



Guarantee All Exams 100% Pass One Time!

# **2018 NEW Microsoft 70-761: Querying Data with Transact-SQL Exam Questions and Answers RELEASED in Braindump2go.com Online IT Study Website Today!**

**2018 Braindump2go Valid Microsoft 70-761 Exam Preparation Materials:**

2018 Latest **135Q&As** 70-761 PDF Dumps and VCE Dumps:

<https://www.braindump2go.com/70-761.html>

## **QUESTION 52**

Note: This question is part of a series of questions that use the same scenario. For your convenience, the scenario is repeated in each question. Each question presents a different goal and answer choices, but the text of the scenario is exactly the same in each question in this series.

Drag and Drop Question

You are developing a database to track customer orders.

The database contains the following tables:

Sales.Customers, Sales.Orders, and Sales.OrderLines.

The following table describes the columns in Sales.Customers.

Column name	Data type	Constraints
CustomerID	int	primary key
CustomerName	nvarchar(100)	does not allow null values
PhoneNumber	nvarchar(20)	does not allow null values
AccountOpenedDate	date	does not allow null values
StandardDiscountPercentage	decimal(18,3)	does not allow null values
CreditLimit	decimal(18,2)	null values are permitted
IsOnCreditHold	bit	does not allow null values
DeliveryLocation	geography	does not allow null values
PhoneNumber	nvarchar(20)	does not allow null values

The following table describes the columns in Sales.Orders.

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

<https://www.braindump2go.com/70-761.html>

Column name	Data type	Constraints
OrderID	int	primary key
CustomerID	int	foreign key to the Sales.Customers table
OrderDate	date	does not allow null values

The following table describes the columns in Sales.OrderLines.

Column name	Data type	Constraints
OrderLineID	int	primary key
OrderID	int	foreign key to the Sales.Orders table
Quantity	int	does not allow null values
UnitPrice	decimal(18,2)	null values are permitted
TaxRate	decimal(18,3)	does not allow null values

You need to create a stored procedure that inserts data into the Customers table.

The stored procedure must meet the following requirements:

- Data changes occur as a single unit of work.
- Data modifications that are successful are committed and a value of 0 is returned.
- Data modifications that are unsuccessful are rolled back. The exception severity level is set to 16 and a value of -1 is returned.
- The stored procedure uses a built-in scalar function to evaluate the current condition of data modifications.
- The entire unit of work is terminated and rolled back if a run-time error occurs during execution of the stored procedure.

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

NOTE: Each correct selection is worth one point.

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

**Transact-SQL segments**

RAISERROR
THROW
XACT_ABORT
XACT_STATE
@@TRANCOUNT
ROLLBACK
COMMIT
END

**Answer Area**

```
CREATE PROCEDURE Sales.InsertCustomer
    @CustomerName nvarchar(100),
    @PhoneNumber nvarchar(20),
    @AccountOpenedDate date,
    @StandardDiscountPercentage decimal(18,3),
    @CreditLimit decimal(18,2),
    @IsCreditOnHold bit,
    @DeliveryLongitude nvarchar(50),
    @DeliveryLatitude nvarchar(50)
AS
BEGIN
    SET NOCOUNT ON
    SET  ON

    BEGIN TRY
        BEGIN TRANSACTION
        INSERT INTO Sales.Customers (CustomerName, PhoneNumber, AccountOpenedDate,
            StandardDiscountPercentage, CreditLimit, IsOnCreditHold, DeliveryLocation)
        VALUES
            (@CustomerName, @PhoneNumber, @AccountOpenedDate, @StandardDiscountPercentage,
            @CreditLimit, @IsCreditOnHold, geography::Point(ISNULL(@DeliveryLongitude, ''),
            ISNULL(@DeliveryLatitude, ''), 4326))

         TRANSACTION
    END TRY
    BEGIN CATCH
        IF  (<> 0)  TRANSACTION
        PRINT 'Unable to create the customer record.'
        
        RETURN -1
    END CATCH
    RETURN 0
END
```

