**Ques.1. Write a SQL query to fetch the count of employees working in project ‘P1’.**

**Ans.** Here, we would be using aggregate function count() with the SQL where clause-

SELECT COUNT(\*)

FROM EmployeeSalary

WHERE Project = 'P1';

**Ques.2. Write a SQL query to fetch employee names having a salary greater than or equal to 5000 and less than or equal 10000.**

**Ans.** Here, we will use BETWEEN in the ‘where’ clause to return the EmpId of the employees with salary satisfying the required criteria and then use it as subquery to find the fullName of the employee form EmployeeDetails table.

SELECT FullName

FROM EmployeeDetails

WHERE EmpId IN

(SELECT EmpId FROM EmpolyeeSalary

WHERE Salary BETWEEN 5000 AND 10000);

**Ques.3. Write a SQL query to fetch project-wise count of employees sorted by project’s count in descending order.**

**Ans.** The query has two requirements – first to fetch the project-wise count and then to sort the result by that count. For project-wise count, we will be using the GROUP BY clause and for sorting, we will use the ORDER BY clause on the alias of the project-count.

SELECT Project, count(EmpId) EmpProjectCount

FROM EmployeeSalary

GROUP BY Project

ORDER BY EmpProjectCount DESC;

**Ques.4. Write a query to fetch only the first name(string before space) from the FullName column of EmployeeDetails table.**

**Ans.**

SQL Server-Using SUBSTRING

SELECT SUBSTRING(FullName, 0, CHARINDEX(' ',FullName))

FROM EmployeeDetails;

Also, we can use LEFT which returns the left part of a string till a specified number of characters.

SELECT LEFT(FullName, CHARINDEX(' ',FullName) - 1)

FROM EmployeeDetails;

**Ques.5. Write a query to fetch employee names and salary records. Return employee details even if the salary record is not present for the employee.**

**Ans.** Here, we can use left join with EmployeeDetail table on the left side.

SELECT E.FullName, S.Salary

FROM EmployeeDetails E LEFT JOIN EmployeeSalary S

ON E.EmpId = S.EmpId;

**Ques.7. Write a SQL query to fetch all employee records from EmployeeDetails table who have a salary record in EmployeeSalary table.**

**Ans.** Using ‘Exists’-

SELECT \* FROM EmployeeDetails E

WHERE EXISTS

(SELECT \* FROM EmployeeSalary S WHERE E.EmpId = S.EmpId);

**Ques.8. Write a SQL query to fetch duplicate records from a table.**

**Ans.** In order to find duplicate records from the table, we can use GROUP BY on all the fields and then use HAVING clause to return only those fields whose count is greater than 1 i.e. the rows having duplicate records.

SELECT EmpId, Project, Salary, COUNT(\*)

FROM EmployeeSalary

GROUP BY EmpId, Project, Salary

HAVING COUNT(\*) > 1;

**Ques.9. Write a SQL query to remove duplicates from a table without using a temporary table.**

**Ans.** Using rowId-

DELETE FROM tenant

WHERE hmy NOT IN

(SELECT MAX(hmy)

FROM tenant

GROUP BY scode);

**Ques.10. Write a SQL query to fetch only odd rows from the table.**

**Ans.** This can be achieved by using Row\_number in SQL server-

SELECT E.EmpId, E.Project, E.Salary

FROM (

SELECT \*, Row\_Number() OVER(ORDER BY EmpId) AS RowNumber

FROM EmployeeSalary

) E

WHERE E.RowNumber % 2 = 1

**Ques.11. Write a SQL query to fetch only even rows from the table.**

**Ans.** Using the same Row\_Number() and checking that the remainder when divided by 2 is 0-

SELECT E.EmpId, E.Project, E.Salary

FROM (

SELECT \*, Row\_Number() OVER(ORDER BY EmpId) AS RowNumber

FROM EmployeeSalary

) E

WHERE E.RowNumber % 2 = 0

**Ques.12. Write a SQL query to create a new table with data and structure copied from another table.**

**Ans.** Using SELECT INTO command-

SELECT \*

INTO newTable

FROM EmployeeDetails;

**Ques.13. Write a SQL query to create an empty table with the same structure as some other table.**

**Ans.**

SELECT \*

INTO newTable

FROM EmployeeDetails

WHERE 1 = 0;

**Ques.14. Write a SQL query to fetch common records between two tables.**

**Ans.** Using INTERSECT-

SELECT \* FROM EmployeeSalary

INTERSECT

SELECT \* FROM ManagerSalary

**Ques.15. Write a SQL query to fetch records that are present in one table but not in another table.**

**Ans.** Using MINUS-

SELECT \* FROM EmployeeSalary

MINUS

SELECT \* FROM ManagerSalary

**Ques.16. Write a SQL query to find the current date-time.**

**Ans. SQL Server-** SELECT getdate();

**Oracle-** SELECT SYSDATE FROM DUAL;

**Ques.17. Write a SQL query to fetch all the Employees details from EmployeeDetails table who joined in the Year 2016.**

**Ans.**

Using BETWEEN for the date range ’01-01-2016′ AND ’31-12-2016′

SELECT \* FROM EmployeeDetails

WHERE DateOfJoining

BETWEEN '01-01-2016’ AND date '31-12-2016';

Also, we can extract year part from the joining date (using YEAR in mySQL)-

SELECT \* FROM EmployeeDetails

WHERE YEAR(DateOfJoining) = '2016';

**Ques.18. Write a SQL query to fetch top n records?**

**Ans.** In SQL server using TOP command-

SELECT TOP N \* FROM EmployeeSalary

ORDER BY Salary DESC

In Oracle using ROWNUM-

SELECT \* FROM

(SELECT \* FROM EmployeeSalary ORDER BY Salary DESC)

WHERE ROWNUM <= 3;

**Ques.19. Write SQL query to find the nth highest salary from table.**

**Ans.** Using Top keyword (SQL Server)-

SELECT TOP 1 Salary

FROM Employee

ORDER BY Salary DESC

**Ques.20. Write SQL query to find the 3rd highest salary from table without using TOP/limit keyword.**

**Ans.** The below SQL query makes use of correlated subquery wherein in order to find the 3rd highest salary the inner query will return the count of till we find that there are two rows that salary greater than other distinct salaries.

