**ScienceQtech Employee Performance Mapping.**

Course-end Project 1

DESCRIPTION

ScienceQtech is a startup that works in the Data Science field. ScienceQtech has worked on fraud detection, market basket, self-driving cars, supply chain, algorithmic early detection of lung cancer, customer sentiment, and the drug discovery field. With the annual appraisal cycle around the corner, the HR department has asked you (Junior Database Administrator) to generate reports on employee details, their performance, and on the project that the employees have undertaken, to analyze the employee database and extract specific data based on different requirements.

**Objective:**

To facilitate a better understanding, managers have provided ratings for each employee which will help the HR department to finalize the employee performance mapping. As a DBA, you should find the maximum salary of the employees and ensure that all jobs are meeting the organization's profile standard. You also need to calculate bonuses to find extra cost for expenses. This will raise the overall performance of the organization by ensuring that all required employees receive training.

**Note:** You must download the dataset from the course resource section in LMS and create a table to perform the above objective.

**Dataset description:**

**emp\_record\_table:** It contains the information of all the employees.

* EMP\_ID – ID of the employee
* FIRST\_NAME – First name of the employee
* LAST\_NAME – Last name of the employee
* GENDER – Gender of the employee
* ROLE – Post of the employee
* DEPT – Field of the employee
* EXP – Years of experience the employee has
* COUNTRY – Country in which the employee is presently living
* CONTINENT – Continent in which the country is
* SALARY – Salary of the employee
* EMP\_RATING – Performance rating of the employee
* MANAGER\_ID – The manager under which the employee is assigned
* PROJ\_ID – The project on which the employee is working or has worked on

**Proj\_table:**It contains information about the projects.

* PROJECT\_ID – ID for the project
* PROJ\_Name – Name of the project
* DOMAIN – Field of the project
* START\_DATE – Day the project began
* CLOSURE\_DATE – Day the project was or will be completed
* DEV\_QTR – Quarter in which the project was scheduled
* STATUS – Status of the project currently

**Data\_science\_team:** It contains information about all the employees in the Data Science team.

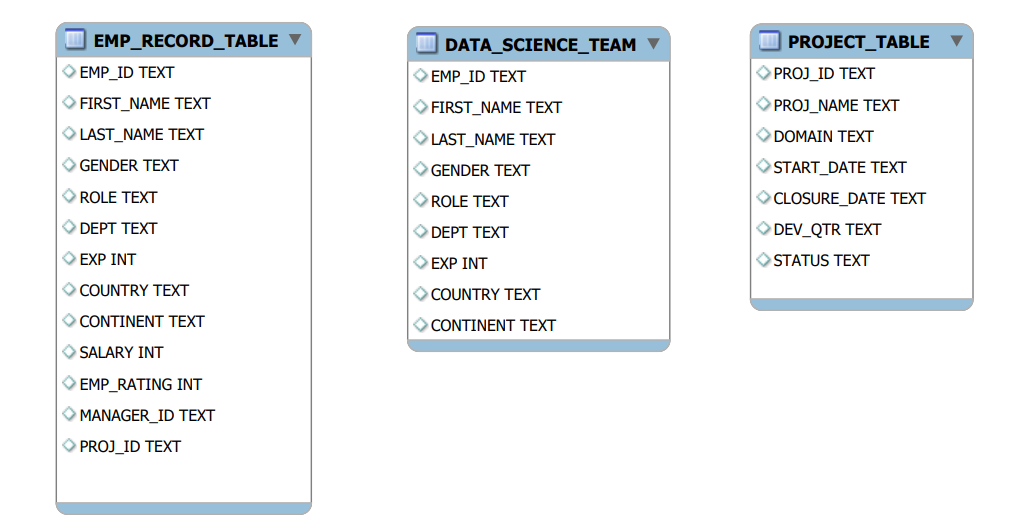
* EMP\_ID – ID of the employee
* FIRST\_NAME – First name of the employee
* LAST\_NAME – Last name of the employee
* GENDER – Gender of the employee
* ROLE – Post of the employee
* DEPT – Field of the employee
* EXP – Years of experience the employee has
* COUNTRY – Country in which the employee is presently living
* CONTINENT – Continent in which the country is

**The task to be performed:**

1. 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.

SQL Query:- CREATE DATABASE employee;

1. Create an ER diagram for the given **employee**database.



1. 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.

