



70-461^{Q&As}

Querying Microsoft SQL Server 2012/2014

Pass Microsoft 70-461 Exam with 100% Guarantee

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

<https://www.pass4itsure.com/70-461.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**

You create a table that has the StudentCode, SubjectCode, and Marks columns to record mid-year marks for students. The table has marks obtained by 50 students for various subjects. You need to ensure that the top half of the students

arranged by their average marks must be given a rank of 1 and the remaining students must be given a rank of 2. Which Transact-SQL query should you use?

- A. `SELECT StudentCode as Code, RANK () OVER (ORDER BY AVG (Marks) DESC) AS Value FROM StudentMarks GROUP BY StudentCode`
- B. `SELECT Id, Name, Marks, DENSE_RANK () OVER (ORDER BY Marks DESC) AS Rank FROM StudentMarks`
- C. `SELECT StudentCode as Code, DENSE_RANK () OVER (ORDER BY AVG (Marks) DESC) AS Value FROM StudentMarks GROUP BY StudentCode`
- D. `SELECT StudentCode as Code, NTILE (2) OVER (ORDER BY AVG (Marks) DESC) AS Value FROM StudentMarks GROUP BY StudentCode`
- E. `SELECT StudentCode AS Code,Marks AS Value FROM (SELECT StudentCode, Marks AS Marks, RANK () OVER (PARTITION BY SubjectCode ORDER BY Marks ASC) AS Rank FROM StudentMarks) tmp WHERE Rank = 1`
- F. `SELECT StudentCode AS Code,Marks AS Value FROM (SELECT StudentCode, Marks AS Marks, RANK() OVER (PARTITION BY SubjectCode ORDER BY Marks DESC) AS Rank FROM StudentMarks) tmp WHERE Rank = 1`
- G. `SELECT StudentCode AS Code,Marks AS Value FROM (SELECT StudentCode, Marks AS Marks, RANK () OVER (PARTITION BY StudentCode ORDER BY Marks ASC) AS Rank FROM StudentMarks) tmp WHERE Rank = 1`
- H. `SELECT StudentCode AS Code,Marks AS Value FROM (SELECT StudentCode, Marks AS Marks, RANK OVER (PARTITION BY StudentCode ORDER BY Marks DESC) AS Rank FROM StudentMarks) tmp WHERE Rank = 1`

Correct Answer: D

QUESTION 2

How do the COMMIT and ROLLBACK commands work with nested transactions in T-SQL? (Choose all that apply.)

- A. A single COMMIT commits the entire nested transaction.
- B. A single ROLLBACK rolls back the entire nested transaction.
- C. A single COMMIT commits only one level of the nested transaction.
- D. A single ROLLBACK rolls back only one level of the nested transaction.

Correct Answer: BC

Single COMMIT commits only the innermost level of a nested transaction.

QUESTION 3



You need to create a query that calculates the total sales of each OrderID from a table named Sales.Details. The table contains two columns named OrderID and ExtendedAmount. The solution must meet the following requirements:

Use one-part names to reference columns.

Start the order of the results from OrderID.

NOT depend on the default schema of a user.

Use an alias of TotalSales for the calculated ExtendedAmount.

Display only the OrderID column and the calculated TotalSales column. Provide the correct code in the answer area.

Correct Answer: Please review the part for this answer

```
SELECT OrderID, SUM(ExtendedAmount) AS TotalSales FROM Sales.Details GROUP BY OrderID ORDER BY OrderID
```

QUESTION 4

You have a database named Sales that contains the tables as shown in the exhibit. (Click the Exhibit button.)



OrderDetails			
	Column Name	Data Type	Allow Nulls
	ListPrice	money	<input type="checkbox"/>
	Quantity	int	<input type="checkbox"/>
			<input type="checkbox"/>

Customers			
	Column Name	Data Type	Allow Nulls
!	CustomerID	int	<input type="checkbox"/>
	FirstName	varchar(100)	<input type="checkbox"/>
	LastName	varchar(100)	<input type="checkbox"/>
			<input type="checkbox"/>

Orders			
	Column Name	Data Type	Allow Nulls
!	OrderID	int	<input type="checkbox"/>
	OrderDate	datetime	<input type="checkbox"/>
	CustomerID	int	<input type="checkbox"/>
			<input type="checkbox"/>

You have a database named Sales that contains the tables shown in the exhibit. (Click the Exhibit button.)

```
CREATE PROCEDURE usp_Customers  
    LastName  
FROM Customers  
ORDER BY LastName
```

Complete the Transact-SQL statement



OrderDetails			
	Column Name	Data Type	Allow Nulls
	ListPrice	money	<input type="checkbox"/>
	Quantity	int	<input type="checkbox"/>
			<input type="checkbox"/>

Customers			
	Column Name	Data Type	Allow Nulls
	CustomerID	int	<input type="checkbox"/>
	FirstName	varchar(100)	<input type="checkbox"/>
	LastName	varchar(100)	<input type="checkbox"/>
			<input type="checkbox"/>

Orders			
	Column Name	Data Type	Allow Nulls
	OrderID	int	<input type="checkbox"/>
	OrderDate	datetime	<input type="checkbox"/>
	CustomerID	int	<input type="checkbox"/>
			<input type="checkbox"/>

You need to create a query for a report. The query must meet the following requirements:

Return the last name of the customer who placed the order.

Return the most recent order date for each customer.

Group the results by CustomerID.

Order the results by the most recent OrderDate.

Use the database name and table name for any table reference.

Use the first initial of the table as an alias when referencing columns in a table.