For nth highest salary-

SELECT Salary

FROM EmployeeSalary Emp1

WHERE N-1 = (

SELECT COUNT( DISTINCT ( Emp2.Salary ) )

FROM EmployeeSalary Emp2

WHERE Emp2.Salary > Emp1.Salary

)

SELECT Salary

FROM EmployeeSalary Emp1

WHERE 2 = (

SELECT COUNT( DISTINCT ( Emp2.Salary ) )

FROM EmployeeSalary Emp2

WHERE Emp2.Salary > Emp1.Salary

)

**Question 1: SQL Query to find second highest salary of Employee**

Answer: There are many ways to find second highest salary of Employee in SQL, you can either use SQL Join or Subquery to solve this problem. Here is SQL query using Subquery:

select MAX(Salary)

from Employee

WHERE Salary NOT IN

(select MAX(Salary) from Employee);

**Question 2: SQL Query to find Max Salary from each department.**

Answer: You can find the maximum salary for each department by grouping all records by DeptId and then using MAX() function to calculate maximum salary in each group or each department.

SELECT DeptID, MAX(Salary)

FROM Employee

GROUP BY DeptID.

These questions become more interesting if Interviewer will ask you to print department name instead of department id, in that case, you need to join Employee table with Department using foreign key DeptID, make sure you do LEFT or RIGHT OUTER JOIN to include departments without any employee as well.  Here is the query

SELECT DeptName, MAX(Salary)

FROM Employee e RIGHT JOIN Department d

ON e.DeptId=d.DeptID

GROUP BYDeptName;

In this query, we have used RIGHT OUTER JOIN because we need the name of the department from Department table which is on the right side of JOIN clause, even if there is no reference of dept\_id on Employee table. **Question 4: Write an SQL Query to check whether date passed to Query is the date of given format or not**.

Answer: SQL has IsDate() function which is used to check passed value is a date or not of specified format, it returns 1(true) or 0(false) accordingly. Remember ISDATE() is an MSSQL function and it may not work on Oracle, MySQL or any other database but there would be something similar.

SELECT ISDATE('1/08/13') AS "MM/DD/YY";

It will return 0 because passed date is not in correct format.

**Question 5: Write an SQL Query to print the name of the distinct employee whose DOB is between 01/01/1960 to 31/12/1975.**

Answer: This SQL query is tricky, but you can use BETWEEN clause to get all records whose date fall between two dates.

SELECT DISTINCTEmpName

FROM Employees

WHERE DOB BETWEEN ‘01/01/1960’AND ‘31/12/1975’;

**Question 6: Write an SQL Query find number of employees according to gender whose DOB is between 01/01/1960 to 31/12/1975.**

Answer:

SELECTCOUNT(\*), sex

from Employees

WHERE DOB BETWEEN '01/01/1960'AND'31/12/1975'

GROUP BY sex;

**Question 7: Write an SQL Query to find an employee whose Salary is equal or greater than 10000**.

Answer:

SELECT EmpName

FROM Employees

WHERE Salary>=10000;

**Question 8: Write an SQL Query to find name of employee whose name Start with ‘M’**

Answer:

SELECT \* FROM Employees

WHERE EmpName like 'M%';

**Question 9: find all Employee records containing the word "Joe", regardless of whether it was stored as JOE, Joe, or joe.**

Answer :

SELECT \* from Employees

WHEREUPPER(EmpName) like '%JOE%';

**Question 10: Write an SQL Query to find  the year from date.**

Answer:  Here is how you can find Year from a Date in SQL Server 2008

SELECT YEAR(GETDATE());

**Question 11: Write SQL Query to find duplicate rows in a database? and then write SQL query to delete them?**  
Answer: You can use the following query to select distinct records:

SELECT \* FROM emp a

WHERE rowid !=

(SELECT MAX(rowid)

FROM EMP b

WHERE a.empno=b.empno)

to Delete:

DELETE FROM emp a

WHERE rowid !=

(SELECT MAX(rowid)

FROM emp b

WHERE a.empno=b.empno)

**Question 12: There is a table which contains two column Student and Marks, you need to find all the students, whose marks are greater than average marks i.e. list of above average students.**  
Answer: This query can be written using subquery as shown below:

SELECT student, marks

from table where marks >

(SELECT AVG(marks) from table)

**Question 13: How do you find all employees which are also manager? .**  
You have given a standard employee table with an additional column mgr\_id, which contains employee id of the manager.

Answer: You need to know about self-join to solve this problem. In Self Join, you can join two instances of the same table to find out additional details as shown below

SELECT e.name, m.name

FROM Employee e, Employee m

WHERE e.mgr\_id = m.emp\_id;

One follow-up is to modify this query to include employees which don't have a manager.

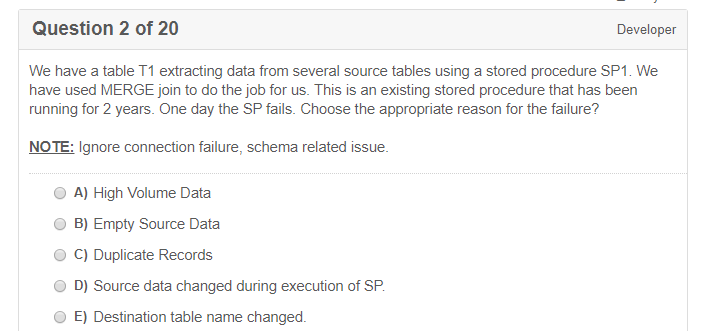
To solve that, instead of using the inner join, just use left outer join, this will also include employees without managers.  
  
