Braindump2go Shares the Latest Free Official Microsoft 70-433 Exam Training Questions and Answers (181-190)

All Latest Updated Questions and Answers in Braindump2go 70-433 Exam Dumps will not take you a lot of time to comprehend and you can easily cover up the entire Microsoft 70-433 syllabus for your examination.Download Braindump2go Free 70-433 Sample Questions Now, Pass 70-433 Exam in advance! Exam Code: 70-433Exam Name: TS: Microsoft SQL Server 2008, Database DevelopmentCertification Provider: MicrosoftKeywords: 70-433 Exam Dumps,70-433 Practice Tests,70-433 Practice Exams,70-433 Exam Questions,70-433 PDF,70-433 VCE Free,70-433 Book,70-433 E-Book,70-433 Study Guide,70-433 Braindump,70-433 Prep Guide

QUESTION 181Drag and Drop QuestionYou are a developer of a Microsoft SQL Server 2008 R2 database instance that supports a web-based order-entry application.You need to create a server-side trace that meets the following requirements:Which four actions should you perform in sequence? (To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.)

Answer:

QUESTION 182You create a stored procedure that contains proprietary formulas.You need to ensure that no users are able to access Microsoft SQL Server Management Studio to view the definition of the stored procedure.Which code segment should you use?

A.   Option AB.   Option BC.   Option CD.   Option D Answer: B QUESTION 183Drag and Drop QuestionYou administer a Microsoft SQL Server 2008 database that includes a table named Products.The Products table has the following schema:

```
CREATE TABLE dbo.Pr
(
    ProductID nchar(4)
    ProductName nvarc
    ProductDescriptio
    UnitCost money NU
    ...
    CONSTRAINT PK_Pro
    (
        ProductID ASC
    )
)
GO
```

You create a User-Defined Function (UDF) in the same database.The UDF has the following schema:

```
CREATE FUNCTION dbo.CalculateProductProfit(@ProductID nchar(4))
RETURNS Money
WITH SCHEMABINDING
A
B
    SELECT @Profit = UnitPrice - UnitCost FRO dbo.Produc WHERE Produc
    RETURN @Profit;
END
```

You need to meet the following requirements:Which three Transact-SQL statements should you use? (To answer, move the appropriate statements from the list of statements to the answer area and arrange them in the correct order.

Answer:

QUESTION 184You need to identify which products will be inserted when you execute the following code block.Which products will be inserted?

```
BEGIN TRANSACTION
    INSERT INTO Product (ProductName) VALUES ('food')
    ...
    COMMIT TRANSACTION
ROLLBACK TRANSACTION
```

A.   BeverageB.   Food and beverageC.   NoneD.   Food Answer: C QUESTION 185You develop a new stored procedure for an

existing database. You create two tables named Customer and Orders.The tables have the following definitions:

```
CREATE TABLE Customer
(CustomerID int NOT NULL PRIMARY KEY
CustomerName nvarchar(255) NOT NULL,
CustomerAddress nvarchar(1024) NOT
CREATE TABLE Orders
(OrderID int NOT NULL PRIMARY KEY CLU
CustomerID int NOT NULL FOREIGN KEY
OrderDetails nvarchar(MAX))
```

Users are restricted from accessing table objects directly. You need to ensure that users are able to retrieve customer data.You need to create a stored procedure that meets the following requirements:Which Transact-SQL statement or statements should you use?

```
C A.   INSERT INT
       (4, 'sale
       (5, 'supp
       (6, 'supp

C B.   INSERT INT
       (4, 'supp
       (5, 'supp
       (6, 'supp

C C.   INSERT IN
       (6, 'supp

C D.   CREATE PRO
       AS

       SELECT c.C
       FROM Custo
       INNER JOIN
          ON c.Cus
       GROUP BY c

       GO
```

