Project 1: ScienceQtech Employee Performance Mapping

Objectives

To facilitate a better understanding, managers have provided ratings for each employee which will help the HR department to finalize the employee performance mapping.

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 the project that the employees have undertaken, to analyze the employee database and extract specific data based on differentrequirements.

Problem Statement and Motivation

As a DBA (Database Administrator), 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 the extra cost for expenses.

What motivates the company to do this? This will raise the overall performance of the organization by ensuring that all required employees receive training.
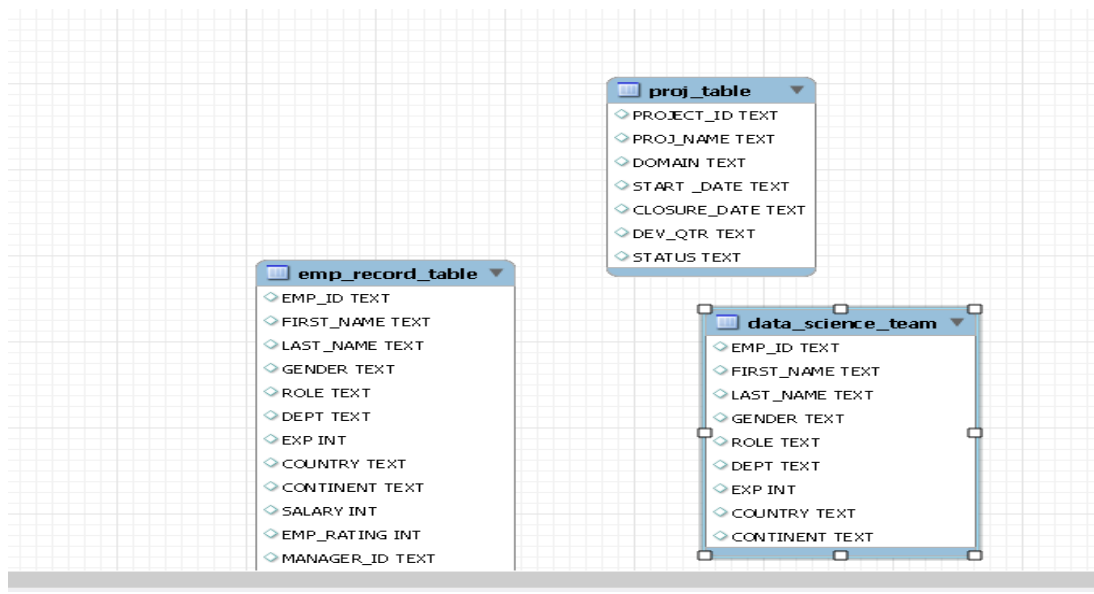
Task

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.

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

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

```
7 •    select  EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT,EMP_RATING
8      from  emp_record_table
9      where EMP_RATING <2 or EMP_RATING >4 or EMP_RATING between 2 and 4 ;
10
11
12
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	DEPT	EMP_RATING
E001	Arthur	Black	M	ALL	5
E005	Eric	Hoffman	M	FINANCE	3
E010	William	Butler	M	AUTOMOTIVE	2
E052	Dianna	Wilson	F	HEALTHCARE	5
E057	Dorothy	Wilson	F	HEALTHCARE	1

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.

```
11 •    select concat(FIRST_NAME,' ',LAST_NAME) as NAME
12      from emp_record_table
13      where DEPT='FINANCE';
14
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

NAME
Eric Hoffman
Emily Grove
Steve Hoffman

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

```
17 •    select Role,count(Role) as EMP_COUNT from emp_record_table
18      group by role;
19
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Role	EMP_COUNT
PRESIDENT	1
LEAD DATA SCIENTIST	2
SENIOR DATA SCIENTIST	5
MANAGER	5
ASSOCIATE DATA SCIENTIST	4

.esult 53 ×

7. 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
select * from emp_record_table where DEPT='healthcare'
union
select * from emp_record_table where DEPT='finance';
```

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT	SALARY	EMP_
E083	Patrick	Voltz	M	MANAGER	HEALTHCARE	15	USA	NORTH AMERICA	9500	5
E505	Chad	Wilson	M	ASSOCIATE DATA SCIENTIST	HEALTHCARE	5	CANADA	NORTH AMERICA	5000	2
E005	Eric	Hoffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA	8500	3
E103	Emily	Grove	F	MANAGER	FINANCE	14	CANADA	NORTH AMERICA	10500	4

Result 50

Output

Action Output

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.

```sql
select  EMP_ID, FIRST_NAME, LAST_NAME,ROLE, GENDER, DEPT,max(EMP_RATING)
    from emp_record_table group by DEPT;
```

EMP_ID	FIRST_NAME	LAST_NAME	ROLE	GENDER	DEPT	max(EMP_RATING)
E001	Arthur	Black	PRESIDENT	M	ALL	5
E005	Eric	Hoffman	LEAD DATA SCIENTIST	M	FINANCE	4
E010	William	Butler	LEAD DATA SCIENTIST	M	AUTOMOTIVE	5
E052	Dianna	Wilson	SENIOR DATA SCIENTIST	F	HEALTHCARE	5
E245	Nian	Zhen	SENIOR DATA SCIENTIST	M	RETAIL	4