The solution must support the ANSI SQL-99 standard and must NOT use object identifiers.



```
1 SELECT LastName,  
2 MAX(OrderDate) AS MostRecentOrderDate
```

Part of the correct T-SQL statement has been provided in the answer area. Complete the SQL statement.

Correct Answer:

```
SELECT o.LastName, MAX (o.OrderData) AS MostRecentOrderData FROM Sales.Orders AS o GROUP BY  
o.CustomerID ORDER BY o.OrderDate DESC
```

QUESTION 5

You have a SQL database that contains a table named Products.

You are implementing a stored procedure that retrieves the list of products, performs custom business logic and then retrieve the list of products again.

The custom business logic in the stored procedure does not modify data from the Products table.

The stored procedure contains the following:

```
01  
02 GO  
03 BEGIN TRANSACTION;  
04 GO  
05 SELECT *  
06 FROM Products  
07 WHERE ProductID=123 ;  
08 GO  
09 ...  
10 SELECT  
11 FROM Products  
12 WHERE ProductID=123;  
13 GO  
14 ...  
15 COMMIT TRANSACTION ;  
16 GO
```

You need to complete line 01 of the stored procedure to ensure that when the transaction occurs, the data read from the SELECT * FROM Products statement on line 05 is identical to the data read from the SELECT * FROM Products

statement on line 10. The solution must maximize concurrency.

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.

```
1 SET TRANSACTION ISOLATION LEVEL ;
```





Keywords

ADD	DESC	KILL	ROW_NUMBER
ALL	DISK	LEFT	ROWGUIDCOL
ALTER	DISTINCT	LIKE	RULE
AND	DISTRIBUTED	LINENO	SAVESHEMA
ANY	DOUBLE	LOAD	SCHEMABINDING
AS	DROP	MAX	SECURITYAUDIT
ASC	DUMP	MERGE	SELECT
AUTHORIZATION	ELSE	NATIONAL	SEMANTICKEYPHRASETABLE
BACKUP	END	NOCHECK	SEMANTICSIMILARITYDETAILSTABLE
BEGIN	ERRLVL	NONCLUSTEREDNOT	SEMANTICSIMILARITYTABLE
BETWEEN	ERROR_NUMBER	NULL	SESSION_USER
BREAK	ESCAPE	NULLIF	SET
BROWSE	ESCEPT	OF	SETUSER
BULK	EXEC	OFF	SHUTDOWN
BY	EXECUTE	OFFSETS	SNAPSHOT
CASCADE	EXISTS	ON	SOME
CASE	EXIT	OPEN	STATISTICS
CAST	EXTERNAL	OPENDATASOURCE	SYSTEM_USER
CATCH	FETCH	OPENQUERY	TABLE
CHECK	FILE	OPENROWSET	TABLESAMPLE
CHECKPOINT	FILESTREAM	OPENXML	TEXTSIZE
CLOSE	FILLFACTOR	OPTION	THEN
CLUSTERED	FOR	OR	TO
COALESCE	FORFOREIGN	ORDER	TOP
COLLATE	FREETEXT	OUTER	TRAN
COLUMN	FREETEXTTABLE	OVER	TRANSACTION
COMMIT	FROM	PERCENT	TRIGGER
COMPUTE	FULL	PERSISTED	TRUNCATE
CONCAT	FUNCTION	PIVOT	TRY_CONVERT
CONSTRAINT	GETDATE	PLAN	TSEQUAL
CONTAINS	GO	PRECISION	UNION
CONTAINSTABLE	GOTO	PRIMARY	UNIQUE
CONTINUE	GRANT	PRINT	UNPIVOT
CONVERT	GROUP	PROC	UPDATE
CREATE	HAVING	PROCEDURE	UPDATETEXT
CROSS	HOLDLOCK	PUBLIC	USE
CREATE	IDENTITY	RAISERROR	USER
CROSS	IDENTITY_INSERT	RANK	VALUES
CURRENT	IDENTITYCOL	READ	VARYING
CURRENT_DATE	IF	READTEXT	VIEW
CURRENT_TIME	IFF	RECONFIGURE	WAITFOR
CURRENT_TIMESTAMP	IN	REFERENCES	WHEN
CURRENT_USER	INDEX	REPEATABLE	WHERE
CURSOR	INNER	REPLICATION	WHILE
DATABASE	INSERT	RESTORE	WITH
DATETIME	INT	RESTRICT	WITHIN GROUP
DBCC	INTERSECT	RETURNREVERT	WRITETEXT
DEALLOCATE	INTO	REVOKE	XML
DECLAREDEFAULT	IS	RIGHT	
DELETE	ISNULL	ROLLBACK	
DENSE_RANK	JOIN	ROWCOUNT	
DENY	KEY		



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

Correct Answer: REPEATABLE READ



Add REPEATABLE READ to line 1 to get:

SET TRANSACTION ISOLATION LEVEL REPEATABLE READ;

REPEATABLE READ specifies that statements cannot read data that has been modified but not yet committed by other transactions and that no other transactions can modify data that has been read by the current transaction until the current

transaction completes.

[70-461 VCE Dumps](#)

[70-461 Practice Test](#)

[70-461 Exam Questions](#)



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.pass4itsure.com/allproducts>

Need Help

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

To update a previously submitted ticket:



 One Year Free Update <p>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>	 Money Back Guarantee <p>To ensure that you are spending on quality products, we provide 100% money back guarantee for 30 days from the date of purchase.</p>	 Security & Privacy <p>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 © pass4itsure, All Rights Reserved.