NEW QUESTION 1
You are creating a database solution to track sales achievements of your training courses. You run the following Transact-SQL statements:

```
CREATE TABLE Courses (  
    CourseID INT IDENTITY(1,1) NOT NULL,  
    Course VARCHAR(50) NULL,  
    TrainerRatingScore DECIMAL(18, 0) NULL
)
CREATE TABLE CourseParticipants (  
    CourseID INT NOT NULL,  
    CourseDate DATE NOT NULL,  
    LocationDescription VARCHAR(100) NOT NULL,  
    NumParticipants INT NOT NULL
)
```

You plan to add courses to table named Highlighted Courses. You must add courses that have been delivered to more than 100 participants only. If the total number of participants for a course is lower than 100, the course must not be added to the HighlightedCourses table. In addition, an error message must be displayed and remaining Transact-SQL code must not run.

```
DECLARE @CourseID INT  
DECLARE @TotalParticipants INT  
SET @CourseID = 1  
SET @TotalParticipants = (SELECT SUM(cp.NumParticipants)  
                    FROM Courses c INNER JOIN  
                    CourseParticipants cp ON c.CourseID = cp.CourseID  
                    WHERE c.CourseID = @CourseID  
                    GROUP BY c.CourseID)  
PRINT @TotalParticipants
BREAK
CONTINUE
RAISERROR('Course is not admissible', 16, 0)  
THROW 50000, 'Course is not admissible', 0
```

Answer:

```
DECLARE @CourseID INT  
DECLARE @TotalParticipants INT  
SET @CourseID = 1  
SET @TotalParticipants = (SELECT SUM(cp.NumParticipants)  
                    FROM Courses c INNER JOIN  
                    CourseParticipants cp ON c.CourseID = cp.CourseID  
                    WHERE c.CourseID = @CourseID  
                    GROUP BY c.CourseID)  
PRINT @TotalParticipants
BREAK
CONTINUE
RAISERROR('Course is not admissible', 16, 0)  
THROW 50000, 'Course is not admissible', 0
```

Explanation:

NEW QUESTION 2
Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the
stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution. After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You have a database that contains a single table named tblVehicleRegistration. The table is defined as follows:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VehicleId</td>
<td>int</td>
<td>the primary key for the table</td>
</tr>
<tr>
<td>RegistrationNumber</td>
<td>varchar(5)</td>
<td>a vehicle registration number that contains only letters and numbers</td>
</tr>
<tr>
<td>RegistrationDate</td>
<td>date</td>
<td>the vehicle registration date</td>
</tr>
<tr>
<td>UserId</td>
<td>int</td>
<td>an identifier for the vehicle owner</td>
</tr>
</tbody>
</table>

You run the following query:

```
SELECT UserId FROM tblVehicleRegistration
WHERE RegistrationNumber = '20012'
AND RegistrationDate > '2016-01-01'
```

The query output window displays the following error message: "Conversion failed when converting the varchar value ‘AB012’ to data type int."

You need to resolve the error.

Solution: You modify the Transact-SQL statement as follows:

```
SELECT UserId FROM tblVehicleRegistration
WHERE RegistrationNumber = CAST('20012' AS varchar(5))
AND RegistrationDate > '2016-01-01'
```

Does the solution meet the goal?

