**Author:** Divya Pardeshi

**Date:** 22-11-2023

**ScienceQtech Employee Performance Mapping- SQL Project**

**Q1) Create a database named employee, then import data\_science\_team.csv proj\_table.csv and emp\_record\_table.csv into the employee database from the given resources.**

**Ans.** #Create the employee database if it doesn't exist

CREATE DATABASE IF NOT EXISTS employee;

#Switch to the employee database

USE employee;

#Show tables in the employee database

SHOW TABLES;

A screenshot of a computer

Description automatically generated

**Q2) Create an ER diagram for the given employee database.**

**Ans.**

**A screenshot of a computer

Description automatically generated**

**Q3) 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.**

**Ans.** SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPT

FROM emp\_record\_table;

A screenshot of a computer

Description automatically generated

**Q4) 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**

**Ans**. **a) less than two:** Employees with EMP\_RATING less than two:

SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPT, EMP\_RATING

FROM emp\_record\_table

WHERE EMP\_RATING < 2;

**A screenshot of a computer

Description automatically generated**

**b) greater than four:** Employees with EMP\_RATING greater than four:

SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPT, EMP\_RATING

FROM emp\_record\_table

WHERE EMP\_RATING > 4;

A screenshot of a computer

Description automatically generated

**c) between two and four:** Employees with EMP\_RATING between two and four:

SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPT, EMP\_RATING

FROM emp\_record\_table

WHERE EMP\_RATING >= 2 AND EMP\_RATING <= 4;

A screenshot of a computer

Description automatically generated

**Q5) 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.**

**Ans.** SELECT CONCAT(FIRST\_NAME, ' ', LAST\_NAME) AS NAME

FROM emp\_record\_table

WHERE DEPT = 'Finance';

**A screenshot of a computer

Description automatically generated**

**Q6) Write a query to list only those employees who have someone reporting to them. Also, show the number of reporters (including the President).**

**Ans.** SELECT E.EMP\_ID, E.FIRST\_NAME, E.LAST\_NAME, E.ROLE, E.EXP,

(SELECT COUNT(\*)

FROM emp\_record\_table R

WHERE R.MANAGER\_ID = E.EMP\_ID

) AS NUM\_REPORTERS

FROM emp\_record\_table E

WHERE E.EMP\_ID = 1 OR E.EMP\_ID IN (SELECT DISTINCT MANAGER\_ID FROM emp\_record\_table WHERE MANAGER\_ID IS NOT NULL)

ORDER BY E.EMP\_ID;

**A screenshot of a computer

Description automatically generated**

**Q7)** **Write a query to list down all the employees from the healthcare and finance departments using union. Take data from the employee record table.**

**Ans.** SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, DEPT

FROM emp\_record\_table

WHERE DEPT = 'HEALTHCARE'

UNION

SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, DEPT

FROM emp\_record\_table

WHERE DEPT = 'FINANCE'

ORDER BY DEPT,EMP\_ID;

**A screenshot of a computer

Description automatically generated**

**Q8)** **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.**

**Ans.**

SELECT

EMP\_ID,

FIRST\_NAME,

LAST\_NAME,

ROLE,

DEPT,

EMP\_RATING,

MAX(EMP\_RATING) OVER (PARTITION BY DEPT) AS MAX\_DEPT\_RATING

FROM

emp\_record\_table

ORDER BY

DEPT;

**A screenshot of a computer

Description automatically generated**

**Q9)** **Write a query to calculate the minimum and the maximum salary of the employees in each role. Take data from the employee record table.**

**Ans.** SELECT ROLE, MAX(SALARY) AS MAX\_SALARY, MIN(SALARY) AS MIN\_SALARY

FROM emp\_record\_table

WHERE ROLE IN ('PRESIDENT', 'LEAD DATA SCIENTIST', 'SENIOR DATA SCIENTIST', 'MANAGER', 'ASSOCIATE DATA SCIENTIST', 'JUNIOR DATA SCIENTIST')

GROUP BY ROLE;

**A screenshot of a computer

Description automatically generated**

**Q10)** **Write a query to assign ranks to each employee based on their experience. Take data from the employee record table.**

**Ans.** SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, EXP,

RANK() OVER (ORDER BY EXP) AS EXP\_RANK

FROM emp\_record\_table;

**A screenshot of a computer

Description automatically generated**

**Q11) 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.**

**Ans.** CREATE VIEW employees\_in\_various\_countries AS

SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, COUNTRY, SALARY

FROM emp\_record\_table

WHERE SALARY > 6000;

SELECT \*FROM employees\_in\_various\_countries;

**A screenshot of a computer

Description automatically generated**

**Q12) Write a nested query to find employees with experience of more than ten years. Take data from the employee record table.**

**Ans.** SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, EXP

FROM emp\_record\_table

WHERE EXP > 10;

**A screenshot of a computer

Description automatically generated**

**Q13)** **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.**

**Ans.** DELIMITER &&

CREATE PROCEDURE get\_experience\_details()

BEGIN

SELECT EMP\_ID,FIRST\_NAME,LAST\_NAME,EXP FROM emp\_record\_table WHERE EXP>3;

END &&

CALL get\_experience\_details();

**A screenshot of a computer

Description automatically generated**

**Q14) 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'.**

**Ans.** DELIMITER //

CREATE FUNCTION Employee\_ROLE(EXP INT)

RETURNS VARCHAR(40)

DETERMINISTIC

BEGIN

DECLARE EmployeeRole VARCHAR(40);

IF EXP <= 2 THEN

SET EmployeeRole = 'JUNIOR DATA SCIENTIST';

ELSEIF EXP > 2 AND EXP <= 5 THEN

SET EmployeeRole = 'ASSOCIATE DATA SCIENTIST';

ELSEIF EXP > 5 AND EXP <= 10 THEN

SET EmployeeRole = 'SENIOR DATA SCIENTIST';

ELSEIF EXP > 10 AND EXP <= 12 THEN

SET EmployeeRole = 'LEAD DATA SCIENTIST';

ELSEIF EXP > 12 AND EXP <= 16 THEN

SET EmployeeRole = 'MANAGER';

ELSE

SET EmployeeRole = 'UNKNOWN';

END IF;

RETURN EmployeeRole;

END //

DELIMITER ;

SELECT EXP, Employee\_ROLE(EXP) AS EmployeeRole

FROM data\_science\_team;

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**Q15)** **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.**

**Ans.**

CREATE INDEX idx\_first\_name

ON emp\_record\_table(FIRST\_NAME(20));

SELECT \* FROM emp\_record\_table

WHERE FIRST\_NAME='Eric';

**A screenshot of a computer

Description automatically generated**

**Q16) Write a query to calculate the bonus for all the employees, based on their ratings and salaries (Use the formula: 5% of salary \* employee rating).**

**Ans.**

SELECT

EMP\_ID,

FIRST\_NAME,

LAST\_NAME,

SALARY,

EMP\_RATING,

(SALARY \* EMP\_RATING \* 0.05) AS BONUS

FROM

emp\_record\_table;

**A screenshot of a computer

Description automatically generated**

**Q17) Write a query to calculate the average salary distribution based on the continent and country. Take data from the employee record table.**

**Ans.**

SELECT

CONTINENT,

COUNTRY,

AVG(SALARY) AS AVERAGE\_SALARY

FROM

emp\_record\_table

GROUP BY

CONTINENT, COUNTRY;

**A screenshot of a computer

Description automatically generated**