**Answer:**

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

#### Transact-SQL segments

RAISERROR  
THROW  
XACT\_ABORT  
XACT\_STATE  
@@TRANCOUNT  
ROLLBACK  
COMMIT  
END

#### Answer Area

```
CREATE PROCEDURE Sales.InsertCustomer
    @CustomerName nvarchar(100),
    @PhoneNumber nvarchar(20),
    @AccountOpenedDate date,
    @StandardDiscountPercentage decimal(18,3),
    @CreditLimit decimal(18,2),
    @IsCreditOnHold bit,
    @DeliveryLongitude nvarchar(50),
    @DeliveryLatitude nvarchar(50)
AS
BEGIN
    SET NOCOUNT ON
    SET XACT_ABORT ON

    BEGIN TRY
        BEGIN TRANSACTION
        INSERT INTO Sales.Customers (CustomerName, PhoneNumber, AccountOpenedDate,
            StandardDiscountPercentage, CreditLimit, IsOnCreditHold, DeliveryLocation)
        VALUES
            (@CustomerName, @PhoneNumber, @AccountOpenedDate, @StandardDiscountPercentage,
            @CreditLimit, @IsCreditOnHold, geography::Point(ISNULL(@DeliveryLongitude, ''), 4326))
        COMMIT TRANSACTION
    END TRY
    BEGIN CATCH
        IF XACT_STATE() <> 0 ROLLBACK TRANSACTION
        PRINT 'Unable to create the customer record.'
        THROW
        RETURN -1
    END CATCH
    RETURN 0
END
```

#### Explanation:

Box 1: XACT\_ABORT

XACT\_ABORT specifies whether SQL Server automatically rolls back the current transaction when a Transact-SQL statement raises a run-time error. When SET XACT\_ABORT is ON, if a Transact-SQL statement raises a run-time error, the entire transaction is terminated and rolled back.

Box 2: COMMIT

Commit the transaction.

Box 3: XACT\_STATE

Box 4: ROLLBACK

Rollback the transaction

Box 5: THROW

THROW raises an exception and the severity is set to 16. Requirement: Data modifications that are unsuccessful are rolled back. The exception severity level is set to 16 and a value of -1 is returned.

References:

<https://msdn.microsoft.com/en-us/library/ms188792.aspx>

<https://msdn.microsoft.com/en-us/library/ee677615.aspx>

#### QUESTION 53

**Note:** This question is part of a series of questions that use the same scenario. For your convenience, the scenario is repeated in each question. Each question presents a different goal and answer choices, but the text of the scenario is exactly the same in each question in this series.

Drag and Drop Question

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE**

**Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

You are developing a database to track customer orders.

The database contains the following tables:

Sales.Customers, Sales.Orders, and Sales.OrderLines.

The following table describes the columns in Sales.Customers.

Column name	Data type	Constraints
CustomerID	int	primary key
CustomerName	nvarchar(100)	does not allow null values
PhoneNumber	nvarchar(20)	does not allow null values
AccountOpenedDate	date	does not allow null values
StandardDiscountPercentage	decimal(18,3)	does not allow null values
CreditLimit	decimal(18,2)	null values are permitted
IsOnCreditHold	bit	does not allow null values
DeliveryLocation	geography	does not allow null values
PhoneNumber	nvarchar(20)	does not allow null values

The following table describes the columns in Sales.Orders.

Column name	Data type	Constraints
OrderID	int	primary key
CustomerID	int	foreign key to the Sales.Customers table
OrderDate	date	does not allow null values

The following table describes the columns in Sales.OrderLines.

Column name	Data type	Constraints
OrderLineID	int	primary key
OrderID	int	foreign key to the Sales.Orders table
Quantity	int	does not allow null values
UnitPrice	decimal(18,2)	null values are permitted
TaxRate	decimal(18,3)	does not allow null values

You need to create a function that calculates the highest tax rate charged for an item in a specific order. Which five Transact-SQL segments should you use to develop the solution? To answer, move the appropriate Transact-SQL segments from the list of Transact-SQL segments to the answer area and arrange them in the correct order.

