[100% Pass Unlimited Passleader Microsoft 70-461 Exam Braindump For Free (46-60)

Free Download 100% Pass Ensure 70-461 New Exam Dumps: PassLeader now supplying the new version of 70-461 vce dumps, we ensure our 70-461 136q exam questions are the most complete and authoritative compared with others', which will ensure your 70-461 exam 100% pass, and now we are offering the free new version VCE Player along with the VCE format 70-461 136q braindump, also the PDF format 70-461 practice test is available now, welcome to choose. keywords: 70-461,70-461 exam,70-461 exam dumps,70-461 136q exam questions,70-461 pdf dumps,70-461 136q practice test,70-461 vce file,Querying Microsoft SQL Server 2012

QUESTION 46 You use a Microsoft SQL Server 2012 database that contains a table named BlogEntry that has the following columns(Id is the Primary Key):

Column name	Data type
Id	bigint
EntryDateTime	datetime
Summary	nvarchar(max)

You need to append the "This is in a draft stage" string to the Summary column of the recent 10 entries based on the values in EntryDateTime. Which Transact-SQL statement should you use? A.   UPDATE TOP(10) BlogEntry SET Summary.WRITE(N' This is in a draft stage', NULL, 0) B.   UPDATE BlogEntry SET Summary = CAST(N' This is in a draft stage' as nvarchar(max)) WHERE Id IN(SELECT TOP(10) Id FROM BlogEntry ORDER BY EntryDateTime DESC) C.   UPDATE BlogEntry SET Summary.WRITE(N' This is in a draft stage', NULL, 0) FROM (SELECT TOP(10) Id FROM BlogEntry ORDER BY EntryDateTime DESC) AS s WHERE BlogEntry.Id = s.ID D.   UPDATE BlogEntrySET Summary.WRITE(N' This is in a draft stage', 0, 0) WHERE Id IN(SELECT TOP(10) Id FROM BlogEntry ORDER BY EntryDateTime DESC) Answer: B QUESTION 47 You use Microsoft SQL Server 2012 to develop a database application. You create a stored procedure named DeleteJobCandidate. You need to ensure that if DeleteJobCandidate encounters an error, the execution of the stored procedure reports the error number. Which Transact-SQL statement should you use?

A.
```
DECLARE @ErrorVar INT;
DECLARE @RowCountVar INT;

EXEC DeleteJobCandidate

SELECT @ErrorVar = @@ERROR,
  @RowCountVar = @@ROWCOUNT;
IF (@ErrorVar <> 0)
PRINT N'Error = ' + CAST(@@ErrorVar AS NVARCHAR(8))+ N', Rows Deleted = ' + CAST(@@RowCountVar AS NVARCHAR
(8));
GO
```

B.
```
DECLARE @ErrorVar INT;
DECLARE @RowCountVar INT;
EXEC DeleteJobCandidate
SELECT @ErrorVar = ERROR_STATE(),
  @RowCountVar = @@ROWCOUNT;
IF (@ErrorVar <> 0)
PRINT N'Error = ' + CAST(ERROR_STATE() AS NVARCHAR(8))+ N', Rows Deleted = ' + CAST(@RowCountVar AS NVARCHAR
(8));
GO
```

C.
```
EXEC DeleteJobCandidate
IF (ERROR_STATE() != 0)
  PRINT N'Error = ' + CAST(@@ERROR AS NVARCHAR(8))+ N', Rows Deleted = ' + CAST(@@ROWCOUNT AS NVARCHAR(8));
GO
```

D.
```
EXEC DeleteJobCandidate
PRINT N'Error = ' + CAST(@@ERROR AS NVARCHAR(8))+ N', Rows Deleted = ' + CAST(@@ROWCOUNT AS NVARCHAR(8));
GO
```