A.　Option AB.　Option BC.　Option CD.　Option D Answer: B QUESTION 186Drag and Drop QuestionYou administer a Microsoft SQL Server 2008 database that contains a table named Sales.SalesOrderHeader.The Sales.SalesOrderHeader table has the following definition:

```
CREATE TABLE [Sales].[SalesOrderHeader](
    [SalesOrderID] [int] IDENTITY(1,1) NOT NULL,
    [OrderDate] [datetime] NOT NULL,
    [DueDate] [datetime] NOT NULL,
    [ShipDate] [datetime] NULL,
    [Status] [tinyint] NOT NULL,
    [CustomerID] [int] NOT NULL,
    [SalesPersonID] [int] NULL,
    [SubTotal] [money] NOT NULL,
    [TaxAmt] [money] NOT NULL,
    CONSTRAINT [PK_OrderHeader_SalesOrderID]
    PRIMARY KEY CLUSTERED
    (
        [SalesOrderID] ASC
    )
) ON [PRIMARY]
GO
```

You want to generate an execution plan of XML output document for a query that displays all Sales.SalesOrderHeader records containing orders that have not been shipped.You need to ensure that the following requirements are met:Which three Transact-SQL statements should you use? (To answer, move the appropriate statements from the list of statements to the answer area and arrange them in the correct order.)

```
SET SHOWPLAN_XML ON;
GO

SET SHOWPLAN_XML OFF;
GO

SELECT *
FROM Sales.SalesOrderHeader
WHERE ShipDate = NULL
GO

GRANT SHOWPLAN XML ON;
GO

GRANT SHOWPLAN XML OFF;
GO

SET STATISTICS XML ON;
GO

SET STATISTICS XML OFF;
GO
```

Answer:

QUESTION 187You administer a Microsoft SQL Server 2008 database named AdventureWork that contains a table named Production.Product.The table has the following definition:

```
CREATE TABLE [Production].[Product](
    [ProductID] [int] IDENTITY(1,1) NOT NULL,
    [ProductName] [nvarchar](50) NOT NULL,
    [ListPrice] [money] NOT NULL,
    [ProductionDate] [datetime] NULL,
    [UserCreated] [nvarchar](128) NOT NULL,
    [DateCreated] [datetime] NOT NULL DEFAULT GETDATE(),
    CONSTRAINT [PK_Product_ProductID] PRIMARY KEY CLUSTERED
(
    [ProductID] ASC
) ON [PRIMARY]
) ON [PRIMARY]
GO
```

You want to add a new product named Widget and a list price of 10.50U.S. dollars to the product table.You need to add a record for the product information. You also need to set the DateCreated field to the current date and the UserCreated field to your Windows login identification name.Which Transact-SQL statement should you use?

A.
```
INSERT INTO [Production].[Product]
    (ProductName, ListPrice, DateCreated)
    values
    ('Widget', 10.50, GETDATE())
```

B.
```
INSERT INTO [Production].[Product]
    (ProductID, ProductName, ListPrice, UserCreated, DateCreated)
    select 1, 'Widget', 10.50, system_user, GETDATE()
```

C.
```
INSERT INTO [Production].[Product]
    (ProductName, ListPrice, UserCreated, ProductionDate)
    select 'Widget', 10.50, user_id(), GETDATE()
```

D.
```
INSERT INTO [Production].[Product]
    (ProductName, ListPrice, UserCreated)
    values
    ('Widget', 10.50, system_user)
```

A. Option AB. Option BC. Option CD. Option D Answer: B QUESTION 188You administer a Microsoft SQL Server 2008 R2 database that has a table named Customer.The table has the following definition:

```
CREATE TABLE Customer
    (CustomerID int NOT NULL PRIMARY KEY,
    FirstName varchar(255) NOT NULL,
    LastName varchar(255) NOT NULL,
    CustomerAddress varchar(1024))
```

The database also has a table named PreferredCustomerList. Data will be added to the PreferredCustomerList table regularly.The PreferredCustomerList table has the following definition:

```
CREATE TABLE PreferredCustomerList
    (CustomerID int NOT NULL,
    LastName varchar(255) NOT NULL)
```