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

**Transact-SQL segments**

```
RETURNS decimal(18,2)

CREATE FUNCTION Sales.CalculateTaxRate ()

CREATE FUNCTION Sales.CalculateTaxRate (
    @OrderID int
)

RETURN @CalculatedRate
END

SET @CalculatedTaxRate = (
    SELECT 1 + (MAX(TaxRate)
        / 100)
    FROM Sales.OrderLines
    WHERE OrderID = @OrderID

RETURNS Table
END

AS
BEGIN
declare @CalculatedTaxRate
decimal(18,2)
```

**Answer Area**



**Answer:**

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**



**Transact-SQL segments**

```
CREATE FUNCTION Sales.CalculateTaxRate ()
```

```
RETURNS Table  
END
```

**Answer Area**

```
CREATE FUNCTION Sales.CalculateTaxRate (  
    @OrderID int  
)
```

```
RETURNS decimal(18,2)
```

```
AS  
BEGIN  
    declare @CalculatedTaxRate  
    decimal(18,2)
```

```
SET @CalculatedTaxRate = (  
    SELECT 1 + (MAX(TaxRate)  
        / 100)  
    FROM Sales.OrderLines  
    WHERE OrderID = @OrderID
```

```
RETURN @CalculatedRate  
END
```

**Explanation:**

Box 1: CREATE FUNCTION...@OrderID

Include definition for the ...@OrderID parameter.

Box 2: RETURNS decimal(18,2)

The function is defined to return a scalar value.

Box 3: AS BEGIN ...

Declare the local variables of the function.

Box 4: SET @CalculatedTaxRate = (..

Calculate the tax rate.

Box 5: RETURN @CalculatedRate END

Return a scalar value.

References: <https://msdn.microsoft.com/en-us/library/ms186755.aspx>

**QUESTION 54**

**Note:** This question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other questions in this series. Information and details provided in a question apply only to that question.

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

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

```
CREATE TABLE Customers (  
    CustomerID int NOT NULL PRIMARY KEY CLUSTERED,  
    FirstName nvarchar(100) NOT NULL,  
    LastName nvarchar(100) NOT NULL,  
    TaxIdNumber varchar(20) NOT NULL,  
    Address nvarchar(1024) NOT NULL,  
    AnnualRevenue decimal(19,2) NOT NULL,  
    DateCreated datetime2(2) NOT NULL,  
    ValidFrom datetime2(2) GENERATED ALWAYS AS ROW START NOT NULL,  
    ValidTo datetime2(2) GENERATED ALWAYS AS ROW END NOT NULL,  
    PERIOD FOR SYSTEM_TIME(ValidFrom, ValidTo)  
)  
WITH (SYSTEM_VERSIONING = ON (HISTORY_TABLE = CustomersHistory))
```

You need to audit all customer data.  
Which Transact-SQL statement should you run?

- A 

```
SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, AnnualRevenue, DateCreated  
FROM Customers  
GROUP BY GROUPING SETS(FirstName, LastName), (Address), (CustomerID, AnnualRevenue), (CustomerID), ( )  
ORDER BY CustomerID, FirstName, LastName, Address, AnnualRevenue
```
- B 

```
SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, AnnualRevenue, DateCreated, ValidFrom, ValidTo  
FROM Customers  
FOR SYSTEM_TIME ALL ORDER BY ValidFrom
```
- C 

```
SELECT c.CustomerID, c.FirstName, c.LastName, c.Address, c.ValidFrom, c.ValidTo  
FROM Customers AS c  
ORDER BY c.CustomerID  
FOR JSON AUTO, ROOT('Customers')
```
- D 

```
SELECT * FROM (SELECT CustomerID, FirstName, LastName, Address, AnnualRevenue, DateCreated  
FROM Customers) AS Customers PIVOT(AVG(AnnualRevenue)  
FOR DateCreated IN([2014])) AS PivotCustomers  
ORDER BY LastName, FirstName
```
- E 

```
SELECT CustomerID, AVG(AnnualRevenue)  
AS AverageAnnualRevenue, FirstName, LastName, Address, DateCreated  
FROM Customers WHERE YEAR(DateCreated) >= 2014  
GROUP BY CustomerID, FirstName, LastName, Address, DateCreated
```
- F 

```
SELECT c.CustomerID, c.FirstName, c.LastName, c.Address, c.ValidFrom, c.ValidTo  
FROM Customers AS c ORDER BY c.CustomerID  
FOR XML PATH ('CustomerData'), root ('Customers')
```
- G 

```
SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, ValidFrom, ValidTo  
FROM Customers FOR SYSTEM_TIME  
BETWEEN '2014-01-01 00:00:00.000000' AND '2015-01-01 00:00:00.000000'
```
- H 

```
SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, ValidFrom, ValidTo  
FROM Customers  
WHERE DateCreated  
BETWEEN '20140101' AND '20141231'
```