**Question 14: You have a composite index of three columns, and you only provide the value of two columns in WHERE clause of a select query? Will Index be used for this operation?** For example if Index is on EmpId, EmpFirstName, and EmpSecondName and you write query like

SELECT \* FROM Employee

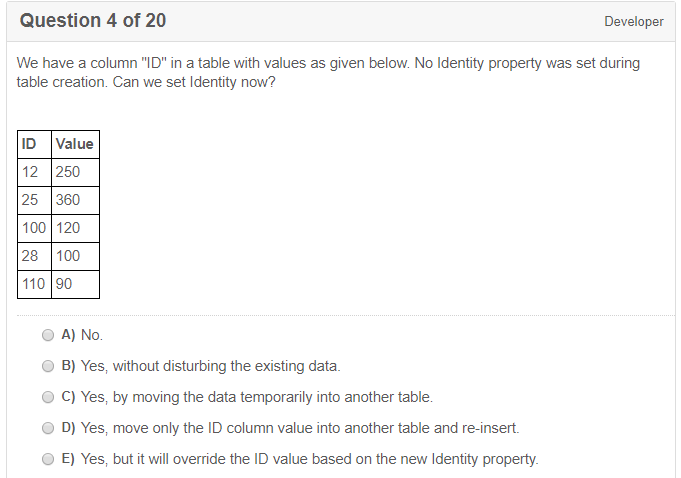
WHERE EmpId=2 and EmpFirstName='Radhe'

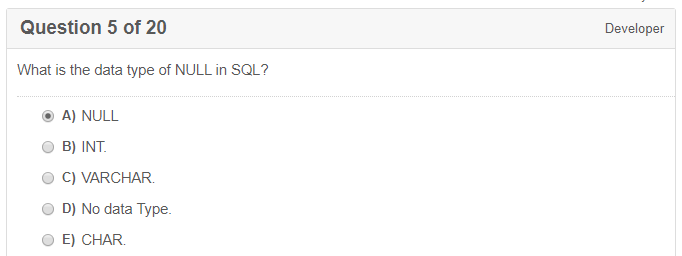
If the given two columns are secondary index column then the index will not invoke, but if the given 2 columns contain the primary index(first column while creating index) then the index will invoke. In this case, Index will be used because EmpId and EmpFirstName are primary columns

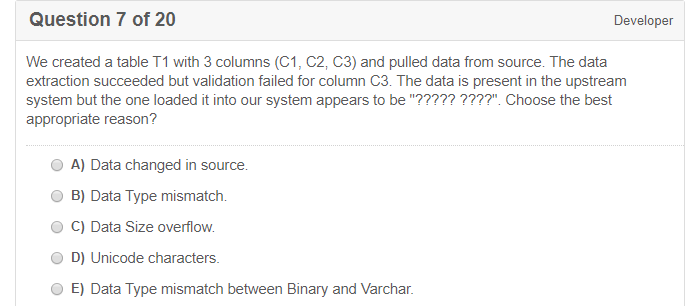
**Search for scenario based SQL Questions**



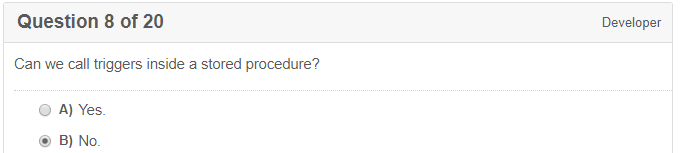
C/D/E, I think - SSIS merge

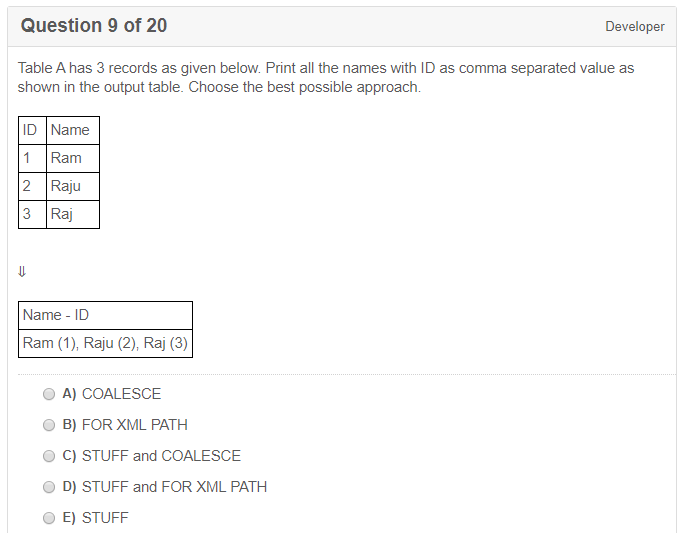
**A**

****



D, I think



**D**

STUFF,

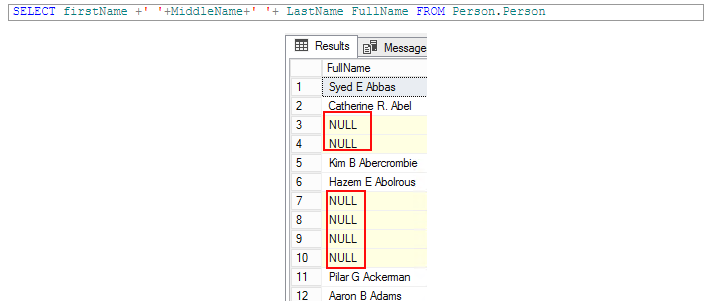
SELECT STUFF('SQL Tutorial', 1, 3, 'HTML'); -> 'HTML Tutorial'

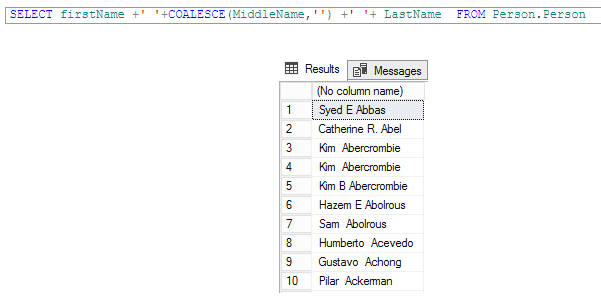
COALESCE

The COALESCE() function returns the first non-null value in a list.