A. Yes  
B. No

**Answer:** B


**NEW 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 by running the following Transact-SQL statement:

```sql
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. 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

NEW QUESTION 4

Guaranteed success with Our exam guides
visit - https://www.certshared.com
NEW QUESTION 5

You have a database that includes the tables shown in the exhibit. (Click the exhibit button.)

You need to create a list of all customers and the date that the customer placed their last order. For customers who have not placed orders, you must substitute 01/01/1990 for the date.

Which Transact-SQL statement should you run?
NEW QUESTION 6
Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.
After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.
You are building a stored procedure that will be used by hundreds of users concurrently.
You need to store rows that will be processed later by the stored procedure. The object that stores the rows must meet the following requirements:
Be indexable
Contain up-to-date statistics
Be able to scale between 10 and 100,000 rows
The solution must prevent users from accessing one another's data. Solution: You create a global temporary table in the stored procedure. Does this meet the goal?
A. Yes
B. No

Answer: A

NEW QUESTION 7
Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.
After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.
You create a table named Customer by running the following Transact-SQL statement:
You must insert the following data into the Customer table:

<table>
<thead>
<tr>
<th>Record</th>
<th>First name</th>
<th>Last name</th>
<th>Date of Birth</th>
<th>Credit limit</th>
<th>Town ID</th>
<th>Created Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Yvonne</td>
<td>McKay</td>
<td>1984-05-25</td>
<td>9,000</td>
<td>no town details</td>
<td>current date and time</td>
</tr>
<tr>
<td>2</td>
<td>Jossef</td>
<td>Goldberg</td>
<td>1995-06-03</td>
<td>5,500</td>
<td>no town details</td>
<td>current date and time</td>
</tr>
</tbody>
</table>

You need to ensure that both records are inserted or neither record is inserted. Solution: You run the following Transact-SQL statement:

```sql
INSERT INTO Customer (FirstName, LastName, DateOfBirth, CreditLimit, CreatedDate)
VALUES ('Yvonne', 'McKay', '1984-05-25', 9000, GETDATE());
INSERT INTO Customer (FirstName, LastName, DateOfBirth, CreditLimit, CreatedDate)
VALUES ('Jossef', 'Goldberg', '1995-06-03', 5500, GETDATE());
```

Does the solution meet the goal?

A. Yes  
B. No

Answer: B

Explanation: As there are two separate INSERT INTO statements we cannot ensure that both or neither records is inserted.

NEW QUESTION 8

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 query a database that includes two tables: Project and Task. The Project table includes the following columns:

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProjectId</td>
<td>int</td>
<td>This is a unique identifier for a project.</td>
</tr>
<tr>
<td>ProjectName</td>
<td>varchar(100)</td>
<td>A nonclustered index exists for this column.</td>
</tr>
<tr>
<td>StartTime</td>
<td>datetime2(7)</td>
<td>A null value indicates the project is not finished yet.</td>
</tr>
<tr>
<td>EndTime</td>
<td>datetime2(7)</td>
<td>A null value indicates the task is not completed yet.</td>
</tr>
</tbody>
</table>

The Task table includes the following columns:

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>TaskId</td>
<td>int</td>
<td>This is a unique identifier for a task.</td>
</tr>
<tr>
<td>TaskName</td>
<td>varchar(100)</td>
<td>Each task may or may not have a parent task.</td>
</tr>
<tr>
<td>ParentTaskId</td>
<td>int</td>
<td>A null value indicates the task is not assigned to a specific project.</td>
</tr>
<tr>
<td>ProjectId</td>
<td>int</td>
<td>A null value indicates the task is not completed yet.</td>
</tr>
<tr>
<td>StartTime</td>
<td>datetime2(7)</td>
<td>Identifies the owner of the task.</td>
</tr>
</tbody>
</table>

You plan to run the following query to update tasks that are not yet started: UPDATE Task SET StartTime = GETDATE() WHERE StartTime IS NULL

You need to return the total count of tasks that are impacted by this UPDATE operation, but are not associated with a project.

What set of Transact-SQL statements should you run?
NEW QUESTION 9

You have a table named HR.Employees as shown in the exhibit. (Click the exhibit button.)

You need to write a query that will change the value of the job title column to Customer Representative for any employee who lives in Seattle and has a job title of Sales Representative. If the employee does not have a manager defined, you must not change the title.

Which three 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.

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

Answer: B
NEW QUESTION 10
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.
Multiple processes use the data from a table named Sales and place it in other databases across the organization. Some of the processes are not completely aware of the data types in the Sales table. This leads to data type conversion errors.
You need to implement a method that returns a NULL value if data conversion fails instead of throwing an error.
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

Answer: H

Explanation: TRY_CONVERT returns a value cast to the specified data type if the cast succeeds; otherwise, returns null. References: https://docs.microsoft.com/en-us/sql/t-sql/functions/try-convert-transact-sql

NEW QUESTION 11
You have the following Transact-SQL statement: DELETE FROM Person
WHERE PersonID = 5
You need to implement error handling.
How should you complete Transact-SQL statement? To answer, select the appropriate options in the answer area.
NOTE: Each correct selection is worth one point.
BEGIN TRANSACTION

END TRY
COMMIT TRANSACTION
END CATCH
BEGIN TRY
ROLLBACK TRANSACTION
BEGIN CATCH

DELETE FROM Person
WHERE PersonID = 5

END TRY
COMMIT TRANSACTION
END CATCH
BEGIN TRY
ROLLBACK TRANSACTION
BEGIN CATCH

END TRY
COMMIT TRANSACTION
END CATCH
BEGIN TRY
ROLLBACK TRANSACTION
BEGIN CATCH

IF @TRANCOUNT > 0

END TRY
COMMIT TRANSACTION
END CATCH
BEGIN TRY
ROLLBACK TRANSACTION
BEGIN CATCH

END TRY
COMMIT TRANSACTION
END CATCH
BEGIN TRY
ROLLBACK TRANSACTION
BEGIN CATCH

IF @TRANCOUNT > 0

COMMIT TRANSACTION

Answer:

Explanation:
NEW QUESTION 12

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 contains tables named Customer_CRMSystem and Customer_HRSystem. Both tables use the following structure:

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Allow null</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerID</td>
<td>int</td>
<td>No</td>
</tr>
<tr>
<td>CustomerCode</td>
<td>char(4)</td>
<td>Yes</td>
</tr>
<tr>
<td>CustomerName</td>
<td>varchar(50)</td>
<td>No</td>
</tr>
</tbody>
</table>

The tables include the following records: Customer_CRMSystem

<table>
<thead>
<tr>
<th>CustomerID</th>
<th>CustomerCode</th>
<th>CustomerName</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CUS1</td>
<td>Roya</td>
</tr>
<tr>
<td>2</td>
<td>CUS9</td>
<td>Almudena</td>
</tr>
<tr>
<td>3</td>
<td>CUS4</td>
<td>Jack</td>
</tr>
<tr>
<td>4</td>
<td>NULL</td>
<td>Jane</td>
</tr>
<tr>
<td>5</td>
<td>NULL</td>
<td>Francisco</td>
</tr>
</tbody>
</table>

Customer_HRSystem

<table>
<thead>
<tr>
<th>CustomerID</th>
<th>CustomerCode</th>
<th>CustomerName</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CUS1</td>
<td>Roya</td>
</tr>
<tr>
<td>2</td>
<td>CUS2</td>
<td>Jose</td>
</tr>
<tr>
<td>3</td>
<td>CUS9</td>
<td>Almudena</td>
</tr>
<tr>
<td>4</td>
<td>NULL</td>
<td>Jane</td>
</tr>
</tbody>
</table>
Records that contain null values for CustomerCode can be uniquely identified by CustomerName. You need to display a Cartesian product, combining both tables. Which Transact-SQL statement should you run?

A. 
```
SELECT c.CustomerCode, c.CustomerName, h.CustomerCode, h.CustomerName
FROM Customer_CRMSystem c
INNER JOIN Customer_HRSysystem h
```

B. 
```
SELECT CustomerCode, CustomerName
FROM Customer_CRMSystem
INTERSECT
SELECT CustomerCode, CustomerName
FROM Customer_HRSysystem
```

C. 
```
SELECT c.CustomerCode, c.CustomerName
FROM Customer_CRMSysystem c
LEFT OUTER JOIN Customer_HRSysystem h
ON c.CustomerCode = h.CustomerCode