A. Option A

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE  
Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**



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

**Answer: B**

**Explanation:**

The FOR SYSTEM\_TIME ALL clause returns all the row versions from both the Temporal and History table.

Note: A system-versioned temporal table defined through is a new type of user table in SQL Server 2016, here defined on the last line WITH (SYSTEM\_VERSIONING = ON..., is designed to keep a full history of data changes and allow easy point in time analysis.

To query temporal data, the SELECT statement FROM<table> clause has a new clause FOR SYSTEM\_TIME with five temporal-specific sub-clauses to query data across the current and history tables.

References: <https://msdn.microsoft.com/en-us/library/dn935015.aspx>

**QUESTION 55**

**Note: This question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other questions in this series. Information and details provided in a question apply only to that question.**

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

```
CREATE TABLE Customers (  
    CustomerID int NOT NULL PRIMARY KEY CLUSTERED,  
    FirstName nvarchar(100) NOT NULL,  
    LastName nvarchar(100) NOT NULL,  
    TaxIdNumber varchar(20) NOT NULL,  
    Address nvarchar(1024) NOT NULL,  
    AnnualRevenue decimal(19,2) NOT NULL,  
    DateCreated datetime2(2) NOT NULL,  
    ValidFrom datetime2(2) GENERATED ALWAYS AS ROW START NOT NULL,  
    ValidTo datetime2(2) GENERATED ALWAYS AS ROW END NOT NULL,  
    PERIOD FOR SYSTEM_TIME(ValidFrom, ValidTo)  
)  
WITH (SYSTEM_VERSIONING = ON (HISTORY_TABLE = CustomersHistory))
```

You need to return normalized data for all customers that were added in the year 2014.

Which Transact-SQL statement should you run?

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

- A `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, AnnualRevenue, DateCreated  
FROM Customers  
GROUP BY GROUPING SETS(FirstName, LastName), (Address), (CustomerID, AnnualRevenue), (CustomerID), (}  
ORDER BY CustomerID, FirstName, LastName, Address, AnnualRevenue`
- B `SELECT FirstName, LastName, Address  
FROM Customers  
FOR SYSTEM_TIME ALL ORDER BY ValidFrom`
- C `SELECT c.CustomerID, c.FirstName, c.LastName, c.Address, c.ValidFrom, c.ValidTo  
FROM Customers AS c  
ORDER BY c.CustomerID  
FOR JSON AUTO, ROOT('Customers')`
- D `SELECT * FROM (SELECT CustomerID, FirstName, LastName, Address, AnnualRevenue, DateCreated  
FROM Customers) AS Customers PIVOT(AVG(AnnualRevenue)  
FOR DateCreated IN([2014])) AS PivotCustomers  
ORDER BY LastName, FirstName`
- E `SELECT CustomerID, AVG(AnnualRevenue)  
AS AverageAnnualRevenue, FirstName, LastName, Address, DateCreated  
FROM Customers WHERE YEAR(DateCreated) >= 2014  
GROUP BY CustomerID, FirstName, LastName, Address, DateCreated`
- F `SELECT c.CustomerID, c.FirstName, c.LastName, c.Address, c.ValidFrom, c.ValidTo  
FROM Customers AS c ORDER BY c.CustomerID  
FOR XML PATH ('CustomerData'), root ('Customers')`
- G `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, ValidFrom, ValidTo  
FROM Customers FOR SYSTEM_TIME  
BETWEEN '2014-01-01 00:00:00.000000' AND '2015-01-01 00:00:00.000000'`
- H `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, ValidFrom, ValidTo  
FROM Customers  
WHERE DateCreated  
BETWEEN '20140101' AND '20141231'`

