEDURE usp_GetOpenings
02   @keyword varchar(max),
03   @minsalary decimal(18,0) = 0
04 AS
05 DECLARE @plural varchar(max);
06 DECLARE @ing varchar(max);
07 SET @plural = @keyword + 's';
08 SET @ing = @keyword + 'ing';
09 SELECT o.Title, o.Salary, c.Name, o.Description
10 FROM Openings o
11 INNER JOIN Companies c ON c.CompanyID = o.CompanyID
12 WHERE (o.Description LIKE '%' + @keyword + '%'
13   OR o.Description LIKE '%' + @plural + '%'
14   OR o.Description LIKE '%' + @ing + '%')
15 AND o.Salary >= @minsalary;

```

Opening Update

Updates to the Openings table are performed by using the following stored procedure named usp_UpdateOpening:

```

01 CREATE PROCEDURE usp_UpdateOpening
02     @openingID int,
03     @title varchar(100),
04     @salary decimal(18,0),
05     @description varchar(8000)
06 AS
07 UPDATE Openings
08 SET Title = @title,
09     Salary = @salary,
10    Description = @description
11 WHERE OpeningID = @openingID;

```

Problems and Reported Issues

Concurrency Problems

You discover that deadlocks frequently occur.

You identify that a stored procedure named usp_AcceptCandidate and a stored procedure named usp_UpdateCandidate generate deadlocks. The following is the code for usp_AcceptCandidate:

```

01 CREATE PROCEDURE usp_AcceptCandidate
02     @applicationID int
03 AS
04     DECLARE @date datetime;
05     SET @date = GETDATE();
06     UPDATE Applications
07         SET AcceptedDate = @date
08         WHERE ApplicationID = @applicationID;
09     SELECT Name, email
10         FROM Candidates c
11         INNER JOIN Applications a
12             ON a.CandidateID = c.CandidateID
13         WHERE a.AcceptedDate IS NOT NULL;

```

Salary Query Issues

Users report that when they perform a search for job openings without specifying a minimum salary, only job openings that specify a minimum salary are displayed.

Log File Growth Issues

The current log file for DB1 grows constantly. The log file fails to shrink even when the daily SQL Server Agent Shrink Database task runs.

Performance Issues

You discover that a stored procedure named usp_ExportOpenings takes a long time to run and executes a table scan when it runs.

You also discover that the usp_GetOpenings stored procedure takes a long time to run and that the non-clustered index on the Description column is not being used.

Page Split Issues

On DB1, many page splits per second spike every few minutes.

Requirements

Security and Application Requirements

Litware identifies the following security and application requirements:

- Only the Administrator, Company, and CompanyWeb database users must be able to execute the usp_UpdateOpening stored procedure.
- Changes made to the database must not affect WebApp1.

Locking Requirements

Litware identifies the following locking requirements:

- The usp_GetOpenings stored procedure must not be blocked by the usp_UpdateOpening stored procedure.
- If a row is locked in the Openings table, usp_GetOpenings must retrieve the latest version of the row, even if the row was not committed yet.

Integration Requirements

Litware exports its job openings to an external company as XML data. The XML data uses the following format:

```
<Opening title="web programmer" salary="75000">
    This is the description of the opening
</Opening>
```

A stored procedure named usp_ExportOpenings will be used to generate the XML data. The following is the code for usp_ExportOpenings:

```
01 CREATE PROCEDURE usp_ExportOpenings
02     @lastPost datetime
03 AS
04 SELECT Description
05     , Title
06     , Salary
07 FROM Openings
08 WHERE PostDate > @lastPost
09     AND FilledDate IS NULL
```

The stored procedure will be executed by a SQL Server Integration Services (SSIS) package named Package1.

The XML data will be written to a secured folder named Folder1. Only a dedicated Active Directory account named Account1 is assigned the permissions to read from or write to Folder1.

Refactoring Requirements

Litware identifies the following refactoring requirements:

- New code must be written by reusing the following query:

```
01 SELECT Title, Salary, Description
02 FROM Openings
03 WHERE Salary >= @minsalary
04     AND FilledData IS NULL
```

- The results from the query must be able to be joined to other queries.

Upload Requirements

Litware requires users to upload their job experience in a Word file by using WebApp1. WebApp1 will send the Word file to DB1 as a stream of bytes. DB1 will then convert the Word file to text before the contents of the Word file is saved to the Candidates table.

A database developer creates an assembly named Conversions that contains the following:

- A class named Convert in the SqlConversions namespace
- A method named ConvertToText in the Convert class that converts Word files to text

The ConvertToText method accepts a stream of bytes and returns text. The method is used in the following stored procedure:

```
01 CREATE PROCEDURE usp_UpdateCandidate
02     @candidateID int,
03     @wordResume varbinary(max)
04 AS
05 DECLARE @textResume varchar(8000);
06 SET @textResume = ConvertToText(@wordResume);
07 UPDATE Candidates SET Resume = @textResume
08     WHERE CandidateID = @candidateID;
09 SELECT OpeningID, ApplicationDate
10 FROM Applications
11 WHERE CandidateID = @candidateID;
```

Job Application Requirements

A candidate can only apply to each job opening once.

Data Recovery Requirements

All changes to the database are performed by using stored procedures. WebApp1 generates a unique transaction ID for every stored procedure call that the application makes to the database.

If a server fails, you must be able to restore data to a specified transaction.

Question: 1

You need to identify the cause of the page split issues.

Which SQL Server feature should you use?

- A. DBCC REINDEX
- B. SQL Server Profiler
- C. Extended Events
- D. DBCC TRACEOFF

Answer: C

Question: 2

You need to implement a solution that meets the job application requirements.

What should you do?

- A. Create a one-to-one relationship between the Openings table and the Applications table.
- B. Create a one-to-one relationship between the Candidates table and the Applications table.
- C. Add a UNIQUE constraint to the Applications table on the ApplicationID column and CandidateID column.
- D. Add a UNIQUE constraint to the Applications table on the OpeningID column and the CandidateID column.

Answer: D

Question: 3

You need to implement a solution that resolves the salary query issue.

Which statement should you execute on DB1?

- Ⓐ A. UPDATE Openings SET Salary=0 WHERE Salary IS NULL;
GO
ALTER TABLE Openings
WITH NOCHECK
MODIFY COLUMN Salary NOT NULL;
GO
ALTER TABLE Openings
WITH NOCHECK
ADD CONSTRAINT DF_SALARY
DEFAULT 0 FOR Salary;
GO

- Ⓑ B. ALTER TABLE Openings
WITH NOCHECK
ADD CONSTRAINT DF_SALARY
DEFAULT 0 FOR Salary;
GO
ALTER TABLE Openings
WITH NOCHECK
MODIFY COLUMN Salary NULL;
GO
UPDATE Openings SET Salary=0 WHERE Salary IS NULL;
GO

- Ⓒ C. UPDATE Openings SET Salary=0 WHERE Salary IS NULL;
GO
ALTER TABLE Openings
WITH NOCHECK
ADD CONSTRAINT CT_SALARY
CHECK (Salary>=0);
GO
ALTER TABLE Openings
WITH NOCHECK
MODIFY COLUMN Salary NOT NULL;
GO

- Ⓓ D. ALTER TABLE Openings
WITH NOCHECK
ADD CONSTRAINT CT_SALARY
CHECK (Salary>=0);
GO
ALTER TABLE Openings
WITH NOCHECK
MODIFY COLUMN Salary NOT NULL;
GO
UPDATE Openings SET Salary=0 WHERE Salary IS NULL;
GO

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: A

Question: 4

You need to implement a solution that addresses the upload requirements.

Which code segment should you use to implement the Conversions assembly?

- A.

```
CREATE FUNCTION ConvertToText (@wordResume varchar(8000))
RETURNS varbinary(max)
AS EXTERNAL NAME SqlConversions.Conversions.ConvertToText;
```
- B.

```
CREATE PROCEDURE ConvertToText (@wordResume varbinary(max))
AS EXTERNAL NAME Conversions.SqlConversions.ConvertToText;
```
- C.

```
CREATE PROCEDURE ConvertToText (@wordResume varchar(8000))
AS EXTERNAL NAME SqlConversions.Conversions.ConvertToText;
```
- D.

```
CREATE FUNCTION ConvertToText (@wordResume varbinary(max))
RETURNS varchar(8000)
AS EXTERNAL NAME Conversions.SqlConversions.ConvertToText;
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: D

Question: 5

You need to design a solution that meets the refactoring requirements.

Which type of object should you include in the solution?

- A. An indexed view
- B. An aggregate function
- C. A distributed view
- D. A table-valued function

Answer: D

Question: 6

You need to create a script that automates the export of the XML data. The script must meet the integration requirements.

What should you include in the script?

- A. The CREATE SERVER ROLE command and the sp_reassign_proxy, sp_add_job, sp_add_jobstep, and

sp_grant_login_to_proxy system stored procedures.

- B. The CREATE CREDENTIAL command and the sp_add_proxy, sp_add_job, sp_add_jobstep, and sp_grant_proxy_to_subsystem system stored procedures.
- C. The CREATE CREDENTIAL command and the sp_reassign_proxy, sp_add_job, sp_add_jobstep, and sp_grant_login_to_proxy system stored procedures.
- D. The CREATE SERVER ROLE command and the sp_add_proxy, sp_add_job, sp_add_jobstep, and sp_grant_proxy_to_subsystem system stored procedures.

Answer: B

Question: 7

You need to implement a solution that meets the security requirements.

Which statement should you execute?

- A. REVOKE EXEC ON usp_UpdateOpening FROM Candidates;
 - B. DENY EXEC ON usp_UpdateOpening TO Candidates;
 - C. ALTER PROCEDURE usp_UpdateOpening
 @openingIDint,
 @titlevarchar(100),
 @salarydecimal(18,0),
 @descriptionvarchar(8000)
 WITH EXECUTE AS Administrator
 AS
 ...
 - D. ALTER PROCEDURE usp_UpdateOpening
 @openingIDint,
 @titlevarchar(100),
 @salarydecimal(18,0),
 @descriptionvarchar(8000)
 WITH EXECUTE AS Company
 AS
 ...
- A. Option A
B. Option B
C. Option C
D. Option D

Answer: A

Question: 8

You need to implement a change to usp_ExportOpenings that meets the integration requirements.

What should you modify in usp_ExportOpenings? (Each correct answer presents part of the solution. Choose all that apply?)

- A. To the end of line 04, add [Opening].
- B. To the end of line 05, add [Opening! title].
- C. To line 10, add FOR XML RAW.
- D. To line 10, add FOR XMLEXPLICIT.
- E. To line 10, add FOR XML AUTO.
- F. To the end of line 04, add [Opening!ELEMENT].
- G. To the end of line 06, add [Opening!salary!ELEMENT].
- H. To the end of line 05, add [Opening!title!ELEMENT].
- I. To the end of line 06, add [Opening! salary].

Answer: A, B, E, I

Explanation:

The AUTO mode generates nesting in the resulting XML by using heuristics based on the way the SELECT statement is specified. You have minimal control over the shape of the XML generated. The nested FOR XML queries can be written to generate XML hierarchy beyond the XML shape that is generated by AUTO mode heuristics.

Question: 9

You need to recommend a solution that meets the concurrency problems.

What should you include in the recommendation?

- A. Modify the stored procedures to use the SERIALIZABLE isolation level.
- B. Modify the order in which usp_AcceptCandidate accesses the Applications table and the Candidates table.
- C. Modify the order in which usp_UpdateCandidate accesses the Applications table and the Candidates table.
- D. Modify the stored procedures to use the REPEATABLE READ isolation level.

Answer: C

Question: 10

You need to resolve the performance issues of the usp_ExportOpenings stored procedure. The solution must minimize the amount of hard disk space used.

Which statement should you execute on DB1?

- A. EXEC sp_dboption 'DB1', 'auto create statistics', 'TRUE';
- B. CREATE INDEX IX_Exp_Openings ON Openings(PostDate, FilledDate) INCLUDE (Description, Title, Salary);
- C. CREATE INDEX IX_Exp_Openings ON Openings(PostDate) INCLUDE (Description, Title, Salary) WHERE FilledDate IS NULL;
- D. EXEC sp_recompile 'usp_ExportOpenings';

Answer: C

Question: 11

You need to implement a solution that meets the locking requirements.
Which line of code should you modify?

- A. Change line 07 in usp_UpdateOpening to:

UPDATE Openings WITH (UPDLOCK)

- B. Change line 09 in usp_GetOpenings to:

FROM Openings o (ROWLOCK)

- C. Change line 07 in usp_UpdateOpening to:

UPDATE Openings WITH (READPAST)

- D. Change line 09 in usp_GetOpenings to:

FROM Openings o (NOLOCK)

Answer: D

Question: 12

You need to implement a solution that meets the data recovery requirements.

You update each stored procedure to accept a parameter named @transactionID.

What should you add next to the beginning of each stored procedure?

- A. SAVE TRANSACTION WITH MARK @transactionID

- B. ROLLBACK DISTRIBUTED TRANSACTION @transactionID

- C. BEGIN TRANSACTION WITH MARK @transactionID

- D. COMMIT TRANSACTION @transactionID

Answer: C

Case Study: 6

Coho Winery

Overview

You are a database developer for a company named Coho Winery. Coho Winery has an office in London. Coho Winery has an application that is used to process purchase orders from customers and retailers in 10 different countries.

The application uses a web front end to process orders from the Internet. The web front end adds orders to a database named Sales. The Sales database is managed by a server named Server1.

An empty copy of the Sales database is created on a server named Server2 in the London office. The database will store sales data for customers in Europe.

A new version of the application is being developed. In the new version, orders will be placed either by using the existing web front end or by loading an XML file.

Once a week, you receive two files that contain the purchase orders and the order details of orders from offshore facilities.

You run the usp_ImportOrders stored procedure and the usp_ImportOrderDetails stored procedure to copy the offshore facility orders to the Sales database.

The Sales database contains a table named Orders that has more than 20 million rows.

Database Definitions

Database and Tables

The following scripts are used to create the database and its tables:

```

01 CREATE DATABASE Sales;
02 GO
03 USE Sales;
04 GO
05 CREATE TABLE Products
06 (
07     ProductID int IDENTITY(1,1) NOT NULL,
08     Name nvarchar(100) NOT NULL,
09     UnitPrice decimal(18,2) NOT NULL,
10     Discontinued bit NOT NULL DEFAULT 0,
11     CONSTRAINT PK_Products PRIMARY KEY (ProductID)
12 );
13 GO
14
15 CREATE TABLE Customers
16 (
17     CustomerID int IDENTITY(1,1) NOT NULL,
18     Name nvarchar(200) NOT NULL,
19     Email nvarchar(200) NOT NULL,
20     Phone nvarchar(10) NOT NULL,
21     Address1 nvarchar(200) NOT NULL,
22     Address2 nvarchar(200) NULL,
23     City nvarchar(200) NOT NULL,
24     State char(2) NOT NULL,
25     ZIP char(5) NOT NULL,
26     CONSTRAINT PK_Customers PRIMARY KEY (CustomerID)
27 );
28 GO
29
30 CREATE TABLE Orders
31 (
32     OrderID int IDENTITY(1,1) NOT NULL,
33     CustomerID int NOT NULL,
34     OrderDate datetime NOT NULL DEFAULT GETDATE(),
35     DeliveryDate datetime NOT NULL,
36     ShipDate datetime NULL,
37     Amount decimal(18,2) NOT NULL,
38     CONSTRAINT PK_Orders PRIMARY KEY(OrderID)
39 );
40 GO
41
42 ALTER TABLE Orders
43     ADD CONSTRAINT FK_Orders_Customers
44         FOREIGN KEY(CustomerID)
45             REFERENCES Customers(CustomerID);
46 GO
47
48 CREATE TABLE OrderDetails
49 (
50     OrderID int NOT NULL,
51     LineItem int NOT NULL,
52     ProductID int NOT NULL,
53     Quantity int NOT NULL,
54     UnitPrice decimal(18,2) NOT NULL,
55     Total decimal(18,2) NOT NULL,
56     Discount decimal(18,2) NULL,
57     CONSTRAINT PK_OrderDetails PRIMARY KEY(OrderID, LineItem)
58 );
59 GO
60
61 ALTER TABLE OrderDetails
62     ADD CONSTRAINT FK_OrderDetails_Orders
63         FOREIGN KEY(OrderID)
64             REFERENCES Orders(OrderID);
65 GO
66
67 ALTER TABLE OrderDetails
68     ADD CONSTRAINT FK_OrderDetails_Products
69         FOREIGN KEY(ProductID)
70             REFERENCES Products(ProductID);
71 GO

```

Stored Procedures

The following are the definitions of the stored procedures used in the database:

```

01 CREATE PROCEDURE usp_AddOrder
02   @customerID int,
03   @deliveryDate datetime,
04   @items ItemsTable READONLY,
05   @orderID int OUTPUT
06 AS
07 SET TRANSACTION ISOLATION LEVEL READ COMMITTED;
08 BEGIN TRANSACTION;
09   DECLARE @amount decimal(18,2);
10   SELECT @amount = SUM(Quantity * UnitPrice) FROM @items;
11   INSERT INTO Orders (CustomerID, DeliveryDate, Amount)
12     VALUES (@customerID, @deliveryDate, @amount);
13   SELECT @orderID = @@IDENTITY;
14   INSERT INTO OrderDetails
15     SELECT @orderID, LineItem, ProductID, Quantity,
16           UnitPrice, Total, Discount
17     FROM @items;
18 COMMIT TRANSACTION;
19 GO

20 CREATE PROCEDURE usp_AddXMLOrder
21   @customerID int,
22   @deliverDate datetime,
23
24   @orderID int OUTPUT
25 AS
26 SET TRANSACTION ISOLATION LEVEL READ COMMITTED;
27 BEGIN TRANSACTION;
28   DECLARE @itemsTable ItemsTable;
29   EXEC usp_ValidateAndGetItems @schema, @items, @itemsTable;
30   UPDATE Orders SET originalOrder = @items
31     WHERE OrderID = @orderID;
32 COMMIT TRANSACTION;
33 GO
34 CREATE PROCEDURE usp_ValidateAndGetItems
35   @schema sysname,
36   @items XML,
37   @itemsTable ItemsTable Output
38 AS

39 CREATE PROCEDURE usp_GetOrdersAndItems
40 AS
41
42   SELECT o.OrderID, o.OrderDate, o.DeliveryDate, o.Amount,
43         od.LineItem, od.Quantity, od.UnitPrice, p.Name
44   FROM Orders o
45   INNER JOIN OrderDetails od ON od.OrderID=o.OrderID
46   INNER JOIN Products p ON p.ProductID=od.ProductID
47   WHERE o.ShipDate IS NULL
48     AND o.DeliveryDate >= GETDATE() - 30;
49 GO

```

```

50 CREATE PROCEDURE usp_GetOrders
51 AS
52   SELECT OrderID, DeliveryDate, Amount
53   FROM Orders
54   WHERE ShipDate IS NULL
55   ORDER BY DeliveryDate;
56 GO
57
58 CREATE PROCEDURE usp_GetOrdersByProduct
59   @productID int
60
61 AS
62 SELECT OrderID, LineItem, Quantity,
63   UnitPrice, Total, Discount
64 FROM OrderDetails
65
66 WHERE ProductID = @productID;
67 GO
68
69 CREATE PROCEDURE usp_ImportOrders
70 AS
71 BULK INSERT Orders
72   FROM 'f:\orders\orders.tbl'
73   WITH
74   (
75     FIELDTERMINATOR = '|',
76     ROWTERMINATOR = '\n'
77   );
78 GO

79 CREATE PROCEDURE usp_ImportOrderDetails
80   @firstRow int
81 AS
82 BULK INSERT OrderDetails
83   FROM 'f:\orders\details.tbl'
84   WITH
85   (
86     FIRSTROW = @firstRow,
87     FIELDTERMINATOR = '|',
88     ROWTERMINATOR = '\n'
89   );
90
91 GO

```

Indexes

The following indexes are part of the Sales database:

```

01 CREATE INDEX IX_Orders_ShipDate
02   ON Orders(Shipdate)
03
04   INCLUDE (CustomerID, OrderDate, Amount);
05 GO

```

Data Import

The XML files will contain the list of items in each order. Each retailer will have its own XML schema and will be able to use different types of encoding. Each XML schema will use a default namespace. The default namespaces are not guaranteed to be unique.

For testing purposes, you receive an XSD file from a customer.

For testing purposes, you also create an XML schema collection named ValidateOrder. ValidateOrder contains schemas for all of the retailers.

The new version of the application must validate the XML file, parse the data, and store the parsed data along with the original XML file in the database. The original XML file must be stored without losing any data.

Reported Issues

Performance Issues

You notice the following for the usp_GetOrdersAndItems stored procedure:

- The stored procedure takes a long time to complete.
- Less than two percent of the rows in the Orders table are retrieved by usp_GetOrdersAndItems.
- A full table scan runs when the stored procedure executes.
- The amount of disk space used and the amount of time required to insert data are very high.

You notice that the usp_GetOrdersByProduct stored procedure uses a table scan when the stored procedure is executed.

Page Split Issues

Updates to the Orders table cause excessive page splits on the IX_Orders_ShipDate index.

Requirements

Site Requirements

Users located in North America must be able to view sales data for customers in North America and Europe in a single report. The solution must minimize the amount of traffic over the WAN link between the offices.

Bulk Insert Requirements

The usp_ImportOrderDetails stored procedure takes more than 10 minutes to complete. The stored procedure runs daily. If the stored procedure fails, you must ensure that the stored procedure restarts from the last successful set of rows.

Index Monitoring Requirements

The usage of indexes in the Sales database must be monitored continuously. Monitored data must be maintained if a server restarts. The monitoring solution must minimize the usage of memory resources and processing resources.

Question: 1

You need to implement a solution that meets the site requirements.

What should you implement?

- A. A non-indexed view on Server1
- B. A non-indexed view on Server2
- C. A distributed view on Server1
- D. A distributed view on Server2

Answer: C

Question: 2

You need to modify usp_GetOrdersAndItems to ensure that an order is NOT retrieved by usp_GetOrdersAndItems while the order is being updated.

What should you add to usp_GetOrdersAndItems?

- A. Add SET TRANSACTION ISOLATION LEVEL SERIALIZABLE to line 03.
- B. Add SET TRANSACTION ISOLATION LEVEL SNAPSHOT to line 03.
- C. Add (UPDLOCK) to the end of line 06.
- D. Add (READPAST) to the end of line 06.

Answer: D

Question: 3

You need to implement a solution that addresses the performance issues of the usp_GetOrdersByProduct stored procedure.

Which statement should you execute?

- A. CREATE INDEX IX_OrderDetails_ByProduct
ON OrderDetails (ProductID)
INCLUDE (OrderID, LineItem, UnitPrice, Total, Discount)
 - B. CREATE INDEX IX_OrderDetails_ByProduct
ON OrderDetails (ProductID)
INCLUDE (LineItem, Quantity, UnitPrice, Total, Discount)
 - C. CREATE INDEX IX_OrderDetails_ByProduct
ON OrderDetails (ProductID)
 - D. CREATE INDEX IX_OrderDetails_ByProduct
ON OrderDetails (ProductID)
INCLUDE (LineItem, Quantity, UnitPrice, Discount)
- A. Option A
B. Option B
C. Option C
D. Option D

Answer: C

Question: 4

You need to implement a solution that addresses the bulk insert requirements.

What should you add to line 08 in usp_ImportOrderDetails?

- A. LASTROW=0.
- B. BATCHSIZE=0.
- C. BATCHSIZE=1000.
- D. LASTROW = 1000.

Answer: C

Question: 5

You discover that the usp_GetOrdersAndItems stored procedure takes a long time to complete while usp_AddOrder or usp_AddXMLOrder run.

You need to ensure that usp_GetOrdersAndItems completes as quickly as possible.

What should you do? (Each correct answer presents part of the solution. Choose all that apply.)

- A. Set the isolation level of the usp_GetOrdersAndItems stored procedure to SERIALIZABLE.
- B. Execute the ALTER DATABASE Sales SET ALLOW_SNAPSHOT_ISOLATION ON statement.
- C. Set the isolation level of the usp_AddOrder stored procedure to SERIALIZABLE.
- D. Set the isolation level of the usp_GetOrdersAndItems stored procedure to SNAPSHOT.
- E. Set the isolation level of the usp_AddOrder stored procedure to SNAPSHOT.
- F. Execute the ALTER DATABASE Sales SET ALLOW_SNAPSHOT_ISOLATION OFF statement.

Answer: B, D

Question: 6

You need to modify the Orders table to store the XML data used by the retailers.

Which statement should you execute?