WHERE h.CustomerCode IS NULL AND c.CustomerCode IS NOT NULL
```

D. 
```
SELECT CustomerCode, CustomerName
FROM Customer_CRMSystem
EXCEPT
SELECT CustomerCode, CustomerName
FROM Customer_HRSysystem
```

E. 
```
SELECT CustomerCode, CustomerName
FROM Customer_CRMSystem
UNION
SELECT CustomerCode, CustomerName
FROM Customer_HRSysystem
```

F. 
```
SELECT CustomerCode, CustomerName
FROM Customer_CRMSystem
UNION ALL
SELECT CustomerCode, CustomerName
FROM Customer_HRSysystem
```

G. 
```
SELECT c.CustomerCode, c.CustomerName, h.CustomerCode, h.CustomerName
FROM Customer_CRMSystem c
CROSS JOIN Customer_HRSysystem h
```

H. 
```
SELECT c.CustomerCode, c.CustomerName, h.CustomerCode, h.CustomerName
FROM Customer_CRMSystem c
FULL OUTER JOIN Customer_HRSysystem h
```

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: A cross join that does not have a WHERE clause produces the Cartesian product of the tables involved in the join. The size of a Cartesian product result set is the number of rows in the first table multiplied by the number of rows in the second table.

NEW QUESTION 13
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 on this series.
Details for the Sales.Customers table are shown in the following table:

<table>
<thead>
<tr>
<th>Column</th>
<th>Data type</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerId</td>
<td>int</td>
<td>primary key</td>
</tr>
<tr>
<td>CustomerCategoryId</td>
<td>int</td>
<td>foreign key to the Sales.CustomerCategories table</td>
</tr>
<tr>
<td>PostalCityID</td>
<td>int</td>
<td>foreign key to the Application.Cities table</td>
</tr>
<tr>
<td>DeliveryCityID</td>
<td>int</td>
<td>foreign key to the Application.Cities table</td>
</tr>
<tr>
<td>AccountOpenedDate</td>
<td>datetime</td>
<td>does not allow values</td>
</tr>
<tr>
<td>StandardDiscountPercentage</td>
<td>int</td>
<td>does not allow values</td>
</tr>
<tr>
<td>CreditLimit</td>
<td>decimal(18,2)</td>
<td>null values are permitted</td>
</tr>
<tr>
<td>IsOnCreditHold</td>
<td>bit</td>
<td>does not allow values</td>
</tr>
<tr>
<td>DeliveryLocation</td>
<td>geography</td>
<td>does not allow values</td>
</tr>
<tr>
<td>PhoneNumber</td>
<td>nvarchar(20)</td>
<td>does not allow values</td>
</tr>
<tr>
<td>ValidFrom</td>
<td>datetime2(7)</td>
<td>does not allow values, GENERATED ALWAYS AS ROW START</td>
</tr>
<tr>
<td>ValidTo</td>
<td>datetime2(7)</td>
<td>does not allow values, GENERATED ALWAYS AS ROW END</td>
</tr>
</tbody>
</table>

Details for the Application.Cities table are shown in the following table:

<table>
<thead>
<tr>
<th>Column</th>
<th>Data type</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CityID</td>
<td>int</td>
<td>primary key</td>
</tr>
<tr>
<td>LatestRecordedPopulation</td>
<td>bigint</td>
<td>null values are permitted</td>
</tr>
</tbody>
</table>

Details for the Sales.CustomerCategories table are shown in the following table:

<table>
<thead>
<tr>
<th>Column</th>
<th>Data type</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerCategoryId</td>
<td>int</td>
<td>primary key</td>
</tr>
<tr>
<td>CustomerCategoryName</td>
<td>nvarchar(50)</td>
<td>does not allow null values</td>
</tr>
</tbody>
</table>

You are creating a report to show when the first customer account was opened in each city. The report contains a line chart with the following characteristics:
- The chart contains a data point for each city, with lines connecting the points.
- The X axis contains the position that the city occupies relative to other cities.
- The Y axis contains the date that the first account in any city was opened. An example chart is shown below for five cities:
During a sales promotion, customers from various cities open new accounts on the same date. You need to write a query that returns the data for the chart. How should you complete the Transact-SQL statement? To answer, drag the appropriate Transact-SQL segments to the correct locations. 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.

**Answer:**

**Explanation:** Box 1: RANK() OVER
RANK returns the rank of each row within the partition of a result set. The rank of a row is one plus the number of ranks that come before the row in question. ROW_NUMBER and RANK are similar. ROW_NUMBER numbers all rows sequentially (for example 1, 2, 3, 4, 5).

**NEW QUESTION 14**
You have a date related query that would benefit from an indexed view. You need to create the indexed view. Which two Transact-SQL functions can you use? Each correct answer presents a complete solution. NOTE: Each correct selection is worth one point

A. DATEADD
B. AT TIME ZONE
C. GETUTCDATE
D. DATEDIFF

**Answer:** A
NEW QUESTION 15
Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.
After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.
You are creating indexes in a data warehouse.
You have a dimension table named Table1 that has 10,000 rows. The rows are used to generate several reports. The reports join a column that is the primary key.
The execution plan contains bookmark lookups for Table1. You discover that the reports run slower than expected.
You need to reduce the amount of time it takes to run the reports. Solution: You create a clustered index on the primary key column. Does this meet the goal?
A. Yes
B. No

Answer: A

NEW QUESTION 16
Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.
After you answer a question in this section. You will NOT be able to return to it. As a result, these questions will not appear in the review screen.
You have a table named Products that stores information about products your company sells. The table has a column named ListPrice that stores retail pricing information for products.
Some products are used only internally by the company. Records for these products are maintained in the Products table for inventory purposes. The price for each of these products is $0.00. Customers are not permitted to order these products.
You need to increase the list price for products that cost less than $100 by 10 percent. You must only increase pricing for products that customers are permitted to order.
Solution: You run the following Transact-SQL statement:

```
UPDATE Production.Products
SET ListPrice = ListPrice * 1.1
WHERE ListPrice BETWEEN 0 and 100
```

Does the solution meet the goal?
A. Yes
B. No

Answer: B

NEW QUESTION 17
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 query a database that includes two tables: Project and Task. The Project table includes the following columns:

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProjectId</td>
<td>int</td>
<td>This is a unique identifier for a project.</td>
</tr>
<tr>
<td>ProjectName</td>
<td>varchar(100)</td>
<td></td>
</tr>
<tr>
<td>StartTime</td>
<td>datetime2(7)</td>
<td></td>
</tr>
<tr>
<td>EndTime</td>
<td>datetime2(7)</td>
<td>A null value indicates the project is not finished yet.</td>
</tr>
<tr>
<td>UserId</td>
<td>int</td>
<td>Identifies the owner of the project.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>TaskId</td>
<td>int</td>
<td>This is a unique identifier for a task.</td>
</tr>
<tr>
<td>TaskName</td>
<td>varchar(100)</td>
<td>A nonclustered index exists for this column.</td>
</tr>
<tr>
<td>ParentTaskId</td>
<td>int</td>
<td>Each task may or may not have a parent task.</td>
</tr>
<tr>
<td>ProjectId</td>
<td>int</td>
<td>A null value indicates the task is not assigned to a specific project.</td>
</tr>
<tr>
<td>StartTime</td>
<td>datetime2(7)</td>
<td></td>
</tr>
<tr>
<td>EndTime</td>
<td>datetime2(7)</td>
<td>A null value indicates the task is not completed yet.</td>
</tr>
<tr>
<td>UserId</td>
<td>int</td>
<td>Identifies the owner of the task.</td>
</tr>
</tbody>
</table>

Task level is defined using the following rules:

<table>
<thead>
<tr>
<th>Task category</th>
<th>Task level definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tasks that have no parent task</td>
<td>Task Level = 0</td>
</tr>
<tr>
<td>Tasks that have a parent task</td>
<td>Task Level = Parent Task’s Level + 1</td>
</tr>
</tbody>
</table>

You need to determine the task level for each task in the hierarchy.
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.

Guaranteed success with Our exam guides visit - https://www.certshared.com
**Answer:**

**Explanation:**

Box 1: `SELECT CAST (NULL AS INT) AS ParentTaskID, etc.`
This statement selects all tasks with task level 0.
The ParentTaskID could be null so we should use `CAST (NULL AS INT) AS ParentTaskID`. Box 2: `UNION` We should use `UNION` and not `UNION ALL` as we do not want duplicate rows. `UNION` specifies that multiple result sets are to be combined and returned as a single result set. Incorrect: Not `UNION ALL`: `ALL` incorporates all rows into the results. This includes duplicates. If not specified, duplicate rows are removed. Box 3, Box 4, Box 5: These statements select all tasks with task level > 0. References: https://msdn.microsoft.com/en-us/library/ms180026.aspx

**NEW QUESTION 18**

You work for an organization that monitors seismic activity around volcanos. You have a table named GroundSensors. The table stores data collected from seismic sensors. It includes the columns described in the following table:

<table>
<thead>
<tr>
<th>Name</th>
<th>Data Type</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>SensorID</td>
<td>int</td>
<td>primary key</td>
</tr>
<tr>
<td>Location</td>
<td>geography</td>
<td>do not allow null values</td>
</tr>
<tr>
<td>Tremor</td>
<td>int</td>
<td>do not allow null values</td>
</tr>
<tr>
<td>NormalizedReading</td>
<td>float</td>
<td>allow null values</td>
</tr>
</tbody>
</table>

The database also contains a scalar value function named NearestMountain that accepts a parameter of type geography and returns the name of the mountain that is nearest to the sensor.

You need to create a query that shows the average of the normalized readings from the sensors for each mountain. The query must meet the following requirements:

- Return the average normalized readings named AverageReading.
- Return the nearest mountain name named Mountain.
- Do not return any other columns.
- Exclude sensors for which no normalized reading exists. Construct the query using the following guidelines:
  - Use one part names to reference tables, columns and functions.
  - Do not use parentheses unless required.
  - Define column headings using the AS keyword.
  - Do not surround object names with square brackets.
Certshared now are offering 100% pass ensure 70-761 dumps!

https://www.certshared.com/exam/70-761/ (200 Q&As)

Guaranteed success with Our exam guides
visit - https://www.certshared.com
Part of the correct Transact-SQL has been provided in the answer area below. Enter the code in the answer area that resolves the problem and meets the stated goals or requirements. You can add code within the code that has been provided as well as below it.