A.   Option A B.   Option B C.   Option C D.   Option D Answer: A QUESTION 48 A table named Profits stores the total profit made each year within a territory. The Profits table has columns named Territory, Year, and Profit. You need to create a report that displays the profits

made by each territory for each year and its preceding year. Which Transact-SQL query should you use? A.   SELECT Territory, Year, Profit, LAG(Profit, 1, 0) OVER(PARTITION BY Year ORDER BY Territory) AS NextProfit FROM Profits B.   SELECT Territory, Year, Profit, LAG(Profit, 1, 0) OVER(PARTITION BY Territory ORDER BY Year) AS NextProfit FROM Profits C.   SELECT Territory, Year, Profit, LEAD(Profit, 1, 0) OVER(PARTITION BY Territory ORDER BY Year) AS NextProfit FROM Profits D.   SELECT Territory, Year, Profit, LEAD(Profit, 1, 0) OVER(PARTITION BY Year ORDER BY Territory) AS NextProfit FROM Profits Answer: B QUESTION 49 You use Microsoft SQL Server 2012 to create a stored procedure as shown in the following code segment. (Line numbers are included for reference only.) The procedure can be called within other transactions. You need to ensure that when the DELETE statement from the HumanResourcesJobCandidate table succeeds, the modification is retained even if the insert into the Audit.Log table fails. Which code segment should you add to line 14?

```
01  CREATE PROCEDURE DeleteCandidate
02  @InputCandidateID INT;
03  AS
04  BEGIN
05    BEGIN TRANSACTION;
06    BEGIN TRY
07      DELETE HumanResources.JobCandi
08      WHERE JobCandidateID = @Input
09      INSERT INTO Audit.Log(Operatio
10      VALUES('Delete', ABS(@@IDENTI
11      COMMIT TRANSACTION;
12    END TRY
13    BEGIN CATCH
14
15      COMMIT TRANSACTION
16    ELSE
17      ROLLBACK TRANSACTION;
18  END CATCH
19  END;
```

A.   IF @@TRANCOUNT = 0 B.   IF (XACT_STATE ( )) = 0 C.   IF (XACT_STATE ( )) = 1 D.   IF @@TRANCOUNT = l Answer: C QUESTION 50 You use Microsoft SQL Server 2012 to develop a database application. Your application sends data to an NVARCHAR(MAX) variable named @var. You need to write a Transact-SQL statement that will find out the success of a cast to a decimal (36,9). Which code segment should you use?

A.
```
BEGIN TRY
SELECT
  convert (decimal(36,9), @var) as Value,
  'True' As BadCast
END TRY
BEGIN CATCH
SELECT
  convert (decimal(36,9), @var) as Value,
  'False' As BadCast
END CATCH
```