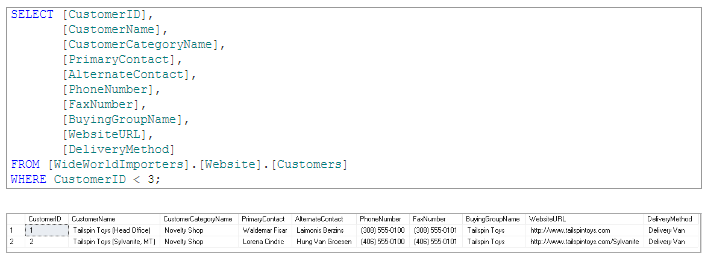
SELECT COALESCE(NULL, NULL, NULL, 'W3Schools.com', NULL, 'Example.com');

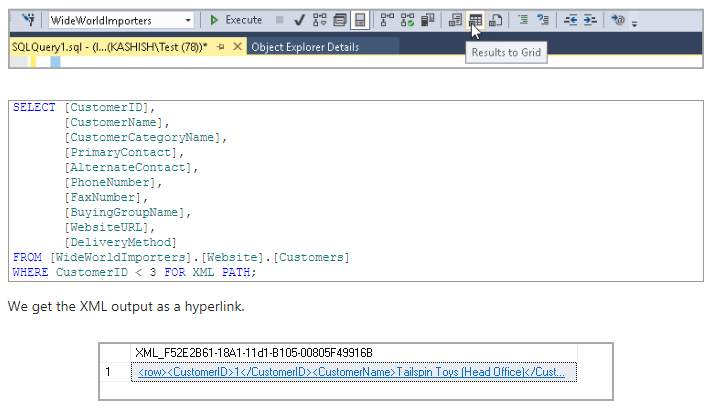
-> W3Schools.com

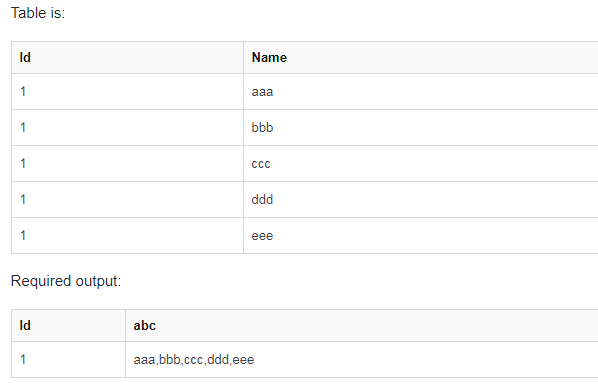




**FOR XML PATH**

****

****

****

SELECT ID,

abc = STUFF(

(SELECT ',' + name FROM temp1 FOR XML PATH ('')), 1, 1, ''

)

FROM temp1 GROUP BY id

**1. Get XML element string with FOR XML**

Adding FOR XML PATH to the end of a query allows you to output the results of the query as XML elements, with the element name contained in the PATH argument.

**SELECT ',' + name**

**FROM temp1**

**FOR XML PATH ('')**

By passing in a blank string (FOR XML PATH('')), we get the following instead:

**,aaa,bbb,ccc,ddd,eee**

**2. Remove leading comma with STUFF**

The STUFF statement literally "stuffs” one string into another, replacing characters within the first string. We, however, are using it simply to remove the first character of the resultant list of values.

SELECT abc = STUFF((

SELECT ',' + NAME

FROM temp1

FOR XML PATH('')

), 1, 1, '')

FROM temp1

The parameters of STUFF are:

* The string to be “stuffed” (in our case the full list of name with a leading comma)
* The location to start deleting and inserting characters (1, we’re stuffing into a blank string)
* The number of characters to delete (1, being the leading comma)

So we end up with: aaa,bbb,ccc,ddd,eee

**3. Join on id to get full list**

Next we just join this on the list of id in the temp table, to get a list of IDs with name:

SELECT ID, abc = STUFF(

(SELECT ',' + name

FROM temp1 t1

WHERE t1.id = t2.id

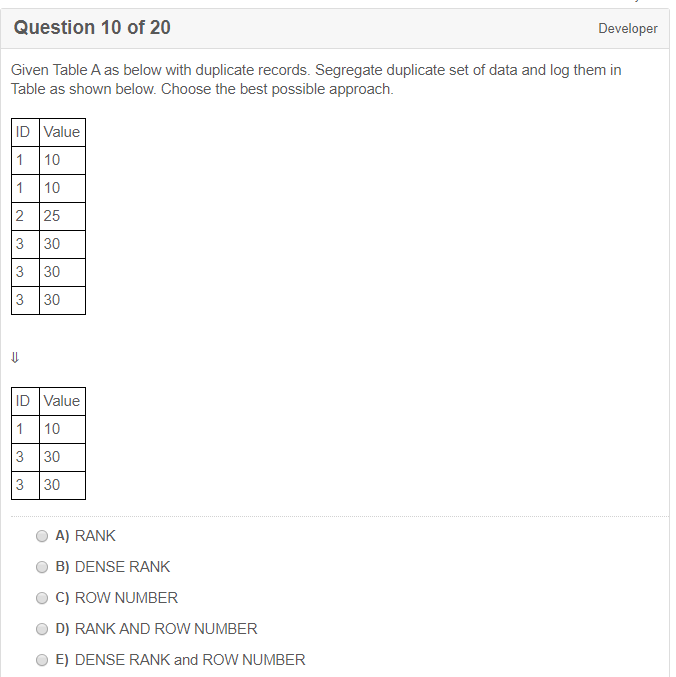
FOR XML PATH (''))

