**SQL Job Preparation Assignment 4**

1. EMP TABLE'S 3 DYNAMIC COLUMNS SHOULD BE DISPLAYED. Instead of "Select EMP, SAL from EMP," your application should prompt you to choose the column you want to select at runtime.

A-

*SELECT GROUP\_CONCAT(DISTINCT(COLUMN\_NAME) SEPARATOR ', ') FROM INFORMATION\_SCHEMA.COLUMNS WHERE TABLE\_NAME='YOUR\_TABLE\_NAME' AND COLUMN\_NAME LIKE '%name%' INTO @colnames;*

*PREPARE stmt FROM @query;*

*EXECUTE stmt;*

1. Write a query to retrieve the data, and column names should be given at Runtime. Sort the query based on the first column.

A-

*SELECT `COLUMN\_NAME`*

*FROM `INFORMATION\_SCHEMA`.`COLUMNS`*

*WHERE `TABLE\_SCHEMA`='yourdatabasename'*

*AND `TABLE\_NAME`='yourtablename';*

Note: As we are hardcoding the name of the column, we cannot use "Order By EMP" as the Column name is not yet selected

1. WHAT IS THE DIFF BETWEEN "&" AND "&&"?

A-

“&” is used to create a temporary substitution variable. You will be prompted to enter the value every time the variable is referenced. The single ampersand operator (&) evaluates both sides of the operator before arriving at its answer.

“&&” is used to create a permanent substitution variable. You need to enter the value only once. The double ampersand operator (&& – also known as the conditional-AND operator) evaluates the RHS only if the LHS is true.

Both operators produce a Boolean result, both operators function as in this truth table.

|  |  |  |
| --- | --- | --- |
| LHS (Left-hand side) | RHS (Right-hand Side) | Result |
| False | False | False |
| True | False | False |
| False | True | False |
| True | True | True |

1. Write a query to demonstrate '&&' and '&' in a single statement.

A-

*SELECT &SORTCOL, SALARY*

*FROM &&MYTABLE*

*WHERE SALARY>12000;*

1. Write a Query for the below statements Table DEPT:



------------------------------ DEPTNO DNAME LOC 10 AC HDC

20 RES CDC

------------------------------

Table DEPT1:

------------------------------ DEPTNO DNAME LOC 10 AC HDC

------------------------------

Use the SET operator to merge the above two tables; also, no repetition data should be there.

A-Table DEPT:

*Create table DEPT(*

*DEPTNO varchar(50) NOT NULL,*

*DNAME varchar(50) UNIQUE,*

*Primary Key(DEPTNO))*

Table DEPT1:

*Create table DEPT1(*

*DEPTNO varchar(50),*

*DNAME varchar(50) NOT NULL,*

*Primary Key(DNAME),*

*FOREIGN KEY (DEPTNO) REFERENCES DEPT(DEPTNO))*

MERGING OF THE 2 TABLES:

*SELECT DEPTNO,DNAME FROM DEPT*

*UNION*

*SELECT DEPTNO,DNAME FROM DEPT1;*