B.
```
TRY(
  SELECT convert (decimal(36,9), @var)
  SELECT 'True' As BadCast
)
CATCH(
  SELECT 'False' As BadCast
)
```

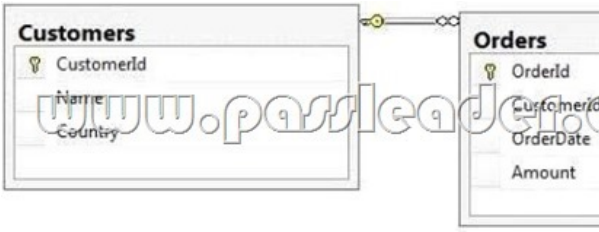

C.
```
SELECT
  CASE
  WHEN convert (decimal(36,9), @var) IS NULL
  THEN 'True'
  ELSE 'False'
  END
As BadCast
```

D.
```
SELECT
  IIF(TRY_PARSE(@var AS decimal(36,9)) IS NULL,
  'True',
  'False'
  )
AS BadCast
```

A.   Option A B.   Option B C.   Option C D.   Option D Answer: D QUESTION 51 You are writing a set of queries against a FILESTREAM-enabled database. You create a stored procedure that will update multiple tables within a transaction. You need to ensure that if the stored procedure raises a run-time error, the entire transaction is terminated and rolled back. Which Transact-SQL statement should you include at the beginning of the stored procedure? A.   SET TRANSACTION ISOLATION LEVEL SERIALIZABLE B.   SET XACT_ABORT OFF C.   SET TRANSACTION ISOLATION LEVEL SNAPSHOT D.   SET IMPLICIT_TRANSACTIONS ON E.   SET XACT_ABORT ON F.   SET IMPLICIT TRANSACTIONS OFF Answer: E QUESTION 52 You develop a Microsoft SQL Server 2012 database. The database is used by two web applications that access a table named Products. You want to create an object that will prevent the applications from accessing the table directly while still providing access to the required data. You need to ensure that the following requirements are met: - Future modifications to the table definition will not affect the applications' ability to access data. - The new object can accommodate data retrieval and data modification. You need to achieve this goal by using the minimum amount of changes to the applications. What should you create for each application? A.   Synonyms B.   Common table expressions C.   Views D.   Temporary tables Answer: C

http://www.passleader.com/70-461.html QUESTION 53 You administer a Microsoft SQL Server 2012 database named

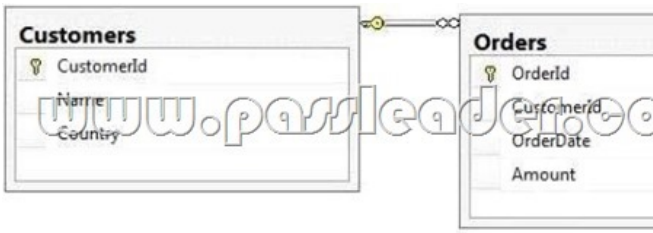
ContosoDb. Tables are defined as shown in the exhibit. (Click the Exhibit button.)



You need to display rows from the Orders table for the Customers row having the CustomerIdvalue set to 1 in the following XML format.

```
<Orders OrderId="1" OrderDate="2000-01-01T00:00:00" Amount="3400.00">
  <Customers Name="Customer A" Country="Australia" />
</Orders>
<Orders OrderId="2" OrderDate="2001-01-01T00:00:00" Amount="4300.00">
  <Customers Name="Customer A" Country="Australia" />
</Orders>
```

Which Transact-SQL query should you use? A.   SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML RAW B.   SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers=CustomerId = 1 FOR XML RAW, ELEMENTS C.   SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML AUTO D.   SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO, ELEMENTS E.   SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO F.   SELECT Name, Country, Orderld, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO, ELEMENTS G.   SELECT Name AS '@Name', Country AS '@Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML PATH ('Customers') H.   SELECT Name AS 'Customers/Name', Country AS 'Customers/Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML PATH ('Customers') Answer: C QUESTION 54 You administer a Microsoft SQL Server 2012 database named ContosoDb. Tables are defined as shown in the exhibit:



You need to display rows from the Orders table for the Customers row having the CustomerId value set to 1 in the following XML format. <CUSTOMERS Name="Customer A" Country="Australia"> <ORDERS OrderID="1" OrderDate="2001-01-01" Amount="3400.00" /> <ORDERS OrderID="2" OrderDate="2002-01-01" Amount="4300.00" /> </CUSTOMERS> Which Transact-SQL query should you use? A.   SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML RAW B.   SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML RAW, ELEMENTS C.   SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML AUTO D.   SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId - Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO, ELEMENTS E.   SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO F.

SELECT Name, Country, Orderld, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO, ELEMENTS G.   SELECT Name AS '@Name', Country AS '@Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML PATH ('Customers') H.   SELECT Name AS'Customers/Name', Country AS 'Customers/Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML PATH ('Customers') Answer: E QUESTION 55 You administer a Microsoft SQL Server 2012 database named ContosoDb. Tables are defined as shown in the exhibit:

You need to display rows from the Orders table for the Customers row having the CustomerId value set to 1 in the following XML format.

```
<Orders>
  <OrderId>1</OrderId>
  <OrderDate>2000-01-01T00:00:00</OrderDate>
  <Amount>3400.00</Amount>
  <Customers>
    <Name>Customer A</Name>
    <Country>Australia</Country>
  </Customers>
</Orders>
<Orders>
  <OrderId>2</OrderId>
  <OrderDate>2001-01-01T00:00:00</OrderDate>
  <Amount>4300.00</Amount>
  <Customers>
    <Name>Customer A</Name>
    <Country>Australia</Country>
  </Customers>
</Orders>
```


Which Transact-SQL query should you use? A.   SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML RAW B.   SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML RAW, ELEMENTS C.   SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML AUTO D.   SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO, ELEMENTS E.   SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId- 1 FOR XML AUTO F.   SELECT Name, Country, Orderld, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO, ELEMENTS G.   SELECT Name AS '@Name', Country AS '@Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId=Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML PATH ('Customers') H.   SELECT Name AS'Customers/Name', Country AS 'Customers/Country', OrderId,OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML PATH ('Customers') Answer: D QUESTION 56 You develop a Microsoft SQL Server 2012 server database that supports an application. The application contains a table that has the following definition: CREATE TABLE Inventory (ItemID int NOT NULL PRIMARY KEY, ItemsInStore int NOT NULL, ItemsInWarehouse int NOT NULL) You need to create a computed column that returns the sum total of the ItemsInStore and ItemsInWarehouse values for each row. The new column is

expected to be queried heavily, and you need to be able to index the column. Which Transact-SQL statement should you use?
A.   ALTER TABLE Inventory ADD TotalItems AS ItemslnStore + ItemsInWarehouse
B.   ALTER TABLE Inventory ADD TotalItems AS ItemsInStore + ItemsInWarehouse PERSISTED
C.   ALTER TABLE Inventory ADD TotalItems AS SUM(ItemsInStore,ItemsInWarehouse) PERSISTED
D.   ALTER TABLE Inventory ADD TotalItems AS SUM(ItemsInStore, ItemsInWarehouse) Answer: B
QUESTION 57 You develop a Microsoft SQL Server 2012 database that contains a table named Customers. The Customers table has the following definition:

```
CREATE TABLE [dbo].[Customers](
  [CustomerId] [bigint] NOT NULL,
  [MobileNumber] [nvarchar](25) NOT NULL,
  [HomeNumber] [nvarchar](25) NULL,
  [Name] [nvarchar](50) NOT NULL,
  [Country] [nvarchar](26) NOT NULL,
  CONSTRAINT [PK_Customers] PRIMARY KEY CLUSTERED
  (
    [CustomerId] ASC
  ) ON [PRIMARY]
) ON [PRIMARY]
```


You need to create an audit record only when either the MobileNumber or HomeNumber column is updated. Which Transact-SQL query should you use? A.   CREATE TRIGGER TrgPhoneNumberChange ON Customers FOR UPDATE AS IF COLUMNS_UPDATED (HomeNumber, MobileNumber) - - Create Audit Records
B.   CREATE TRIGGER TrgPhoneNumberChange ON Customers FOR UPDATE AS IF EXISTS(SELECT HomeNumber FROM inserted) OR EXISTS (SELECT MobileNumber FROM inserted) - - Create Audit Records
C.   CREATE TRIGGER TrgPhoneNumberChange ON Customers FOR UPDATE AS IF COLUMNS_CHANGED (HomeNumber, MobileNumber) - - Create Audit Records D.   CREATE TRIGGER TrgPhoneNumberChange ON Customers FOR UPDATE AS IF UPDATE (HomeNumber) OR UPDATE (MobileNumber) - - Create Audit Records Answer: D QUESTION 58 You develop a Microsoft SQL Server 2012 database that has two tables named SavingAccounts and LoanAccounts. Both tables have a column named AccountNumber of the nvarchar data type. You use a third table named Transactions that has columns named TransactionId AccountNumber, Amount, and TransactionDate. You need to ensure that when multiple records are inserted in the Transactions table, only the records that have a valid AccountNumber in the SavingAccounts or LoanAccounts are inserted. Which Transact-SQL statement should you use?

