**SQL – PROJECT**

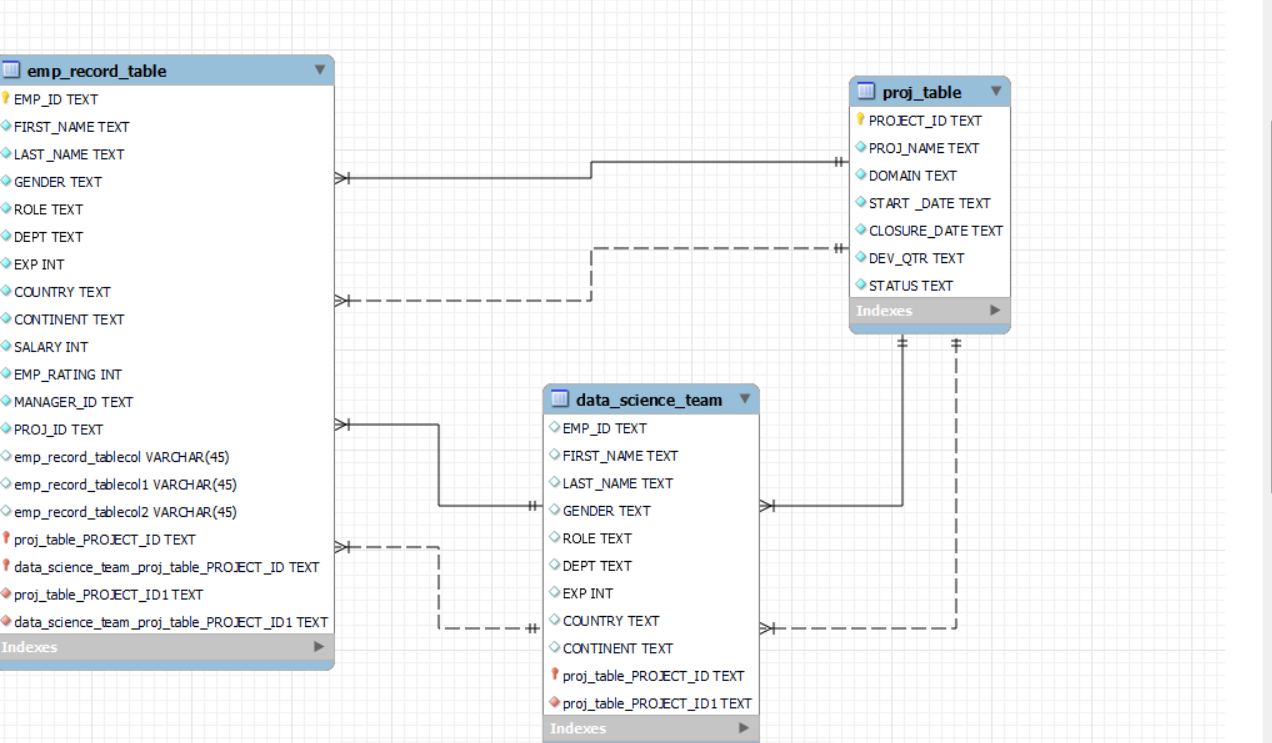
**Science Qtech Employee Performance Mapping.**

* 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.

Query :- **create database employee;**

**use employee;**

* Create an ER diagram for the given employee database.



* 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.

**Query :- select emp\_id, first\_name, last\_name, gender, dept from employee.emp\_record\_table;**

**select first\_name, last\_name, dept from employee.emp\_record\_table;**

* 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.

**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;**

* 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.

**Query :- select first\_name, last\_name, dept, concat(first\_name,' ',Last\_name) as Name**

**From employee.emp\_record\_table where dept = 'Finance';**

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

#### Method 1

**Query :- SELECT emp\_id, first\_name, last\_name, manager\_id as reporters, role**

**FROM emp\_record\_table group by manager\_id;**

#### Method 2

**Query :- SELECT DISTINCT emp\_id, role FROM emp\_record\_table**

**WHERE emp\_id IN (SELECT manager\_id FROM emp\_record\_table);**

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

**Query:- select first\_name, last\_name, dept, emp\_id from employee.emp\_record\_table where**

**DEPT = "Healthcare"**

**union**

**select first\_name, LAST\_NAME, DEPT,EMP\_ID from employee.emp\_record\_table where**

**DEPT = "Finance";**

* 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.

**Query** :- **select emp\_id, first\_name, Last\_name, role, dept, emp\_rating, max(emp\_rating) from**

**employee.emp\_record\_table group by DEPT;**

* Write a query to calculate the minimum and the maximum salary of the employees in each role.

Take data from the employee record table.

**Query :-** **select emp\_id, first\_name, last\_name, role, min(salary) from employee.emp\_record\_table**

**group by role;**

**select emp\_id, first\_name, last\_name, role, max(salary) from employee.emp\_record\_table**

**group by role;**

**select min(salary), max(salary), role from employee.emp\_record\_table group by role;**

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

**Query :- select first\_name, Last\_name, emp\_id, exp, rank() over (order by exp desc) as emp\_rank**

**From employee.emp\_record\_table;**

* 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.

Take data from the employee record table

**Query :-** **create view emp\_country as select emp\_id, concat(first\_name,' ', last\_name), salary, country**

**from employee.emp\_record\_table where salary > 6000;**

**select \* from emp\_country;**

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

**Query :- select exp, concat (first\_name,' ', last\_name) as emp\_name from employee.emp\_record\_table**

**where exp in (select exp from employee.emp\_record\_table where exp > 10);**

* 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.

**Query :- delimiter //**

**create procedure exp\_more\_than\_10 ()**

**begin**

**select \* from employee.emp\_record\_table where exp > 3;**

**end //**

**delimeter;**

**call exp\_more\_than\_10 ();**

* 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'.

**Query :- delimiter $$**

**create procedure get\_emp ( in eid varchar(10), out role varchar (50))**

**begin**

**declare gain int default 1;**

**select exp into gain from employee.emp\_record\_table where emp\_id = eid;**

**if gain <= 2 then set role = 'Junior Data Scientist';**

**elseif gain >2 and gain <= 5 then set role = 'Associate Data Scientist';**

**elseif gain >5 and gain <= 10 then set role = 'Senior Data Scientist' ;**

**elseif gain >10 and gain <=12 then set role = 'Lead Data Scientist';**

**elseif gain >12 and gain < 16 then set role = 'Manager';**

**else set role = 'President';**

**end if;**

**end $$**

**call get\_emp ('E002', @role);**

**select @role ;**

* 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.

**Query :- create index idx on emp\_record\_table(exp);**

**select \* from emp\_record\_table where FIRST\_NAME = 'Eric';**

* 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).

**Query :- select emp\_id, first\_name, last\_name, role, exp, salary, emp\_rating, 0.05\* salary\* emp\_rating as**

**bonus from emp\_record\_table order by bonus;**

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

**Query :- select round(avg(salary)) as avg\_sal, country, continent from emp\_record\_table group by country, CONTINENT;**