- A. ALTER Orders
ADD originalOrder XML (ValidateOrder);
- B. ALTER Orders
ADD originalOrder XML;
- C. ALTER Orders
ADD originalOrder varchar(max);
- D. ALTER Orders
ADD originalOrder varbinary(max);

Answer: D

Question: 7

You plan to create a stored procedure that inserts data from an XML file to the OrderDetails table. The following is the signature of the stored procedure:

```
CREATE PROCEDURE usp_InsertItems
    @items XML (ValidateOrder)
```

The following is the XSD file used to create the ValidateOrder schema collection:

```
<?xml version="1.0" encoding="UTF-16"?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" >
<xsd:element name="root">
  <xsd:complexType mixed="true">
    <xsd:sequence>
      <xsd:element name="Product"
        minOccurs="1" maxOccurs="unbounded">
        <xsd:complexType mixed="true">
          <xsd:sequence>
            <xsd:element name="UnitPrice" type="xsd:decimal"
              minOccurs="1" maxOccurs="1" />
            <xsd:element name="Quantity" type="xsd:integer"
              minOccurs="1" maxOccurs="1" />
          </xsd:sequence>
          <xsd:attribute name="lineItem"
            type="xsd:integer" use="required"/>
          <xsd:attribute name="productID"
            type="xsd:integer" use="required"/>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
    <xsd:attribute name="numberItems"
      type="xsd:integer" use="required"/>
  </xsd:complexType>
</xsd:element>
</xsd:schema>
```

You develop a code segment that retrieves the number of items and loops through each item. Each time the loop runs, a variable named @itemNumber is incremented.

You need to develop a code segment that retrieves the product ID of each item number in the loop.

Which code segment should you develop?

- A. SET @productID = @items.value'/Root/Product/productID', int)
- B. SET @productID = @items.value'/Root/Product['+ @itemNumber+ ']/@productID', int)
- C. SET @productID = @items.value'/Root/Product['+ @itemNumber+ ']/productID', int)
- D. SET @productID = @items.value'/Root/Product/@productID', int)

Answer: B

Question: 8

You need to ensure that a new execution plan is used by usp_GetOrdersByProduct each time the stored procedure runs.

What should you do?

- A. Execute sp_help 'usp_GetOrdersByProduct'.
- B. Execute sp_recompile 'usp_GetOrdersByProduct'.
- C. Add WITH RECOMPILE to line 03 in usp_GetOrdersByProduct.
- D. Add WITH (FORCESEEK) to line 07 in usp_GetOrdersByProduct.

Answer: C

Ref: [http://msdn.microsoft.com/en-us/library/ms190439\(v=sql.90\).aspx](http://msdn.microsoft.com/en-us/library/ms190439(v=sql.90).aspx)

Question: 9

You need to implement a solution that addresses the page split issues.
Which statement should you execute?

- A. ALTER INDEX IX_Orders_ShipDate ON Orders
REBUILD WITH (PAD_INDEX=OFF, DROP_EXISTING = ON);
- B. ALTER INDEX IX_Orders_ShipDate ON Orders
REBUILD WITH (FILLFACTOR=50, DROP_EXISTING = ON);
- C. ALTER INDEX IX_Orders_ShipDate ON Orders
REBUILD WITH (FILLFACTOR = 0, DROP_EXISTING = ON);
- D. ALTER INDEX IX_Orders_ShipDate ON Orders
REBUILD WITH (PAD_INDEX=ON, DROP_EXISTING = ON);

Answer: B

Question: 10

You need to ensure that usp_AddXMLOrder can be used to validate the XML input from the retailers.
Which parameters should you add to usp_AddXMLOrder on line 04 and line 05? (Each correct answer presents part of the solution. Choose all that apply.)

- A. @schema varbinary(100).
- B. @items varchar(max).
- C. @schema sysname.
- D. @items varbinary(max).
- E. @items xml.
- F. @schema xml.

Answer: C, E

Question: 11

You need to implement a solution that addresses the index monitoring requirements.
What should you do?

- A. Schedule a SQL Server Agent job that saves data from the dynamic management views to a table in the database.
- B. Create a SQL Server Audit that saves data to a log file, and then create a SQL Server Audit Specification that gathers data from the DATABASE_OPERATION group.
- C. Create a performance monitor Data Collector Set (DCS) that monitors the SQL Server counters.
- D. Schedule a SQL Server Profiler trace, and then save the trace data to a table in the database.

Answer: A

Question: 12

You need to implement a solution that solves the performance issues of
usp_GetOrdersAndItems.

Which statements should you execute?

- A. CREATE INDEX IX_Orders_Active ON Orders(ShipDate, DeliveryDate, Amount)
- B. CREATE INDEX IX_Orders_Active ON Orders(DeliveryDate) INCLUDE(Amount)
WHERE ShipDate IS NULL
- C. CREATE INDEX IX_Orders_Active ON Orders(DeliveryDate, Amount) WHERE ShipDate
IS NULL
- D. CREATE INDEX IX_Orders_Active ON Orders(ShipDate, DeliveryDate)
INCLUDE(Amount)

Answer: B

Question: 13

You need to modify usp.GetOrdersAndItems to ensure that an order is NOT retrieved by usp_GetOrdersAndItems while the order is being updated.

What should you add to usp.GetOrdersAndItems?

- A. Add WITH (NOLOCK) to the end of line 47.
- B. Add SET TRANSACTION ISOLATION LEVEL READ COMMITTED to line 44.
- C. Add SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED to line 44.
- D. Add WITH (READPAST) to the end of line 47.

Answer: B

Question: 14

You need to ensure that a new execution plan is used by usp_GetOrdersByProduct each time the stored procedure runs.

What should you do?

- A. Execute sp_help usp_GetOrdersByProduct\
- B. Add WITH (FORCESEEK) to line 69 in usp.GetOrdersByProduct.
- C. Add WITH RECOMPILE to line 64 in usp.GetOrdersByProduct.
- D. Execute sp_recompile usp.GetOrdersByProduct'.

Answer: B

Case Study: 7

Fourth Coffee

Background

Corporate Information

Fourth Coffee is global restaurant chain. There are more than 5,000 locations worldwide.

Physical Locations

Currently a server at each location hosts a SQL Server 2012 instance. Each instance contains a database called StoreTransactions that stores all transactions from point of sale and uploads summary batches nightly. Each server belongs to the COFFECORP domain. Local computer accounts access the StoreTransactions database at each store using sysadmin and datareaderwriter roles.

Planned changes

Fourth Coffee has three major initiatives:

- The IT department must consolidate the point of sales database infrastructure.
- The marketing department plans to launch a mobile application for micropayments.
- The finance department wants to deploy an internal tool that will help detect fraud.

Initially, the mobile application will allow customers to make micropayments to buy coffee and other items on the company web site. These micropayments may be sent as gifts to other users and redeemed within an hour of ownership transfer. Later versions will generate profiles based on customer activity that will push texts and ads generated by an analytics application.

When the consolidation is finished and the mobile application is in production, the micropayments and point of sale transactions will use the same database.

Existing Environment

Existing Application Environment

Some stores have been using several pilot versions of the micropayment application. Each version currently is in a database that is independent from the point of sales systems. Some versions have been used in field tests at local stores, and others are hosted at corporate servers. All pilot versions were developed by using SQL Server 2012.

Existing Support Infrastructure

The proposed database for consolidating micropayments and transactions is called CoffeeTransactions. The database is hosted on a SQL Server 2014 Enterprise Edition instance and has the following file structures:

Database name:	CoffeeTransactions										
Owner:	BUILTIN\Administrators										
<input checked="" type="checkbox"/> Use full-text indexing											
Database files:											
<table border="1"><thead><tr><th>Logical Name</th><th>File Type</th><th>Filegroup</th></tr></thead><tbody><tr><td>CoffeeTrans...</td><td>ROWS Data</td><td>PRIMARY</td></tr><tr><td>CoffeeTrans...</td><td>LOG</td><td>Not Applicable</td></tr></tbody></table>			Logical Name	File Type	Filegroup	CoffeeTrans...	ROWS Data	PRIMARY	CoffeeTrans...	LOG	Not Applicable
Logical Name	File Type	Filegroup									
CoffeeTrans...	ROWS Data	PRIMARY									
CoffeeTrans...	LOG	Not Applicable									

Business Requirements

General Application Solution Requirements

The database infrastructure must support a phased global rollout of the micropayment application and consolidation.

The consolidated micropayment and point of sales database will be into a CoffeeTransactions database. The infrastructure also will include a new CoffeeAnalytics database for reporting on content from CoffeeTransactions.

Mobile applications will interact most frequently with the micropayment database for the following activities:

- Retrieving the current status of a micropayment;

- Modifying the status of the current micropayment; and
- Canceling the micropayment.

The mobile application will need to meet the following requirements:

- Communicate with web services that assign a new user to a micropayment by using a stored procedure named `usp_AssignUser`.
- Update the location of the user by using a stored procedure named `usp_AddMobileLocation`.

The fraud detection service will need to meet the following requirements:

- Query the current open micropayments for users who own multiple micropayments by using a stored procedure named `usp_LookupConcurrentUsers`.
- Persist the current user locations by using a stored procedure named `usp_MobileLocationSnapshot`.
- Look at the status of micropayments and mark micropayments for internal investigations.
- Move micropayments to `dbo.POSException` table by using a stored procedure named `ups_DetectSuspiciousActivity`.
- Detect micropayments that are flagged with a `StatusId` value that is greater than 3 and that occurred within the last minute.

The CoffeeAnalytics database will combine imports of the `POSTransaction` and `MobileLocation` tables to create a `UserActivity` table for reports on the trends in activity. Queries against the `UserActivity` table will include aggregated calculations on all columns that are not used in filters or groupings.

Micropayments need to be updated and queried for only a week after their creation by the mobile application or fraud detection services.

Performance

The most critical performance requirement is keeping the response time for any queries of the `POSTransaction` table predictable and fast.

Web service queries will take a higher priority in performance tuning decisions over the fraud detection agent queries.

Scalability

Queries of the user of a micropayment cannot return while the micropayment is being updated, but can show different users during different stages of the transaction.

The fraud detection service frequently will run queries over the micropayments that occur over different time periods that range between 30 seconds and ten minutes.

The `POSTransaction` table must have its structure optimized for hundreds of thousands of active micropayments that are updated frequently.

All changes to the `POSTransaction` table will require testing in order to confirm the expected throughput that will support the first year's performance requirements.

Updates of a user's location can tolerate some data loss.

Initial testing has determined that the `POSTransaction` and `POSException` tables will be migrated to an in-memory optimized table.

Availability

In order to minimize disruption at local stores during consolidation, nightly processes will restore the databases to a staging server at corporate headquarters.

Technical Requirements

Security

The sensitive nature of financial transactions in the store databases requires certification of the

COFFECORP\Auditors group at corporate that will perform audits of the data. Members of the COFFECORP\Auditors group cannot have sysadmin or datawriter access to the database.

Compliance requires that the data stewards have access to any restored StoreTransactions database without changing any security settings at a database level.

Nightly batch processes are run by the services account in the COFFECORP\StoreAgent group and need to be able to restore and verify the schema of the store databases match.

No Windows group should have more access to store databases than is necessary.

Maintainability

You need to anticipate when POSTransaction table will need index maintenance.

When the daily maintenance finishes, micropayments that are one week old must be available for queries in UserActivity table but will be queried most frequently within their first week and will require support for in-memory queries for data within first week.

The maintenance of the UserActivity table must allow frequent maintenance on the day's most recent activities with minimal impact on the use of disk space and the resources available to queries. The processes that add data to the UserActivity table must be able to update data from any time period, even while maintenance is running.

The index maintenance strategy for the UserActivity table must provide the optimal structure for both maintainability and query performance.

All micropayments queries must include the most permissive isolation level available for the maximum throughput.

In the event of unexpected results, all stored procedures must provide error messages in text message to the calling web service.

Any modifications to stored procedures will require the minimal amount of schema changes necessary to increase the performance.

Performance

Stress testing of the mobile application on the proposed CoffeeTransactions database uncovered performance bottlenecks. The sys.dm_os_wait_stats Dynamic Management View (DMV) shows high wait_time values for WRTTELOG and PAGEIOLATCHJJP wait types when updating the MobileLocation table.

Updates to the MobileLocation table must have minimal impact on physical resources.

Supporting Infrastructure

The stored procedure usp_LookupConcurrentUsers has the current implementation:

```

CREATE PROCEDURE usp_LookupConcurrentUsers
AS BEGIN
    --summary table
    CREATE TABLE #POSTransactionTemp (
        POSTransactionId int NOT NULL,
        UserId int NOT NULL,
        StatusID int NOT NULL,
        POSLocation int NOT NULL,
        CreateDate datetime2 NOT NULL,
        Price money
    )
    DECLARE @timewindow datetime2
    SET @timewindow = GETDATE();

    WITH concurrentusers
    AS
    (
        SELECT UserId, COUNT(*) concurrentsessions
        FROM dbo.POTransaction
        WHERE CreateDate >= dateadd(second,-60, @timewindow )
        GROUP BY UserId
        HAVING COUNT(*) > 1
    )
    INSERT INTO #POSTransactionTemp
    (
        POSTransactionId, UserId,
        StatusID, POSLocation,
        CreateDate, Price
    )

    SELECT d.*
    FROM dbo.POTransaction d
    JOIN concurrentusers c
    on d.UserID = c.UserId
    WHERE d.CreateDate >= dateadd(second,-60, @timewindow )
    ...
    SELECT * FROM #POSTransactionTemp
END

```

The current stored procedure for persisting a user location is defined in the following code:

```

CREATE PROCEDURE dbo.usp_MobileLocationSnapshot
AS
BEGIN

    INSERT INTO CoffeeAnalytics.dbo.MobileLocationLog
    SELECT * FROM CoffeeTransactions.dbo.MobileLocation

END

```

The current stored procedure for managing micropayments needing investigation is defined in the following code:

```

01 CREATE PROCEDURE dbo.usp_DetectSuspiciousActivity
02 WITH NATIVE_COMPILATION, SCHEMABINDING, EXECUTE AS OWNER
03 AS
04 BEGIN ATOMIC
05   WITH (TRANSACTION ISOLATION LEVEL = SNAPSHOT,
06 LANGUAGE = 'us_english')
07   IF EXISTS(SELECT POSTransactionId FROM dbo.POTransaction
08 WHERE StatusID >= 4 and CreateDate >= dateadd(second,-60,
09 GETDATE() ))
10   MERGE dbo.POException AS target
11   USING (SELECT POSTransactionId, StatusID, UserId,
12 POSLocation, CreateDate, Price FROM dbo.POTransaction
13 WHERE StatusID >= 4 and
14 CreateDate >= dateadd(second,-60, GETDATE() ))
15   AS source (POTransactionId, StatusID, UserId,
16 POSLocation, CreateDate, Price)
17   ON (target.POTransactionId = source.POTransactionId)
18 WHEN MATCHED THEN
19   UPDATE SET StatusID = source.StatusID
20 WHEN NOT MATCHED THEN
21   INSERT (POTransactionId, StatusID, UserId,
22 POSLocation, CreateDate, Price)
23   VALUES (source.POTransactionId, source.StatusID,
24 source.UserId, source.POSLocation,
25 source.CreateDate, source.Price);
26 END

```

The current table, before implementing any performance enhancements, is defined as follows:

```

CREATE TABLE dbo.POTransaction (
    POTransactionId int NOT NULL PRIMARY KEY,
    UserId int NOT NULL,
    POSLocation int NOT NULL,
    StatusID int NOT NULL,
    CreateDate datetime2 NOT NULL,
    Price money
)
CREATE INDEX ix_UserID on dbo.POTransaction(UserId)

```

Question: 1

You need to monitor the health of your tables and indexes in order to implement the required index maintenance strategy.

What should you do?

- A. Query system DMVs to monitor avg_chain_length and max_chain_length. Create alerts to notify you when these values converge.
- B. Create a SQL Agent alert when the File Table: Avg time per file I/O request value is increasing.
- C. Query system DMVs to monitor total_bucket_count. Create alerts to notify you when this value increases.
- D. Query system DMVs to monitor total_bucket_count. Create alerts to notify you when this value decreases.

Answer: A

From scenario:

- * You need to anticipate when POSTransaction table will need index maintenance.
- * The index maintenance strategy for the UserActivity table must provide the optimal structure for both maintainability and query performance.

Question: 2

DRAG DROP

You need to design the UserActivity table.

Which three steps should you perform in sequence? To answer, move the appropriate three actions from the list of actions to the answer area and arrange them in the correct order.

Actions	Answer Area
Create a nonclustered hash index.	
Create a clustered columnstore index.	
Create a partitioning scheme for use by the table.	
Use an ALTER INDEX REBUILD on a specific partition.	
Use an ALTER INDEX REORGANIZE on a specific partition.	

Answer:

Box 1:

Create a clustered columnstore index.

Box 2:

Create a partitioning scheme for use by the table.

Box 3:

Use an ALTER INDEX REORGANIZE on a specific partition.

Note:

- * Creating a partitioned table or index typically happens in four parts:

Create a filegroup or filegroups and corresponding files that will hold the partitions specified by the partition scheme.

Create a partition function that maps the rows of a table or index into partitions based on the values of a specified column.

Create a partition scheme that maps the partitions of a partitioned table or index to the new filegroups.

Create or modify a table or index and specify the partition scheme as the storage location.

- * Reorganizing an index uses minimal system resources.

* From scenario:

/ The index maintenance strategy for the UserActivity table must provide the optimal structure for both maintainability and query performance.

/ The CoffeeAnalytics database will combine imports of the POSTransaction and MobileLocation tables to create a UserActivity table for reports on the trends in activity. Queries against the UserActivity table will include aggregated

calculations on all columns that are not used in filters or groupings.

/ When the daily maintenance finishes, micropayments that are one week old must be available for queries in UserActivity table but will be queried most frequently within their first week and will require support for in-memory queries for data within first week.

The maintenance of the UserActivity table must allow frequent maintenance on the day's most recent activities with minimal impact on the use of disk space and the resources available to queries. The processes that add data to the UserActivity table must be able to update data from any time period, even while maintenance is running.

* Columnstore indexes work well for mostly read-only queries that perform analysis on large data sets. Often, these are queries for data warehousing workloads. Columnstore indexes give high performance gains for queries that use full table scans, and are not well-suited for queries that seek into the data, searching for a particular value.

Question: 3

You need to implement security for the restore and audit process. What should you do?

- A. Grant the COFFECORP\Auditors group ALTER ANY CONNECTION and SELECT ALL USER SECURABLES permissions.
Grant the COFFECORP\StoreAgent group ALTER ANY CONNECTION and IMPERSONATE ANY LOGIN permissions.
- B. Grant the COFFECORP\Auditors group CONNECT ANY DATABASE and IMPERSONATE ANY LOGIN permissions. Grant the COFFECORP\StoreAgent group CONNECT ANY DATABASE and SELECT ALL USER SECURABLES permissions.
- C. Grant the COFFECORP\Auditors group ALTER ANY CONNECTION and IMPERSONATE ANY LOGIN permissions. Grant the COFFECORP\StoreAgent group ALTER ANY CONNECTION and SELECT ALL USER SECURABLES permissions.
- D. Grant the COFFECORP\Auditors group CONNECT ANY DATABASE and SELECT ALL USER SECURABLES permissions.
Grant the COFFECORP\StoreAgent group CONNECT ANY DATABASE and IMPERSONATE ANY LOGIN permissions.

Answer: A

Question: 4

DRAG DROP

You need to create the usp.AssignUser stored procedure.

Develop the solution by selecting and arranging the required code blocks in the correct order. You may not need all of the code blocks.

Code Blocks	Answer Area
<pre>IF @StatusID IS NULL RAISERROR (N'The transaction does not exist.',16,1)</pre>	
<pre>WITH NATIVE_COMPILATION, SCHEMABINDING, EXECUTE AS OWNER</pre>	
<pre>CREATE PROCEDURE dbo.usp_AssignUser @UserId int, @POSTransactionId int</pre>	
<pre>WITH (TRANSACTION ISOLATION LEVEL = READ COMMITTED, LANGUAGE = N'us_english')</pre>	
<pre>UPDATE dbo.POTransaction SET UserId=@UserId WHERE POTransactionId=@POTransactionId END</pre>	
<pre>AS BEGIN</pre>	
<pre>DECLARE @StatusID int SELECT @StatusID=StatusId FROM dbo.POTransaction WHERE POTransactionId=@POTransactionId</pre>	
<pre>IF @StatusID IS NULL THROW 51000, N'The transaction does not exist.', 1</pre>	
<pre>WITH (TRANSACTION ISOLATION LEVEL = REPEATABLE READ, LANGUAGE = N'us_english')</pre>	
<pre>AS BEGIN ATOMIC</pre>	

Answer:

Box 1:

```
CREATE PROCEDURE dbo.usp_AssignUser
@UserId int, @POSTransactionId int
```

Box 2:

```
WITH
NATIVE_COMPILATION, SCHEMABINDING,
EXECUTE AS OWNER
```

Box 3:

```
AS
BEGIN ATOMIC
```

Box 4:

```
WITH (TRANSACTION ISOLATION LEVEL =
REPEATABLE READ, LANGUAGE
= N'us_english')
```

Box 5:

```
UPDATE dbo.POTransaction
SET UserId=@UserId
WHERE POTransactionId=@POTransactionId
END
```

Box 6:

```
DECLARE @StatusID int
SELECT @StatusID=StatusId
FROM dbo.POTransaction
WHERE POTransactionId=@POTransactionId
```

Box 7:

```
IF @StatusID IS NULL
THROW 51000, N'The transaction
does not exist.', 1
```

Note:

* From scenario: The mobile application will need to meet the following requirements:

/Communicate with web services that assign a new user to a micropayment by using a stored procedure named usp_AssignUser.

* Example:

```
create procedure dbo.OrderInsert(@OrdNo integer, @CustCode nvarchar(5))
with native_compilation, schemabinding, execute as owner
as
begin atomic with
(transaction isolation level = snapshot,
language = N'English')
declare @OrdDate datetime = getdate();
insert into dbo.Org (OrdNo, CustCode, OrdDate) values (@OrdNo, @CustCode, @OrdDate);
end
go
```

* 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.

* READ COMMITTED versus REPEATABLE READ

Read committed is an isolation level that guarantees that any data read was committed at the moment is read. It simply restricts the reader from seeing any intermediate, uncommitted, 'dirty' read. IT makes no promise whatsoever that if the transaction re-issues the read, will find the same data, data is free to change after it was read.

Repeatable read is a higher isolation level, that in addition to the guarantees of the read committed level, it also guarantees that any data read cannot change, if the transaction reads the same data again, it will find the previously read data in place, unchanged, and available to read.

* Both RAISERROR and THROW statements are used to raise an error in Sql Server.

The journey of RAISERROR started from Sql Server 7.0, where as the journey of THROW statement has just began with Sql Server 2012. obviously, Microsoft suggesting us to start using THROW statement instead of RAISERROR. THROW statement seems to be simple and easy to use than RAISERROR.

* Explicit transactions. The user starts the transaction through an explicit BEGIN TRAN or BEGIN ATOMIC. The transaction is completed following the corresponding COMMIT and ROLLBACK or END (in the case of an atomic block).

Question: 5

DRAG DROP

You need to implement a new version of usp_AddMobileLocation. Develop the solution by selecting and arranging the required code blocks in the correct order. You may not need all of the code blocks.

Code Blocks	Answer Area
DELAYED_DURABILITY = ON , TRANSACTION ISOLATION LEVEL = SNAPSHOT	
CREATE PROCEDURE dbo.usp_AddMobileLocation @POSTransactionId int, @Long float, @Lat float WITH	
NATIVE_COMPILATION ...	
DELAYED_DURABILITY = OFF , TRANSACTION ISOLATION LEVEL = READ UNCOMMITTED	
DELAYED_DURABILITY = ON , TRANSACTION ISOLATION LEVEL = READ UNCOMMITTED	
Insert into dbo.MobileLocation (POSTransactionId, Longitude, Latitude, CreateDate) VALUES (@POSTransactionId, @Long, @Lat, GETDATE()) END	
, LANGUAGE = N'English')	
AS BEGIN ATOMIC WITH (
DELAYED_DURABILITY = OFF , TRANSACTION ISOLATION LEVEL = SNAPSHOT	

Answer:

Box 1:

```

CREATE PROCEDURE dbo.usp_AddMobileLocation
    @POSTransactionId int, @Long
    float, @Lat float
    WITH

```

Box 2:

```

    NATIVE_COMPILATION
    ...

```

Box 3:

```

    AS
    BEGIN ATOMIC WITH
    (

```

Box 4:

```

    ,DELAYED_DURABILITY = OFF
    ,TRANSACTION ISOLATION LEVEL
    = READ UNCOMMITTED

```

Box 5:

```

    ,LANGUAGE = N'English'
)

```

Box 6:

```

Insert into dbo.MobileLocation
(
    POSTransactionId,
    Longitude,
    Latitude,
    CreateDate
)
VALUES
(
    @POSTransactionId,
    @Long,
    @Lat,
    GETDATE()
)
END

```

Note:

* From scenario:

The mobile application will need to meet the following requirements:

- Update the location of the user by using a stored procedure named usp_AddMobileLocation.

* DELAYED_DURABILITY

SQL Server transaction commits can be either fully durable, the SQL Server default, or delayed durable (also known as lazy commit).

Fully durable transaction commits are synchronous and report a commit as successful and return control to the client only after the log records for the transaction are written to disk. Delayed durable transaction commits are asynchronous and report a commit as successful before the log records for the transaction are written to disk. Writing the transaction log entries to disk is required for a transaction to be durable. Delayed durable transactions become durable when the transaction log entries are flushed to disk.

Question: 6

You need to modify the usp_DetectSuspiciousActivity stored procedure.

