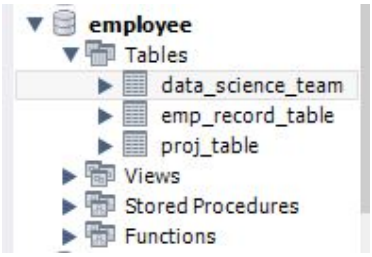
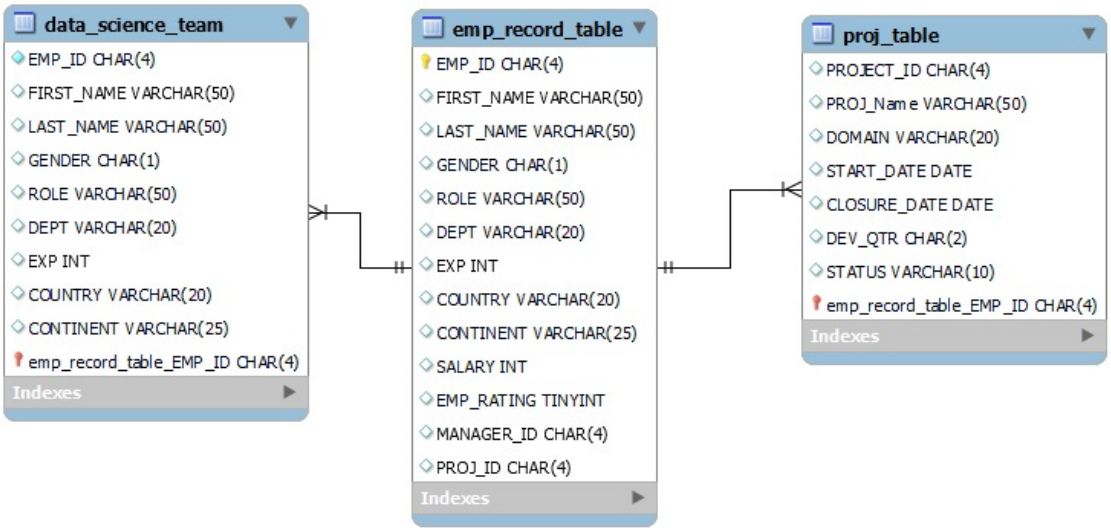
PROJECT 1 - ScienceQtech Employee Performance Mapping.

Screenshots of tasks 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.



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.

```
67 • SELECT EMP_ID,FIRST_NAME,LAST_NAME,GENDER,DEPT
68   FROM emp_record_table;
69
70
```

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

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	DEPT
E001	Arthur	Black	M	ALL
E005	Eric	Hoffman	M	FINANCE
E010	William	Butler	M	AUTOMOTIVE
E052	Dianna	Wilson	F	HEALTHCARE
E057	Dorothy	Wilson	F	HEALTHCARE
E083	Patrick	Voltz	M	HEALTHCARE

emp_record_table 8

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

```
81 • SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING
82   FROM emp_record_table WHERE EMP_RATING > 4;
```

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

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	DEPT	EMP_RATING
E001	Arthur	Black	M	ALL	5
E052	Dianna	Wilson	F	HEALTHCARE	5
E083	Patrick	Voltz	M	HEALTHCARE	5
E204	Karene	Nowak	F	AUTOMOTIVE	5

emp_record_table 10

```
76 • SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING
77   FROM emp_record_table WHERE EMP_RATING < 2;
```

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

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	DEPT	EMP_RATING
E057	Dorothy	Wilson	F	HEALTHCARE	1
E532	Claire	Brennan	F	AUTOMOTIVE	1
E620	Katrina	Allen	F	RETAIL	1

emp_record_table 9

```
86 •  SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING
87    FROM emp_record_table WHERE EMP_RATING BETWEEN 2 AND 4 ;
```

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

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	DEPT	EMP_RATING
E010	William	Butler	M	AUTOMOTIVE	2
E103	Emily	Grove	F	FINANCE	4
E245	Nian	Zhen	M	RETAIL	2
E260	Roy	Collins	M	RETAIL	3
E403	Steve	Hoffman	M	FINANCE	3
E428	Pete	Allen	M	AUTOMOTIVE	4
E478	David	Smith	M	RETAIL	4
E505	Chad	Wilson	M	HEALTHCARE	2
E583	Janet	Hale	F	RETAIL	2
E612	Tracy	Norris	F	RETAIL	4
E640	Jenifer	Jhones	F	RETAIL	4

emp_record_table 11 ×

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.

```
91 •  SELECT
92    CONCAT(FIRST_NAME,' ',LAST_NAME) AS NAME FROM emp_record_table
93    WHERE DEPT='FINANCE';
```

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

NAME
Eric Hoffman
Emily Grove
Steve Hoffman

Result 13 ×

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

```
97 •  SELECT m.EMP_ID,m.FIRST_NAME,m.LAST_NAME,m.ROLE,
98    m.EXP,COUNT(e.EMP_ID) as "EMP_COUNT"
99    FROM emp_record_table m
100   INNER JOIN emp_record_table e
101   ON m.EMP_ID = e.MANAGER_ID
102   GROUP BY m.EMP_ID
103   ORDER BY m.EMP_ID;
```