, 1, 1, '') from temp1 t2

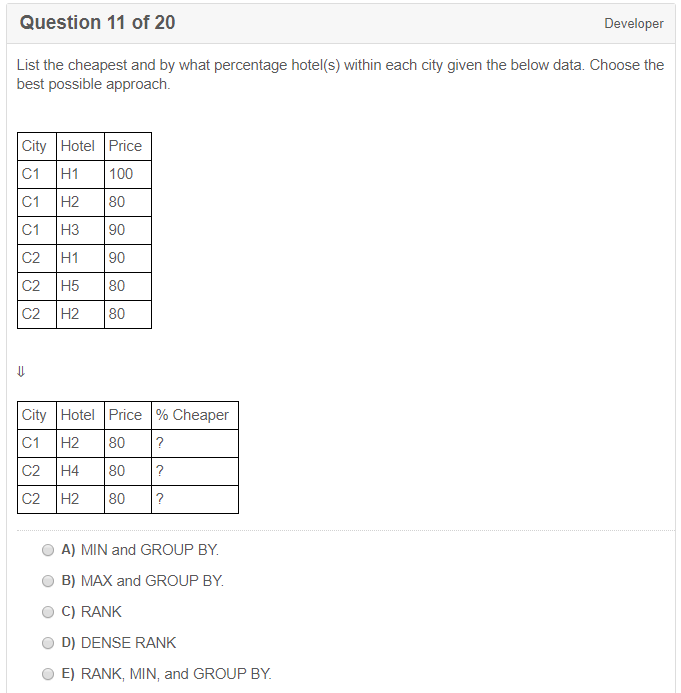
group by id;

And we have our result:

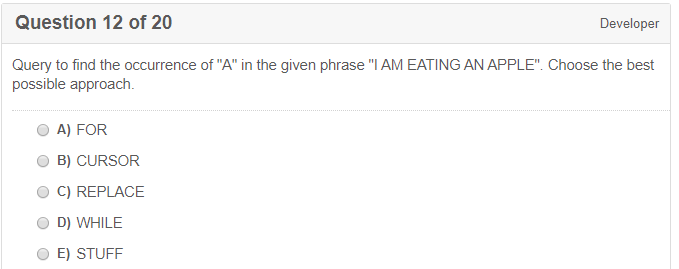




C, I think



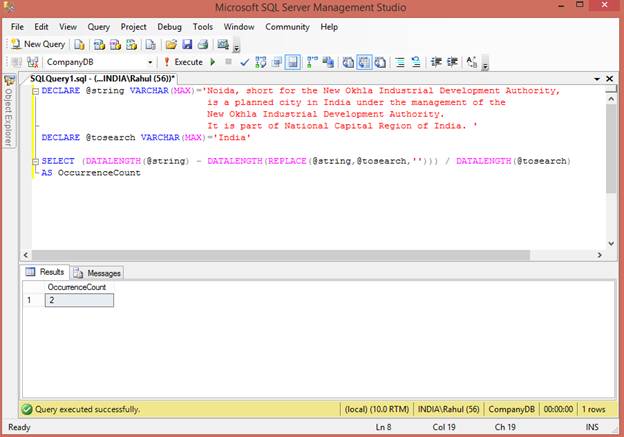
**B, i think**

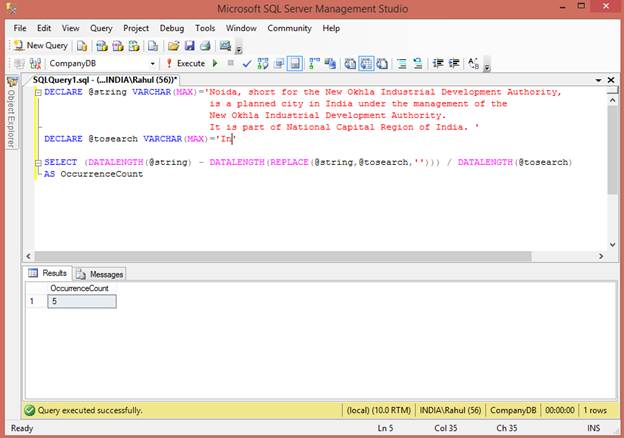
**C**

declare @string varchar(100)

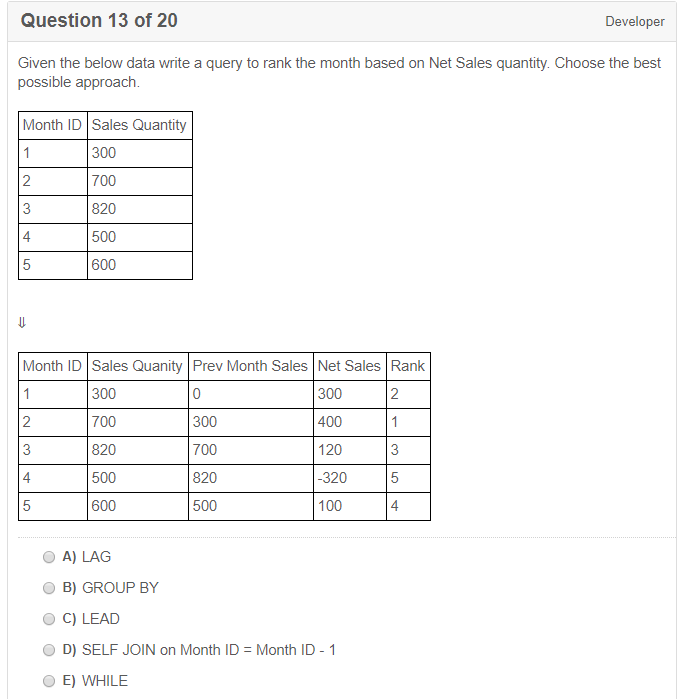
select @string = 'sfdasadhfasjfdlsajflsadsadsdadsa'