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

- A. Replace lines 04-06 with the following code:

```
BEGIN ATOMIC WITH
(
    DELAYED_DURABILITY = ON,
    TRANSACTION ISOLATION LEVEL = READ UNCOMMITTED,
    LANGUAGE = N'English'
)
```

- B. Replace lines 04-06 with the following code:

```
BEGIN ATOMIC WITH
(
    DELAYED_DURABILITY = ON,
    TRANSACTION ISOLATION LEVEL = REPEATABLE READ
)
```

- C. Change the logic of the stored procedure to use separate UPDATE and INSERT statements.

- D. Replace lines 07-09 with the following code:

```
DECLARE @exists BIT = 0
IF EXISTS ( SELECT TOP 1 * FROM POTransaction (NOLOCK) WHERE StatusID = 4 and CreateDate
>= dateadd(second,-60, GETDATE() ))
```

- E. Replace lines 04-06 with the following code:

```
BEGIN ATOMIC WITH
(
    TRANSACTION ISOLATION LEVEL = READ UNCOMMITTED,
    LANGUAGE = N'English'
)
```

- F. Replace lines 07-09 with the following code:

```
DECLARE @exists BIT = 0
SELECT TOP 1 @exists = 1 FROM POTransaction WHERE StatusID >= 4 and CreateDate >= dateadd
(second,-60, GETDATE() )
IF @exists = 1
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D
- E. Option E
- F. Option F

Answer: DE

Note:

* Move micropayments to dbo.POException table by using a stored procedure named ups_DetectSuspiciousActivity.

Question: 7

DRAG DROP

You need to redesign the system to meet the scalability requirements of the application.

Develop the solution by selecting and arranging the required code blocks in the correct order.

You may not need all of the code blocks.

Code Blocks

Answer Area

```
,  
    UserId int NOT NULL  
    INDEX ix_UserId NONCLUSTERED  
    HASH WITH (BUCKET_COUNT=2),
```

```
,  
    UserId int NOT NULL  
    INDEX x_UserId NONCLUSTERED  
    HASH WITH (BUCKET_COUNT=900000),
```

```
    POSLocation int NOT NULL,  
    StatusID int NOT NULL,  
    CreateDate datetime2 NOT NULL,  
    Price money  
)
```

```
    POSTransactionId int NOT NULL  
    PRIMARY KEY CLUSTERED
```

```
    POSTransactionId int NOT NULL
```

```
ALTER DATABASE CoffeeTransactions  
ADD FILEGROUP [CoffeeTransactions_inmem]  
] CONTAINS MEMORY_OPTIMIZED_DATA
```

```
ON [CoffeeTransactions_inmem]
```

```
WITH (MEMORY_OPTIMIZED=ON,  
DURABILITY=SCHEMA_ONLY)
```

```
    POSTransactionId int NOT NULL  
    PRIMARY KEY CLUSTERED  
    HASH WITH (BUCKET_COUNT=1000000)
```

```
,  
    UserId int NOT NULL  
    NONCLUSTERED INDEX ix_UserId,
```

```
CREATE TABLE dbo.POSTransaction (
```

```
    POSTransactionId int NOT NULL  
    PRIMARY KEY NONCLUSTERED  
    HASH WITH (BUCKET_COUNT=1)
```

Answer:

Box 1:

```
ALTER DATABASE CoffeeTransactions  
ADD FILEGROUP [CoffeeTransactions_inmem]  
] CONTAINS MEMORY_OPTIMIZED_DATA
```

Box 2:

```
CREATE TABLE dbo.POSTransaction (
```

Box 3:

```
,  
    UserId int NOT NULL  
    INDEX ix_UserId NONCLUSTERED  
    HASH WITH (BUCKET_COUNT=900000),
```

Box 4:

```
    POSTransactionId int NOT NULL  
    PRIMARY KEY CLUSTERED  
    HASH WITH (BUCKET_COUNT=1000000)
```

Box 5:

```
    POSLocation int NOT NULL,  
    StatusID int NOT NULL,  
    CreateDate datetime2 NOT NULL,  
    Price money  
)
```

Box 6:

```
WITH (MEMORY_OPTIMIZED=ON,  
DURABILITY=SCHEMA_ONLY)
```

Box 7:

```
ON [CoffeeTransactions_inmem]
```

Note:

* MEMORY_OPTIMIZED_DATA

First create a memory-optimized data filegroup and add a container to the filegroup.

Then create a memory-optimized table.

* You must specify a value for the BUCKET_COUNT parameter when you create the memory-optimized table. In most cases the bucket count should be between 1 and 2 times the number of distinct values in the index key.

* Example:

```
-- create a durable (data will be persisted) memory-optimized table  
-- two of the columns are indexed
```

```
CREATE TABLE dbo.ShoppingCart (  
    ShoppingCartId INT IDENTITY(1,1) PRIMARY KEY NONCLUSTERED,  
    UserId INT NOT NULL INDEX ix_UserId NONCLUSTERED HASH WITH (BUCKET_COUNT=1000000),  
    CreatedDate DATETIME2 NOT NULL,  
    TotalPrice MONEY  
) WITH (MEMORY_OPTIMIZED=ON)  
GO
```

Question: 8

DRAG DROP

You need to optimize the index and table structures for POSTransaction.

Which task should you use with each maintenance step? To answer, drag the appropriate tasks to the correct maintenance steps. Each task 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.

Tasks	Maintenance Steps
an identity for UserActivityID starting at the next value	Convert UserActivity to use
a sequence for UserActivityID starting at the next value	Task
on-disk tables using the partitioning scheme	Copy UserActivity metadata to create UserActivity_Archive as
in-memory tables using the partitioning scheme	Task
UserActivity and UserActivity_Archive	After copying UserActivity metadata to create UserActivity_Staging, create a view on top of
UserActivity, UserActivity_Staging, and UserActivity_Archive	Task
Alter the partition function and UserActivity_Staging constraints	After switching a new partition from UserActivity_Staging into UserActivity_Archive,
Alter the partition function and UserActivity_Archive constraints	Task

Answer:

Maintenance Steps

Convert UserActivity to use	a sequence for UserActivityID starting at the next value
Copy UserActivity metadata to create UserActivity_Archive as	on-disk tables using the partitioning scheme
After copying UserActivity metadata to create UserActivity_Staging, create a view on top of	UserActivity and UserActivity_Archive
After switching a new partition from UserActivity_Staging into UserActivity_Archive,	Alter the partition function and UserActivity_Archive constraints

Question: 9

You need to modify the stored procedure usp_LookupConcurrentUsers. What should you do?

- A. Add a clustered index to the summary table.
- B. Add a nonclustered index to the summary table.
- C. Add a clustered columnstore index to the summary table.
- D. Use a table variable instead of the summary table.

Answer: A

Scenario: Query the current open micropayments for users who own multiple micropayments by using a stored procedure named usp.LookupConcurrentUsers

Question: 10

You need to optimize the index structure that is used by the tables that support the fraud detection services. What should you do?

- A. Add a hashed nonclustered index to CreateDate.
- B. Add a not hash nonclustered index to CreateDate.
- C. Add a not hash clustered index on POSTransactionId and CreateDate.
- D. Add a hashed clustered index on POSTransactionId and CreateDate.

Answer: A

The fraud detection service will need to meet the following requirement (among others):

- * Detect micropayments that are flagged with a StatusId value that is greater than 3 and that occurred within the last minute.

Question: 11

You need to modify the stored procedure usp_LookupConcurrentUsers.

What should you do?

- A. Use the summary table as an in-memory optimized table with a non-hash clustered index.
- B. Use the summary table as an in-memory optimized table with a non-hash nonclustered index.
- C. Use a type variable instead of the summary table.
- D. Add a clustered index to the summary table.

Answer: A

Case Study: 8

Mix Questions

Question: 1

DRAG DROP

You have a table named Table1 that contains 1 million rows. Table1 contains a column named Column1 that stores sensitive information. Column1 uses the nvarchar (16) data type.

You have a certificate named Cert1.

You need to replace Column1 with a new encrypted column named Column2 that uses one-way hashing.

Which code segment should you execute before you remove Column1?

To answer, move the appropriate code segments from the list of code segments to the answer area and arrange them in the correct order.

```

OPEN SYMMETRIC KEY Key1
DECRYPTION BY CERTIFICATE Cert1;

CREATE SYMMETRIC KEY Key1 WITH ALGORITHM = S
HA1
ENCRYPTION BY CERTIFICATE Cert1;

ALTER TABLE Table1
ADD Column2 nvarchar(256);

ALTER TABLE Table1
ADD Column2 varbinary(256);

CLOSE SYMMETRIC KEY;

CREATE CREDENTIAL Cred1 WITH IDENTITY = 'Use
r1', SECRET = 'P@ssw0rd';

UPDATE table1 SET Column2 = EncryptByKey
(Key_GUID('Key1'),Column1);

CREATE SYMMETRIC KEY Key1 WITH ALGORITHM = A
ES_256
ENCRYPTION BY CERTIFICATE Cert1;

```

Answer:

Box 1:

```

CREATE SYMMETRIC KEY Key1 WITH ALGORITHM = S
HA1
ENCRYPTION BY CERTIFICATE Cert1;

```

First create a hash key using the certificate.

Not AES: AES is not based on hashing.

Box 2:

```

ALTER TABLE Table1
ADD Column2 varbinary(256);

```

Add a column with varbinary data type.

Box 3:

```

OPEN SYMMETRIC KEY Key1
DECRYPTION BY CERTIFICATE Cert1;

```

Box 4:

```

UPDATE table1 SET Column2 = EncryptByKey
(Key_GUID('Key1'),Column1);

```

Box 5:

```

CLOSE SYMMETRIC KEY;

```

Note:

- * There are a few different hashing algorithms available in SQL Server 2005: MD2, MD4, MD5, SHA, SHA1, with each having pros and cons.

- * In cryptography, SHA-1 is a cryptographic hash function designed by the United States National Security Agency and published by the United States NIST as a US Federal Information Processing Standard. SHA stands for "secure hash algorithm". The four SHA algorithms are structured differently and are distinguished as SHA-0, SHA-1, SHA-2, and SHA-3. SHA-1 is very similar to SHA-0, but corrects an error in the original SHA hash specification that led to significant weaknesses. The SHA-0 algorithm was not adopted by many applications. SHA-2 on the other hand significantly differs from the SHA-1 hash function.

SHA-1 is the most widely used of the existing SHA hash functions, and is employed in several widely used applications and protocols.

- * To encrypt a column of data using a simple symmetric encryption

In Object Explorer, connect to an instance of Database Engine.

On the Standard bar, click New Query.

Copy and paste the following example into the query window and click Execute.

```
USE AdventureWorks2012;
--If there is no master key, create one now.
IF NOT EXISTS
    (SELECT * FROM sys.symmetric_keys WHERE symmetric_key_id = 101)
    CREATE MASTER KEY ENCRYPTION BY
        PASSWORD = '23987hxJKL95QYV4369#ghf0%lekjg5k3fd117r$$#1946kcj$n44ncjhdij'
GO
CREATE CERTIFICATE Sales09
    WITH SUBJECT = 'Customer Credit Card Numbers';
GO
CREATE SYMMETRIC KEY CreditCards_Key11
    WITH ALGORITHM = AES_256
    ENCRYPTION BY CERTIFICATE Sales09;
GO
-- Create a column in which to store the encrypted data.
ALTER TABLE Sales.CreditCard
    ADD CardNumber_Encryptedvarbinary(128);
GO
-- Open the symmetric key with which to encrypt the data.
OPEN SYMMETRIC KEY CreditCards_Key11
    DECRYPTION BY CERTIFICATE Sales09;
-- Encrypt the value in column CardNumber using the
-- symmetric key CreditCards_Key11.
-- Save the result in column CardNumber_Encrypted.
UPDATE Sales.CreditCard
    SET CardNumber_Encrypted = EncryptByKey(Key_GUID('CreditCards_Key11')
        , CardNumber, 1, HashBytes('SHA1', CONVERT(varbinary
        , CreditCardID)));
GO
Reference: SQL Server 2012, Encrypt a Column of Data
Ref: http://www.mssqltips.com/sqlservertip/2431/sql-server-column-level-encryption-example-using-symmetric-keys/
```

Question: 2

You use SQL Azure to store data used by an e-commerce application.
You develop a stored procedure named sp1. Sp1 is used to read and change the price of all the products sold on the e-commerce site.
You need to ensure that other transactions are blocked from updating product data while sp1 is executing.
Which transaction isolation level should you use in sp1?

- A. Repeatable read
- B. Read committed
- C. Serializable
- D. Snapshot

Answer: C

Question: 3

You review a query that runs slowly. The query accesses data in a table named Schema1.Table1. The following is the relevant portion of the execution plan for the query:

```
<MissingIndexes>
  <MissingIndexGroup Impact="95.8296">
    <MissingIndex Database="DB1" Schema="Schema1" Table="Table1">
      <ColumnGroup Usage="EQUALITY">
        <Column Name="Column1" ColumnId="14" />
      </ColumnGroup>
      <ColumnGroup Usage="INEQUALITY">
        <Column Name="Column2" ColumnId="17" />
        <Column Name="Column3" ColumnId="21" />
      </ColumnGroup>
      <ColumnGroup Usage="INCLUDE">
        <Column Name="Column4" ColumnId="11" />
      </ColumnGroup>
    </MissingIndex>
  </MissingIndexGroup>
</MissingIndexes>
```

You need to create the missing index.

Which code segment should you execute?

- A. CREATE NONCLUSTERED INDEX IX1 on Schema1.Table1 (Column1) INCLUDE (Column4) WHERE Column2 <> Column3
- B. CREATE NONCLUSTERED INDEX IX1 on Schema1.Table1 (Column1)
- C. CREATE NONCLUSTERED INDEX IX1 on Schema1.Table1 (Column1, Column2, Column3) INCLUDE (Column4)
- D. CREATE NONCLUSTERED INDEX IX1 on schema1.Table1 (Column1) INCLUDE (Column4)

Answer: C

Question: 4

You use SQL Server 2012 to maintain the data used by the applications at your company.

You plan to create a table named Table1 by using the following statement. (Line numbers are included for reference only.)

```
01 CREATE TABLE dbo.table1(
02   ID int IDENTITY(1,1) NOT NULL,
03
04   Email varchar(100) NULL,
05   CONSTRAINT PK_table1 PRIMARY KEY CLUSTERED(ID ASC)
06 )
```

You need to ensure that Table1 contains a column named UserName. The UserName column will:

Store string values in any language.

Accept a maximum of 200 characters.

Be case-insensitive and accent-insensitive.

Which code segment should you add at line 03?

- A. UserName nvarchar(200) COLLATE Latin1_General_CS_AS NOT NULL,
- B. UserName varchar(200) COLLATE Latin1_General_CI_AI NOT NULL,
- C. UserName varchar(200) COLLATE Latin1_General_CS_AS NOT NULL,
- D. UserName nvarchar(200) COLLATE Latin1_General_CI_AI NOT NULL,

Answer: D

Question: 5

You execute the following code:

```
CREATE TABLE dbo.Projects
( Id int,
details XML);
GO

INSERT INTO Projects (Id,details)
VALUES
(1,
N'<Project Name="Project1">
<Tasks>
  <Task Name="T1"><IsFinished>true</IsFinished></Task>
  <Task Name="T2"><IsFinished>true</IsFinished></Task>
</Tasks>
</Project>'),
(2,
N'<Project Name="Project2">
<Tasks>
  <Task Name="T_1"><IsFinished>false</IsFinished></Task>
</Tasks>
</Project>');
```

You need to select the task that has an IsFinished value of true from the Project that has an Id value of 1.
Which code segment should you use?

- A.

```
SELECT Projects.details.query('Project/Tasks/Task[@IsFinished="true"]')
FROM Projects
WHERE Projects.Id = 1;
```
- B.

```
SELECT Projects.details.query('//Task/IsFinished="true"')
FROM Projects
WHERE Projects.Id = 1;
```
- C.

```
SELECT Projects.details
FROM Projects
WHERE Projects.Id = 1 AND Details LIKE '%true%';
```
- D.

```
SELECT Projects.details.query('//Task[@IsFinished="true"]')
FROM Projects
WHERE Projects.Id = 1;
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: B

Question: 6

You are creating a table to support an application that will cache data outside of SQL Server.

The application will detect whether cached values were changed before it updates the values. You need to create the table, and then verify that you can insert a row into the table. Which code segment should you use?

- C A. CREATE TABLE Table1


```

        (
          ID int IDENTITY(1,1),
          Name varchar(100),
          Version uniqueidentifier DEFAULT (NEWID())
        )
        INSERT INTO Table1 (Name, Version)
        VALUES ('Smith, Ben', NEWID())
      
```
- C B. CREATE TABLE Table1


```

        (
          ID int IDENTITY(1,1),
          Name varchar(100),
          Version uniqueidentifier DEFAULT (NEWID())
        )
        INSERT INTO Table1 (Name)
        VALUES ('Smith, Ben')
      
```
- C C. CREATE TABLE Table1


```

        (
          ID int IDENTITY(1,1),
          Name varchar(100),
          Version rowversion
        )
        INSERT INTO Table1 (Name)
        VALUES ('Smith, Ben')
      
```
- C D. CREATE TABLE Table1


```

        (
          ID int IDENTITY(1,1),
          Name varchar(100),
          Version rowversion
        )
        INSERT INTO Table1 (Name, Version)
        VALUES ('Smith, Ben', NEWID())
      
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: C

Explanation:
<http://msdn.microsoft.com/en-us/library/ms182776.aspx>
<http://msdn.microsoft.com/en-us/library/ms187942.aspx>

<http://msdn.microsoft.com/en-us/library/ms190348.aspx>

Question: 7

Your company has a SQL Azure subscription.
You implement a database named Database1. Database1 has two tables named Table1 and Table2.
You create a stored procedure named sp1. Sp1 reads data from Table1 and inserts data into Table2.
A user named User1 informs you that he is unable to run sp1.
You verify that User1 has the SELECT permission on Table1 and Table2.
You need to ensure that User1 can run sp1. The solution must minimize the number of permissions assigned to User1.
What should you do?

A. Change sp1 to run as the saUser.
B. Grant User1 the EXECUTE permission on sp1.
C. Add User1 to the db_datawriter role.
D. Grant User1 the INSERT permission on Table2.

Answer: B

Explanation:

<http://msdn.microsoft.com/en-us/library/ms191291.aspx>

Question: 8

DRAG DROP

You run the following code segment:

```
CREATE TABLE dbo.Customers
(
    Id int CONSTRAINT Check_ID PRIMARY KEY,
    CustomerName varchar(50),
    Details xml
) ;
GO
CREATE PRIMARY XML INDEX PXML_Customers
    ON dbo.Customers (Details);
GO
```

After you add 10,000 rows to Customers, you discover that the index is fragmented.

You need to defragment the index in the least amount of time.

Which code segment should you execute?

To answer, drag the appropriate value to the correct location in the code segment in the answer area. (Answer choices may be used once, more than once, or not at all.)

Values	Answer Area
<input type="button" value="ON"/> <input type="button" value="OFF"/>	<pre>ALTER INDEX ALL ON Customers REBUILD WITH (ONLINE = <input type="button" value="Value"/>, STATISTICS_NORECOMPUTE <input type="button" value="Value"/>);</pre>

Answer:

Values	Answer Area
<input type="button" value="OFF"/> <input type="button" value="ON"/>	<pre>ALTER INDEX ALL ON Customers REBUILD WITH (ONLINE = <input type="button" value="OFF"/>, STATISTICS_NORECOMPUTE <input type="button" value="ON"/>);</pre>

Note:

Locking the table during the process and not recomputing statistics would be the fastest.

* Online = OFF

Table locks are applied for the duration of the index operation. An offline index operation that creates, rebuilds, or drops a clustered, spatial, or XML index, or rebuilds or drops a nonclustered index, acquires a Schema modification (Sch-M) lock on the table. This prevents all user access to the underlying table for the duration of the operation. An offline index operation that creates a nonclustered index acquires a Shared (S) lock on the table. This prevents updates to the underlying table but allows read operations, such as SELECT statements.

* STATISTICS_NORECOMPUTE = ON

Out-of-date statistics are not automatically recomputed.

Reference: ALTER INDEX (Transact-SQL)

Question: 9

You execute the following code:

```
CREATE TABLE dbo.Customers
(
    id int PRIMARY KEY,
    CustomerName char(10)
)
```

You create a nonclustered index named IX_CustomerName on the CustomerName column.

You execute the following query:

```
SELECT * FROM dbo.Customers
WHERE LEFT(CustomerName,1) = 'a'
```

You need to reduce the amount of time it takes to execute the query.

What should you do?

- A. Partition the table and use the CustomerName column for the partition scheme.
- B. Replace IX_CustomerName with a clustered index.
- C. Replace LEFT(CustomerName ,1) = 'a' with CustomerName LIKE 'a%'.
- D. Replace LEFT(CustomerName ,1) = 'a' with SUBSTRING(CustomerName ,1,1) - 'a'.

Answer: C

Explanation:

<http://msdn.microsoft.com/en-us/library/ms179859.aspx>
<http://msdn.microsoft.com/en-us/library/ms187748.aspx>

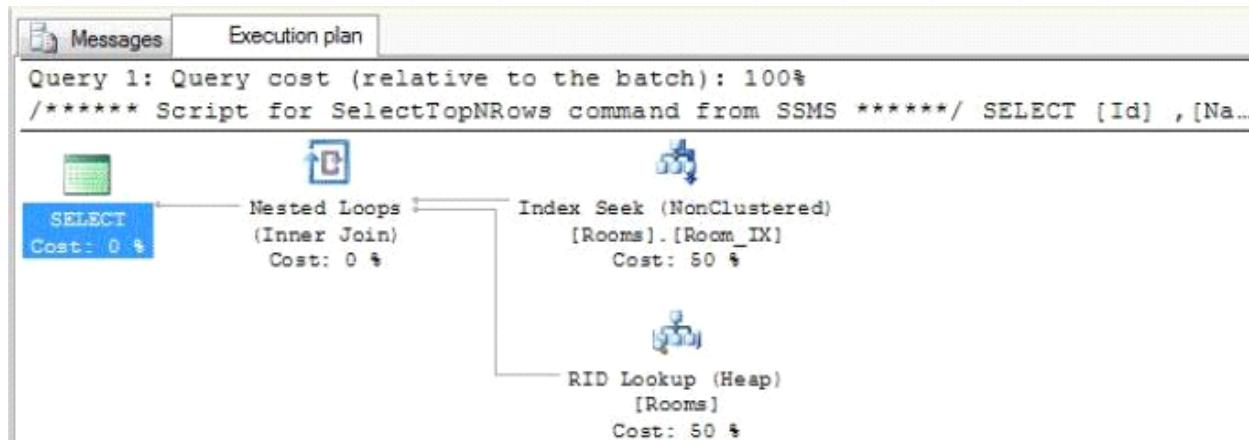
Question: 10

You have a table named Rooms that contains three columns.

You execute the following query:

```
SELECT [Id],
       [RoomName],
       [Position]
  FROM [dbo].[Rooms]
 WHERE [RoomName] = 'Room1'
```

You discover the execution plan shown in the exhibit. (Click the Exhibit button.)



You need to recommend a solution to reduce the amount of time it takes to execute the query.

What should you do?

More than one answer choice may achieve the goal. Select the BEST answer.

- A. Include the RoomName column and the Position column in the Room_IX index.
- B. Create a nonclustered index for RoomName, Id, and Position.
- C. Create a clustered index for Id.
- D. Use the WITH (INDEX(Room_IX),NOLOCK) query hint.

Answer: B

Question: 11

You have a database named database1.

Database developers report that there are many deadlocks.

You need to implement a solution to monitor the deadlocks. The solution must meet the following requirements:

Support real-time monitoring.

Be enabled and disabled easily.
Support querying of the monitored data.
What should you implement?
More than one answer choice may achieve the goal. Select the BEST answer.

- A. Log errors by using trace flag 1222
- B. Log errors by using trace flag 1204
- C. A SQL Server Profiler template
- D. An Extended Events session

Answer: D

Explanation:

<http://www.sqlservercentral.com/blogs/james-sql-footprint/2012/08/12/monitor-deadlock-in-sql-2012/>
http://blogs.technet.com/b/mspfe/archive/2012/06/28/how_2d00_to_2d00_monitor_2d00_deadlocks_2d00_in_2d00_sql_2d00_server.aspx

Question: 12

You execute the following code:

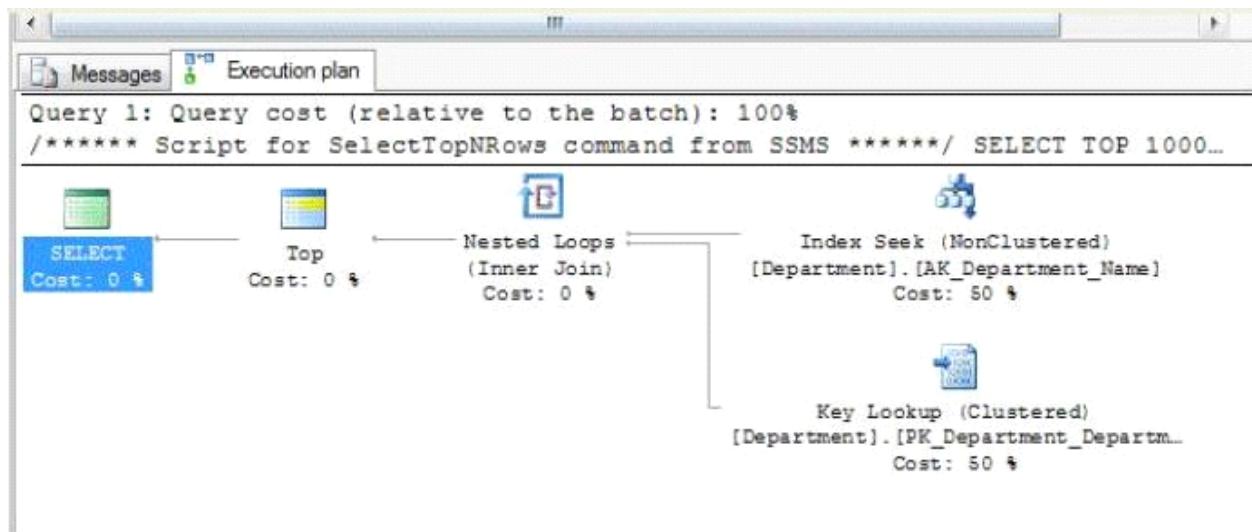
```
CREATE TABLE Department(
    DepartmentID smallint IDENTITY(1,1) NOT NULL,
    DepartmentName varchar(100) NOT NULL,
    GroupName varchar(100) NOT NULL,
    CONSTRAINT PK_Department_DepartmentID
        PRIMARY KEY CLUSTERED (DepartmentID ASC)
);
GO
```

```
CREATE UNIQUE NONCLUSTERED INDEX
    AK_Department_DepartmentName ON
    Department
(
    DepartmentName ASC
);
GO
```

You run the following query:

```
SELECT DepartmentID
    ,DepartmentName
    ,GroupName
FROM Department
WHERE DepartmentName = '1234';
```

The execution plan for the query is shown in the exhibit. (Click the Exhibit button.)



You need to prevent the key lookup.

What should you modify?

More than one answer choice may achieve the goal. Select the BEST answer.

- A.

