



70-761^{Q&As}

Querying Data with Transact-SQL

Pass Microsoft 70-761 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.pass4lead.com/70-761.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Microsoft
Official Exam Center

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers



**QUESTION 1**

DRAG DROP

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

You 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(18,2)	does not allow null values

The following table describes the columns in Sales.Orders. The following table describes the columns in Sales.OrderLines.

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 to the calling procedure.



Data modifications that are unsuccessful are rolled back. You must display a message that uses severity level 16 and a value of -1.

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.

Select and Place:

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

```

Correct Answer:



Transact-SQL segments

Answer Area

RAISERROR

@@TRANCOUNT

END

```

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, ''),
            ISNULL(@DeliveryLatitude, ''), 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

```

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 2

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

Information and details provided in a question apply only to that question.

You have a database that is denormalized. Users make frequent changes to data in a primary table.

You need to ensure that users cannot change the tables directly, and that changes made to the primary table also update any related tables.

What should you implement?

- A. the COALESCE function
- B. a view



- C. a table-valued function
- D. the TRY_PARSE function
- E. a stored procedure
- F. the ISNULL function
- G. a scalar function
- H. the TRY_CONVERT function

Correct Answer: B

Using an Indexed View would allow you to keep your base data in properly normalized tables and maintain data-integrity while giving you the denormalized "view" of that data. References: <http://stackoverflow.com/questions/4789091/updating-redundant-denormalized-data-automatically-in-sql-server>

QUESTION 3

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?



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

Correct Answer: C

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

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 4

HOTSPOT

You have the following subqueries: Subquery1, Subquery2, and Subquery3.

You need to replace the three subqueries with named result sets or temporary tables. The following requirements must



be met:

Subquery name	Requirements
Subquery1	The result set of this subquery must use the execution scope of a SELECT statement.
Subquery2	The result set of this subquery must be visible to other session users before disconnected.
Subquery3	The result set of this subquery must be accessible to other statements in the same session but must not be visible to other sessions.

Which replacement techniques should you use? To answer, select the appropriate options in the answer area.

Hot Area:

Answer Area

Subquery name

Subquery replacement

Subquery1

▼
common table expression (CTE)
local temporary table
global temporary table

Subquery2

▼
common table expression (CTE)
local temporary table
global temporary table

Subquery3

▼
common table expression (CTE)
local temporary table
global temporary table

Correct Answer:



Answer Area

Subquery name	Subquery replacement
Subquery1	<div style="border: 1px solid gray; padding: 5px;"> <div style="background-color: #f0f0f0; padding: 2px; display: flex; justify-content: space-between; align-items: center;"> ▼ </div> <div style="padding: 2px;"> <p>common table expression (CTE)</p> <p>local temporary table</p> <p>global temporary table</p> </div> </div>
Subquery2	<div style="border: 1px solid gray; padding: 5px;"> <div style="background-color: #f0f0f0; padding: 2px; display: flex; justify-content: space-between; align-items: center;"> ▼ </div> <div style="padding: 2px;"> <p>common table expression (CTE)</p> <p>local temporary table</p> <p>global temporary table</p> </div> </div>
Subquery3	<div style="border: 1px solid gray; padding: 5px;"> <div style="background-color: #f0f0f0; padding: 2px; display: flex; justify-content: space-between; align-items: center;"> ▼ </div> <div style="padding: 2px;"> <p>common table expression (CTE)</p> <p>local temporary table</p> <p>global temporary table</p> </div> </div>

Subquery1: common table expression (CTE)

A common table expression (CTE) can be thought of as a temporary result set that is defined within the execution scope of a single SELECT, INSERT, UPDATE, DELETE, or CREATE VIEW statement. A CTE is similar to a derived table in

that it is not stored as an object and lasts only for the duration of the query. Unlike a derived table, a CTE can be self-referencing and can be referenced multiple times in the same query.

Subquery2: global temporary table

Global temporary tables are visible to any user and any connection after they are created, and are deleted when all users that are referencing the table disconnect from the instance of SQL Server.

Subquery3: local temporary table

Local temporary tables are visible only to their creators during the same connection to an instance of SQL Server as when the tables were first created or referenced. Local temporary tables are deleted after the user disconnects from the

instance of SQL Server.

References:

[https://technet.microsoft.com/en-us/library/ms190766\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms190766(v=sql.105).aspx)



<https://technet.microsoft.com/en-us/library/ms186986.aspx>

QUESTION 5

You run the following Transact-SQL statements:

```
CREATE TABLE CourseParticipants
(
  CourseID INT NOT NULL,
  CourseDate DATE NOT NULL,
  LocationDescription VARCHAR(100) NOT NULL,
  NumParticipants INT NOT NULL
)
```

You need to create a query that returns the total number of attendees for each combination of CourseID, CourseDate, and the following locations: Lisbon, London, and Seattle. The result set should resemble the following:

	CourseID	CourseDate	Lisbon	London	Seattle
1	1	2018-02-01	NULL	NULL	15
2	2	2018-02-01	33	NULL	NULL
3	1	2018-02-02	NULL	20	NULL
4	1	2018-02-03	20	10	NULL
5	2	2018-02-03	NULL	20	NULL

Which Transact-SQL code segment should you run?

- A. `SELECT * FROM CourseParticipants PIVOT(SUM(NumParticipants) FOR LocationDescription IN (Lisbon, London, Seattle))`
- B. `SELECT * FROM CourseParticipants PIVOT(SUM(NumParticipants) FOR LocationDescription IN (Lisbon, London, Seattle)) as PVTTable`
- C. `SELECT * FROM CourseParticipants UNPIVOT(SUM(NumParticipants) FOR LocationDescription IN (Lisbon, London, Seattle))`
- D. `SELECT * FROM CourseParticipants UNPIVOT(SUM(NumParticipants) FOR LocationDescription IN (Lisbon, London, Seattle)) AS PVTTable`

Correct Answer: B

References: https://www.techonthenet.com/sql_server/pivot.php



To Read the [Whole Q&As](#), please purchase the [Complete Version](#) from [Our website](#).

Try our product !

100% Guaranteed Success

100% Money Back Guarantee

365 Days Free Update

Instant Download After Purchase

24x7 Customer Support

Average 99.9% Success Rate

More than 800,000 Satisfied Customers Worldwide

Multi-Platform capabilities - [Windows](#), [Mac](#), [Android](#), [iPhone](#), [iPod](#), [iPad](#), [Kindle](#)

We provide exam PDF and VCE of Cisco, Microsoft, IBM, CompTIA, Oracle and other IT Certifications. You can view Vendor list of All Certification Exams offered:

<https://www.pass4lead.com/allproducts>

Need Help

Please provide as much detail as possible so we can best assist you.

To update a previously submitted ticket:



 <p>One Year Free Update Free update is available within One Year after your purchase. After One Year, you will get 50% discounts for updating. And we are proud to boast a 24/7 efficient Customer Support system via Email.</p>	 <p>Money Back Guarantee To ensure that you are spending on quality products, we provide 100% money back guarantee for 30 days from the date of purchase.</p>	 <p>Security & Privacy We respect customer privacy. We use McAfee's security service to provide you with utmost security for your personal information & peace of mind.</p>
---	---	--

Any charges made through this site will appear as Global Simulators Limited.

All trademarks are the property of their respective owners.

Copyright © pass4lead, All Rights Reserved.