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

**Answer: G**

**Explanation:**

The following query searches for row versions for Employee row with EmployeeID = 1000 that were active at least for a portion of period between 1st January of 2014 and 1st January 2015 (including the upper boundary):

`SELECT * FROM Employee`

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE**

**Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

FOR SYSTEM\_TIME  
BETWEEN '2014-01-01 00:00:00.0000000' AND '2015-01-01 00:00:00.0000000' WHERE  
EmployeeID = 1000 ORDER BY ValidFrom;  
References: <https://msdn.microsoft.com/en-us/library/dn935015.aspx>

**QUESTION 56**

**Note:** This question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other questions in this series. Information and details provided in a question apply only to that question.

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

```
CREATE TABLE Customers (  
    CustomerID int NOT NULL PRIMARY KEY CLUSTERED,  
    FirstName nvarchar(100) NOT NULL,  
    LastName nvarchar(100) NOT NULL,  
    TaxIdNumber varchar(20) NOT NULL,  
    Address nvarchar(1024) NOT NULL,  
    AnnualRevenue decimal(19,2) NOT NULL,  
    DateCreated datetime2(2) NOT NULL,  
    ValidFrom datetime2(2) GENERATED ALWAYS AS ROW START NOT NULL,  
    ValidTo datetime2(2) GENERATED ALWAYS AS ROW END NOT NULL,  
    PERIOD FOR SYSTEM_TIME(ValidFrom, ValidTo)  
)  
WITH (SYSTEM_VERSIONING = ON (HISTORY_TABLE = CustomersHistory))
```

You are developing a report that displays customer information.

The report must contain a grand total column.

You need to write a query that returns the data for the report.

Which Transact-SQL statement should you run?

- A 

```
SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, AnnualRevenue, DateCreated  
FROM Customers  
GROUP BY GROUPING SETS(FirstName, LastName), (Address), (CustomerID, AnnualRevenue), (CustomerID), (}  
ORDER BY CustomerID, FirstName, LastName, Address, AnnualRevenue
```
- B 

```
SELECT FirstName, LastName, Address  
FROM Customers  
FOR SYSTEM_TIME ALL ORDER BY ValidFrom
```
- C 

```
SELECT c.CustomerID, c.FirstName, c.LastName, c.Address, c.ValidFrom, c.ValidTo  
FROM Customers AS c  
ORDER BY c.CustomerID  
FOR JSON AUTO, ROOT('Customers')
```
- D 

```
SELECT * FROM (SELECT CustomerID, FirstName, LastName, Address, AnnualRevenue, DateCreated  
FROM Customers) AS Customers PIVOT(AVG(AnnualRevenue)  
FOR DateCreated IN([2014])) AS PivotCustomers  
ORDER BY LastName, FirstName
```
- E 

```
SELECT CustomerID, AVG(AnnualRevenue)  
AS AverageAnnualRevenue, FirstName, LastName, Address, DateCreated  
FROM Customers WHERE YEAR(DateCreated) >= 2014  
GROUP BY CustomerID, FirstName, LastName, Address, DateCreated
```

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE  
Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

- F `SELECT c.CustomerID, c.FirstName, c.LastName, c.Address, c.ValidFrom, c.ValidTo  
FROM Customers AS c ORDER BY c.CustomerID  
FOR XML PATH ('CustomerData'), root ('Customers')`
- G `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, ValidFrom, ValidTo  
FROM Customers FOR SYSTEM_TIME  
BETWEEN '2014-01-01 00:00:00.000000' AND '2015-01-01 00:00:00.000000'`
- H `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, ValidFrom, ValidTo  
FROM Customers  
WHERE DateCreated  
BETWEEN '20140101' AND '20141231'`
- A. Option A  
B. Option B  
C. Option C  
D. Option D  
E. Option E  
F. Option F  
G. Option G  
H. Option H

**Answer:** A

**Explanation:**

Calculate aggregate column through AVG function and GROUP BY clause.