```
DROP INDEX AK_Department_DepartmentName
ON Department;
GO
CREATE INDEX AK_Department_DepartmentName
ON Department (DepartmentName, GroupName);
GO
```
- B. the SELECT statement to use the WITH(INDEX(PK_Department_DepartmentID)) query hint
- C.

```
DROP INDEX AK_Department_DepartmentName
ON Department;
GO
CREATE INDEX AK_Department_DepartmentName
ON Department (DepartmentName)
INCLUDE (GroupName);
GO
```
- D. the SELECT statement to use the WITH(INDEX(AK_Department_DepartmentName)) query hint

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: C

Question: 13

DRAG DROP

You have a table named Table1. Table1 has 1 million rows.

Table1 has a columnstore index for a column named Column1.

You need to import data to Table1. The solution must minimize the amount of time it takes to import the data.

What should you do?

To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

- Switch Table2 to Table1.
- Create a table named Table2 by using the same schema as Table1.
- Partition Table1.
- Import the data to Table2.
- Import the data to Table1.
- Create a columnstore index on Table2 for Column1.
- Create the columnstore index on Table1.

Answer:

Box 1: Create a table named Table2 by using the same schema as Table1.

Note: Table2 is the staging table.

Box 2: Partition Table1

Box 3: Import the data to Table2.

Box 4: Create a columnstore index on Table2 for Column1.

Box 5: Switch Table2 to Table1

Note:

* An xVelocity memory optimized columnstore index, groups and stores data for each column and then joins all the columns to complete the whole index.

Columnstore indexes can transform the data warehousing experience for users by enabling faster performance for common data warehousing queries such as filtering, aggregating, grouping, and star-join queries.

* Tables that have a columnstore index cannot be updated.

There are three ways to work around this problem.

A) To update a table with a columnstore index, drop the columnstore index, perform any required INSERT, DELETE, UPDATE, or MERGE operations, and then rebuild the columnstore index.

B) (applies in this scenario) Partition the table and switch partitions. For a bulk insert, insert data into a staging table, build a columnstore index on the staging table, and then switch the staging table into an empty partition. For other updates, switch a partition out of the main table into a staging table, disable or drop the columnstore index on the staging table, perform the update operations, rebuild or re-create the columnstore index on the staging table, and then switch the staging table back into the main table.

C) Place static data into a main table with a columnstore index, and put new data and recent data likely to change, into a separate table with the same schema that does not have a columnstore index.

Reference: Best Practices: Updating Data in a Columnstore Index

Question: 14

You have a database for a mission-critical web application. The database is stored on a SQL Server 2012 instance and is the only database on the instance.

The application generates all T-SQL statements dynamically and does not use stored procedures.

You need to maximize the amount of memory available for data caching.

Which advanced server option should you modify?

- A. Optimize for Ad hoc Workloads
- B. Enable Contained Databases
- C. Allow Triggers to Fire Others
- D. Scan for Startup Procs

Answer: A

Question: 15

DRAG DROP

You are designing two stored procedures named Procedure1 and Procedure2.

You identify the following requirements:

Procedure1 must take a parameter that ensures that multiple rows of data can pass into the stored procedure.

Procedure2 must use business logic that resides in a Microsoft .NET Framework assembly.

You need to identify the appropriate technology for each stored procedure.

Which technologies should you identify?

To answer, drag the appropriate technology to the correct stored procedure in the answer area. (Answer choices may be used once, more than once, or not at all.)

Technologies	Answer Area
Common language runtime (CLR)	Procedure1 Technology
Extensible Markup Language (XML)	Procedure2 Technology
a table-valued parameter (TVP)	

Answer:

Technologies	Answer Area
	Procedure1 a table-valued parameter (TVP)
Extensible Markup Language (XML)	Procedure2 Common language runtime (CLR)

Explanation:

<http://msdn.microsoft.com/en-us/library/ms131102.aspx>

<http://msdn.microsoft.com/en-us/library/bb522446.aspx>

<http://msdn.microsoft.com/en-us/library/bb510489.aspx>

Question: 16

You have an application that uses a view to access data from multiple tables.

You need to ensure that you can insert rows into the underlying tables by using the view.

What should you do?

- Create an INSTEAD OF trigger on the view.
- Define the view by using the SCHEMABINDING option.
- Define the view by using the CHECK option.
- Materialize the view.

Answer: C

Explanation:

<http://msdn.microsoft.com/en-us/library/ms180800.aspx>

<http://msdn.microsoft.com/en-us/library/ms187956.aspx>

Question: 17

You have a database that contains a user-defined function named Schema1.Udf1 and two tables named Schema1.Table1 and Schema1.Table2.

Schema1.Table1 has 1 million rows. The schema for Schema1.Table1 is configured as shown in the following table.

Column	Data type
CountryID	int
CustomerName	varchar(50)

Schema1.Udf1 was defined by using the following code:

```
CREATE FUNCTION Schema1.Udf1(@CountryID int)
RETURNS TABLE
AS
RETURN
SELECT Country
FROM Shemal.Table2
WHERE CountryID = @CountryID
```

You need to write a query that will contain the following columns:

Country
CountryID
CustomerName

The solution must meet the following requirements:

Rows must be returned only if the function returns data.

The amount of time it takes the query to execute must be minimized.

Which query should you use?

- A.

```
SELECT t.CountryID,
      u.Country,
      t.CustomerName
   FROM Schema1.Table1 AS t
INNER JOIN Schema1.Udf1(t.CountryID) AS u;
```
- B.

```
SELECT t.CountryID,
      u.Country,
      t.CustomerName
   FROM Schema1.Table1 AS t
CROSS APPLY Schema1.Udf1(t.CountryID) AS u;
```
- C.

```
SELECT t.CountryID,
      u.Country,
      t.CustomerName
   FROM Schema1.Table1 AS t
OUTER APPLY Schema1.Udf1(t.CountryID) AS u;
```
- D.

```
SELECT t.CountryID,
      u.Country,
      t.CustomerName
   FROM Schema1.Table1 AS t
LEFT JOIN Schema1.Udf1(t.CountryID) AS u;
```

- A. Option A
 B. Option B
 C. Option C
 D. Option D

Answer: B

Question: 18

You have a database hosted on SQL Azure.

You are developing a script to create a view that will be used to update the data in a table.

The following is the relevant portion of the script. (Line numbers are included for reference only.)

```
01 CREATE VIEW View1
02 AS
03 SELECT
04 ...
05 WHERE Column1 = 'City1'
06
```

You need to ensure that the view can update the data in the table, except for the data in Column1.

Which code segment should you add at line 06?

- A. WITH CHECK OPTION
 B. WITH VIEW_METADATA
 C. WITH ENCRYPTION
 D. WITH SCHEMABINDING

Answer: A

Explanation:

The question concerning the view that has a clause "WHERE Column1 = 'City1' is wrong. That's not what the CHECK option is made for. Actually you will be able to update ONLY the rows satisfied by that WHERE clause, that is, only the rows with the Column1 being 'City1'.

None of the answers are valid from that question. You need a trigger to achieve that.

<http://msdn.microsoft.com/en-us/library/ms187956.aspx>

Question: 19

DRAG DROP

You are planning two stored procedures named SProc1 and SProc2. You identify the following requirements:

SProc1 must return a table.

SProc2 must return a status code.

You need to identify which options must be implemented to meet each stored procedure requirement.

Which options should you identify?

To answer, drag the appropriate option to the correct requirement in the answer area. (Answer choices may be used once, more than once, or not at all.)

Options	Answer Area
a raise error	SProc1 Option
a return value	
a SELECT statement	SProc2 Option
a table-valued parameter (TVP)	

Answer:

Options	Answer Area
a raise error	SProc1 a SELECT statement
	SProc2 a return value
a table-valued parameter (TVP)	

Question: 20

You have a Microsoft SQL Azure database.

You have the following stored procedure:

```
01 CREATE PROC up_employees
02   @ID int,
03   @Name nvarchar(50)
04 AS
05
06 SELECT Name AS OriginalName
07 FROM HR.Employees
08
09 WHERE ID = @ID;
10
11 UPDATE HR.Employees
12 SET Name = @Name
13
14 WHERE ID = @ID;
```

You discover that the stored procedure periodically fails to update HR.Employees.

You need to ensure that HR.Employees is always updated when up_employees executes.

The solution must minimize the amount of time required for the stored procedure to execute and the number of locks held.

What should you do?

- A. Add the following line of code to line 05:
SET TRANSACTION ISOLATION LEVEL SNAPSHOT
- B. Add the following line of code to line 13:
WITH (UPDLOCK)
- C. Add the following line of code to line 05:
SET TRANSACTION ISOLATION LEVEL SERIALIZABLE
- D. Add the following line of code to line 08:
WITH (UPDLOCK)

Answer: D

Question: 21

You have a database named Database1. Database1 has two stored procedures named Proc1 and Proc2 and a table named Table1. Table1 has millions of rows.

Proc1 updates data in Table1. Proc2 reads data from Table1.

You discover that when Proc1 is executed to update more than 4,000 rows, Proc2 is blocked. The block affects all rows, including those that are not being updated by Proc1.

You need to ensure that when Proc1 is executing, Proc2 can access the data in Table1 that Proc1 is not updating.

What should you change Proc1 to do?

More than one answer choice may achieve the goal. Select the BEST answer.

- A. Update less than 4,000 rows simultaneously.
- B. Use the PAGLOCK table hint.
- C. Wait for Proc2 to complete.
- D. Use the ROWLOCK table hint.

Answer: A

Question: 22

You need to implement a solution that meets the data recovery requirements.
You update each stored procedure to accept a parameter named @transactionID.
What should you add next to the beginning of each stored procedure?

- A. SAVE TRANSACTION WITH MARK @transactionID
- B. ROLLBACK DISTRIBUTED TRANSACTION @transactionID
- C. BEGIN TRANSACTION WITH MARK @transactionID
- D. COMMIT TRANSACTION @transactionID

Answer: C

Question: 23

You plan to create a database that has multiple tables. The tables will contain product information. Each product has a stock-keeping unit (SKU).
You need to recommend a solution to ensure that each SKU starts with the letters "ADV" and is followed by 10 digits.
The solution must minimize the amount of development effort required.
What should you include in the recommendation?

- A. A FOREIGN KEY constraint
- B. A trigger
- C. A user-defined data type
- D. A CHECK constraint

Answer: C

Question: 24

You have a database that contains a table named Department. Department contains the names and locations of each department.
You have a table-valued function named ProjectList() that returns a list of all the projects assigned to a department.
The name of the department is passed as an argument to the ProjectList() function.
You need to create a query that returns a list of all the department names and the project names. The solution must return only departments that are associated to projects.
What should you add to the query?

- A. OUTER APPLY
- B. OUTER JOIN
- C. CROSS JOIN
- D. CROSS APPLY

Answer: D

Question: 25

DRAG DROP
You plan to create a custom aggregation function named Function1.
You plan to deploy Function1 to SQL Server 2012.

You need to ensure that Function1 can access a web service. The solution must minimize the number of changes made to the database.

You create a Microsoft .NET Framework class that contains the definition of Function1.

You upload a certificate to SQL Server.

What three tasks should you perform next?

To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

Answer Area

Use the certificate to add a digital signature to the assembly.

Execute the CREATE FUNCTION statement.

Execute the CREATE ASSEMBLY statement.

Execute the CREATE AGGREGATE statement.

Modify the TRUSTWORTHY property of the database.

Answer:

Box 1: Modify the TRUSTWORTHY property of the database-

Box 2: Execute the CREATE ASSEMBLY statement.

Box 3: Execute the CREATE AGGREGATE statement.

Note:

* TRUSTWORTHY CREATE signature

The TRUSTWORTHY property indicates whether the instance of SQL Server trusts the database and the contents within it.

* CREATE AGGREGATE

Creates a user-defined aggregate function whose implementation is defined in a class of an assembly in the .NET Framework. For the Database Engine to bind the aggregate function to its implementation, the .NET Framework assembly that contains the implementation must first be uploaded into an instance of SQL Server by using a CREATE ASSEMBLY statement.

* Example:

```
ALTER DATABASE [DatabaseName] SET TRUSTWORTHY ON  
GO  
CREATE ASSEMBLY [CLR.Utilities] FROM 'C:\Path\To\File\CLR.Utilities.dll' WITH PERMISSION_SET = UNSAFE  
GO  
CREATE AGGREGATE [dbo].[Concatenate] (@Input nvarchar(max)) RETURNS nvarchar(max)  
EXTERNAL NAME [CLR.Utilities].[CLR.Utilities.Concatenate]  
GO
```

Question: 26

You have a database that uses the following management views:

Sys.dm_os_volume_stats

Sys.dm_db_partition_stats

Sys.dm_db_file_space_usage

Sys.fulltext_indexes

You plan to migrate the database to Microsoft SQL Azure.

You need to identify which view can be used in SQL Azure.

Which view should you identify?

- A. sys.fulltext_indexes
- B. sys.dm_db_file_space_usage
- C. sys.dm_os_volume_stats
- D. sys.dm_db_partition_stats

Answer: D

Question: 27

You have a SQL Server 2012 database named Database1. Database1 has a table named Customers. Customers contains more than 1 million rows. The database has a stored procedure that was created by using the following script:

```
CREATE PROCEDURE up_customers
    @CustomerTypeID nvarchar(400)
AS
SELECT CustomerID,
    FirstName,
    LastName
FROM dbo.customers
WHERE CustomerTypeID IN (@CustomerTypeID);
```

You need to ensure that up_customers returns rows when the following statement is executed:

```
EXECUTE up_customers'1,2,3,4,5';
```

What should you do?

- A. Update @CustcmerTypelist to use the int data type.
- B. Convert @CustomerTypeList to a table variable.
- C. Convert @CustomerTypeList to an XML variable.
- D. Update @CustomerTypeList to use the XML data type.

Answer: B

Question: 28

DRAG DROP

You have a SQL Server 2012 database named database1.

Users report that queries that usually take less than one second to execute, take more than 30 seconds to execute.

You need to view the server resource consumption when the queries are executed.

What should you do?

To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

Import the performance data into SQL Server Profiler.
Save the SQL Server Profiler trace.
Start a SQL Server Profiler trace.
Save the Performance Monitor data.
Start a data collection by using Performance Monitor.

Answer:

- Box 1: Start a SQL Server Profiler trace.
Box 2: Start a data collection by using Performance monitor.
Box 3: Save the SQL Profiler trace.
Box 4: Save the Performance Monitor data.
Box 5: Import the performance data into SQL Server Profiler.

Note:

- * (step1, step 2) Both the Profiler trace and the Performance Monitor logs should be started and stopped at about the same time.
- * (step 3, step 4) Once you have completed capturing the data for both tools, you are ready to perform the correlation analysis.

* (step 5) How to Correlate SQL Server Profiler Data with Performance Monitor Data

Correlating Performance Monitor and Profiler data is a straightforward process that simply involves importing both sets of data into Profiler. Start Profiler and load the trace file you want to correlate.

From the main menu of Profiler, select File | Import Performance Data,

* With SQL Server Profiler, we have the tools to identify the causes of such spikes. We can import Performance Monitor log data and compare it directly with Profiler activity. If we see a spike in CPU utilization, we can identify which statement or statements were running at the same time, and diagnose potential problems.

Question: 29

DRAG DROP

You have a SQL Server 2012 database named Database1. Database1 has a data file named Database1_data.mdf and a transaction log named Database1log.ldf. Database1_data.mdf is 1.5 GB. Database1log.ldf is 1.5 terabytes.

A full backup of Database1 is performed every day.

You need to reduce the size of the log file. The solution must ensure that you can perform transaction log backups in the future.

Which code segment should you execute?

To answer, move the appropriate code segments from the list of code segments to the answer area and arrange them in the correct order.

DBCC SHRINKFILE(database1_log,1)
ALTER DATABASE database1 SET RECOVERY FULL
ALTER DATABASE database1 SET RECOVERY SIMPLE
BACKUP LOG database1 WITH TRUNCATE_ONLY
DBCC SHRINKFILE(database1_data,1)

Answer:

Box 1:

ALTER DATABASE database1 SET RECOVERY SIMPLE

Box 2:

DBCC SHRINKFILE(database1_log,1)

Box 3:

ALTER DATABASE database1 SET RECOVERY FULL

Question: 30

You need to identify which long running transactions use an index.

Which dynamic management view should you use?

- A. sys.dm_exec_query_optimizer_info
- B. sys.dm_exec_connections
- C. sys.dm_exec_query_stats
- D. sys.dm_exec_sessions

Answer: A

Question: 31

You create a table named Customers by using the following code segment:

```
CREATE TABLE dbo.Customers
(
    id int primary key,
    name char(10)
)
```

You create a non-clustered index named IX_Name on the name column.

You write the following query to retrieve all of the customers that have a name that starts with the letters SMI:

```
SELECT * FROM dbo.Customers
WHERE 'smi' = LEFT(name,3)
```

You discover that the query performs a table scan.

You need to ensure that the query uses the index.

What should you do?

- A. Replace LEFT(name,3) = 'smi' by using name like 'smi%'
- B. Replace LEFT(name,3) = 'smi' by using substring(name,1,3) = 'smi'
- C. Recreate IX_Name as a unique index
- D. Recreate IX Name as a clustered index

Answer: A

Question: 32

You plan to execute the following code:

```
01 CREATE TABLE dbo.Table1
02 (
03     datavalue varchar(20)
04 );
05 GO
06 BEGIN TRANSACTION;
07 INSERT INTO Table1 VALUES('entry1');
08 BEGIN TRANSACTION;
09     INSERT INTO Table1 VALUES('entry2');
10 COMMIT TRANSACTION;
11 INSERT INTO Table1 VALUES('entry3');
12 ROLLBACK TRANSACTION;
13 Go
```

You need to identify how many rows will be in dbo.Table1 after you execute the code.
How many rows should you identify?

- A. 0
- B. 1
- C. 2
- D. 3

Answer: A

Question: 33

You plan to design an application that temporarily stores data in a SQL Azure database.
You need to identify which types of database objects can be used to store data for the application. The solution must ensure that the application can make changes to the schema of a temporary object during a session.
Which type of objects should you identify?

- A. Common table expressions (CTEs)
- B. Temporary stored procedures
- C. Temporary tables
- D. Table variables

Answer: C

Explanation:

<http://msdn.microsoft.com/en-us/library/ms175972.aspx>
<http://msdn.microsoft.com/en-us/library/ms189084.aspx>
<http://msdn.microsoft.com/en-us/library/ms175010.aspx>
<http://msdn.microsoft.com/en-us/library/bb510489.aspx>
<http://msdn.microsoft.com/en-us/library/ms187926.aspx>
<http://zacksfiasco.com/post/2010/01/21/SQL-Server-Temporary-Stored-Procedures.aspx>

Question: 34

DRAG DROP

You have a table named Customers that has a clustered index defined on the ID column.

You write a script to create a stored procedure.

You need to complete the script for the stored procedure. The solution must minimize the number of locks and deadlocks.

What should you do?

To answer, drag the appropriate option to the correct location in the answer area. (Answer choices may be used once, more than once, or not at all.)

```

READ COMMITTED CREATE PROCEDURE Proc1 (@ParamID int)
SERIALIZABLE AS
WITH(UPDLOCK) SET TRANSACTION ISOLATION LEVEL
WITH(XLOCK) BEGIN TRANSACTION
DECLARE @var as NCHAR(10)
Select @var = Name
FROM dbo.Customers
WHERE ID = @ParamID
...
UPDATE dbo.Customers
SET Name = @var
WHERE ID = @ParamID
COMMIT TRANSACTION;
GO

```

Answer:

```

SERIALIZABLE CREATE PROCEDURE Proc1 (@ParamID int)
AS
SET TRANSACTION ISOLATION LEVEL READ COMMITTED
WITH(XLOCK) BEGIN TRANSACTION
DECLARE @var as NCHAR(10)
Select @var = Name
FROM dbo.Customers WITH(UPDLOCK)
WHERE ID = @ParamID
...
UPDATE dbo.Customers
SET Name = @var
WHERE ID = @ParamID
COMMIT TRANSACTION;
GO

```

Note:

- * Optimized bulk load operations on heaps block queries that are running under the following isolation levels:
- SNAPSHOT
- READ UNCOMMITTED

READ COMMITTED using row versioning

* 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.

* SERIALIZABLE (more locks)

Specifies the following:

Statements cannot read data that has been modified but not yet committed by other transactions.

No other transactions can modify data that has been read by the current transaction until the current transaction completes.

Other transactions cannot insert new rows with key values that would fall in the range of keys read by any statements in the current transaction until the current transaction completes.

* UPDLOCK

Specifies that update locks are to be taken and held until the transaction completes. UPDLOCK takes update locks for read operations only at the row-level or page-level. If UPDLOCK is combined with TABLOCK, or a table-level lock is taken for some other reason, an exclusive (X) lock will be taken instead.

When UPDLOCK is specified, the READCOMMITTED and READCOMMITTEDLOCK isolation level hints are ignored. For example, if the isolation level of the session is set to SERIALIZABLE and a query specifies (UPDLOCK, READCOMMITTED), the READCOMMITTED hint is ignored and the transaction is run using the SERIALIZABLE isolation level.

* XLOCK

Specifies that exclusive locks are to be taken and held until the transaction completes. If specified with ROWLOCK, PAGLOCK, or TABLOCK, the exclusive locks apply to the appropriate level of granularity.

Reference: Table Hints (Transact-SQL)

Question: 35

You execute the following code:

```
CREATE TABLE UserInfo
(
    ID int NOT NULL IDENTITY (1, 1)
    CONSTRAINT PK_UserInfo PRIMARY KEY CLUSTERED,
    UserName varchar(100) NOT NULL,
    Manager varchar(100) NULL,
    HireDate date NOT NULL,
    PerformanceReviewScore int NULL
);
```

You have a stored procedure that includes the following SELECT statement:

```
SELECT UserName, PerformanceReviewScore
FROM UserInfo
WHERE Manager = 'Ben Smith';
```

You need to create a covering index on UserInfo.

Which code segment should you execute?

- A. CREATE NONCLUSTERED INDEX [IX_Covering_Index] ON UserInfo
(
 [Manager] ASC
)
- B. CREATE NONCLUSTERED INDEX [IX_Covering_Index] ON UserInfo
(
 [UserName] ASC,
 [PerformanceReviewScore] ASC,
)
- C. CREATE NONCLUSTERED INDEX [IX_Covering_Index] ON UserInfo
(
 [Manager] ASC,
 [PerformanceReviewScore] ASC,
 [UserName] ASC
)
- D. CREATE NONCLUSTERED INDEX [IX_Covering_Index] ON UserInfo
(
 [UserName] ASC,
 [Manager] ASC
)

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: C

Question: 36

You have an index for a table in a SQL Azure database. The database is used for Online Transaction Processing (OLTP). You discover that the index consumes more physical disk space than necessary. You need to minimize the amount of disk space that the index consumes. What should you set from the index options?

- A. STATISTICS_NORECOMPUTE = OFF
- B. FILLFACTOR = 80
- C. FILLFACTOR = 0
- D. STATISTICS_NORECOMPUTE = ON

Answer: C

Explanation:

<http://msdn.microsoft.com/en-us/library/ms177459.aspx>
<http://msdn.microsoft.com/en-us/library/ms188783.aspx>

Question: 37

You run the following code:

```
CREATE TABLE dbo.Orders
(
    Id int CONSTRAINT PK_Order_Id PRIMARY KEY,
    Amount decimal,
    Details xml
);
```

You need to ensure that the root node of the XML data stored in the Details column is <Order_Details>.

What should you implement?

More than one answer choice may achieve the goal. Select the BEST answer.

- A. A user-defined data type
- B. An XML index
- C. A Data Definition Language (DDL) trigger
- D. A data manipulation language (DML) trigger
- E. An XML schema collection

Answer: E

Explanation:

<http://msdn.microsoft.com/en-us/library/ms187856.aspx>

Question: 38

You are creating a table named Orders.

You need to ensure that every time a new row is added to the Orders table, a user-defined function is called to validate the row before the row is added to the table.

What should you use?

More than one answer choice may achieve the goal. Select the BEST answer.

- A. A data manipulation language (DML) trigger
- B. A DEFAULT constraint
- C. A Data Definition Language (DDL) trigger
- D. A CHECK constraint
- E. A FOREIGN KEY constraint

Answer: D

Explanation:

<http://www.techrepublic.com/blog/programming-and-development/comparing-sql-serverconstraints-and-dmltriggers/402>

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

Question: 39

You have a SQL Azure database.

You execute the following code:

```
CREATE SCHEMA Sales;
GO

CREATE TABLE Sales.Customers
(
    CustomerID int IDENTITY(1,1) PRIMARY KEY,
    FaxNumber char(10) SPARSE NULL,
    CustomerName varchar(100) NOT NULL,
    EmailAddress varchar(100) NOT NULL
);
GO
```

```
CREATE PROCEDURE Sales.CustomersByFaxNumber
    @FaxNumber char(10)
AS
SELECT CustomerID,
    CustomerName
FROM Sales.Customers
WHERE FaxNumber = @FaxNumber
```

The Sales.Customers table will contain 100,000 rows. You expect the FaxNumber column to contain a null value for 70 percent of the rows.

You need to create an index to support Sales.CustomersByFaxNumber. The solution must minimize the disk storage requirements.

Which code segment should you execute?

- A. CREATE INDEX IX_Customers ON Customers (FaxNumber)
WHERE FaxNumber IS NOT NULL
- B. CREATE INDEX IX_Customers ON Customers (FaxNumber)
WITH FILLFACTOR=0
- C. CREATE INDEX IX_Customers ON Customers (CustomerName)
INCLUDE (FaxNumber)
- D. CREATE INDEX IX_Customers ON Customers (FaxNumber)
- E. CREATE INDEX IX_Customers ON Customers (FaxNumber)
WHERE FaxNumber IS NULL

Answer: A

Question: 40

You plan to create a new column in a table. The column must meet the following requirements:

Be able to store images that are larger than 1 MB each.

Be able to access the images from Microsoft .NET Framework applications.

You need to recommend which data type must be used in the column.

Which data type should you recommend?

More than one answer choice may achieve the goal. Select the BEST answer.

- A. nvarchar
- B. varbinary
- C. image
- D. FileStream

Answer: D

Question: 41

You plan to modify a procedure that contains hundreds of lines of code.
The modification must support the following guidelines:
Use only tables that are not persistent in the database.
Minimize the amount of time required to execute and recompile procedures.
You need to identify which type of table must be used to support the planned modification.
Which type of table should you identify?

- A. A system table
- B. A partitioned table
- C. A table variable
- D. A temporary table

Answer: C

Question: 42

DRAG DROP

You have a SQL Server 2012 database named DB1. DB1 contains four filegroups named FG1, FG2, FG3, and FG4. You execute the following code:

```
CREATE PARTITION FUNCTION PF1 (int)
AS RANGE LEFT FOR VALUES (20120331, 20120630, 20120930);
GO
CREATE PARTITION SCHEME PS1
AS PARTITION PF1
TO (FG1, FG2, FG3, FG4);
GO

CREATE TABLE dbo.Sales
(
    Date_key int NOT NULL,
    Customer_key int,
    Amount money
) ON PS1(Date_key);
GO
```

Two million rows are added to dbo.Sales.

You need to move the data from the first partition to a new table named SalesHistory and, starting on December 31, 2012, repartition dbo.Sales to support new sales data for three months.

Which code segment should you execute?

To answer, move the appropriate code segments from the list of code segments to the answer area and arrange them in the correct order.

```
ALTER PARTITION FUNCTION PF1 MERGE RANGE
(20120331);
```

```
CREATE PARTITION SCHEME PS1
AS PARTITION PF1
TO (FG1, FG2, FG3, FG4);
```

```
DROP PARTITION SCHEME PS1;
```

```
CREATE PARTITION FUNCTION PF1 (int)
AS RANGE LEFT FOR VALUES
(20120630, 20120930, 20121231);
```

```
CREATE TABLE SalesHistory
(
    Date_key int NOT NULL,
    Customer_key int,
    Amount money
) ON PS1(Date_key);
```

```
ALTER TABLE SalesHistory SWITCH 1 TO Sales;
```

```
DROP PARTITION FUNCTION PF1
```

```
ALTER PARTITION FUNCTION PF1 SPLIT RANGE
(20121231);
```

```
CREATE TABLE SalesHistory
(
    Date_key int NOT NULL,
    Customer_key int,
    Amount money
) ON FG1;
```

```
ALTER TABLE Sales SWITCH 1 TO SalesHistory;
```

Answer:

Box 1:

```
CREATE TABLE SalesHistory
(
    Date_key int NOT NULL,
    Customer_key int,
    Amount money
) ON PS1(Date_key);
```

Box 2:

```
ALTER TABLE Sales SWITCH 1 TO SalesHistory;
```

Box 3:

```
DROP PARTITION SCHEME PS1;
```

Box 4:

```
DROP PARTITION FUNCTION PF1
```

Box 5:

```
CREATE PARTITION FUNCTION PF1 (int)
AS RANGE LEFT FOR VALUES
(20120630, 20120930, 20121231);
```

Box 6:

```
CREATE PARTITION SCHEME PS1
AS PARTITION PF1
TO (FG1, FG2, FG3, FG4);
```

Note:

* Box 1 – Box 2:

/ You need to move the data from the first partition to a new table named SalesHistory.

/ First create the new table, then move the contents of the first partition.

*(Box 3 Box 4) Drop the partition scheme and then the partition function and the recreate them (box 5-box6). First recreate the partition function.

/ You need, starting on December 31, 2012, repartition dbo.Sales to support new sales data for three months.

/ A partition function can be dropped only if there are no partition schemes currently using the partition function. If there are partition schemes using the partition function, DROP PARTITION FUNCTION returns an error.

Question: 43

DRAG DROP

You have a database named database1. Each table in database1 has one index per column.

Users often report that creating items takes a long time.

You need to perform the following maintenance tasks:

Identify unused indexes.

Identify indexes that need to be defragmented.

What should you use?

To answer, drag the appropriate function to the correct management task in the answer area. (Answer choices may be used once, more than once, or not at all.)

Functions	Answer Area
sys.dm_db_index_usage_stats	Identify unused indexes.
sys.dm_db_index_operational_stats	Identify indexes that need to be defragmented.
sys.dm_db_index_physical_stats	
sys.dm_db_missing_index_columns	
sys.dm_db_missing_index_details	
sys.dm_db_missing_index_groups	

Answer:

Functions	Answer Area
	Identify unused indexes.
sys.dm_db_index_operational_stats	sys.dm_db_index_usage_stats
	Identify indexes that need to be defragmented.
sys.dm_db_missing_index_columns	sys.dm_db_index_physical_stats
sys.dm_db_missing_index_details	
sys.dm_db_missing_index_groups	

Note:

* sys.dm_db_index_usage_stats

Returns counts of different types of index operations and the time each type of operation was last performed.

* sys.dm_db_index_physical_stats

Returns size and fragmentation information for the data and indexes of the specified table or view.

Question: 44

You have a SQL Azure database.

You execute the following script:

```
CREATE TABLE dbo.Table1
(
    Column1 int PRIMARY KEY,
    Column2 varchar(50) SPARSE NULL
)
```

You add 1 million rows to Table1. Approximately 85 percent of all the rows have a null value for Column2.

You plan to deploy an application that will search Column2.

You need to create an index on Table1 to support the planned deployment. The solution must minimize the storage requirements.

Which code segment should you execute?

- A. CREATE INDEX IX_Table1 ON Table1 (Column2)
WITH FILLFACTOR=0
- B. CREATE INDEX IX_Table1 ON Table1 (Column1)
INCLUDE (Column2)
- C. CREATE INDEX IX_Table1 ON Table1 (Column2)
WHERE Column2 IS NULL
- D. CREATE INDEX IX_Table1 ON Table1 (Column2)
WHERE Column2 IS NOT NULL

Answer: D

Explanation:

<http://msdn.microsoft.com/en-us/library/ms188783.aspx>

<http://msdn.microsoft.com/en-us/library/cc280372.aspx>

Question: 45

You are creating a table named Orders.

You need to ensure that every time a new row is added to the Orders table, a table that is used for auditing is updated.

What should you use?

More than one answer choice may achieve the goal. Select the BEST answer.

- A. A Data Definition Language (DDL) trigger
- B. A DEFAULT constraint
- C. A CHECK constraint
- D. A FOREIGN KEY constraint
- E. A data manipulation language (DML) trigger

Answer: E

Explanation:

<http://www.techrepublic.com/blog/programming-and-development/comparing-sql-serverconstraints-and-dmltriggers/402>
<http://msdn.microsoft.com/en-us/library/ms178110.aspx>

Question: 46

You have a text file that contains an XML Schema Definition (XSD).

You have a table named Schema1.Table1.

You have a stored procedure named Schema1.Proc1 that accepts an XML parameter named Param1.

You need to store validated XML data in Schema1.Table1. The solution must ensure that only valid XML data is accepted by Param1.

What should you do? (Each correct answer presents part of the solution. Choose all that apply.)

- A. Define an XML column in Table1 by using an XML schema collection.
- B. Create an XML schema collection in the database from the text file.
- C. Declare Param1 var1 as type XML and associate the variable to the XML schema collection.
- D. use the modify method to insert the XML schema into each row of the XML column in Table1.

Answer: A, B, D

Explanation/Reference:

<http://msdn.microsoft.com/en-us/library/bb510420.aspx>
<http://msdn.microsoft.com/en-us/library/ms187856.aspx>
<http://msdn.microsoft.com/en-us/library/ms176009.aspx>
<http://msdn.microsoft.com/en-us/library/hh403385.aspx>
<http://msdn.microsoft.com/en-us/library/ms184277.aspx>

Question: 47

You have an index for a table in a SQL Azure database. The database is used for Online Transaction Processing (OLTP).

You discover that many page splits occur when records are inserted or updated in the table.

You need to minimize the number of page splits.

What should you set from the index options?

- A. FILLFACTOR = 0
- B. STATISTICS_NORECOMPUTE = OFF
- C. STATISTICS_NORECOMPUTE = ON
- D. FILLFACTOR = 80

Answer: D

Explanation:

<http://msdn.microsoft.com/en-us/library/ms188783.aspx>
<http://msdn.microsoft.com/en-us/library/ms177459.aspx>

Question: 48

You have a SQL Server 2012 database named Database1.

You execute the following code:

```
CREATE TABLE Sales
(
    ID int IDENTITY(1,1) NOT NULL PRIMARY KEY,
    OrderDate char(10) NOT NULL,
    Amount decimal
);
GO

CREATE INDEX IX_Sales_OrderDate
    ON Sales(OrderDate)
    INCLUDE (ID, Amount);
GO

CREATE PROC usp_Proc1(
    @date1 datetime,
    @date2 datetime
)
AS
SELECT ID, OrderDate, Amount
    FROM Sales
    WHERE CAST(OrderDate AS datetime)
        BETWEEN @date1 AND @date2
    ORDER BY ID;
GO
```

You insert 3 million rows into Sales.

You need to reduce the amount of time it takes to execute Proc1.

What should you do?

C A. Change the query inside Proc1 to:

```
SELECT ID, OrderDate, Amount
FROM Sales
WHERE OrderDate BETWEEN CONVERT(char(10), @date1, 112)
AND CONVERT(char(10), @date2, 112)
ORDER BY ID;
```

C B. Change the definition of Proc1 to:

```
CREATE PROC usp_Proc1(
@date1 int, @date2 int
)
```

C C. Change the query inside Proc1 to:

```
SELECT ID, OrderDate, Amount
FROM Sales
WHERE CAST(OrderDate AS datetime) < @date1
AND CAST(OrderDate AS datetime) > @date2
ORDER BY ID;
```

C D. Change the definition of Proc1 to:

```
CREATE PROC usp_Proc1(
@date1 date, @date2 date
)
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: A

Question: 49

DRAG DROP

You execute the following code:

```
CREATE TABLE Customers
(
    id int primary key,
    name nchar(10)
)
GO
```

You discover that the Customers table was created in the dbo schema.

You need to create a code segment to move the table to another schema named Schema2.

What should you create?

To answer, drag the appropriate code segments to the correct location in the answer area. (Answer choices may be used once, more than once, or not at all.)

Code Segments	Answer Area			
ALTER SCHEMA	Code	Code	Code	Code
ALTER TABLE				
dbo				
dbo.Customers				
EXEC sp_rename				
TRANSFER				
Schema2				

Answer:

Code Segments	Answer Area			
	ALTER SCHEMA	Schema2	TRANSFER	dbo.Customers
ALTER TABLE				
dbo				
EXEC sp_rename				

Explanation:

<http://msdn.microsoft.com/en-us/library/ms173423.aspx>

Question: 50

DRAG DROP

You plan to deploy two stored procedures named SP1 and SP2 that read data from the database.

Your company identifies the following requirements for each stored procedure:

SP1 must allow dirty reads.

SP2 must place range locks on the data to ensure read consistency.

You need to identify which isolation level you must set for each stored procedure. The solution must minimize the

number of locks.

Which isolation level should you identify?

To answer, drag the appropriate isolation level to the correct stored procedure in the answer area. (Answer choices may be used once, more than once, or not at all.)

Isolation Levels	Answer Area
read committed	SP1 Isolation level
read uncommitted	SP2 Isolation level
repeatable read	
serializable	
snapshot	

Answer:

Isolation Levels	Answer Area
read committed	SP1 read uncommitted
repeatable read	SP2 serializable
snapshot	

Note:

* READ UNCOMMITTED

Specifies that statements can read rows that have been modified by other transactions but not yet committed.

Transactions running at the READ UNCOMMITTED level do not issue shared locks to prevent other transactions from modifying data read by the current transaction. READ UNCOMMITTED transactions are also not blocked by exclusive locks that would prevent the current transaction from reading rows that have been modified but not committed by other transactions. When this option is set, it is possible to read uncommitted modifications, which are called dirty reads. Values in the data can be changed and rows can appear or disappear in the data set before the end of the transaction. This option has the same effect as setting NOLOCK on all tables in all SELECT statements in a transaction. This is the least restrictive of the isolation levels.

* SERIALIZABLE

Specifies the following:

Statements cannot read data that has been modified but not yet committed by other transactions.

No other transactions can modify data that has been read by the current transaction until the current transaction

completes.

Other transactions cannot insert new rows with key values that would fall in the range of keys read by any statements in the current transaction until the current transaction completes.

Range locks are placed in the range of key values that match the search conditions of each statement executed in a transaction. This blocks other transactions from updating or inserting any rows that would qualify for any of the statements executed by the current transaction. This means that if any of the statements in a transaction are executed a second time, they will read the same set of rows. The range locks are held until the transaction completes. This is the most restrictive of the isolation levels because it locks entire ranges of keys and holds the locks until the transaction completes. Because concurrency is lower, use this option only when necessary.

Reference: SET TRANSACTION ISOLATION LEVEL (Transact-SQL)

Question: 51

You have database objects that were created by using the following script:

```
CREATE TABLE dbo.Customer
(
    CustomerID int IDENTITY(1,1),
    FirstName nvarchar(50) NOT NULL,
    LastName nvarchar(50) NOT NULL,
    CreationDate datetime NOT NULL,
    CONSTRAINT PK_Customer
        PRIMARY KEY (CustomerID)
);
GO
CREATE NONCLUSTERED INDEX IX_Customers_CreationDate
ON dbo.Customer (CreationDate)
INCLUDE (LastName, FirstName)
WHERE CreationDate >= '1/1/2008';
GO
CREATE PROCEDURE dbo.usp_GetCustomersByDate
    @CreationDate datetime
AS
SELECT LastName,
    FirstName,
    CreationDate
FROM dbo.Customer
WHERE CreationDate > @CreationDate;
GO
```

The dbo.Customers table has 1 million rows.

You discover that usp_GetCustomersByDate takes a long time to complete.

The query plan used by the stored procedure is shown in the exhibit. (Click the Exhibit button.)

Select Distinct myData From dbo.Customer Where CustomerCreateDate >= @Startdate;



You need to ensure that usp_GetCustomersByDate completes as quickly as possible.

What should you do?

- Modify the stored procedure to include the OPTIMIZE FOR UNKNOWN query hint.
- Execute the sp_recompile 'dbo.GetCustomersByDate' statement.
- Execute the ALTER INDEXIX_Customers_CreationDate WITH REBUILD statement.
- Modify the stored procedure to include the OPTIMIZE FOR('1/1/2008') query hint.

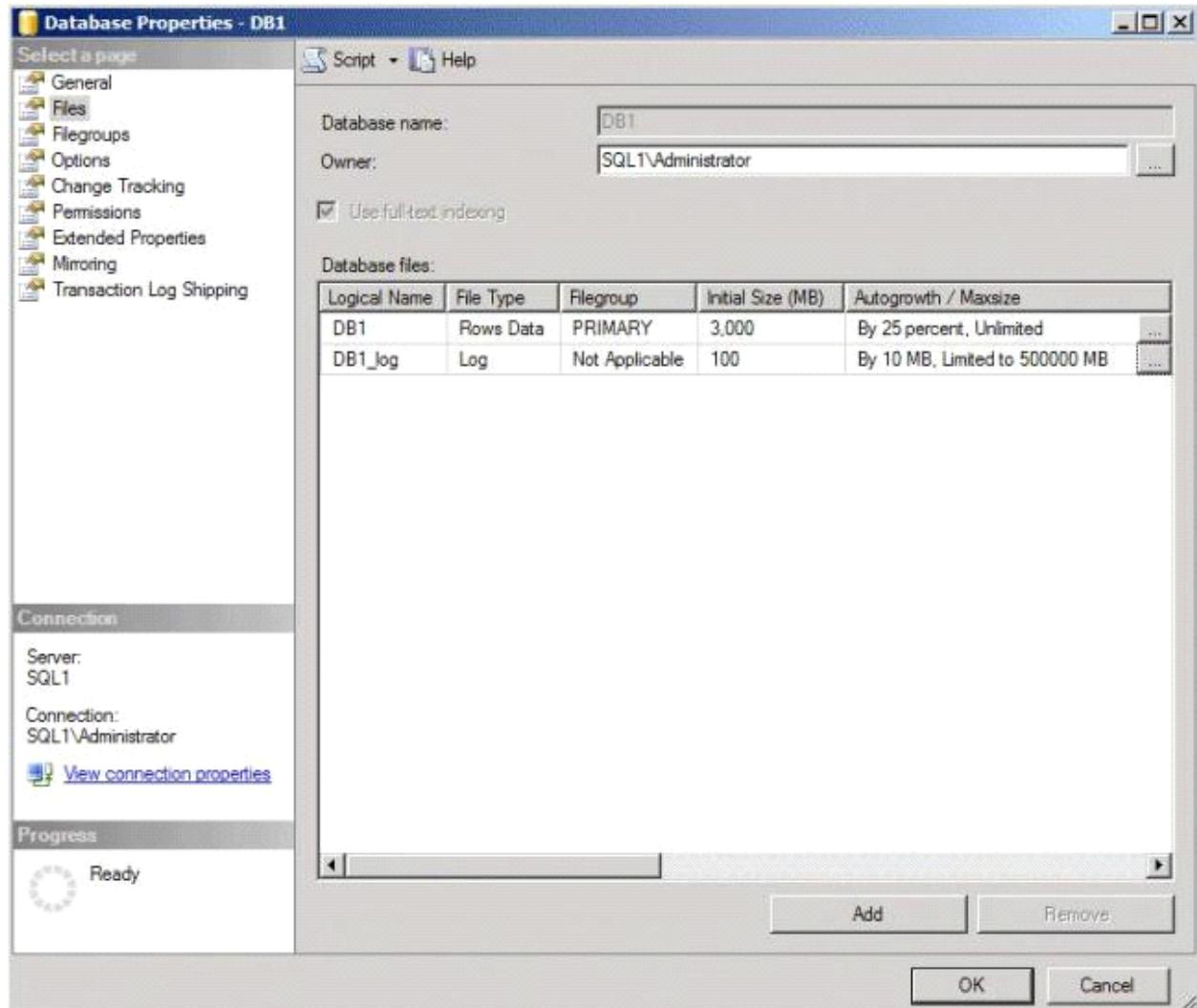
Answer: A

Question: 52

You have a database named DB1.

You plan to configure change data capture on the existing tables in DB1.

The database file settings for the DB1 are shown in the exhibit. (Click the Exhibit button.)



You need to minimize disk contention caused by change data capture.

What should you do?

More than one answer choice may achieve the goal. Select the BEST answer.

- A. Increase the autogrowth value of the database file.
- B. Set the database recovery model to simple.
- C. Increase the autogrowth value of the log file.
- D. Configure change data capture to use to a secondary filegroup.

Answer: D

Question: 53

You have a SQL Server 2012 database named DB1 that is accessed by 650 concurrent users.

You need to log all of the queries to DB1 that become deadlocked. The solution must meet the following requirements:

Provide a representation of the deadlock in XML format.

Minimize the impact on the server.

What should you create?

- A. A SQL Server Profiler trace
- B. A SQL Server Agent job that retrieves information from the sys.dm_tran_session_transactions dynamic management views
- C. A SQL Server Agent job that retrieves information from the sys.dm_tran_active_transactions dynamic management views
- D. A script that enables trace flags

Answer: A

Question: 54

You have a database named Database1.

You execute the following code:

```
CREATE TABLE dbo.table1
(
    ID int IDENTITY(1,1) NOT NULL PRIMARY KEY,
    FirstName varchar(50) NOT NULL,
    LastName varchar(50) NOT NULL,
    EmailAddress varchar(200) NULL,
    Notes nvarchar(MAX) NULL,
    LastContactDate datetime NULL
)
```

You have the following query. (Line numbers are included for reference only.)

```
01 SELECT FirstName + ' ' + LastName AS Name
02 FROM dbo.table1
03 WHERE Notes LIKE '% call%' AND
04 LastContactDate >= '1/1/2010'
```

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

You create a full-text index for the Notes column.

You need to recommend changes to the query to reduce the amount of time it takes for the query to complete.

Which code segment should you use to replace line 03?

- A. WHERE FREETEXT(notes, '%call%') AND
B. INNER JOIN FREETEXTTABLE(dbo.table1, notes, 'call')
AS t2 ON dbo.table1.ID = t2.key WHERE
- C. WHERE CONTAINS(notes, 'call*') AND
- D. WHERE CONTAINS(notes, *%call%> AND

Answer: A

Question: 55

You have a SQL Server 2012 instance.

You plan to create an application that uses spatial data.

You need to create an object that will support the representation of the surface area of all the oceans.

Which code segment should you use?

- A. `DECLARE @g GEOGRAPHY =
GEOGRAPHY::STGeomFromText(
'FULLGLOBE',4326
) ;`
- B. `DECLARE @g GEOGRAPHY =
GEOGRAPHY::STGeomFromText(
'POLYGON(0 0, 0 10, 10 10, 10 0, 0 0)',4326
) ;`
- C. `DECLARE @g GEOGRAPHY =
GEOGRAPHY::STGeomFromText(
'COMPOUNDCURVE(
CIRCULARSTRING(0 -50, 90 0, 0 50),
CIRCULARSTRING(0 50, 45 50, -90 50),
CIRCULARSTRING(-90 50, 0 0, -90 -50),
CIRCULARSTRING(-90 -50, 45 -50, 0 -50),4326
)'
) ;`
- D. `DECLARE @g GEOGRAPHY =
GEOGRAPHY::STGeomFromText(
'CIRCULARSTRING(0 50, 90 50, 180 50)',4326
) ;`

A. Option A

B. Option B

C. Option C

D. Option D

Answer: A

Question: 56

You use SQL Server 2012 to store data used by an e-commerce application.

You develop a stored procedure named sp1. Sp1 is used to read the price of all the products sold on the e-commerce site.

You need to ensure that sp1 can read data even while another transaction is modifying the price of a product. Sp1 must only read committed data.

Which transaction isolation level should you use in sp1?

- A. Serializable
- B. Snapshot
- C. Repeatable read
- D. Read committed

Answer: B

Question: 57

DRAG DROP

You have a database named database1. Each table in database1 has one index per column.

Users often report that creating items takes a long time.

You need to perform the following maintenance tasks:

Identify unused indexes.

Identify indexes that need to be defragmented.

What should you use?

To answer, drag the appropriate function to the correct management task in the answer area. (Answer choices may be used once, more than once, or not at all.)