**SQL Query**:- SELECT EMP\_ID,FIRST\_NAME,LAST\_NAME,GENDER,DEPT FROM EMPLOYEE.EMP\_RECORD\_TABLE;

1. 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

SQL Query:- SELECT EMP\_ID,FIRST\_NAME,LAST\_NAME,GENDER,DEPT,EMP\_RATING FROM EMPLOYEE.EMP\_RECORD\_TABLE WHERE EMP\_RATING<2;

SELECT EMP\_ID,FIRST\_NAME,LAST\_NAME,GENDER,DEPT,EMP\_RATING FROM EMPLOYEE.EMP\_RECORD\_TABLE WHERE EMP\_RATING>4;

SELECT EMP\_ID,FIRST\_NAME,LAST\_NAME,GENDER,DEPT,EMP\_RATING FROM EMPLOYEE.EMP\_RECORD\_TABLE WHERE EMP\_RATING BETWEEN 2 AND 4;

1. 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.

SQL Query:- SELECT FIRST\_NAME,LAST\_NAME,CONCAT(FIRST\_NAME,' ', LAST\_NAME) AS NAME FROM EMPLOYEE.EMP\_RECORD\_TABLE;

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

SQL Query:- SELECT MANAGER\_ID,COUNT(EMP\_ID) FROM EMPLOYEE.EMP\_RECORD\_TABLE GROUP BY MANAGER\_ID ;

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

SQL Query:- SELECT EMP\_ID,DEPT FROM EMPLOYEE.EMP\_RECORD\_TABLE WHERE DEPT='HEALTHCARE' UNION SELECT EMP\_ID,DEPT FROM EMPLOYEE.EMP\_RECORD\_TABLE WHERE DEPT='FINANCE' ;

1. 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.

SQL Query:- SELECT EMP\_ID,FIRST\_NAME,LAST\_NAME,ROLE,DEPT,EMP\_RATING,MAX(EMP\_RATING) OVER ( PARTITION BY DEPT) MAX\_RATING FROM EMPLOYEE.EMP\_RECORD\_TABLE ;

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

SQL Query:- SELECT MIN(SALARY) OVER (PARTITION BY ROLE) MIN\_SALARY,MAX(SALARY) OVER (PARTITION BY ROLE) MAX\_SALARY,ROLE FROM EMPLOYEE.EMP\_RECORD\_TABLE;

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

SQL Query:- SELECT EMP\_ID,EXP,dense\_rank() OVER(ORDER BY EXP DESC) RANKING FROM EMPLOYEE.EMP\_RECORD\_TABLE;

1. 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.

 SQL Query:- CREATE OR REPLACE VIEW EMP\_RECORD\_VIEW AS SELECT EMP\_ID,COUNTRY,SALARY FROM EMPLOYEE.EMP\_RECORD\_TABLE WHERE SALARY > 6000;

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

SQL Query:- select emp\_id,first\_name ,exp from employee.emp\_record\_table where exp in(select exp from employee.emp\_record\_table where exp>10);

1. 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.

SQL Query:- DELIMITER //

CREATE PROCEDURE EMPEXP\_PROC()

BEGIN

SELECT EMP\_ID,FIRST\_NAME,EXP FROM EMPLOYEE.EMP\_RECORD\_TABLE WHERE EXP>3;

END //

DELIMITER ;;

CALL EMPEXP\_PROC();

1. 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'.

SQL Query:- DELIMITER $$

drop function IF EXISTS employee.emp\_details;

CREATE FUNCTION emp\_details(exp int)

RETURNS VARCHAR(2255) DETERMINISTIC

BEGIN DECLARE emp\_details VARCHAR(2255);

IF exp <= 2 THEN SET emp\_details = 'JUNIOR DATA SCIENTIST';

ELSEIF exp > 2 and exp<=5 THEN SET emp\_details = 'ASSOCIATE DATA SCIENTIST';

ELSEIF exp > 5 and exp<=10 THEN SET emp\_details = 'SENIOR DATA SCIENTIST';

ELSEIF exp > 10 and exp<= 12 THEN SET emp\_details = 'LEAD DATA SCIENTIST';

END IF; RETURN (emp\_details); END$$ DELIMITER $$;

SELECT first\_name, last\_name, dept, emp\_details(exp) as designation FROM

employee.emp\_record\_table ORDER BY exp;

1. 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.

SQL Query:- create index idx\_word on emp\_record\_table(first\_name);

select \* from employee.emp\_record\_table where first\_name='Eric';

1. 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).

SQL Query:- SELECT EMP\_ID,EMP\_RATING,SALARY,0.05\*SALARY\*EMP\_RATING AS BONUS FROM EMPLOYEE.EMP\_RECORD\_TABLE;

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

SQL Query:- SELECT EMP\_ID,COUNTRY,CONTINENT,AVG(SALARY) OVER (PARTITION BY COUNTRY) AVG\_SAL\_COUNTRY,AVG(SALARY) OVER (PARTITION BY CONTINENT) AVG\_SAL\_CONTINENT FROM EMPLOYEE.EMP\_RECORD\_TABLE ;