#### QUESTION 57

**Note:** This question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other questions in this series. Information and details provided in a question apply only to that question.

You create a table named Customers. Data stored in the table must be exchanged between web pages and web servers by using AJAX calls that use REST endpoint.

You need to return all customer information by using a data exchange format that is text-based and lightweight.

Which Transact-SQL statement should you run?

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE  
Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

- A `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, AnnualRevenue, DateCreated  
FROM Customers  
GROUP BY GROUPING SETS(FirstName, LastName), (Address), (CustomerID, AnnualRevenue), (CustomerID), ( )  
ORDER BY CustomerID, FirstName, LastName, Address, AnnualRevenue`
- B `SELECT FirstName, LastName, Address  
FROM Customers  
FOR SYSTEM_TIME ALL ORDER BY ValidFrom`
- C `SELECT c.CustomerID, c.FirstName, c.LastName, c.Address, c.ValidFrom, c.ValidTo  
FROM Customers AS c  
ORDER BY c.CustomerID  
FOR JSON AUTO, ROOT('Customers')`
- D `SELECT * FROM (SELECT CustomerID, FirstName, LastName, Address, AnnualRevenue, DateCreated  
FROM Customers) AS Customers PIVOT(AVG(AnnualRevenue)  
FOR DateCreated IN([2014])) AS PivotCustomers  
ORDER BY LastName, FirstName`
- E `SELECT CustomerID, AVG(AnnualRevenue)  
AS AverageAnnualRevenue, FirstName, LastName, Address, DateCreated  
FROM Customers WHERE YEAR(DateCreated) >= 2014  
GROUP BY CustomerID, FirstName, LastName, Address, DateCreated`
- F `SELECT c.CustomerID, c.FirstName, c.LastName, c.Address, c.ValidFrom, c.ValidTo  
FROM Customers AS c ORDER BY c.CustomerID  
FOR XML PATH ('CustomerData'), root ('Customers')`
- G `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, ValidFrom, ValidTo  
FROM Customers FOR SYSTEM_TIME  
BETWEEN '2014-01-01 00:00:00.000000' AND '2015-01-01 00:00:00.000000'`
- H `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, ValidFrom, ValidTo  
FROM Customers  
WHERE DateCreated  
BETWEEN '20140101' AND '20141231'`

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

**Answer: C**

**Explanation:**

JSON can be used to pass AJAX updates between the client and the server.

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE  
Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**



Export data from SQL Server as JSON, or format query results as JSON, by adding the FOR JSON clause to a SELECT statement.

When you use the FOR JSON clause, you can specify the structure of the output explicitly, or let the structure of the SELECT statement determine the output.

References: <https://msdn.microsoft.com/en-us/library/dn921882.aspx>

#### QUESTION 58

**Note:** This question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other questions in this series. Information and details provided in a question apply only to that question.

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

```
CREATE TABLE Customers (  
    CustomerID int NOT NULL PRIMARY KEY CLUSTERED,  
    FirstName nvarchar(100) NOT NULL,  
    LastName nvarchar(100) NOT NULL,  
    TaxIdNumber varchar(20) NOT NULL,  
    Address nvarchar(1024) NOT NULL,  
    AnnualRevenue decimal(19,2) NOT NULL,  
    DateCreated datetime2(2) NOT NULL,  
    ValidFrom datetime2(2) GENERATED ALWAYS AS ROW START NOT NULL,  
    ValidTo datetime2(2) GENERATED ALWAYS AS ROW END NOT NULL,  
    PERIOD FOR SYSTEM_TIME(ValidFrom, ValidTo)  
)  
WITH (SYSTEM_VERSIONING = ON (HISTORY_TABLE = CustomersHistory))
```

You are developing a report that aggregates customer data only for the year 2014.

