**Sql question for project 1**

**The task to be performed:**

1. Create a database named *project* and *employee*, then import **data\_science\_team.csv**and **proj\_table.csv** into the **project** database and **emp\_record\_table.csv** into the **employee** database from the given resources.

2. Create an ER diagram for the given **project** and the **employee** databases.

3. Write a query to fetch EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, and DEPARTMENT from the employee record table, and make a list of employees and details of their department.

4. Write a query to fetch EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPARTMENT, and EMP\_RATING if the EMP\_RATING is:

* less than two
* greater than four
* between two and four

5. Write a query to concatenate the FIRST\_NAME and the LAST\_NAME of employees in the *Finance* department from the employee table and then give the resultant column alias as NAME.

7. Write a query to list down all the employees from the healthcare and finance departments main using union. Take data from the employee record table.

8. Write a query to list down employee details such as EMP\_ID, FIRST\_NAME, LAST\_NAME, ROLE, DEPARTMENT, and EMP\_RATING grouped by dept. Also include the respective employee rating along with the max emp rating for the department.

9. Write a query to calculate the minimum and the maximum salary of the employees in each role. Take data from the employee record table.

10. Write a query to assign ranks to each employee based on their experience. Take data from the employee record table.

11. Write a query to create a view that displays employees in various countries whose salary is more than six thousand**.** Take data from the employee record table.

12. Write a nested query to find employees with experience of more than ten years. Take data from the employee record table.

13. Write a query to create a stored procedure to retrieve the details of the employees whose experience is more than three years. Take data from the employee record table.

14. Write a query using stored functions in the project table to check whether the job profile assigned to each employee in the data science team matches the organization’s set standard.

     The standard being:

For an employee with experience less than or equal to 2 years assign 'JUNIOR DATA SCIENTIST',

For an employee with the experience of 2 to 5 years assign 'ASSOCIATE DATA SCIENTIST',

For an employee with the experience of 5 to 10 years assign 'SENIOR DATA SCIENTIST',

For an employee with the experience of 10 to 12 years assign 'LEAD DATA SCIENTIST',

For an employee with the experience of 12 to 16 years assign 'MANAGER'.

15. Create an index to improve the cost and performance of the query to find the employee whose FIRST\_NAME is ‘Eric’ in the employee table after checking the execution plan.

Codes\_

select \* from assignment.data\_science\_team;

select \* from assignment.emp\_record\_table;

select EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER,DEPT from assignment.emp\_record\_table;

select EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER,DEPT,EMP\_RATING from assignment.emp\_record\_table where EMP\_RATING <2;

select EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER,DEPT,EMP\_RATING from assignment.emp\_record\_table where EMP\_RATING >4;

select EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER,DEPT,EMP\_RATING from assignment.emp\_record\_table where EMP\_RATING between 2 and 4;

select concat(FIRST\_NAME," ",LAST\_NAME) AS NAME FROM assignment.emp\_record\_table WHERE DEPT="FINANCE";

select FIRST\_NAME,LAST\_NAME,DEPT from assignment.emp\_record\_table where DEPT= "FINANCE" or DEPT= "HEALTHCARE";

select FIRST\_NAME,LAST\_NAME,ROLE,DEPT,EMP\_RATING from assignment.emp\_record\_table group by DEPT order by EMP\_RATING desc;

select FIRST\_NAME,LAST\_NAME,ROLE,DEPT,SALARY, dense\_rank() over(order by SALARY desc) as salary\_rank from assignment. emp\_record\_table;

select FIRST\_NAME,LAST\_NAME,ROLE,DEPT,EXP , case

when EXP<=20 and EXP>=15 then "1"

when EXP<=14 and EXP>=10 then "2"

when EXP<=9 and EXP>=5 then "3"

when EXP<=4 and EXP>=2 then "4"

when EXP<=1 then "5"

end as exp\_rating

from assignment.emp\_record\_table;

create view slry as select \* from assignment.emp\_record\_table where SALARY>6000;

select \* from slry;

select \* from assignment.emp\_record\_table where EXP>10;

select \* from assignment.emp\_record\_table where EXP>3;

select \* from assignment.data\_science\_team;

select EMP\_ID,FIRST\_NAME,LAST\_NAME,GENDER,DEPT,EXP,

CASE

WHEN EXP>= 13 AND EXP<= 16 THEN "MANAGER"

WHEN EXP>= 11 AND EXP<= 12 THEN "LEAD DATA SCIENTIST"

WHEN EXP>= 6 AND EXP<= 10 THEN "SENIOR DATA SCIENTIST"

WHEN EXP>= 3 AND EXP<= 5 THEN "ASSOCIATE DATA SCIENTIST"

WHEN EXP<=2 THEN "JUNIOR DATA SCIENTIST"

ELSE "ENTER CORRECT ENTRY"

END AS DEGNISATION

FROM assignment.data\_science\_team;

select \* from assignment.emp\_record\_table where FIRST\_NAME like "Eric";