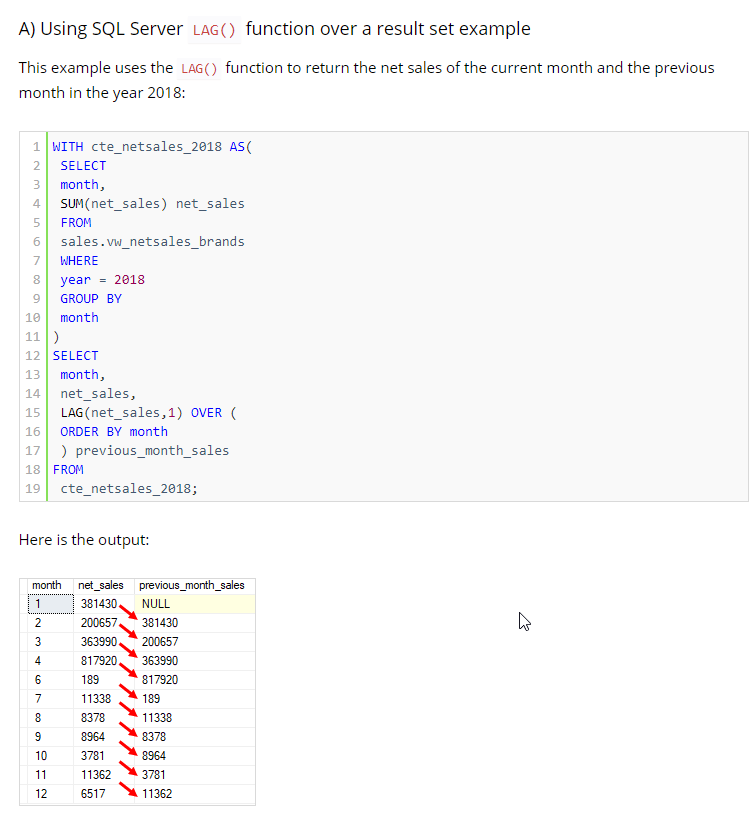
SELECT LEN(@string) - LEN(REPLACE(@string, 'd', '')) AS D\_Count



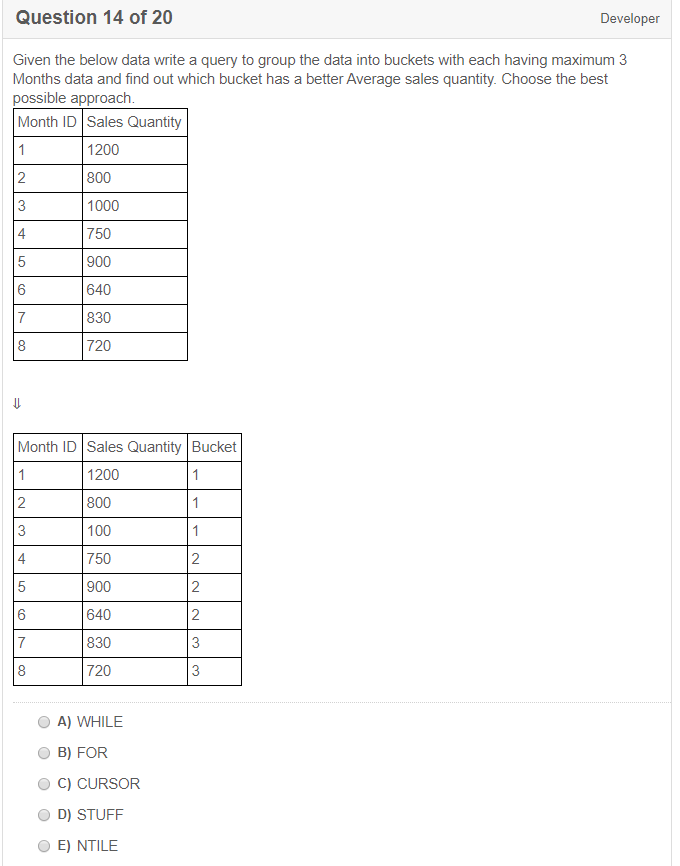


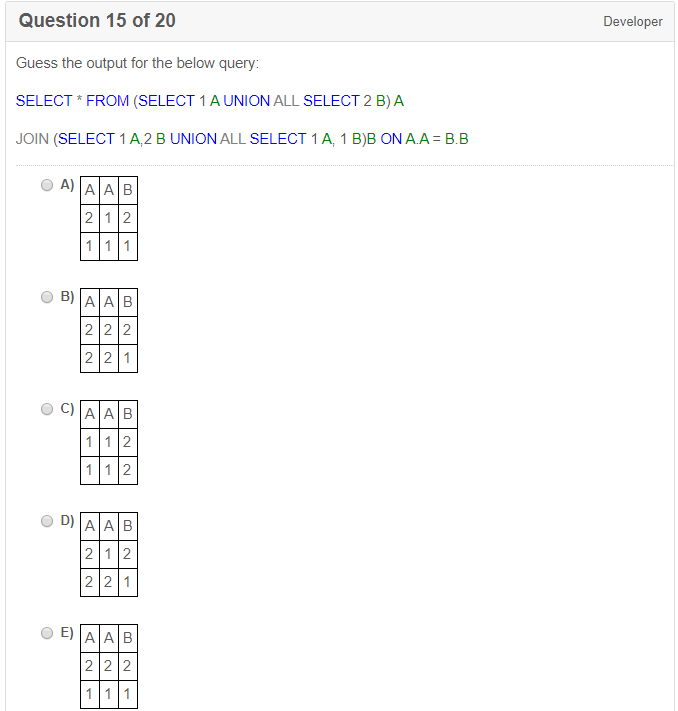
1. **DECLARE** @tosearch **VARCHAR**(**MAX**)='In'
3. **SELECT** (DATALENGTH(@string)-DATALENGTH(REPLACE(@string,@tosearch,'')))/DATALENGTH(@tosearch)
4. **AS** OccurrenceCount