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

EMP_ID	FIRST_NAME	LAST_NAME	ROLE	EXP	EMP_COUNT
E001	Arthur	Black	PRESIDENT	20	5
E083	Patrick	Voltz	MANAGER	15	3
E103	Emily	Grove	MANAGER	14	2
E428	Pete	Allen	MANAGER	14	3
E583	Janet	Hale	MANAGER	14	3

Result 15

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.

```
107 •  SELECT EMP_ID,FIRST_NAME,LAST_NAME,DEPT FROM emp_record_table
108    WHERE DEPT = "HEALTHCARE"
109    UNION
110    SELECT EMP_ID,FIRST_NAME,LAST_NAME,DEPT FROM emp_record_table
111    WHERE DEPT = "FINANCE"
112    ORDER BY DEPT,EMP_ID;
```

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

EMP_ID	FIRST_NAME	LAST_NAME	DEPT
E005	Eric	Hoffman	FINANCE
E103	Emily	Grove	FINANCE
E403	Steve	Hoffman	FINANCE
E052	Dianna	Wilson	HEALTHCARE
E057	Dorothy	Wilson	HEALTHCARE

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.

```
116 •  SELECT  m.EMP_ID,m.FIRST_NAME,m.LAST_NAME,m.ROLE,m.DEPT,m.EMP_RATING,max(m.EMP_RATING)
117    OVER(PARTITION BY m.DEPT)
118    AS "MAX_DEPT_RATING"
119    FROM emp_record_table m
120    ORDER BY DEPT;
```

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

EMP_ID	FIRST_NAME	LAST_NAME	ROLE	DEPT	EMP_RATING	MAX_DEPT_RATING
E001	Arthur	Black	PRESIDENT	ALL	5	5
E010	William	Butler	LEAD DATA SCIENTIST	AUTOMOTIVE	2	5
E204	Karene	Nowak	SENIOR DATA SCIENTIST	AUTOMOTIVE	5	5
E428	Pete	Allen	MANAGER	AUTOMOTIVE	4	5
E532	Claire	Brennan	ASSOCIATE DATA SCIENTIST	AUTOMOTIVE	1	5

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.

```
.24 •  SELECT ROLE, MIN(SALARY) AS MIN_SALARY, MAX(SALARY) AS MAX_SALARY
.25    FROM emp_record_table GROUP BY ROLE;
```

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

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

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

```
129 •  SELECT EMP_ID,FIRST_NAME,LAST_NAME,EXP,
130    RANK() OVER(ORDER BY EXP) EXP_RANK
131    FROM emp_record_table;
```

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

EMP_ID	FIRST_NAME	LAST_NAME	EXP	EXP_RANK
E640	Jenifer	Jhones	1	1
E620	Katrina	Allen	2	2
E478	David	Smith	3	3
E532	Claire	Brennan	3	3
E403	Steve	Hoffman	4	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.

```
135 •  CREATE VIEW employees_in_various_countries AS
136    SELECT EMP_ID,FIRST_NAME,LAST_NAME,COUNTRY,SALARY
137    FROM emp_record_table
138    WHERE SALARY>6000;
139
140 •  SELECT *FROM employees_in_various_countries;
```

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

EMP_ID	FIRST_NAME	LAST_NAME	COUNTRY	SALARY
E001	Arthur	Black	USA	16500
E005	Eric	Hoffman	USA	8500
E010	William	Butler	FRANCE	9000
E057	Dorothy	Wilson	USA	7700
E083	Patrick	Voltz	USA	9500

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

```
144 •  SELECT EMP_ID,FIRST_NAME,LAST_NAME,EXP FROM emp_record_table
145    WHERE EMP_ID IN(SELECT manager_id FROM emp_record_table);
```

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

EMP_ID	FIRST_NAME	LAST_NAME	EXP
E001	Arthur	Black	20
E083	Patrick	Voltz	15
E103	Emily	Grove	14
E428	Pete	Allen	14
E583	Janet	Hale	14

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.

```
149     DELIMITER &&
150 •   CREATE PROCEDURE get_experience_details()
151     BEGIN
152     SELECT EMP_ID,FIRST_NAME,LAST_NAME,EXP FROM emp_record_table WHERE EXP>3;
153     END &&
154 •   CALL get_experience_details();
```

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

EMP_ID	FIRST_NAME	LAST_NAME	EXP
E001	Arthur	Black	20
E005	Eric	Hoffman	11
E010	William	Butler	12
E052	Dianna	Wilson	6
E057	Dorothy	Wilson	9