The report requires that the data be denormalized.

You need to return the data for the report.

Which Transact-SQL statement should you run?

- A 

```
SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, AnnualRevenue, DateCreated  
FROM Customers  
GROUP BY GROUPING SETS(FirstName, LastName), (Address), (CustomerID, AnnualRevenue), (CustomerID), (())  
ORDER BY CustomerID, FirstName, LastName, Address, AnnualRevenue
```
- B 

```
SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, AnnualRevenue, DateCreated, ValidFrom, ValidTo  
FROM Customers  
FOR SYSTEM_TIME ALL ORDER BY ValidFrom
```
- C 

```
SELECT c.CustomerID, c.FirstName, c.LastName, c.Address, c.ValidFrom, c.ValidTo  
FROM Customers AS c  
ORDER BY c.CustomerID  
FOR JSON AUTO, ROOT('Customers')
```
- D 

```
SELECT * FROM (SELECT CustomerID, FirstName, LastName, Address, AnnualRevenue, DateCreated  
FROM Customers) AS Customers PIVOT(AVG(AnnualRevenue)  
FOR DateCreated IN([2014])) AS PivotCustomers  
ORDER BY LastName, FirstName
```
- E 

```
SELECT CustomerID, AVG(AnnualRevenue)  
AS AverageAnnualRevenue, FirstName, LastName, Address, DateCreated  
FROM Customers WHERE YEAR(DateCreated) >= 2014  
GROUP BY CustomerID, FirstName, LastName, Address, DateCreated
```

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE**

**Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

- 'F' `SELECT c.CustomerID, c.FirstName, c.LastName, c.Address, c.ValidFrom, c.ValidTo  
FROM Customers AS c ORDER BY c.CustomerID  
FOR XML PATH ('CustomerData'), root ('Customers')`
- 'G' `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, ValidFrom, ValidTo  
FROM Customers FOR SYSTEM_TIME  
BETWEEN '2014-01-01 00:00:00.000000' AND '2015-01-01 00:00:00.000000'`
- 'H' `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, ValidFrom, ValidTo  
FROM Customers  
WHERE DateCreated  
BETWEEN '20140101' AND '20141231'`
- A. Option A  
B. Option B  
C. Option C  
D. Option D  
E. Option E  
F. Option F  
G. Option G  
H. Option H

**Answer: G**

#### QUESTION 59

**Note:** This question is part of a series of questions that use the same or similar answer choices. An answer choice may be correct for more than one question in the series. Each question is independent of the other questions in this series. Information and details provided in a question apply only to that question.

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

```
CREATE TABLE Customers (  
    CustomerID int NOT NULL PRIMARY KEY CLUSTERED,  
    FirstName nvarchar(100) NOT NULL,  
    LastName nvarchar(100) NOT NULL,  
    TaxIdNumber varchar(20) NOT NULL,  
    Address nvarchar(1024) NOT NULL,  
    AnnualRevenue decimal(19,2) NOT NULL,  
    DateCreated datetime2(2) NOT NULL,  
    ValidFrom datetime2(2) GENERATED ALWAYS AS ROW START NOT NULL,  
    ValidTo datetime2(2) GENERATED ALWAYS AS ROW END NOT NULL,  
    PERIOD FOR SYSTEM_TIME(ValidFrom, ValidTo)  
)  
WITH (SYSTEM_VERSIONING = ON (HISTORY_TABLE = CustomersHistory))
```

You need to develop a query that meets the following requirements:

- Output data by using a tree-like structure.
- Allow mixed content types.
- Use custom metadata attributes.

Which Transact-SQL statement should you run?