Functions	Answer Area
sys.dm_db_index_usage_stats	Identify unused indexes.
sys.dm_db_index_operational_stats	Identify indexes that need to be defragmented.
sys.dm_db_index_physical_stats	
sys.dm_db_missing_index_columns	
sys.dm_db_missing_index_details	
sys.dm_db_missing_index_groups	

Answer:

Functions	Answer Area
	Identify unused indexes.
sys.dm_db_index_operational_stats	sys.dm_db_index_usage_stats
	Identify indexes that need to be defragmented.
sys.dm_db_missing_index_columns	sys.dm_db_index_physical_stats
sys.dm_db_missing_index_details	
sys.dm_db_missing_index_groups	

Note:

* sys.dm_db_index_usage_stats

Returns counts of different types of index operations and the time each type of operation was last performed.

* sys.dm_db_index_physical_stats

Returns size and fragmentation information for the data and indexes of the specified table or view.

Question: 58

DRAG DROP

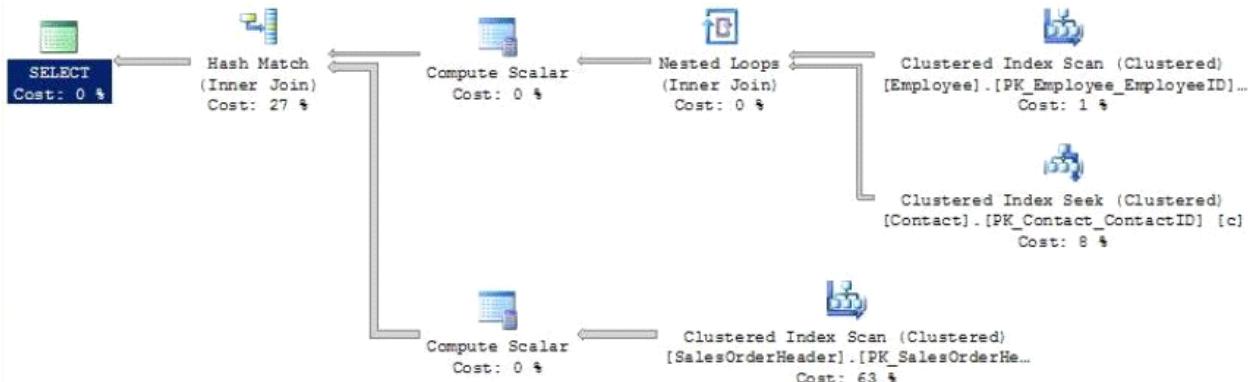
You have a database that contains three tables. The tables are configured as shown in the following table.

Table	Primary key index
SalesOrderHeader	PK_SalesOrderHeader_SalesOrderID
Employee	PK_Employee_EmployeeID
Contact	PK_Contact_ContactID

You have the following query:

```
SELECT soh.SalesPersonID,
       c.FirstName + ' ' + c.LastName AS FullName,
       c.EmailAddress,
       e.Title,
       soh.SubTotal,
       YEAR(soh.OrderDate) AS Year
  FROM SalesOrderHeader soh
 INNER JOIN Employee e
   ON soh.SalesPersonID = e.EmployeeID
 INNER JOIN Contact c
   ON e.ContactID = c.ContactID
 WHERE soh.OrderDate >= '1/1/2012'
```

The execution plan for the query is shown in the exhibit. (Click the Exhibit button.)



You need to create one index to minimize the amount of time it takes to execute the query.

What should you do?

To answer, drag the appropriate columns to the correct locations in the answer area. (Answer choices may be used once, more than once, or not at all.)

Columns	Answer Area
Contact.EmailAddress	Indexed Columns Column
Contact.FirstName	Included Columns Column
Contact.LastName	
Employee.Title	
SalesOrderHeader.OrderDate	
SalesOrderHeader.SalesPersonID	
SalesOrderHeader.SubTotal	

Answer:

Box 1:

Contact.EmailAddress

Box 2:

Contact.FirstName

Box 3:

Contact.LastName

Note:

Covering index: A type of index that includes all the columns that are needed to process a particular query. For example, your query might retrieve the FirstName and LastName columns from a table, based on a value in the ContactID column. You can create a covering index that includes all three columns.

Question: 59

You have a SQL Server 2012 instance that hosts a single-user database.

The database does not contain user-created stored procedures or user-created functions.

You need to minimize the amount of memory used for query plan caching.

Which advanced server option should you modify?

- A. Scan for Startup Procs
- B. Enable Contained Databases
- C. Optimize for Ad hoc Workloads
- D. Allow Triggers to Fire Others

Answer: C

Question: 60

You execute the following code.

```
CREATE TABLE HumanResources.Employees
(
    EmployeeID int IDENTITY(1,1) PRIMARY KEY,
    ContactID int NOT NULL
        FOREIGN KEY REFERENCES Person.Contact(ContactID),
    JobTitle varchar(100)
);
GO
```

```
CREATE INDEX IX_Employees
ON HumanResources.Employee(JobTitle);
GO
```

After populating the Employees table with 10,000 rows, you execute the following query:

```
SELECT EmployeeID, JobTitle
FROM HumanResources.Employee
WHERE SUBSTRING(JobTitle,1,1) = 'C'
```

You need to reduce the amount of time it takes to execute the query.

What should you do?

- A. Partition the table and use the JobTitle column for the partition scheme.
- B. Change SUBSTRING(JobTitle,1,1) = 'C' to JobTitle LIKE 'c%'
- C. Change SUBSTRING (JobTitle, 1,1] = 'c' to LEFT(JobTitle ,1) = 'c'.
- D. Replace IX_Employees with a clustered index.

Answer: B

Question: 61

You have a server that has SQL Server 2012 installed.

You need to identify which parallel execution plans are running in serial.

Which tool should you use?

- A. Performance Monitor
- B. Database Engine Tuning Advisor
- C. Data Profile Viewer
- D. Extended Events

Answer: D

Explanation:

<http://msdn.microsoft.com/en-us/library/bb677278.aspx>
<http://msdn.microsoft.com/en-us/library/bb630282.aspx>
<http://www.sql-server-performance.com/2006/query-execution-plan-analysis/>
<http://www.simple-talk.com/sql/learn-sql-server/understanding-and-using-parallelism-in-sqlserver/>
<http://www.sqlservercentral.com/articles/SQL+Server+2012/At+last%2c+execution+plans+show+true+thread+reservations./92458/>
http://sqlblog.com/blogs/paul_white/archive/2011/12/23/forcing-a-parallel-query-executionplan.aspx
http://sqlblog.com/blogs/paul_white/archive/2012/05/02/parallel-row-goals-gone-rogue.aspx

<http://msdn.microsoft.com/en-us/library/bb895310.aspx>
<http://msdn.microsoft.com/en-us/library/bb895313.aspx>
<http://msdn.microsoft.com/en-us/library/hh231122.aspx>

Question: 62

You have a SQL Server 2012 database named DB1. You have a backup device named Device1.
You discover that the log file for the database is full.
You need to ensure that DB1 can complete transactions. The solution must not affect the chain of log sequence numbers (LSNs).
Which code segment should you execute?

- A. BACKUP LOG DB1 TO Device1 WITH COPY_ONLY
- B. BACKUP LOG DB1 TO Device1 WITH NORECOVERY
- C. BACKUP LOG DB1 TO Device1 WITH TRUNCATE_ONLY
- D. BACKUP LOG DB1 TO Device1

Answer: D

Explanation:

<http://msdn.microsoft.com/en-us/library/ms186865.aspx>
<http://msdn.microsoft.com/en-us/library/ms179478.aspx>
<http://msdn.microsoft.com/en-us/library/ms190925.aspx>

Question: 63

DRAG DROP

You plan to create a new table that will contain a column named Salary. Salary will contain highly sensitive data.
Salary must meet the following requirements:

Contain numeric data.

Contain only encrypted data that remains encrypted in memory.

You need to identify which encryption type and data type must be used for Salary.

Which encryption type and data type should you identity?

To answer, drag the appropriate encryption type and data type to the correct identifier in the answer area.

Encryption Types		Answer Area	
Transparent data encryption (TDE)		Encryption Type	<input type="text"/>
Encrypting File System (EFS)		Data Type	<input type="text"/>
Cell-level encryption			
BitLocker Drive Encryption (BitLocker)			
Data Types			
decimal			
varchar			
varbinary			
money			

Answer:

Encryption Type	<input type="text"/> Cell-level encryption
Data Type	<input type="text"/> varbinary

Question: 64

You have a Microsoft SQL Azure database named DBAzure1.
 You create a table in DBAzure1 by using the following script:

```

CREATE TABLE dbo.Customers
(
    CustomerId int NOT NULL,
    CustomerName nvarchar(50) NULL,
    CustomerContact nvarchar(50) NULL,
    CustomerDetails nvarchar(200) NULL,
    CONSTRAINT PK_Customers PRIMARY KEY CLUSTERED (CustomerId)
)
ON [PRIMARY]
GO

```

You need to recommend a solution to ensure that each combination of CustomerContact and CustomerDetails is not duplicated.

What should you recommend creating?

- A. A CHECK constraint
- B. A filtered index
- C. A columnstore index
- D. A UNIQUE constraint

Answer: D

Question: 65

DRAG DROP

Your network contains a SQL Server 2012 instance named SQL1. SQL1 contains a database named DB1. DB1 contains three tables.

The tables are configured as shown in the following table.

Table name	Configuration
Table1	<ul style="list-style-type: none"> • Table data will not be updated. • The table will contain historical calculations. • The table will contain 10 million records.
Table2	<ul style="list-style-type: none"> • 20% of the table data will be updated weekly. • The table will contain 25 million records.
Table3	<ul style="list-style-type: none"> • 40% of the table data will be updated weekly. • The table will contain 1 million records.

You plan to create indexes for the tables.

You need to identify which type of index must be created for each table. The solution must minimize the amount of time required to return information from the tables.

Which type of index should you create for each table? To answer, drag the appropriate index type to the correct table in the answer area.

Index Types	Answer Area
Columnstore Index	Table1
Nonclustered Index	Table2
	Table3

Answer:

Table1	Columnstore Index
Table2	Nonclustered Index
Table3	Nonclustered Index

Question: 66

DRAG DROP

You create a table that contains the following script:

```
CREATE TABLE dbo.Customers
(
    id int IDENTITY(1,1) NOT NULL,
    FirstName nvarchar(50) NULL,
    LastName nvarchar(50) NULL,
    EmployeeID int NULL,
        CONSTRAINT PK_Customers PRIMARY KEY CLUSTERED (id)
) ON [PRIMARY]
GO
```

You need to prevent duplicate values in the EmployeeID field.

Which five code segments should you use?

To answer, move the appropriate code segments from the list of code segments to the answer area and arrange them in the correct order.

Answer Area
ADD CONSTRAINT
CHECK
ALTER TABLE dbo.Customers
CK_EmployeeID
UNIQUE
FOREIGN KEY
(EmployeeID)

Answer:

Box 1:

ALTER TABLE dbo.Customers

Box 2:

ADD CONSTRAINT

Box 3:

CK_EmployeeID

Box 4:

UNIQUE

Box 5:

(EmployeeID)

Question: 67

You plan to create a new table that has the following requirements:

Uses a GUID data type as the primary key.

Uses a clustered index as the primary key.

Minimizes fragmentation.

You need to recommend which option to include in the CREATE statement.

Which option should you include?

More than one answer choice may achieve the goal. Select the BEST answer.

- A. NEWID
- B. @@IDENTITY
- C. NEWSEQUENTIALID
- D. IDENTITY

Answer: C

Question: 68

You have a SQL Server 2012 instance named SQL\Instance1. Instance1 contains a database named Database1. You need to recommend an index defragmentation solution for an index named ContentIndex. ContentIndex must meet the following requirements:

- Remain online during the defragmentation.
- Update distribution statistics.
- Perform defragmentation as quickly as possible.

Which type of index defragmentation solution should you include in the recommendation? More than one answer choice may achieve the goal. Select the BEST answer.

- A. DBCC DBREINDEX
- B. REORGANIZE
- C. REBUILD
- D. DBCC INDEXDEFrag

Answer: B

Question: 69

You have a SQL Azure database. You need to identify which keyword must be used to create a view that will be indexed. Which keyword should you identify?

- A. SCHEMABINDING
- B. VIEW_METADATA
- C. DISTINCT
- D. DEFAULT

Answer: A

Explanation:
<http://msdn.microsoft.com/en-us/library/ms187956.aspx>
<http://msdn.microsoft.com/en-us/library/ms191432.aspx>

Question: 70

You have a SQL Server 2012 database named Database1. Database1 contains a table named OrderDetails. For a given sales order, you need to retrieve the OrderID, Quantity, and LineTotal columns for all of the items in the OrderDetails table. The solution must ensure that the results can be joined to other tables. Which code segment should you execute?

- C A. CREATE FUNCTION dbo.GetOrderDetails(@OrderID int)
 RETURNS TABLE
 AS
 RETURN
 (SELECT OrderID, Quantity, LineTotal
 FROM Sales.OrderDetails
 WHERE OrderID = @OrderID);
- C B. CREATE PROC dbo.GetOrderDetails(@OrderID int)
 AS
 SELECT OrderID, Quantity, LineTotal
 FROM Sales.OrderDetails
 WHERE OrderID = @OrderID;
- C C. CREATE FUNCTION dbo.GetOrderDetails(@OrderID int)
 RETURNS @retOrderDetails TABLE
 (
 OrderID int NOT NULL,
 Quantity int NOT NULL,
 LineTotal decimal NULL
)
 AS
 BEGIN
 INSERT @retOrderDetails
 SELECT OrderID, Quantity, LineTotal
 FROM Sales.OrderDetails
 ORDER BY @OrderID;
 RETURN;
 END;
- C D. CREATE VIEW dbo.GetOrderDetails
 AS
 SELECT OrderID, Quantity, LineTotal
 FROM Sales.OrderDetails;

- A. Option A
 B. Option B
 C. Option C
 D. Option D

Answer: A

Question: 71

You create a view by using the following code:

```
CREATE VIEW dbo.View1
WITH VIEW_METADATA
AS
SELECT t1.col1, t1.col2, t2.*
FROM dbo.Table1 AS t1 JOIN dbo.Table2 AS t2 ON t1.col1=t2.col2;
```

Several months after you create the view, users report that the view has started to return unexpected results. You discover that the design of Table2 was modified since you created the view.

You need to ensure that the view returns the correct results.

Which code segment should you run?

- A.

```
EXEC sp_refreshsqlmodule @name = 'dbo.Table2';
```
- B.

```
DROP dbo.View1;
GO
CREATE dbo.View1 WITH SCHEMABINDING, VIEW_METADATA
AS
SELECT t1.col1, t1.col2, t2.*
FROM dbo.Table1 AS t1 JOIN dbo.Table2 AS t2
ON t1.col1=t2.col2;
```
- C.

```
ALTER dbo.View1 WITH SCHEMABINDING, VIEW_METADATA
AS
SELECT t1.col1, t1.col2, t2.*
FROM dbo.Table1 AS t1 JOIN dbo.Table2 AS t2
ON t1.col1=t2.col2;
```
- D.

```
EXEC sp_refreshview @viewname = 'dbo.View1';
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: D

Question: 72

DRAG DROP

You are planning two stored procedures named SProc1 and SProc2.

You identify the following requirements:

SProc1 must return a table.

SProc2 must return a scalar value.

You need to identify which option must be implemented for each stored procedure to return the desired data.

Which options should you identify?

To answer, drag the appropriate option to the correct requirement in the answer area. (Answer choices may be used once, more than once, or not at all.)

Options	Answer Area	
an output parameter		
a raise error		
a SELECT statement		
a table-valued parameter (TVP)		

Answer:

Options	Answer Area
a raise error	SProc1 a SELECT statement
a table-valued parameter (TVP)	SProc2 an output parameter

Note:

- * a table (a set of rows) can be returned through a SELECT statement
- * a scalar can be returned through an output parameter.
- * incorrect: TVP is used for input not output.

Question: 73

Your company has a main office in London and a branch office in New York.

Your network contains a server named Server5 that has SQL Server 2012 installed. Server5 contains a database name ContentDB and a table named ContentTable.

You add an additional server named Server9 that runs SQL Server 2012.

You need to create a distributed partitioned view. The solution must minimize the amount of network traffic. What should you do? (Each correct answer presents part of the solution. Choose all that apply.)

- A. Create the view on Server5.
- B. Add Server9 as a linked server.
- C. Create the view on Server9.
- D. Add the Customers table to Server9.
- E. Add Server9 as a Distributor.
- F. Remove the Customers table from Server5.

Answer: A, B, C, D

Question: 74

You need to encapsulate a T-SQL script into a reusable user-defined object.

The object must meet the following requirements:

- Permit insertions into a table variable.
- Support structured exception handling.
- Prevent changes to the definition of referenced objects.
- Support the use of the APPLY operator on the output of the object.

Which type of object should you use?

- A. An inline table-valued function
- B. A stored procedure

- C. A scalar user-defined function
- D. A multi-statement table-valued function

Answer: C

Question: 75

You plan to migrate an instance of SQL Server 2008 to a new installation of SQL Server 2012.

You need to migrate alerts and e-mail notifications.

Which system stored procedures should you use? (Each correct answer presents part of the solution. Choose all that apply.)

- A. sp_syspolicy_create_job
- B. sp_add_operator
- C. sp_audit_write
- D. sp_add_alert

Answer: B, C

B: sp_add_operator

Creates an operator (notification recipient) for use with alerts and jobs.

C: sp_audit_write

Adds a user-defined audit event to the USER_DEFINED_AUDIT_GROUP. If USER_DEFINED_AUDIT_GROUP is not enabled, sp_audit_write is ignored.

Question: 76

You have a Microsoft SQL Azure database that contains a table named Customers.

You have a table-valued function named TopCustomers that returns a list of all the customers that have purchased items during the last 12 months. The ID of the customer is passed as an argument to the TopCustomers function.

You need to create a query that returns a list of all the Customer names and the purchase dates.

The solution must return only customers that have purchased an item during the last 12 months.

What should you add to the query?

- A. OUTER JOIN
- B. CROSS JOIN
- C. CROSS APPLY
- D. OUTER APPLY

Answer: C

Question: 77

You have a SQL Server 2012 database named Database1.

You execute the following code:

```
CREATE TABLE Sales
(
    ID int IDENTITY(1,1) NOT NULL PRIMARY KEY,
    OrderDate char(10) NOT NULL,
    Amount decimal
);
GO

CREATE INDEX IX_Sales_OrderDate
    ON Sales(OrderDate)
    INCLUDE (ID, Amount);
GO

CREATE PROC usp_Proc1(
    @date1 datetime,
    @date2 datetime
)
AS
SELECT ID, OrderDate, Amount
    FROM Sales
    WHERE CAST(OrderDate AS datetime)
        BETWEEN @date1 AND @date2
    ORDER BY ID;
GO
```

You insert 3 million rows into Sales.

You need to reduce the amount of time it takes to execute Proc1.

What should you do?

C A. Change the query inside Proc1 to:

```
SELECT ID, OrderDate, Amount  
FROM Sales  
WHERE OrderDate BETWEEN CONVERT(char(10), @date1, 112)  
AND CONVERT(char(10), @date2, 112)  
ORDER BY ID;
```

C B. Change the definition of Proc1 to:

```
CREATE PROC usp_Proc1(  
@date1 int, @date2 int  
)
```

C C. Change the query inside Proc1 to:

```
SELECT ID, OrderDate, Amount  
FROM Sales  
WHERE CAST(OrderDate AS datetime) < @date1  
AND CAST(OrderDate AS datetime) > @date2  
ORDER BY ID;
```

C D. Change the definition of Proc1 to:

```
CREATE PROC usp_Proc1(  
@date1 date, @date2 date  
)
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: A

Question: 78

DRAG DROP

You have a SQL Server 2012 database named Database1. Database1 has a data file named database1_data.mdf and a transaction log file named database1_Log.ldf. Database1_Data.mdf is 1.5 GB.

Database1_Log.ldf is 1.5 terabytes. A full backup of Database1 is performed every day.

You need to reduce the size of the log file. The solution must ensure that you can perform transaction log backups in the future. Which code segment should you execute? To answer, move the appropriate code segments from the list of code segments to the answer area and arrange them in the correct order.

Ordered List Title	Answer Choices Title
	<pre data-bbox="817 280 1214 1088"> ALTER DATABASE Database1 SET RECOVERY FULL ; ALTER DATABASE Database1 SET RECOVERY SIMPLE ; DBCC SHRINKFILE (Database1_Log); DBCC SHRINKFILE (Database1_Log, TRUNCATEONLY), BACKUP LOG Database1 TO Database1_Log_Backup; BACKUP LOG Database1 TO Database1_Log_Backup WITH NO_TRUNCATE; USE Database1; USE master; </pre>

Answer:

Box 1:

```
USE Database1;
```

Box 2:

```

ALTER DATABASE
Database1 SET
RECOVERY SIMPLE ;
```

Box 3:

```
DBCC SHRINKFILE
(Database1_Log);
```

Box 4:

```

ALTER DATABASE
Database1 SET
RECOVERY FULL ;
```

Shrinking a log file to a specified target size

The following example shrinks the log file in the AdventureWorks database to 1 MB. To allow the DBCC SHRINKFILE command to shrink the file, the file is first truncated by setting the database recovery model to SIMPLE.

Transact-SQL

```
USE AdventureWorks2012;
```

```
GO
```

```
-- Truncate the log by changing the database recovery model to SIMPLE.
```

```
ALTER DATABASE AdventureWorks2012
```

```
SET RECOVERY SIMPLE;
```

```

GO
-- Shrink the truncated log file to 1 MB.
DBCC SHRINKFILE (AdventureWorks2012_Log, 1);
GO
-- Reset the database recovery model.
ALTER DATABASE AdventureWorks2012
SET RECOVERY FULL;
GO
http://msdn.microsoft.com/en-gb/library/ms189493.aspx

```

Question: 79

DRAG DROP

You plan to deploy SQL Server 2012. You must create two tables named Table1 and Table2 that will have the following specifications:

Table1 will contain a date column named Column1 that will contain a null value approximately 80 percent of the time.

Table2 will contain a column named Column2 that is the product of two other columns in Table2.

Both Table1 and Table2 will contain more than 1 million rows.

You need to recommend which options must be defined for the columns. The solution must minimize the storage requirements for the tables. Which options should you recommend? To answer, drag the appropriate options to the correct column in the answer area.

Options	Answer Area
<input type="checkbox"/> Sparse	Column1 <input type="checkbox"/> Option
<input type="checkbox"/> Computed	Column2 <input type="checkbox"/> Option
<input type="checkbox"/> Persisted computed	

Answer:

Options	Answer Area
<input type="checkbox"/> Sparse	Column1 <input type="checkbox"/> Sparse
<input type="checkbox"/> Computed	Column2 <input type="checkbox"/> Computed
<input type="checkbox"/> Persisted computed	

Explanation:

<http://msdn.microsoft.com/en-us/library/cc280604.aspx>
<http://msdn.microsoft.com/en-us/library/ms186241.aspx>

Question: 80

DRAG DROP

You are designing a database for a university. The database will contain two tables named

Classes and StudentGrades that have the following specifications:

Classes will store brochures in the XPS format.

The brochures must be structured in folders and must be accessible by using UNC paths.

StudentGrades must be backed up on a separate schedule than the rest of the database.

You need to identify which SQL Server technology meets the specifications of each table. Which technologies should you identify? To answer, drag the appropriate technology to the correct table in the answer area.

Technologies	Answer Area	
FileStream	Technology	Classes
FileTable	Technology	StudentGrades
Filegroup		
Partitioned views		
Answer:		
Technologies	Answer Area	
FileStream	FileTable	Classes
FileTable	Filegroup	StudentGrades
Filegroup		
Partitioned views		

Explanation:

<http://msdn.microsoft.com/en-us/library/gg471497.aspx>

<http://msdn.microsoft.com/en-us/library/ff929144.aspx>

<http://msdn.microsoft.com/en-us/library/ms189563.aspx>

<http://msdn.microsoft.com/en-us/library/ms190174.aspx>

<http://msdn.microsoft.com/en-us/library/ms187956.aspx>

Question: 81**DRAG DROP**

You have a SQL Azure database named Database1. You need to design the schema for a table named Table1.

Table1 will have less than one million rows. Table1 will contain the following information for each row:

Column	Description
ID	An incremental numeric value used to identify the row
Name	A string in English
Code	An alphanumeric code that has five characters
ModifiedDate	The date of the last modification

The solution must minimize the amount of space used to store each row. Which data types should you recommend for each column? To answer, drag the appropriate data type to the correct column in the answer area.

Data Types

Answer Area

int

ID

Data type

bigint

Name

Data type

varchar

Code

Data type

nvarchar

ModifiedDate

Data type

char

smalldatetime

date

Answer:

Data Types	Answer Area
int	ID int
bigint	Name varchar
varchar	Code char
nvarchar	ModifiedDate date
char	
smalldatetime	
date	

Explanation:

<http://msdn.microsoft.com/en-US/library/ms187752.aspx>

Question: 82

Your network contains a server named SQL1 that has SQL Server 2012 installed. SQL1 contains a database name DB1 and a table named Customers.

You add an additional server named SQL2 that runs SQL Server 2012.

You need to create a distributed partitioned view. The solution must minimize the amount of network traffic. What should you do? (Each correct answer presents part of the solution. Choose all that apply.)