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'.

```
164     DELIMITER &&
165     CREATE FUNCTION Employee_ROLE(EXP int)
166     RETURNS VARCHAR(40)
167     DETERMINISTIC
168     BEGIN DECLARE Employee_ROLE VARCHAR(40);
169     IF EXP>12 AND 16 THEN SET Employee_ROLE="MANAGER";
170     ELSEIF EXP>10 AND 12 THEN SET Employee_ROLE ="LEAD DATA SCIENTIST";
171     ELSEIF EXP>5 AND 10 THEN SET Employee_ROLE ="SENIOR DATA SCIENTIST";
172     ELSEIF EXP>2 AND 5 THEN SET Employee_ROLE ="ASSOCIATE DATA SCIENTIST";
173     ELSEIF EXP<=2 THEN SET Employee_ROLE ="JUNIOR DATA SCIENTIST";
174     END IF;
175     RETURN (Employee_ROLE);
176     END &&
177
178     SELECT EXP,Employee_ROLE(EXP)
179     FROM data_science_team;
```

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

EXP	Employee_ROLE(EXP)
11	LEAD DATA SCIENTIST
12	LEAD DATA SCIENTIST
6	SENIOR DATA SCIENTIST
9	SENIOR DATA SCIENTIST
8	SENIOR DATA SCIENTIST

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.

```
183     CREATE INDEX idx_first_name
184     ON emp_record_table(FIRST_NAME(20));
185     SELECT * FROM emp_record_table
186     WHERE FIRST_NAME='Eric';
187
```

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

Query cost: 1.55

query_block #1

1.55 | 13 rows

Full Table Scan

data_science_team
Access Type: ALL
Full Table Scan
Cost Hint: Very High - very costly for large tables (not so much for small ones).
No usable indexes were found for the table and the optimizer must search every row.
This could also mean the search range is so broad that the index would be useless.
Used Columns: EXP

Key/Index: -

Rows Examined per Scan: 13
Rows Produced per Join: 13
Filtered (ratio of rows produced per rows examined): 100.00%
Hint: 100% is best, <= 1% is worst
A low value means the query examines a lot of rows that are not returned.

Cost Info
Read: 0.25
Eval: 1.30
Prefix: 1.55
Data Read: 11K

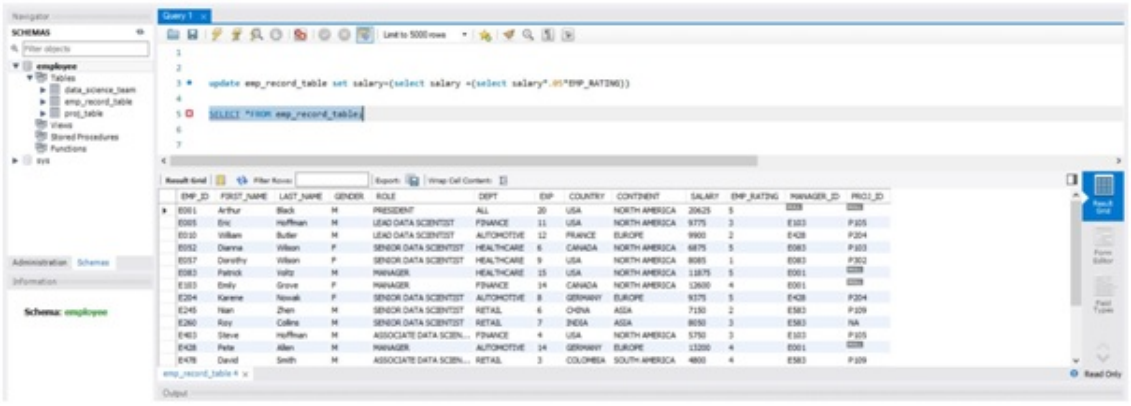
Result 38

Output

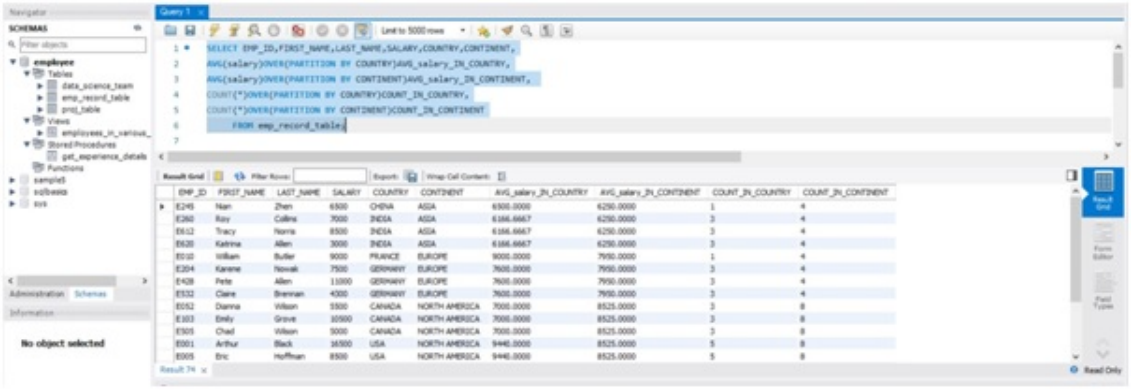
Action Output

#	Time	Action
105	21:41:02	EXPLAIN SELECT EXP,Employee_ROLE(EXP) FROM data_scien
106	21:41:02	EXPLAIN FORMAT=JSON SELECT EXP,Employee_ROLE(EXP)

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)



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



- By Mohammed Jabir V K
jabirasdftv@gmail.com