A.
```
CREATE TRIGGER TrgValidateAccountNumber
ON Transactions
INSTEAD OF INSERT
AS
BEGIN
  INSERT INTO Transactions
  SELECT TransactionId,AccountNumber,Amount,TransactionDate FROM inserted
  WHERE AccountNumber IN
  (SELECT AccountNumber FROM LoanAccounts
   UNION SELECT AccountNumber FROM SavingAccounts)
END
```

B.
```
CREATE TRIGGER TrgValidateAccountNumber
ON Transactions
FOR INSERT
AS
BEGIN
  INSERT INTO Transactions
  SELECT TransactionId,AccountNumber,Amount,TransactionDate FROM inserted
  WHERE AccountNumber IN
  (SELECT AccountNumber FROM LoanAccounts
   UNION SELECT AccountNumber FROM SavingAccounts)
END
```

C.
```
CREATE TRIGGER TrgValidateAccountNumber
ON Transactions
INSTEAD OF INSERT
AS
BEGIN
  IF EXISTS (
   SELECT AccountNumber FROM inserted EXCEPT
  (SELECT AccountNumber FROM LoanAccounts
   UNION SELECT AccountNumber FROM SavingAccounts))
  BEGIN
    ROLLBACK TRAN
  END
END
```


D.
```
CREATE TRIGGER TrgValidateAccountNumber
ON Transactions
FOR INSERT
AS
BEGIN
  IF EXISTS (
  SELECT AccountNumber FROM inserted EXCEPT
  (SELECT AccountNumber FROM LoanAccounts
   UNION SELECT AccountNumber FROM SavingAccounts))
  BEGIN
    ROLLBACK TRAN
  END
END
```

A.   Option A B.   Option B C.   Option C D.   Option D Answer: A QUESTION 59 You develop a Microsoft SQL Server 2012 database. You create a view that performs the following tasks: - Joins 8 tables that contain up to 500,000 records each. - Performs aggregations on 5 fields. - The view is frequently used in several reports. You need to improve the performance of the reports. What should you do? A.   Convert the view into a table-valued function. B.   Convert the view into a Common Table Expression (CTE). C.   Convert the view into an indexed view. D.   Convert the view into a stored procedure and retrieve the result from the stored procedure into a temporary table. Answer: C QUESTION 60 You are a database developer of a Microsoft SQL Server 2012 database. The database contains a table named Customers that has the following definition:

```
CREATE TABLE Customer
(CustomerID INT NOT NULL PRIMARY KEY,
 CustomerName VARCHAR(255) NOT NULL,
 CustomerAddress VARCHAR(1000) NOT NULL)
```


You are designing a new table named Orders that has the following definition:

```
CREATE TABLE Orders
(OrderID INT NOT NULL PRIMARY KEY,
 CustomerID INT NOT NULL,
 OrderDescription VARCHAR(2000))
```

You need to ensure that the CustomerId column in the Orders table contains only values that exist in the CustomerId column of the Customer table. Which Transact-SQL statement should you use? A.   ALTER TABLE Orders ADD CONSTRAINT FX_Orders_CustomerID FOR EIGN KEY (CustomerId) REFERENCES Customer (CustomerId) B.   ALTER TABLE Customer ADD CONSTRAINT FK_Customer_CustomerID FOR EIGN KEY {Cu3tomerID; REFERENCES Orders (CustomerId) C.   ALTER TABLE Orders ADD CONSTRAINT CK_Crders_CustomerID CHECK (CustomerId IN (SELECT CustomerId FROM Customer)) D.   ALTER TABLE Customer ADD OrderId INT NOT NULL; ALTER TABLE Customer ADD CONSTRAINT FK_Customer_OrderID

FOR EIGN KEY (CrderlD) REFERENCES Orders (CrderlD); E.   ALTER TABLE Orders ADD CONSTRAINT PK Orders CustomerId PRIMARY KEY (CustomerID) Answer: A

http://www.passleader.com/70-461.html