- A. Add SQL2 as a Distributor.
- B. Add the Customers table to SQL2.
- C. Add SQL2 as a linked server.
- D. Create the view on SQL1.
- E. Remove the Customers table from SQL1.
- F. Create the view on SQL2.

Answer: B, C, D, F

Question: 83

You have a table named Table1 that stores customer data.

Each customer has a credit limit that can only be discovered by querying multiple tables.

You need to ensure that the value of the credit limit is returned by executing a query on Table1.

What should you create?

- A. A trigger that uses a ranking function

- B. A trigger that uses a table-valued function
- C. A calculated column that uses a table-valued function
- D. A calculated column that uses a scalar function

Answer: C

Question: 84

You are creating a stored procedure named usp1. Usp1 will create a table that will be used during the execution of usp1. Only usp1 will be allowed to access the table.

You need to write the code required to create the table for usp1. The solution must minimize the need to recompile the stored procedure.

Which code segment should you use to create the table?

- A. CREATE TABLE oneTable
- B. CREATE TABLE ##oneTable
- C. CREATE TABLE #oneTable
- D. DECLARE oneTable TABLE

Answer: B

Question: 85

You plan to modify a stored procedure to use temporary data.

The stored procedure must meet the following requirements:

Favor physical memory when physical memory is available.

Be able to roll back changes to the temporary data.

You need to recommend which object to add to the stored procedure.

Which T-SQL command should you recommend?

- A. CREATE TABLE ##Table...
- B. CREATE TABLE Table...
- C. CREATE VIEW Table...
- D. CREATE PARTITION SCHEME Table...
- E. DECLARE TABLE @ Table...

Answer: A

Temporary Tables

You can create local and global temporary tables. Local temporary tables are visible only in the current session, and global temporary tables are visible to all sessions. Temporary tables cannot be partitioned.

Prefix local temporary table names with single number sign (#table_name), and prefix global temporary table names with a double number sign (##table_name)

Question: 86

DRAG DROP

Your network contains a server named Server1 that runs SQL Server 2012. Server1 contains an instance named Instance1. Instance1 contains a database named ContentDatabase.

ContentDatabase uses transaction log backups.

The recovery model of ContentDatabase is set to FULL.

You need to shrink the ContentDatabase_Log log file to 10 MB. The solution must ensure that you can continue to back up the transaction log.

Which three code segments should you execute?

To answer, move the appropriate code segments from the list of code segments to the answer area and arrange them in the correct order.

Answer Area
DBCC SHRINKFILE (ContentDatabase_Log, 10); GO
ALTER DATABASE ContentDatabase SET RECOVERY SIMPLE; GO
ALTER DATABASE ContentDatabase SET RECOVERY FULL; GO
ALTER DATABASE ContentDatabase SET PAGE_VERIFY CHECKSUM; GO
BACKUP LOG ContentDatabase WITH TRUNCATE_ONLY
DBCC SHRINKFILE (ContentDatabase_Log, 7168); GO

Answer:

Explanation:

Box 1:

```
ALTER DATABASE ContentDatabase
SET RECOVERY SIMPLE;
GO
```

Box 2:

```
DBCC SHRINKFILE (ContentDatabase_Log, 10);
GO
```

Box 3:

```
ALTER DATABASE ContentDatabase
SET RECOVERY FULL;
GO
```

Note:

* Shrinking a log file to a specified target size

The following example shrinks the log file in the AdventureWorks database to 1 MB. To allow the DBCC SHRINKFILE command to shrink the file, the file is first truncated by setting the database recovery model to SIMPLE.

Transact-SQL

```
USE AdventureWorks2012;
GO
-- Truncate the log by changing the database recovery model to SIMPLE.
ALTER DATABASE AdventureWorks2012
SET RECOVERY SIMPLE;
GO
```

```
-- Shrink the truncated log file to 1 MB.  
DBCC SHRINKFILE (AdventureWorks2012_Log, 1);  
GO  
-- Reset the database recovery model.  
ALTER DATABASE AdventureWorks2012  
SET RECOVERY FULL;  
GO
```

* If the log file does not shrink (after dbcc shrinkfile)

Typically it is the log file that appears not to shrink. This is usually the result of a log file that has not been truncated. You can truncate the log by setting the database recovery model to SIMPLE, or by backing up the log and then running the DBCC SHRINKFILE operation again.

* DBCC SHRINKFILE shrinks the size of the specified data or log file for the current database, or empties a file by moving the data from the specified file to other files in the same filegroup, allowing the file to be removed from the database.

Arguments include:

target_size

Is the size for the file in megabytes, expressed as an integer.

Question: 87

You have a Microsoft SQL Azure database that contains a table named Employees.

```
CREATE TABLE HR.Employees  
(  
    id int primary key,  
    name varchar(50)  
)
```

You create a non-clustered index named EmployeeName on the name column.

```
SELECT * FROM HR.Employees  
WHERE 'JOH' = LEFT(name, 3)
```

You write the following query to retrieve all of the employees that have a name that starts with the letters JOH:

You discover that the query performs a table scan.

You need to ensure that the query uses EmployeeName.

What should you do?

- A. Recreate EmployeeName as a unique index
- B. Recreate EmployeeName as a clustered index
- C. Replace LEFT(name,3) = 'JOH' by using name like 'JOH%'
- D. Replace LEFT(name,3) = 'JOH' by using substring(name, 1, 3) = 'JOH'

Answer: C

Question: 88

You have a SQL Server 2012 database that contains a table named Users. The Users table contains usernames and passwords.

You need to ensure that all new records have a password.

Which code segment should you use?

More than one answer choice may achieve the goal. Select the BEST answer.

- C A. ALTER TABLE dbo.Users
DROP Password;
GO
ALTER TABLE dbo.Users
ADD Password varchar(30) NOT NULL;
GO
- C B. ALTER TABLE dbo.Users
ADD CONSTRAINT CK_Users_Password
CHECK (Password IS NULL);
GO
- C C. DROP TABLE dbo.Users;
GO
CREATE TABLE dbo.Users (
CustID int PRIMARY KEY,
Name varchar(30),
Password varchar(30),
CONSTRAINT CK_Users_Password
CHECK (Password IS NOT NULL));
GO
- C D. ALTER TABLE dbo.Users
ADD CONSTRAINT CK_Users_Password
CHECK (Password IS NOT NULL);
GO

- A. Option A
B. Option B
C. Option C
D. Option D

Answer: D

Question: 89

You have a SQL Server 2012 environment that contains two servers. The servers are configured as shown in the following table.

Server name	Database	Type
Server1	DB1	Principal
Server2	DB1	Mirror

After the failover is complete, a user receives the following error message when connecting to DB1 on Server2:

"Msg 916, Level 14, State 1, Line 1

The server principal "Account1" is not able to access the database "DB1" under the current security context."

You verify that there is a server login for Account1 on Server2.

You need to ensure that Account1 can connect to DB1.

What should you do?

More than one answer choice may achieve the goal. Select the BEST answer.

- A. Update the SID for Account1 on DB1.
- B. Add Account1 to the db_datareader role.
- C. Create a new database user on DB1.
- D. Implement Windows authentication.

Answer: B

Question: 90

DRAG DROP

You have two existing tables, one named COUNTRY and the other named STATES.

The tables are defined as follows:

```
CREATE TABLE COUNTRY
(
Country_Abbr CHAR(3) PRIMARY KEY CLUSTERED,
Country_Description VARCHAR(30) Not Null
)
CREATE TABLE STATES
(
State_Abbr CHAR(2) PRIMARY KEY CLUSTERED,
State_Description VARCHAR(30) Not Null,
Country_Abbr CHAR(3) Not Null
)
```

You need to set up a rule that every STATE.Country_Abbr must match an existing record in the COUNTRY table.

Develop the solution by selecting and arranging the required code blocks in the correct order.

You may not need all of the code blocks.

Code Blocks	Answer Area
REFERENCES STATES (Country_Abbr)	
REFERENCES COUNTRY (Country_Abbr)	
ON STATES	
FOREIGN KEY (Country_Abbr)	
ON COUNTRY	
ADD CONSTRAINT FK_StateCountry	
ON COUNTRY_ABBR	
ALTER TABLE COUNTRY	
ADD FOREIGN KEY FK_StateCountry	
ALTER TABLE STATES	

Answer:

Box 1:

ALTER TABLE STATES

Box 2:

ADD CONSTRAINT FK_StateCountry

Box 3:

FOREIGN KEY (Country_Abbr)

Box 4:

REFERENCES COUNTRY (Country_Abbr)

Note:

To allow naming of a FOREIGN KEY constraint, and for defining a FOREIGN KEY constraint on multiple columns, use the following SQL syntax:

MySQL / SQL Server / Oracle / MS Access:

ALTER	TABLE	Orders
ADD	CONSTRAINT	fk_PerOrders
FOREIGN	KEY	(P_Id)
REFERENCES Persons(P_Id)		

Question: 91

DRAG DROP

You plan to deploy two stored procedures name USP_1 and USP_2 that read data from a database.

Your company identifies the following requirements for each stored procedure:

USP_1 cannot allow dirty reads.

USP_2 must place range locks on the data to ensure read consistency.

You need to identify which isolation level you must set for each stored procedure. The solution must minimize the number of locks.

Which isolation level should you identify?

To answer, drag the appropriate isolation level to the correct stored procedure in the answer area. (Answer choices may be used once, more than once, or not at all.)

Isolation Levels	Answer Area
read committed	USP_1 Isolation level
read uncommitted	USP_2 Isolation level
repeatable read	
serializable	
snapshot	

Answer:

USP_1	read committed
USP_2	serializable

* read committed

READ COMMITTED

Specifies that shared locks are held while the data is being read to avoid dirty reads, but the data can be changed before the end of the transaction, resulting in nonrepeatable reads or phantom data. This option is the SQL Server default.

* SERIALIZABLE

Places a range lock on the data set, preventing other users from updating or inserting rows into the data set until the transaction is complete. This is the most restrictive of the four isolation levels. Because concurrency is lower, use this option only when necessary. This option has the same effect as setting HOLDLOCK on all tables in all SELECT statements in a transaction.

Question: 92

You use SQL Server 2014 to maintain the data used by applications at your company.

You need to run two separate SQL statements.

You must guarantee that the following three things happen:

1. Either BOTH statements succeed or BOTH statements fail as a batch.
2. If an error occurs on the first statement, SQL should not attempt to run the second statement.
3. Error information should be returned to the client.

What should you do?

- A. SET XACT_ABORT ON
BEGIN TRY
BEGIN TRANSACTION
...Statement 1
...Statement 2
COMMIT TRANSACTION
END TRY
BEGIN CATCH
ROLLBACK TRANSACTION
END CATCH
- B. SET XACT_ABORT OFF
BEGIN TRY
...Statement 1
END TRY
BEGIN TRY
...Statement 2
END TRY
BEGIN CATCH
THROW
END CATCH
- C. SET XACT_ABORT ON
BEGIN TRANSACTION
...Statement 1
...Statement 2
If @@ERROR <> 0
ROLLBACK
ELSE
COMMIT TRANSACTION
- D. SET XACT_ABORT ON
BEGIN TRY
...Statement 1
if @@ERROR <> 0
GOTO CATCH
...Statement 2
if @@ERROR <> 0
GOTO CATCH
END TRY
BEGIN CATCH
THROW
END CATCH

- A. Option A
B. Option B
C. Option C
D. Option D

Answer: A

* SET XACT_ABORT

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

When SET XACT_ABORT is OFF, in some cases only the Transact-SQL statement that raised the error is rolled back and the transaction continues processing.

Question: 93

You have a database that is accessed by 300 concurrent users.

You need to log all of the queries that become deadlocked. The solution must meet the following requirements:

Provide a representation of the deadlock in XML format.

Minimize the impact on the server.

What should you create?

- A. A SQL Server Profiler trace
B. A script that enables trace flags
C. A SQL Server Agent job that retrieves information from the sys.dm_tran_active_transactions dynamic management views
D. A SQL Server Agent job that retrieves information from the sys.dm_tran_session_transactions dynamic management views

Answer: A

Analyze Deadlocks with SQL Server Profiler

Use SQL Server Profiler to identify the cause of a deadlock. A deadlock occurs when there is a cyclic dependency between two or more threads, or processes, for some set of resources within SQL Server. Using SQL Server Profiler, you can create a trace that records, replays, and displays deadlock events for analysis.

To trace deadlock events, add the Deadlock graph event class to a trace. This event class populates the TextData data column in the trace with XML data about the process and objects that are involved in the deadlock. SQL Server Profiler can extract the XML document to a deadlock XML (.xdl) file which you can view later in SQL Server Management Studio.

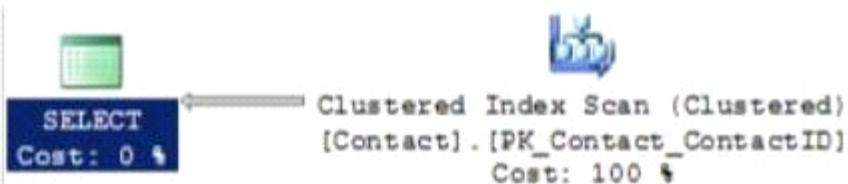
Question: 94

You have the following query on a disk-based table:

```
SELECT ContactID,  
      EmailAddress,  
      LastName  
FROM Person.Contact  
WHERE LastName = N'Johnson'
```

You discover that the query takes a long time to complete.

The execution plan is shown in the Execution Plan exhibit. (Click the Exhibit button.)



The index usage is show in the Index Usage exhibit. (Click the Exhibit button.)

Clustered Index Scan (Clustered)

Scanning a clustered index, entirely or only a range.

Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Actual Execution Mode	Row
Estimated Execution Mode	Row
Actual Number of Rows	730
Actual Number of Batches	0
Estimated I/O Cost	2.04016
Estimated Operator Cost	2.06229 (100%)
Estimated CPU Cost	0.0221262
Estimated Subtree Cost	2.06229
Number of Executions	1
Estimated Number of Executions	1
Estimated Number of Rows	82.1249
Estimated Row Size	78 B
Actual Rebinds	0
Actual Rewinds	0
Ordered	False
Node ID	0

Predicate

[DB1].[Person].[Contact].[LastName]=CONVERT_IMPLICIT
(nvarchar(4000),[@1],0)

Object

[DB1].[Person].[Contact].[PK_Contact_ContactID]

Output List

[DB1].[Person].[Contact].ContactID, [DB1].[Person].[Contact].EmailAddress, [DB1].[Person].[Contact].LastName

You need to reduce the amount of time it takes to complete the query. You must achieve this goal as quickly as possible.

What should you do?

- A. Reorganize the index.
- B. Update statistics.
- C. Create an index on LastName.
- D. Rebuild the index.

Answer: C

Question: 95

DRAG DROP

You are a SQL Server 2014 Developer. A database that you work on contains two tables that are defined as follows:

```
CREATE TABLE Product (
    ProductID int IDENTITY(1,1) PRIMARY KEY,
    ProductName nvarchar(30) NOT NULL,
    LastUpdatedDate smalldatetime,
    LastUpdatedBy nvarchar(128))

CREATE TABLE ProductAudit (
    ProductAuditID int IDENTITY(1,1) PRIMARY KEY,
    OldProductID int NOT NULL,
    OldProductName nvarchar(30) NOT NULL,
    UpdatedDate smalldatetime,
    UpdatedBy nvarchar(128))
```

Product is an important table that has sensitive audit requirements.

You need to create a trigger that supports the following requirements:

1. Every row that is inserted or updated in Product will reflect its actual LastUpdatedDate and LastUpdatedBy values in the Product table.
2. Any row that is updated or deleted must write a new record reflecting the OLD values into the ProductAudit table.
3. Any error that occurs during the course of the trigger's execution must prevent the changes from happening.

Develop the solution by selecting and arranging the required code blocks in the correct order.

You may not need all of the code blocks.

Code Blocks	Answer Area
<pre> DECLARE @OldProductId int, @OldProductName varchar(30) SELECT @OldProductId = ProductId, @OldProductName = ProductName FROM deleted INSERTProductAudit (OldProductID, OldProductName, UpdatedDate, Update dBy) SELECT @OldProductID, @OldProductName, SUSER_NAME() (), GETDATE() UPDATE Product SET LastUpdatedBy = SUSER_NAME(), LastUpdatedDate = GETDATE() FROM Product AS p INNER JOIN inserted AS i ON p.ProductID = i.ProductID UPDATE Product SET LastUpdatedBy = SUSER_NAME(), LastUpdatedDate = GETDATE() FROM Product AS p INNER JOIN inserted AS i ON p.ProductID = i.ProductID INSERTProductAudit (OldProductID, OldProductName, UpdatedDate, Update dBy) SELECT d.ProductID, d.ProductName, SUSER_NAME() (), GETDATE() FROM deleted AS d END COMMIT TRANSACTION IF @@ERROR <> 0 ROLLBACK CREATE TRIGGER ProductAuditTrigger ON Product FOR INSERT, UPDATE, DELETE AS BEGIN </pre>	

Answer:

Box 1:

```

CREATE TRIGGER ProductAuditTrigger ON Product
FOR INSERT, UPDATE, DELETE
AS
BEGIN

```

Box 2:

```

DECLARE @OldProductId int, @OldProductName varchar(30)
SELECT @OldProductId = ProductId,
@OldProductName = ProductName
FROM deleted

INSERT ProductAudit
(OldProductID, OldProductName, UpdatedDate, Update
dBy)
SELECT @OldProductID, @OldProductName, SUSER_NAME()
(), GETDATE()

UPDATE Product
SET LastUpdatedBy = SUSER_NAME(),
LastUpdatedDate = GETDATE()
FROM Product AS p
INNER JOIN inserted AS i ON p.ProductID
= i.ProductID

```

Box 3:

```
COMMIT TRANSACTION
```

Box 4:

```
IF @@ERROR <> 0
ROLLBACK
```

Box 5:

```
END
```

Note:

* Executing a ROLLBACK TRANSACTION or COMMIT TRANSACTION Transact-SQL statement inside a stored procedure or trigger is possible, but doing so may cause errors.

Question: 96

DRAG DROP

You administer a SQL Server 2014 instance.

The server is capable of 10000 IO/second (IOPS). During the time period when the second process executes, the disk IO can reach 7000 IOPS, and CPU use can average 30% over the eight processors.

The first process summarizes the day's activity executed by a login of [SummaryReportLogin]. The second process submits transactions executed by a login of [ETLLogin].

A Resource Governor classifier function has been created to return WG_Low for connections from the [ETLLogin] and [SummaryReportLogin].

You need to set up the Resource Group and Workgroup Pools on the instance.

You have the following requirements:

Both processes must never use more than 50 percent of the CPU at any one time.

The number of active queries that these processes can execute simultaneously should be limited to a maximum of 10.

The SummaryReportLogin process must always achieve the minimum IOPS required to be minimally affected during executing the ETLLogin processes.

Develop the solution by selecting and arranging the required code blocks in the correct order.

You may not need all of the code blocks.

Code Blocks

```
MAX_IOPS_PER_VOLUME=3000
)
```

```
CREATE WORKLOAD GROUP WG_Low
WITH
(
    MAX_DOP = 4
)
USING RP_Low
```

```
CREATE WORKLOAD GROUP WG_Low
WITH
(
GROUP_MAX_REQUESTS=10
)
USING RP_Low
```

```
CREATE WORKLOAD GROUP WG_Low
WITH
(
    REQUEST_MAX_CPU_TIME_SEC = 100,
    MAX_DOP = 4
)
USING RP_Low
```

```
CREATE RESOURCE POOL RP_Low
WITH
(
CAP_CPU_PERCENT=50,
MAX_CPU_PERCENT=30,
```

```
CREATE RESOURCE POOL RP_Low
WITH
(
AFFINITY_SCHEDULER = (0 to 50),
MAX_CPU_PERCENT=30,
```

```
CREATE RESOURCE POOL RP_Low
WITH
(
MAX_CPU_PERCENT=50,
```

```
)
```

Answer Area**Answer:**

Box 1:

```
CREATE RESOURCE POOL RP_Low
WITH
(
CAP_CPU_PERCENT=50,
MAX_CPU_PERCENT=30,
```

Box 2:

```
MAX_IOPS_PER_VOLUME=3000  
)
```

Box 3:

```
CREATE WORKLOAD GROUP WG_Low  
WITH  
(  
GROUP_MAX_REQUESTS=10  
)  
USING RP_Low
```

Note:

CREATE WORKLOAD RESOURCE POOL

* Resource pools. A resource pool, represents the physical resources of the server. You can think of a pool as a virtual SQL Server instance inside of a SQL Server instance.

* Workload groups. A workload group serves as a container for session requests that have similar classification criteria. A workload allows for aggregate monitoring of the sessions, and defines policies for the sessions. Each workload group is in a resource pool.

* CAP_CPU_PERCENT =value

Specifies a hard cap on the CPU bandwidth that all requests in the resource pool will receive. Limits the maximum CPU bandwidth level to be the same as the specified value. value is an integer with a default setting of 100. The allowed range for value is from 1 through 100.

* MIN_IOPS_PER_VOLUME =value

Specifies the minimum I/O operations per second (IOPS) per disk volume to reserve for the resource pool.

* GROUP_MAX_REQUESTS =value

Specifies the maximum number of simultaneous requests that are allowed to execute in the workload group. value must be a 0 or a positive integer.

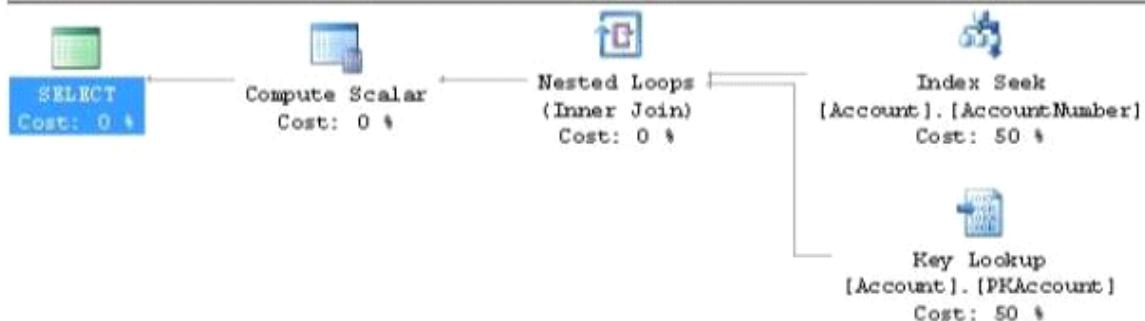
Question: 97

You administer an instance of SQL Server 2014.

You are tasked with tuning a common set of queries. You have the results of several test executions, along with query plans. The schema and the data for all database object(s) used remain unchanged between executions. The QueryTime column is defined as a computed column that uses the GETDATE() system function. The query plans and results are shown below:

```
SELECT *
FROM   dbo.Account
WHERE  AccountNumber = 'A10000001'
```

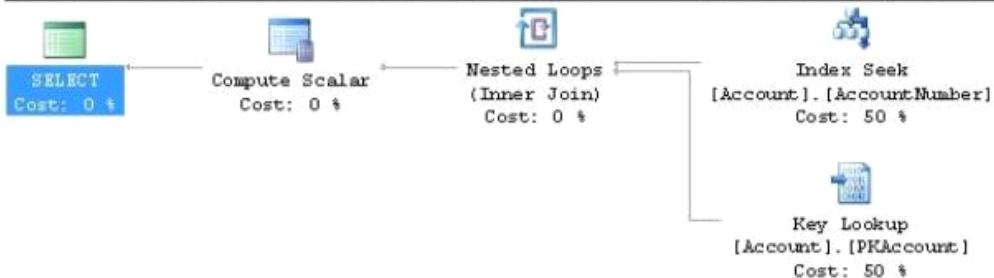
```
Query 1: Query cost (relative to the batch): 100%
SELECT * FROM [dbo].[Account] WHERE [AccountNumber]=@1
```



AccountID	AccountNumber	Name	QueryTime
-----	-----	-----	-----
0F63B176-7257-4480-9D0E-126C45 CEFFF1	A10000001	Don Hall	2014-01-29 18:01:50.923

```
SELECT *
FROM   dbo.Account
WHERE AccountNumber IN( 'A10000001','Q88700323','R00000012')
GO
```

```
Query 1: Query cost (relative to the batch): 100%
SELECT * FROM dbo.Account WHERE AccountNumber IN( 'A10000001', 'Q88700323', 'R00000012' )
```



AccountID	AccountNumber	Name	QueryTime
-----	-----	-----	-----
0F63B176-7257-4480-9D0E-126C45CEFFF1	A10000001	Don Hall	2014-01-29 20:14:05.660
337227AA-3A4B-4B28-8E02-0ADEAD06EA10	Q88700323	Darren Parker	2014-01-29 20:14:05.660
C4980E64-874E-4640-8826-BAF35D8FB845	R00000012	Carol Philips	2014-01-29 20:14:05.660

You need to make an initial diagnosis of the situation, based solely on this input

Which two statements can you make about the performance characteristics of this query? Each correct answer presents a complete solution. Choose two.