```sql
1 SELECT avg (normalizedreading) as AverageReading, location as Mountain
2 FROM GroundSensors
3 WHERE normalizedreading is not null
```

Use the Check Syntax button to verify your work. Any syntax or spelling errors will be reported by line and character position.

**Answer:**

**Explanation:**
1. SELECT avg (normalizedreading) as AverageReading, location as Mountain
2. FROM GroundSensors
3. WHERE normalizedreading is not null

Note: On line 1 change to AverageReading and change to Mountain.

**NEW QUESTION 19**

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 table named AuditTrail that tracks modifications to data in other tables. The AuditTrail table is updated by many processes. Data input into AuditTrail may contain improperly formatted date time values. You implement a process that retrieves data from the various columns in AuditTrail, but sometimes the process throws an error when it is unable to convert the data into valid date time values.

You need to convert the data into a valid date time value using the en-US format culture code. If the conversion fails, a null value must be returned in the column output. The conversion process must not throw an error.

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

**Answer:** H

**Explanation:**
A TRY_CONVERT function returns a value cast to the specified data type if the cast succeeds; otherwise, returns null.


**NEW QUESTION 20**

......
Thank You for Trying Our Product

We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions
2nd - Questions and Answers in PDF Format

70-761 Practice Exam Features:

* 70-761 Questions and Answers Updated Frequently
* 70-761 Practice Questions Verified by Expert Senior Certified Staff
* 70-761 Most Realistic Questions that Guarantee you a Pass on Your First Try
* 70-761 Practice Test Questions in Multiple Choice Formats and Updates for 1 Year

100% Actual & Verified — Instant Download, Please Click
Order The 70-761 Practice Test Here