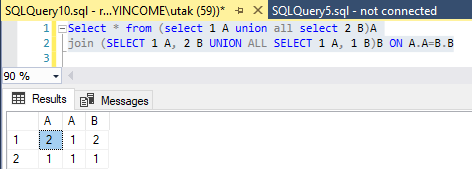
**A**

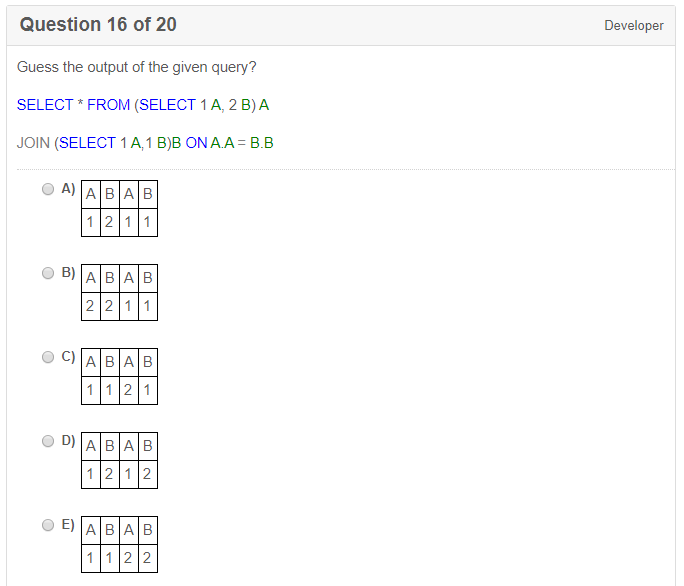
****

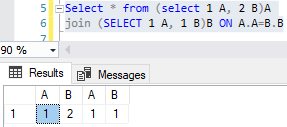
****

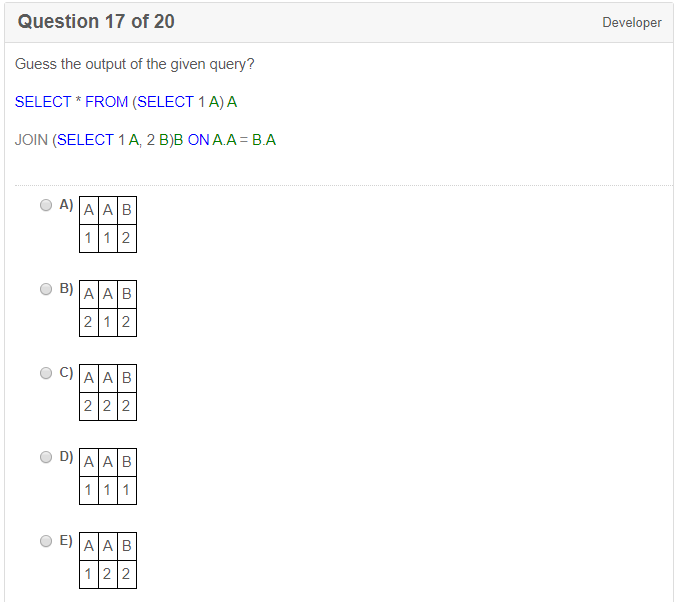
**E**

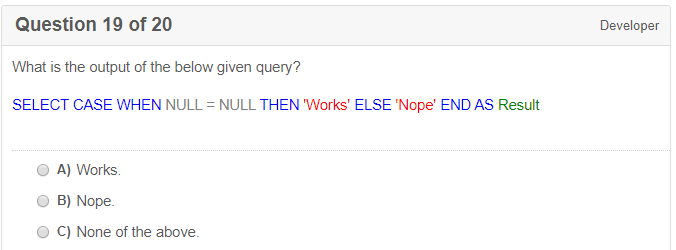




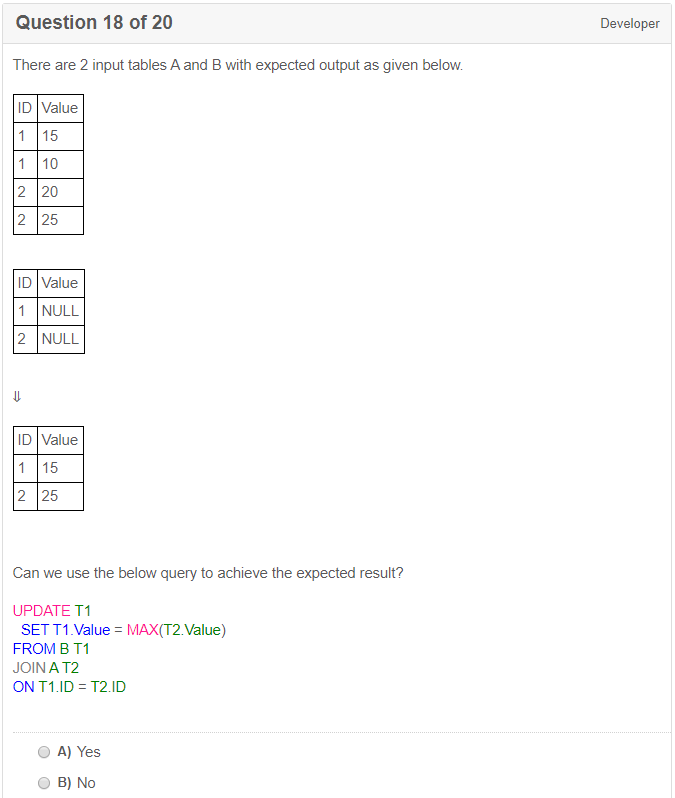


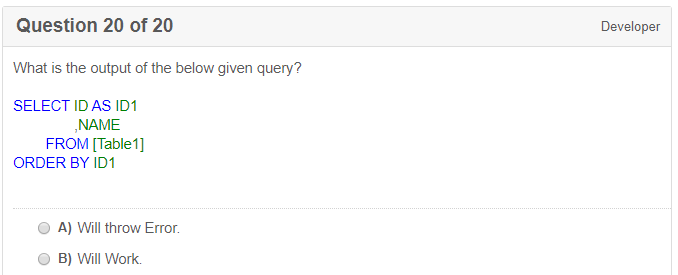


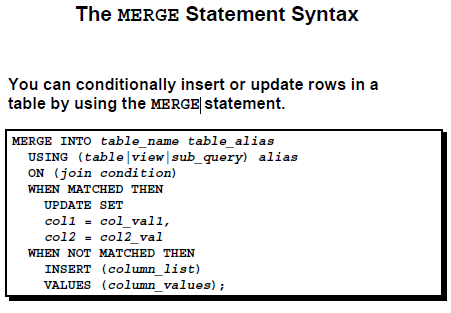
**A**

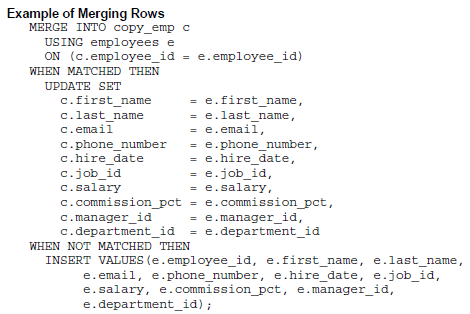


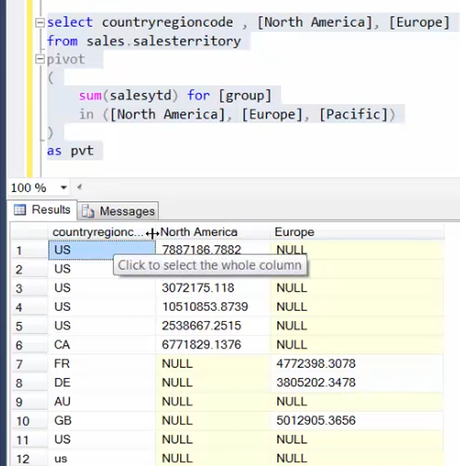
**B. Nope**

**A**

**B**







Q1 How will you go about identifying duplicate records in a table?  
A1 The following SQL query will do the trick.

SELECT code,user\_name,COUNT(user\_name)ASNumOccurrences

FROMtbl\_user

GROUPBY code,user\_name

HAVING (COUNT(user\_name)> 1 )

Q2 How would you go about deleting the duplicate records?  
A2 You could do it in a number of steps as shown below.

* Create a temporary table.
* Insert the unique records into the temporary table.
* Delete the records from the original table.
* Insert the saved single records from the temporary table back to the original tab

Q3 How will you go about searching for table and column names that you don’t know where they really are? For example, search for a column name to find out in which tables they do exist.  
A3  sp\_help tablename

sp\_helptext procedure\_name --definition

select \* from sys.columns

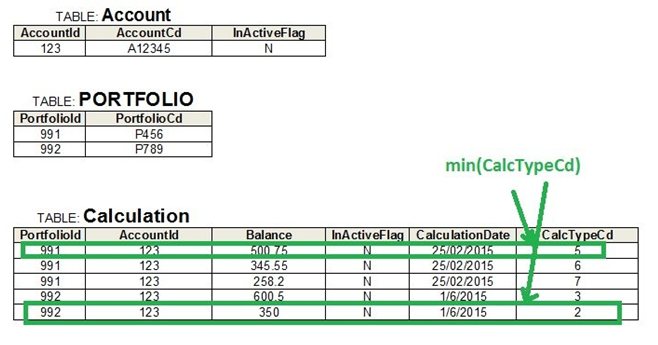
where object\_id = OBJECT\_ID(N'table\_name')

sys.tables

sys.triggers

sys.all\_object

**SQL**query for the following scenario?

****

select acc.AccountCd, calc.Balance

from Calculation calc

inner join Portfolio pf

on pf.PortfolioId=calc.PortfolioId