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.

```sql
select ROLE,min(salary) as Minimum_Salary, max(salary) as Maximum_Salary
    from emp_record_table
    group by ROLE;
```

ROLE	Minimum_Salary	Maximum_Salary
PRESIDENT	16500	16500
LEAD DATA SCIENTIST	8500	9000
SENIOR DATA SCIENTIST	5500	7700
MANAGER	8500	11000
ASSOCIATE DATA SCIENTIST	4000	5000

Result 54

Output

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

```sql
33 •    select FIRST_NAME,ROLE,DEPT,EXP,
34      row_number() over (order by EXP Desc) Ranking
35      from emp_record_table ;
36
37
```

FIRST_NAME	ROLE	DEPT	EXP	Ranking
Arthur	PRESIDENT	ALL	20	1
Patrick	MANAGER	HEALTHCARE	15	2
Emily	MANAGER	FINANCE	14	3
Pete	MANAGER	AUTOMOTIVE	14	4
Janet	MANAGER	RETAIL	14	5

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.

```sql
39
40 •    CREATE VIEW Employee_Country AS
41      SELECT FIRST_NAME,COUNTRY,SALARY
42      FROM emp_record_table
43      WHERE SALARY >6000 ;
44
45 •    select * from employee_country;
46
47
48
```

FIRST_NAME	COUNTRY	SALARY
Arthur	USA	16500
Eric	USA	8500
William	FRANCE	9000
Dorothy	USA	7700
Patrick	USA	9500

12. Write a nested query to find employees with experience of more than ten years. Take data from the employee record table.
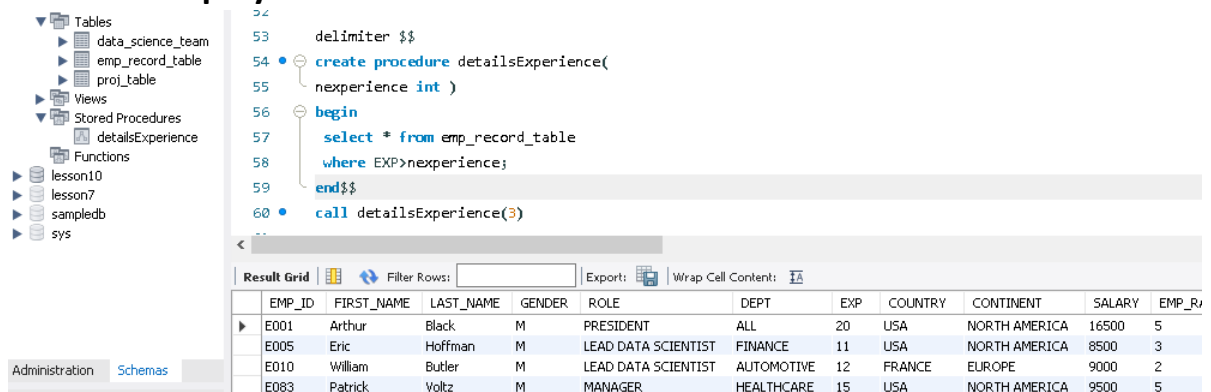
```sql
48 •    select * from emp_record_table
49      where EMP_ID in(
50      select EMP_ID from emp_record_table where Exp>10
51      );
52
53
```

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT	SALARY	EMP_RATI
E001	Arthur	Black	M	PRESIDENT	ALL	20	USA	NORTH AMERICA	16500	5
E005	Eric	Hoffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA	8500	3
E010	William	Butler	M	LEAD DATA SCIENTIST	AUTOMOTIVE	12	FRANCE	EUROPE	9000	2
E083	Patrick	Voltz	M	MANAGER	HEALTHCARE	15	USA	NORTH AMERICA	9500	5

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.**

```
52
53      delimiter $$
54  ●  ⊖  create procedure detailsExperience(
55         nexperience int )
56     ⊖  begin
57           select * from emp_record_table
58           where EXP>nexperience;
59         end$$
60  ●      call detailsExperience(3)
```

Tables
- data_science_team
- emp_record_table
- proj_table

Views

Stored Procedures
- detailsExperience

Functions

lesson10
lesson7
sampledb
sys

Administration   Schemas

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT	SALARY	EMP_R/
E001	Arthur	Black	M	PRESIDENT	ALL	20	USA	NORTH AMERICA	16500	5
E005	Eric	Hoffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA	8500	3
E010	William	Butler	M	LEAD DATA SCIENTIST	AUTOMOTIVE	12	FRANCE	EUROPE	9000	2
E083	Patrick	Voltz	M	MANAGER	HEALTHCARE	15	USA	NORTH AMERICA	9500	5