You need to create a view that returns all records and columns of the Customer table that are also present in the PreferredCustomerList table.Which Transact-SQL statement should you use?

```
⊙ A.    CREATE VIEW vw_ValidCustomer
        AS
        SELECT c.CustomerID,
          c.FirstName,
          c.LastName,
          c.CustomerAddress
        FROM Customer c
        LEFT OUTER JOIN PreferredCustomerList cel
          ON c.Firstname = cel.FirstName
          AND c.LastName = cel.LastName
        WHERE cel.LastName IS NULL

⊙ B.    CREATE VIEW vw_ValidCustomer
        AS
        SELECT c.CustomerID,
          c.FirstName,
          c.LastName,
          c.CustomerAddress
        FROM Customer c
        INTERSECT
        SELECT c.CustomerID,
          c.FirstName,
          c.LastName,
          c.CustomerAddress
        FROM Customer c
        INNER JOIN PreferredCustomerList cel
          ON c.Firstname = cel.FirstName
          AND c.LastName = cel.LastName

⊙ C.    CREATE VIEW vw_ValidCustomer
        AS
        SELECT c.CustomerID,
          c.FirstName,
          c.LastName,
          c.CustomerAddress
        FROM Customer c
        LEFT OUTER JOIN PreferredCustomerList cel
          ON c.Firstname = cel.FirstName
          AND c.LastName = cel.LastName
        WHERE cel.FirstName IS NULL

⊙ D.    CREATE VIEW vw_ValidCustomer
        AS
        SELECT FirstName,
          LastName
        FROM Customer c
        EXCEPT
        SELECT FirstName,
          LastName
        FROM PreferredCustomerList
```

A.   Option AB.   Option BC.   Option CD.   Option D Answer: B QUESTION 189You administer a Microsoft SQL Server 2008 R2 instance configured to use Windows Authentication.The database contains a table named CustomerTransaction that has the following definition:

```
CREATE TABLE dbo.CustomerTransaction
(CustomerTransactionId int NOT NULL PRIMARY KEY,
 ...
 TransactionAmount money NOT NULL )
```

You define the following table:

```
CREATE TABLE dbo.CustomerWarningLog(
  CustomerWarningLogId int NOT NULL identity PRIMARY KEY,
  ...
  LogTime datetime2(0) NOT NULL,
  LogUserName nvarchar(128) NOT NULL)
```

You need to ensure that the following requirements are met:Which Transact-SQL statement or statements should you use?

```
A.  CREATE TRIGGER dbo.CustomerTransaction_InsertUpdateTrigger
    ON dbo.CustomerTransaction
    FOR INSERT, UPDATE
    AS

    IF UPDATE(CustomerID) or UPDATE(TransactionAmount)
       WITH TotalCTE AS
          (SELECT CustomerId, SUM(TransactionAmount) AS Balance
       FROM dbo.CustomerTransaction
       WHERE CustomerId in (SELECT CustomerId
          FROM inserted
          UNION ALL
          SELECT CustomerId
          FROM deleted)
       GROUP BY CustomerId)
       INSERT dbo. CustomerWarningLog(
          CustomerId, Balance, LogTime, LogUserName)
       SELECT CustomerId, Balance, SYSDATETIME(), USER_NAME()
       FROM TotalForCustomer
       WHERE TotalCTE.Amount < 100
```

```
B.  CREATE TRIGGER dbo.CustomerTransaction_InsertUpdateDeleteTrigger
    ON dbo.CustomerTransaction
    FOR INSERT, UPDATE, DELETE
    AS

    WITH TotalCTE AS
    (SELECT CustomerId, SUM(TransactionAmount) AS Balance
    FROM dbo.CustomerTransaction
       WHERE CustomerId in (SELECT CustomerId
          FROM inserted
       )
       GROUP BY CustomerId)
       INSERT dbo. CustomerWarningLog(
          CustomerId, Balance, LogTime, LogUserName)
       SELECT CustomerId, Balance, SYSDATETIME(), SUSER_SNAME()
       FROM TotalForCustomer
       WHERE TotalCTE.Amount <= 100

    GO
```

```
C.  CREATE TRIGGER dbo.CustomerTransaction_InsertUpdateDeleteTrigger
    ON dbo.CustomerTransaction
    FOR INSERT, UPDATE, DELETE
    AS

    DECLARE @CustomerId int, @Balance money

    WITH TotalCTE AS
    (SELECT CustomerId, SUM(TransactionAmount) AS Balance
    FROM dbo.CustomerTransaction
    WHERE CustomerId in (SELECT CustomerId
       FROM inserted
       UNION ALL
       SELECT CustomerId
       FROM deleted)
    GROUP BY CustomerId)

    SELECT @CustomerId = CustomerId,
       @Balance = TransactionAmount
    FROM TotalForCustomer
    WHERE TotalCTE.Amount < 100

    IF @CustomerId IS NULL
    INSERT dbo. CustomerWarningLog(

    GO
```

```
D.  CREATE TRIGGER dbo.CustomerTransaction_InsertUpdateDeleteTrigger
    ON dbo.CustomerTransaction
    FOR INSERT, UPDATE, DELETE
    AS

    IF UPDATE(CustomerID) or UPDATE(TransactionAmount)
       WITH TotalCTE AS
       (SELECT CustomerId, SUM(TransactionAmount) AS Balance
       FROM dbo.CustomerTransaction
       WHERE CustomerId in (SELECT CustomerId
          FROM inserted
          UNION ALL
          SELECT CustomerId
          FROM deleted)
          GROUP BY CustomerId)
          INSERT dbo. CustomerWarningLog(
             CustomerId, Balance, LogTime, LogUserName)
          SELECT CustomerId, Balance, SYSDATETIME(), SUSER_SNAME()
          FROM TotalForCustomer
          WHERE TotalCTE.Amount < 100

    GO
```