- A. The queries would perform better if the index named AccountNumber included the Name and QueryTime column.
- B. The queries would perform worse if the index named AccountNumber included the NameColumn.
- C. The queries would perform better if the index named AccountNumber included the Name column.
- D. The object Account is a table, with an index having a leading column of AccountNumber and a Clustered Index named PKAccount.
- E. The object Account is an indexed view, with an index having a leading column of AccountNumber and a Clustered Index named PKAccount.
- F. The object Account is a view, joining the Account-AccountNumber and Account.PKAccount objects together.

Answer: BD

Question: 98

DRAG DROP

You have a table named Table1 that contains 1 million rows. Table1 contains a column named Column1 that stores sensitive information. Column1 uses the nvarchar(16) data type.

You have a certificate named Cert1.

You need to replace Column1 with a new encrypted column that uses two-way encryption.

Which code segment should you execute before you remove Column1?

To answer, move the appropriate code segments from the list of code segments to the answer area and arrange them in the correct order.

Code segments	Answer Area
ALTER TABLE Table1 ADD Column2 varbinary(256);	
CLOSE SYMMETRIC KEY;	
CREATE SYMMETRIC KEY Key1 WITH ALGORITHM = SHA1 ENCRYPTION BY CERTIFICATE Cert1;	
CREATE CREDENTIAL Cred1 WITH IDENTITY = 'User1', SECRET = 'P@ssw0rd';	
ALTER TABLE Table1 ADD Column2 nvarchar(256);	
OPEN SYMMETRIC KEY Key1 DECRYPTION BY CERTIFICATE Cert1;	
UPDATE table1 SET Column2 = EncryptByKey(Key_GUID ('Key1'),Column1);	
CREATE SYMMETRIC KEY Key1 WITH ALGORITHM = AES_256 ENCRYPTION BY CERTIFICATE Cert1;	

Answer:

Box 1:

```
CREATE SYMMETRIC KEY Key1 WITH ALGORITHM = AES_256
ENCRYPTION BY CERTIFICATE Cert1;
```

Box 2:

```
ALTER TABLE Table1
ADD Column2 varbinary(256);
```

Box 3:

```
OPEN SYMMETRIC KEY Key1
DECRYPTION BY CERTIFICATE Cert1;
```

Box 4:

```
UPDATE table1 SET Column2 = EncryptByKey(Key_GUID
('Key1'),Column1);
```

Box 5:

```
CLOSE SYMMETRIC KEY;
```

Note:

- * Use AES_256 for two-way encryption.
 - * Use varbinary to store key.
 - * CLOSE SYMMETRIC KEY (Transact-SQL)
- Closes a symmetric key, or closes all symmetric keys open in the current session.

- * Example:

```
CREATE SYMMETRIC KEY CreditCards_Key11
WITH ALGORITHM = AES_256
ENCRYPTION BY CERTIFICATE Sales09;
```

GO

-- Create a column in which to store the encrypted data.

```
ALTER TABLE Sales.CreditCard
```

```
ADD CardNumber_Encrypted varbinary(128);
```

```

GO
-- Open the symmetric key with which to encrypt the data.
OPEN SYMMETRIC KEY CreditCards_Key11
    DECRYPTION BY CERTIFICATE Sales09;
-- Encrypt the value in column CardNumber using the
-- symmetric key CreditCards_Key11.
-- Save the result in column CardNumber_Encrypted.
UPDATE Sales.CreditCard
SET CardNumber_Encrypted = EncryptByKey(Key_GUID('CreditCards_Key11')
    , CardNumber, 1, HashBytes('SHA1', CONVERT( varbinary
        , CreditCardID)));
GO

```

Question: 99

The database contains a disk-based table named ContentTable that has 1 million rows and a column named Fax. Fax allows null values.

You need to update Fax to meet the following requirements:

Prevent null values from being used.

Always use an empty string instead of a null value.

Which statement or statements should you execute? (Each correct answer presents part of the solution. Choose all that apply.)

- A. ALTER TABLE dbo.ContentTable
ADD CONSTRAINT
 DF_ContentTable_Fax
 DEFAULT '' FOR Fax
- B. ALTER TABLE ContentTable
ALTER COLUMN
 Fax char(10) NOT NULL
- C. ALTER TABLE ContentTable
ADD COLUMN
 Fax char(10) NOT NULL
- D. ALTER TABLE ContentTable
DROP COLUMN Fax
- E. UPDATE ContentTable
SET Fax = ''
WHERE Fax IS NULL

- A. Option A
- B. Option B
- C. Option C
- D. Option D
- E. Option E

Answer: ABE

E: First change the NULLs to ''.

A: Then set the default to the column to ''.

B: Finally add the NOT NULL constraint to the column.

Question: 100

DRAG DROP

You have a SQL Server 2014 database.

You plan to create a stored procedure that will retrieve the following information:

The XML content of the query plans that is stored in memory

The number of times each query plan is used

You need to identify which dynamic management objects must be used to retrieve the required information for the stored procedure.

Which dynamic management objects should you identify?

To answer, drag the appropriate dynamic management object to the correct requirement in the answer area.

Dynamic Management Objects	Answer Area
sys.dm_exec_sessions	The XML content of the query plans that is stored in memory
sys.dm_exec_cached_plans	The number of times each query plan is used
sys.dm_exec_procedure_stats	
sys.dm_exec_query_plan	
sys.dm_exec_query_optimizer_info	

Answer:

The XML content of the query plans that is stored in memory	sys.dm_exec_query_plan
The number of times each query plan is used	sys.dm_exec_cached_plans

Note:

* sys.dm_exec_query_plan

Returns the Showplan in XML format for the batch specified by the plan handle. The plan specified by the plan handle can either be cached or currently executing.

* sys.dm_exec_cached_plans

Returns a row for each query plan that is cached by SQL Server for faster query execution. You can use this dynamic management view to find cached query plans, cached query text, the amount of memory taken by cached plans, and the reuse count of the cached plans.

Question: 101

DRAG DROP

You plan to create a custom aggregation function named Median.

You plan to deploy Median to a SQL Server 2014 server named Server1.

You need to ensure that Median can access a web service named WebApp1. The solution must minimize the number of changes made to the database.

You create a Microsoft .NET Framework class that contains the definition of Median.

You upload a certificate to Server1.

What three tasks should you perform next?

To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

Actions	Answer Area
Execute the CREATE AGGREGATE statement.	
Modify the TRUSTWORTHY property of the database.	
Execute the CREATE ASSEMBLY statement.	
Execute the CREATE FUNCTION statement.	
Use the certificate to add a digital signature to the assembly.	

Answer:

Box 1: Execute the CREATE ASSEMBLY statement.

Box 2: Use the certificate to add a digital signature to the assembly.

Box 3: Execute the CREATE AGGREGATE statement.

Note:

* CREATE AGGREGATE

Creates a user-defined aggregate function whose implementation is defined in a class of an assembly in the .NET Framework. For the Database Engine to bind the aggregate function to its implementation, the .NET Framework assembly that contains the implementation must first be uploaded into an instance of SQL Server by using a CREATE ASSEMBLY statement.

Question: 102

You have a Microsoft SQL Azure database named DBAzure1. DBAzure1 contains a table named Orders that stores sales data.

Each order has a sales total that can only be discovered by querying multiple tables.

You need to ensure that the value of the sales total is returned by executing a query on Orders.

What should you create?

- A. A calculated column that uses a scalar function
- B. A trigger that uses a table-valued function
- C. A calculated column that uses a table-valued function
- D. A trigger that uses a ranking function

Answer: C

A table-valued parameter is scoped to the stored procedure, function, or dynamic Transact-SQL text, exactly like other parameters. Similarly, a variable of table type has scope like any other local variable that is created by using a DECLARE statement. You can declare table-valued variables within dynamic Transact-SQL statements and pass these variables as table-valued parameters to stored procedures and functions.

Table-valued parameters offer more flexibility and in some cases better performance than temporary tables or other ways to pass a list of parameters.

Incorrect:

Not A: A scalar function would only be able to use other columns from the same table.

Question: 103**DRAG DROP**

You administer a SQL Server 2014 instance.

You have been assigned to determine the cause of frequent long-running transactions that have been tracked to the dbo.Account table, where there are many cases of blocking and deadlocks. The dbo.Account table contains more than one million rows.

Users and processes frequently search for and update data by using the AccountId column, and less frequently the AccountNumber and GovernmentId columns, all of which contain only unique values. Users frequently get lists of AccountNumber values by searching on Last Name and then First Name.

You need to modify the structure of the dbo.Account table to alleviate the issues.

How should you complete the table definition to reduce contention on the table structure? To answer, drag the appropriate code snippets to the correct locations in the CREATE TABLE statement. Each code snippet 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.

Code Snippets	CREATE TABLE Statement
PRIMARY KEY CLUSTERED	CREATE TABLE dbo.Account
(AccountNumber nchar(10) NOT NULL
UNIQUE NONCLUSTERED	
)	AccountId int NOT NULL
INCLUDE (AccountNumber)	
(Lastname, FirstName) INCLUDE (AccountNumber)	GovernmentId nvarchar(11) NOT NULL
(Lastname, FirstName) INCLUDE (AccountNumber)	FirstName nvarchar(20) NOT NULL,
(FirstName, Lastname)	MiddleInitial nvarchar(1) NULL,
	LastName nvarchar(20) NOT NULL
/* No Change To Structure */	
)
	Go
	CREATE NONCLUSTERED INDEX X1 ON dbo.Account

Answer:**CREATE TABLE Statement**

```
CREATE TABLE dbo.Account
(
    AccountNumber nchar(10) NOT NULL
    PRIMARY KEY CLUSTERED
    UNIQUE NONCLUSTERED
        ( Lastname, FirstName ) INCLUDE (AccountNumber)
    AccountId int NOT NULL
    GovernmentId nvarchar(11) NOT NULL
    FirstName nvarchar(20) NOT NULL,
    MiddleInitial nvarchar(1) NULL,
    LastName nvarchar(20) NOT NULL
    )
    Go
    CREATE NONCLUSTERED INDEX X1 ON dbo.Account
```

Note:

Users and processes frequently search for and update data by using the AccountId column (Primary Key Clustered), and less frequently the AccountNumber (Unique Clustered) and GovernmentId(Unique Clustered) columns, all of which contain only unique values. Users frequently get lists of AccountNumber values by searching on Last Name and then First Name (LastName, FirstName) INCLUDE (AccountNumber).

Question: 104

DRAG DROP

You create a disk-based table that contains the following script:

```
CREATE TABLE dbo.Products
(
    ProductId bigint IDENTITY(1,1),
    Name nvarchar(50) NULL,
    Description nvarchar(max) NULL,
    SKU char(10) NULL,
        CONSTRAINT PK_Products PRIMARY KEY CLUSTERED (ProductId)
) ON [PRIMARY]
GO
```

You need to prevent duplicate values in the SKU field.

Which five code segments should you use?

To answer, move the appropriate code segments from the list of code segments to the answer area and arrange them in the correct order.

Code segments	Answer Area
(SKU)	
ALTER TABLE dbo.Products	
CHECK	
FOREIGN KEY	
UNIQUE	
ADD CONSTRAINT	
CK_SKU	

Answer:

Box 1: ALTER TABLE dbo.Products

Box 2: ADD CONSTRAINT

Box 3: CK_SKU

Box 4: UNIQUE

Box 5: (SKU)

Note: The SQL command is:

```
ALTER TABLE <tablename> ADD CONSTRAINT
<constraintname> UNIQUE
(
    <columnname>
)
```

Question: 105

Your network contains a server that has SQL Server 2014 installed.

You create a table by using the following script:

```

CREATE TABLE dbo.Products
(
    id int NOT NULL,
    ProductName nvarchar(50) NULL,
    ProductManufacturer nvarchar(50) NULL,
    ProductDescription nvarchar(200) NULL,
    CONSTRAINT PK_Products PRIMARY KEY CLUSTERED (id)
)
ON [PRIMARY]
GO

```

You need to recommend a solution to ensure that each combination of ProductName and ProductManufacturer is not duplicated.

What should you recommend creating?

- A. A UNIQUE constraint
- B. A filtered index
- C. A columnstore index
- D. A CHECK constraint

Answer: A

Question: 106

You have a Microsoft SQL Azure database.

You have the following stored procedure:

```

01 CREATE PROCEDURE UpdateContact
02     @ContactID int,
03     @LastName nvarchar(50)
04 AS
05
06 SELECT LastName AS OriginalName
07 FROM Person.Contact
08
09 WHERE ContactID = @ContactID;
10 UPDATE Person.Contact
11 SET LastName = @LastName
12
13 WHERE ContactID = @ContactID;

```

You discover that the stored procedure periodically fails to update Person.Contact.

You need to ensure that Person.Contact is always updated when UpdateContact executes. The solution must minimize the amount of time required for the stored procedure to execute and the number of locks held.

What should you do?

- A. Add the following line of code to line 12:
WITH (UPDLOCK)
- B. Add the following line of code to line 05:
SET TRANSACTION ISOLATION LEVEL SERIALIZABLE
- C. Add the following line of code to line 08:
WITH (UPDLOCK)
- D. Add the following line of code to line 05:
SET TRANSACTION ISOLATION LEVEL SNAPSHOT

Answer: C

* Overall, you should use UPDLOCK when you read a value that you plan to update later in the same transaction to prevent the value from changing.

* **UPDLOCK**

Specifies that update locks are to be taken and held until the transaction completes. UPDLOCK takes update locks for read operations only at the row-level or page-level. If UPDLOCK is combined with TABLOCK, or a table-level lock is taken for some other reason, an exclusive (X) lock will be taken instead.

When UPDLOCK is specified, the READCOMMITTED and READCOMMITTEDLOCK isolation level hints are ignored. For example, if the isolation level of the session is set to SERIALIZABLE and a query specifies (UPDLOCK, READCOMMITTED), the READCOMMITTED hint is ignored and the transaction is run using the SERIALIZABLE isolation level.

Question: 107

You use SQL Server 2014. The physical server is a dedicated database server that has 120GB of RAM available. There is approximately 50GB of storage space available on a slow local disk.

You create a new stored procedure. You decide you need to temporarily hold approximately 300,000 rows from two tables, from which you will compute two complex business scores.

The stored procedure will use temporary storage defined as follows:

```
AccountNumber char(10) NOT NULL  
YearToDateSalesTotal decimal(15,2) NULL  
SalesScore int NULL  
FutureSalesExpectationScore int NULL
```

The code will make several passes through the data, applying complex calculations before writing the data to a permanent disk-based table in the same database from which it reads the data.

For this stored procedure, you need to deal with temporary data in the most efficient way to minimize physical disk pressure.

What should you do? More than one answer choice may achieve the goal. Select the BEST answer.

- A.

```
CREATE TYPE dbo.AccountScoringModel as TABLE
(
    AccountNumber char(10) COLLATE Latin1_General_100_BIN2 NOT NULL ,
    YearToDateSalesTotal decimal(15,2) NULL,
    SalesScore int NULL,
    FutureSalesExpectationScore int NULL,
    INDEX AccountNumber HASH (AccountNumber) WITH (BUCKET_COUNT = 25000)
) WITH ( MEMORY_OPTIMIZED = ON )
GO
DECLARE @AccountScoring as dbo.AccountScoringModel
```
- B.

```
DECLARE @AccountScoring as TABLE
(
    AccountNumber char(10) NOT NULL,
    YearToDateSalesTotal decimal(15,2) NULL,
    SalesScore int NULL,
    FutureSalesExpectationScore int NULL
)
```
- C.

```
CREATE TABLE #AccountScoring
(
    AccountNumber char(10) NOT NULL,
    YearToDateSalesTotal decimal(15,2) NULL,
    SalesScore int NULL,
    FutureSalesExpectationScore int NULL
)
```
- D.

```
CREATE TYPE dbo.AccountScoringModel as TABLE
(
    AccountNumber char(10) COLLATE Latin1_General_100_BIN2 NOT NULL ,
    YearToDateSalesTotal decimal(15,2) NULL,
    SalesScore int NULL,
    FutureSalesExpectationScore int NULL,
    INDEX AccountNumber HASH (AccountNumber) WITH (BUCKET_COUNT = 120)
) WITH ( MEMORY_OPTIMIZED = ON )
GO
DECLARE @AccountScoring as dbo.AccountScoringModel
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: A

* You must specify a value for the BUCKET_COUNT parameter when you create the memory-optimized table. In most cases the bucket count should be between 1 and 2 times the number of distinct values in the index key. If the index key contains a lot of duplicate values, on average there are more than 10 rows for each index key value, use a nonclustered index instead

You may not always be able to predict how many values a particular index key may have or will have. Performance should be acceptable if the BUCKET_COUNT value is within 5 times of the actual number of key values.

Question: 108

DRAG DROP

You need to identify which nonclustered indexes are unused by queries.

How should you complete the statement? To answer, drag the appropriate values to the correct locations. Each value

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.

Values

sys.dm_db_index_operational_stats
sys.dm_db_index_physical_stats
sys.dm_db_index_usage_stats
sys.indexes
sys.tables

• • • •

Answer Area

```
SELECT database_id, a.object_id, a.index_id, b.name, a.user_seeks, a.user_scans, a.user_updates
FROM sys.dm_db_index_usage_stats a
join sys.indexes b on a.object_id = b.object_id
and a.index_id = b.index_id
where a.index_id >= 2
and a.database_id = db_id()
and (a.user_seeks = 0
     or a.user_scans = 0)
```

Answer:

Answer Area

```
SELECT database_id, a.object_id, a.index_id, b.name, a.user_seeks, a.user_scans, a.user_updates
FROM sys.dm_db_index_usage_stats a
join sys.indexes b on a.object_id = b.object_id
and a.index_id = b.index_id
where a.index_id >= 2
and a.database_id = db_id()
and (a.user_seeks = 0
     or a.user_scans = 0)
```

Box 1: sys.dm_db_index_usage_stats

sys.dm_db_index_usage_stats shows you how many times the index was used for user queries. It returns counts of different types of index operations and the time each type of operation was last performed in SQL Server.

Box 2: sys.indexes

sys.indexes contains a row per index or heap of a tabular object, such as a table, view, or table-valued function.

References: <https://docs.microsoft.com/en-us/sql/relational-databases/system-dynamic-management-views/sys-dm-db-index-usage-stats-transact-sql>

<https://docs.microsoft.com/en-us/sql/relational-databases/system-catalog-views/sys-indexes-transact-sql>

Question: 109

DRAG DROP

You have a database named database1. Each table in database1 has one index per column.

Users often report that creating items takes a long time.

You need to perform the following maintenance tasks:

What should you use?

To answer, drag the appropriate function to the correct management task in the answer area. (Answer choices may be used once, more than once, or not at all.)

Functions	Answer Area
sys.dm_db_index_usage_stats	Identify unused indexes.
sys.dm_db_index_operational_stats	Identify which indexes should be created.
sys.dm_db_index_physical_stats	
sys.dm_db_missing_index_columns	
sys.dm_db_missing_index_details	
sys.dm_db_missing_index_groups	

Answer:

Answer Area
Identify unused indexes.
Identify which indexes should be created.

Box 1: sys.dm_db_index_usage_stats

sys.dm_db_index_usage_stats shows you how many times the index was used for user queries. It returns counts of different types of index operations and the time each type of operation was last performed in SQL Server.

Box 2: sys.dm_db_missing_index_details

sys.dm_db_missing_index_details returns detailed information about a missing index; for example, it returns the name and identifier of the table where the index is missing, and the columns and column types that should make up the missing index.

References: <https://docs.microsoft.com/en-us/sql/relational-databases/system-dynamic-management-views/sys-dm-db-index-usage-stats-transact-sql>

<https://docs.microsoft.com/en-us/sql/relational-databases/system-catalog-views/sys-indexes-transact-sql>

[https://technet.microsoft.com/en-us/library/ms345524\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms345524(v=sql.105).aspx)

Question: 110

You are designing a new database table that will be used for reporting.

You define the table by using the following statement.

```
CREATE TABLE Customers
(CustomerId BIGINT,
 LastName VARCHAR(255),
 FirstName VARCHAR(255),
 SalesPersonId INT,
 Address1 VARCHAR(255),
 Address2 VARCHAR(255),
 City VARCHAR(255),
 State CHAR(2),
 PostalCode VARCHAR(10),
 Country CHAR(2))
```

You need to store the data in the table by using the least amount of storage space possible.
Which storage option should you use?

- A. a clustered index
- B. a clustered columnstore index
- C. a nonclustered index
- D. In-Memory OLTP

Answer: B

Explanation:

Columnstore indexes work well for mostly read-only queries that perform analysis on large data sets. This would fit this scenario as the table will be used for reporting.

Columnstore Index benefits include Columnstore Index benefits high compression rates, which improve query performance by using a smaller in-memory footprint. In turn, query performance can improve because SQL Server can perform more query and data operations in-memory.

Use the columnstore index to achieve up to 10x query performance gains over traditional row-oriented storage, and up to 7x data compression over the uncompressed data size.

References: [https://msdn.microsoft.com/en-us/library/gg492088\(v=sql.120\).aspx](https://msdn.microsoft.com/en-us/library/gg492088(v=sql.120).aspx)

Question: 111

You have the following stored procedure.

```
CREATE PROCEDURE GetFile
    @FileName nvarchar(512)
AS
SELECT *
FROM Files
WHERE FileName = @FileName
```

The stored procedure takes much longer to execute than expected.

While reviewing the execution plan of the stored procedure, you discover the following predicate for a Clustered Index Scan operator.

```
CONVERT_IMPLICIT(nvarchar(512), [SampleDatabase].[dbo].[Files].[FileName], 0)=[@1]
```

You need to resolve the performance issue.

What should you do?

- A. Change the @FileName parameter from nvarchar(512) to varchar(512).
- B. Change the FileName column from varchar(512) to nvarchar(512).
- C. Add a NOLOCK query hint to the SELECT statement.
- D. Convert the table to a memory-optimized table.
- E. Add a FORCESEEK query hint to the SELECT statement.

Answer: A

Explanation:

When using a variable, make sure that the datatype matches the column's datatype. We suspect that the issue is that the variable is NVARCHAR (512) whilst the table column is VARCHAR (512). This is indicated by the CONVERT_IMPLICIT operator in the execution plan.

References: https://sqlserverperformance.wordpress.com/2009/02/02/beware-of-convert_implicit-in-a-sql-execution-plan/

Question: 112

DRAG DROP

You use the following statement to create a table.

```
CREATE TABLE Employee
(EmployeeID INT PRIMARY KEY IDENTITY(1,1),
LastName varchar(50),
FirstName varchar(50),
DepartmentId INT,
SupervisorId INT,
OfficeId INT,
Address1 varchar(50),
Address2 varchar(50),
City varchar(50),
State char(2),
PostalCode varchar(10),
Country char(2))
```

You have the following queries.

```
SELECT FirstName, LastName, EmployeeId, OfffceID, DepartmentID
FROM Employee
WHERE FirstName = 'Ben' AND LastName = 'Smith'
```

```
SELECT FirstName, LastName, EmployeeId, OfffceID, DepartmentID
FROM Employee
WHERE LastName = 'Smith'
```

You need to create an index to minimize the execution time of the queries.

How should you complete the statement? To answer, drag the appropriate code elements to the correct locations.

Each code element 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.

Code Elements

DepartmentId
EmployeeId
FirstName
LastName
OfficeId

Answer Area

```
CREATE INDEX IX_Index3 ON dbo.Employee
( [Code element], [Code element] )
INCLUDE ( [Code element], [Code element] )
```

Answer:

Answer Area

```
CREATE INDEX IX_Index3 ON dbo.Employee
( LastName, FirstName )
```



```
INCLUDE ( DepartmentId, OfficeId )
```

Box 1: LastName

Redesign nonclustered indexes with a large index key size so that only columns used for searching and lookups are key columns. Make all other columns that cover the query into nonkey columns. In this way, you will have all columns needed to cover the query, but the index key itself is small and efficient.

Box 2: FirstName

Box 3: DepartmentID

Non-key columns, called included columns, can be added to the leaf level of a nonclustered index to improve query performance by covering the query. That is, all columns referenced in the query are included in the index as either key or non-key columns. This allows the query optimizer to locate all the required information from an index scan; the table or clustered index data is not accessed.

Box 4: OfficeID

Question: 113

DRAG DROP

You plan to deploy two stored procedures name USP_1 and USP_2 that read data from a database.

Your company identifies the following requirements for each stored procedure:

You need to identify which isolation level you must set for each stored procedure. The solution must minimize the number of locks.

Which isolation level should you identify?

To answer, drag the appropriate isolation level to the correct stored procedure in the answer area. (Answer choices may be used once, more than once, or not at all.)

Isolation Levels	Answer Area
read committed	
read uncommitted	
repeatable read	
serializable	
snapshot	

Answer:

Answer Area
USP_1 read uncommitted
USP_2 serializable

Box 1: read uncommitted

READ UNCOMMITTED is the least restrictive isolation level because it ignores locks placed by other transactions. Transactions executing under READ UNCOMMITTED can read modified data values that have not yet been committed by other transactions; these are called "dirty" reads.

Box 2: SERIALIZABLE

Places a range lock on the data set, preventing other users from updating or inserting rows into the data set until the transaction is complete. This is the most restrictive of the four isolation levels. Because concurrency is lower, use this option only when necessary. This option has the same effect as setting HOLDLOCK on all tables in all SELECT statements in a transaction.

References: [https://msdn.microsoft.com/en-us/library/tcbchxcb\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/tcbchxcb(v=vs.110).aspx)