inner join Account acc

on acc.AccountId=calc.AccountId

where pf.PortfolioCd='P456'

and calc.CalculationDate='25 Feb 2015'

and calc.InActiveFlag='N'

and acc.InActiveFlag='N'

and calc.CalcTypeCd=

(select min(calc2.CalcTypeCd)

from calculation calc2

where calc2.CalculationDate =calc.CalculationDate

and calc2.InActiveFlag ='N'

and calc2.AccountId =calc.AccountId

group by AccountId)

order by acc.AccountCd

Q5 If you need to map actual values retrieved from the database to some other value and then sort by these translated values as well, how will you go about accomplishing this in your SQL code?

SELECT PortfolioCd, SentDateTime, ExcludedFlag, StatusCd as ActualStatusCd,

case when p.ExcludedFlag='Y' then 'Excluded'

else case when p.SentDateTime isnull then p.StatusCd

else 'Sent' end

end as EvaluatedStatus Cd

FROM Portfolio p

WHERE calculation date>'09 Jan 2013'

and InActiveFlag='N'

ORDERBY

case when p.ExcludedFlag ='Y'then'4'

else case when p.SentDateTime isnotnull then'3'

else case when p.StatusCd='New' then'1'

when p.StatusCd='Processed'then'2' end

end

end,PortfolioCd

Q6 How would you retrieve a date time column converted to string and formatted as dd/mm/yy hh:mm:ss  
A6 You can use specific **functions**provided by your database server. These functions are specific to the database server you are using, hence your code cannot be ported to other database servers. Here is an example in Sybase.

SELECT PortfolioCd,

convert(char(11),p.SentDateTime, 103) +convert(char(12), p.SentDateTime, 108) as SentDateTime

FROM Portfolio p

WHERE calculationdate>'09 Jan 2013'

AND InActiveFlag='N'

**SQL Windows Function**