A.   Option AB.   Option BC.   Option CD.   Option D Answer: D QUESTION 190Drag and Drop QuestionYou administer a database named Contoso running on a Microsoft SQL Server 2008 R2 instance.You plan to implement custom error handling in your application.You need to implement custom error handling that meets the following requirements:Which three Transact-SQL statements should you use? (To answer, move the appropriate statements from the list of statements to the answer area and arrange them in the correct order.)

```
DECLARE @ProcessID INT;
DECLARE @ProcessName NVACHAR(150);
SET @ProcessID = @@SPID;
SET @ProcessName = DB_NAME();

DECLARE @ProcessID INT;
DECLARE @ProcessName INT;
SET @ProcessID = @@SPID;
SET @ProcessName = @@DB_NAME;

EXECUTE sp_addmessage 50025, 10, N'Session
ID: %d returned an error in the database: %
s.'

EXEC
ID:
s.'

RAISERROR (50000, @ProcessID, @ProcessName,
10, 1)

RAISERROR (50025, 10, 1, @ProcessID,
@ProcessName)

RAISERROR (50000, @ProcessID, @ProcessName,
14, 1)

RAISERROR(50025, 10, 1, @@SPID, DB_NAME() )
```

Answer:

Thanks For Trying Braindump2go Latest Microsoft 70-433 Dumps Questions! Braindump2go Exam Dumps ADVANTAGES:? 100% Pass Guaranteed Or Full Money Back!? Instant Download Access After Payment!? One Year Free Updation!? Well Formated: PDF,VCE,Exam Software!? Multi-Platform capabilities ? Windows, Laptop, Mac, Android, iPhone, iPod, iPad.? Professional, Quick,Patient IT Expert Team 24/7/3210 Onlinen Help You!? We served more than 35,000 customers all around the world in last 5 years with 98.99% PASS RATE!? Guaranteed Secure Shopping! Your Transcations are protected by Braindump2go all the time!? Pass any exams at the FIRST try!

Compared Before Buying Microsoft 70-433 PDF & VCE!		
Pass4sure	**Braindump2go** 100% Pass OR Money Back	**Test King**
202 Q&As - Practice	210 Q&As – Real Questions	202 Q&As - Practice
$125.99	$99.99	$124.99
No Discount	Coupon Code: BDNT2014	No Discount

http://www.braindump2go.com/70-433.html