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 is:**
 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'.

```
62      delimiter $$
63  ●  ⊖  create procedure CheckingRole(
64         in experience int,
65         out output varchar(20),
66         out desig varchar(30)
67         )
68     ⊖  begin
69         declare value varchar(80);
70
71         select Role into value
72         from emp_record_table
73         where EXP=experience;
74     ⊖  case
75           When value = 'JUNIOR DATA SCIENTIST' and experience <=2 then
```

```
76        set output ='True';
77        set desig=value;
78
79     when value =  'ASSOCIATE DATA SCIENTIST'  and experience >2 and experience <5 then
80        set output ='True';
81        set desig=value;
82
83     when value =  'SENIOR DATA SCIENTIST'   and experience >=5 and experience <10 then
84        set output ='True';
85        set desig=value;
86
87     when value =   'LEAD DATA SCIENTIST'   and experience >=10 then
88        set output ='True';
89        set desig=value;
90      else
```

```
91        set output='False';
92        set desig=value;
93     end case;
94     end$$
95     delimiter ;
96
97 ●  call CheckingRole(2,@answer,@roles);
98 ●  select @answer,@roles;
99
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA

@answer	@roles
True	JUNIOR DATA SCIENTIST

Views
Stored Procedures
  detailsExperience
  RoleAssign
Functions
lesson10
  Tables
  Views
  Stored Procedures
  Functions
lesson7
sampledb
sys

Administration   Schemas

Information

No object selected

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.

Solution :

Before Index searching execution plan :

```
101 ●  describe emp_record_table ;
102 ●  select *
103     from emp_record_table where FIRST_NAME='Eric';
104
```

Visual Explain   ▾ Display Info: Read + Eval cost   ▾ | Overview: | View Source:

Query cost: 2.15

query_block #1

2.15   19 rows
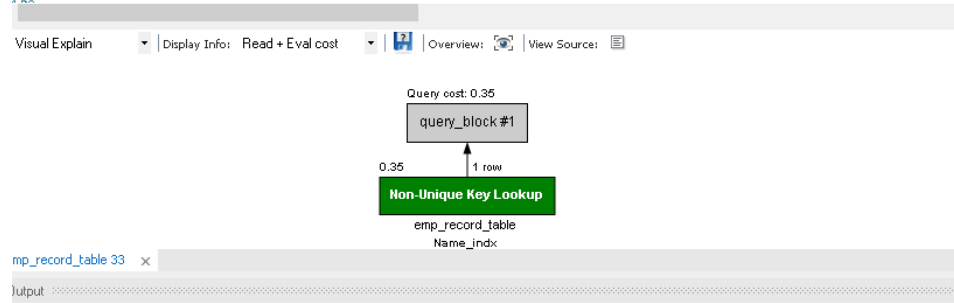
Full Table Scan

emp_record_table

emp_record_table 32 ×

After Index searching execution plan :

```
101 ●   describe emp_record_table ;
102 ●   select *
103     from emp_record_table where FIRST_NAME='Eric';
104
105
106 ●   create index Name_indx  on emp_record_table (FIRST_NAME(10));
107
```

Visual Explain   ▾ | Display Info: Read + Eval cost   ▾ | 🔖 | Overview: 📷 | View Source: 📄

```
                    Query cost: 0.35
                    ┌──────────────┐
                    │ query_block #1│
                    └──────────────┘
             0.35 │        │ 1 row
             ┌──────────────────────┐
             │  Non-Unique Key Lookup │
             └──────────────────────┘
                   emp_record_table
                      Name_indx
```

mp_record_table 33 ✕

Output

16.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).

```
112
113 ●   select FIRST_NAME,ROLE,SALARY,EMP_RATING,
114     ((SALARY * EMP_RATING)*0.05) as bonus
115      from  emp_record_table;
116
```

Result Grid | 🔢 ↩ Filter Rows: [   ] | Export: 🔳 | Wrap Cell Content: 🔤

FIRST_NAME	ROLE	SALARY	EMP_RATING	bonus
Arthur	PRESIDENT	16500	5	4125.00
Eric	LEAD DATA SCIENTIST	8500	3	1275.00
William	LEAD DATA SCIENTIST	9000	2	900.00
Dianna	SENIOR DATA SCIENTIST	5500	5	1375.00
Dorothy	SENIOR DATA SCIENTIST	7700	1	385.00

Result 35 ✕

Output

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

```
117 ●   select country,CONTINENT,
118     avg(salary) over(partition by country )  as Average_Country_Salary,
119     avg(salary) over(partition by CONTINENT)  as Average_CONTINENT_Salary
120     from emp_record_table;
121
```

Result Grid | 🔢 ↩ Filter Rows: [   ] | Export: 🔳 | Wrap Cell Content: 🔤

country	CONTINENT	Average_Country_Salary	Average_CONTINENT_Salary
CHINA	ASIA	6500.0000	6250.0000
INDIA	ASIA	6166.6667	6250.0000
INDIA	ASIA	6166.6667	6250.0000
INDIA	ASIA	6166.6667	6250.0000
FRANCE	EUROPE	9000.0000	7950.0000

Result 42 ✕