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE**

**Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

- A `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, AnnualRevenue, DateCreated  
FROM Customers  
GROUP BY GROUPING SETS(FirstName, LastName), (Address), (CustomerID, AnnualRevenue), (CustomerID), (}  
ORDER BY CustomerID, FirstName, LastName, Address, AnnualRevenue`
- B `SELECT FirstName, LastName, Address  
FROM Customers  
FOR SYSTEM_TIME ALL ORDER BY ValidFrom`
- C `SELECT c.CustomerID, c.FirstName, c.LastName, c.Address, c.ValidFrom, c.ValidTo  
FROM Customers AS c  
ORDER BY c.CustomerID  
FOR JSON AUTO, ROOT('Customers')`
- D `SELECT * FROM (SELECT CustomerID, FirstName, LastName, Address, AnnualRevenue, DateCreated  
FROM Customers) AS Customers PIVOT(AVG(AnnualRevenue)  
FOR DateCreated IN([2014])) AS PivotCustomers  
ORDER BY LastName, FirstName`
- E `SELECT CustomerID, AVG(AnnualRevenue)  
AS AverageAnnualRevenue, FirstName, LastName, Address, DateCreated  
FROM Customers WHERE YEAR(DateCreated) >= 2014  
GROUP BY CustomerID, FirstName, LastName, Address, DateCreated`
- F `SELECT c.CustomerID, c.FirstName, c.LastName, c.Address, c.ValidFrom, c.ValidTo  
FROM Customers AS c ORDER BY c.CustomerID  
FOR XML PATH ('CustomerData'), root ('Customers')`
- G `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, ValidFrom, ValidTo  
FROM Customers FOR SYSTEM_TIME  
BETWEEN '2014-01-01 00:00:00.000000' AND '2015-01-01 00:00:00.000000'`
- H `SELECT CustomerID, FirstName, LastName, TaxIdNumber, Address, ValidFrom, ValidTo  
FROM Customers  
WHERE DateCreated  
BETWEEN '20140101' AND '20141231'`
- A. Option A  
B. Option B  
C. Option C  
D. Option D  
E. Option E  
F. Option F  
G. Option G  
H. Option H

**Answer: F**

**Explanation:**

In a FOR XML clause, you specify one of these modes: RAW, AUTO, EXPLICIT, and PATH.

\* The EXPLICIT mode allows more control over the shape of the XML. You can mix attributes and elements at will in deciding the shape of the XML. It requires a specific format for the resulting

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE**

**Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

rowset that is generated because of query execution. This rowset format is then mapped into XML shape. The power of EXPLICIT mode is to mix attributes and elements at will, create wrappers and nested complex properties, create space-separated values (for example, OrderID attribute may have a list of order ID values), and mixed contents.

\* The PATH mode together with the nested FOR XML query capability provides the flexibility of the EXPLICIT mode in a simpler manner.

References: <https://msdn.microsoft.com/en-us/library/ms178107.aspx>

**QUESTION 60**

Your team is developing a database for a new online travel application.

You need to design tables and other database objects to support the application.

One particular table called Airline\_Schedules needs to store the departure and arrival dates and times of flights along with time zone information.

What should you do?

- A. Use the CAST function
- B. Use the DATETIMEOFFSET data type
- C. Use a user-defined table type
- D. Use the DATETIME2 data type

**Answer: B**

**Explanation:**

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

**QUESTION 61**

As part of a new enterprise project, you're designing a new table to store financial transactions.

This table could eventually store millions of rows and so storage space is very important.

One of the columns in the table will store either a 1 or 0 value. Which data type would be most appropriate?

- A. tinyint
- B. bit
- C. float
- D. numeric

**Answer: B**

**Explanation:**

bit is an integer data type that can take a value of 1, 0, or NULL.

**QUESTION 62**

You're creating a new query that will select rows from a products tables.

The query works out the count of products within each category by grouping on the category, filtering by categories that contain more than one product and then sorting the results in category order.

In which order should these clauses be used in the query?

- A. GROUP BY, HAVING, ORDER BY
- B. HAVING, GROUP BY, ORDER BY
- C. ORDER BY, GROUP BY, HAVING

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE**

**Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**

D. GROUP BY, ORDER BY, HAVING

**Answer: A**

**Free Download Braindump2go Latest 70-761 Exam PDF and VCE  
Dumps 135q from [www.braindump2go.com](http://www.braindump2go.com)**

**100% Pass Guaranteed! 100% Real Exam Questions!**

**<https://www.braindump2go.com/70-761.html>**