By Archana Kumari

POJECT WRITE-UP:

SCIENCEQTECH EMPLOYEE PERFORMANCE MAPPING.

Objectives:

The main goal of this project is to assist the HR department in finalizing employee performance mapping for the annual appraisal cycle. This involves analyzing employee details, their performance ratings, and the projects they have worked on.

Specific tasks include: Identifying the maximum salary of employees. Ensuring all job roles meet the organization's profile standards. Calculating bonuses to estimate extra costs. Providing data to ensure employees receive necessary training. Tools and Techniques Database Management: Creation and manipulation of databases. SQL Queries: To extract and manipulate data. ER Diagrams: To visualize relationships between different data tables. Stored Procedures and Functions: For efficient data processing. Indexing: To optimize query performance.

Dataset Description:

The dataset includes the following features: -

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

PROJECT TASKS:

Q 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.

Solution:

1. Launch MySQL Workbench:

Connect to mysql database. On the left side of MySQL Workbench, right-click and select 'Create Schema'. Name the new schema as employee, then click 'Apply'. Navigate to the newly created schema “employee”, click on it, and proceed to Tables option. Right-click on 'Tables'.

2. Navigate to Import Wizard:

- Go to File -> Import Data - > From File... import data_science_team.csv.

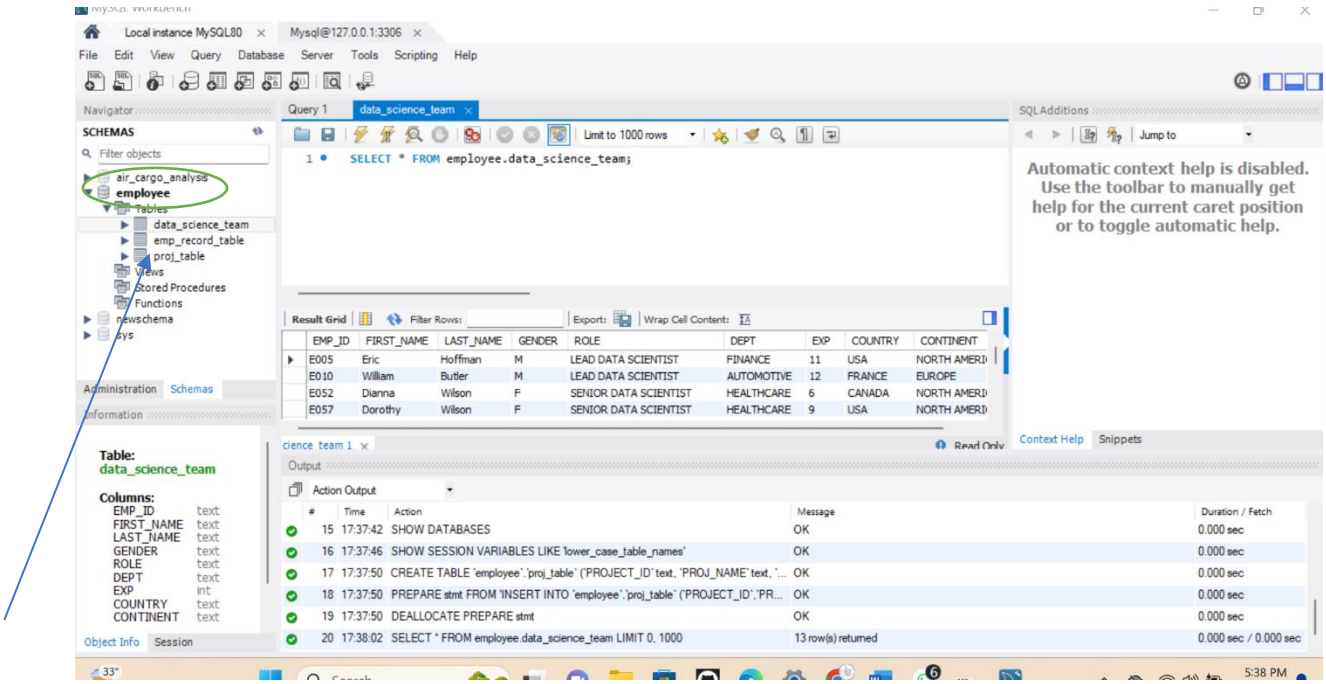
3. Specify CSV File:

- Select your CSV file "import data_science_team.csv "using the file browser.

6. Execute Import:

- Click Import or Next to execute the import process.

Same process repeats from second step for importing other CSV files such as "proj_table.csv" and "emp_record_table.csv" file.



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

Solution:

To create an ER (Entity-Relationship) diagram for the given employee database, we will identify the entities, their attributes, and the relationships between them.

Entities and Attributes:

1. emp_record_table:

- EMP_ID (Primary Key)
- FIRST_NAME
- LAST_NAME
- GENDER
- ROLE
- DEPT
- EXP
- COUNTRY
- CONTINENT
- SALARY
- EMP_RATING
- MANAGER_ID (Foreign Key referencing EMP_ID)
- PROJ_ID (Foreign Key referencing Proj_table.PROJECT_ID)

2. Proj_table:
   - PROJECT_ID (Primary Key)
   - PROJ_Name
   - DOMAIN
   - START_DATE
   - CLOSURE_DATE
   - DEV_QTR
   - STATUS
3. Data_science_team:
   - EMP_ID (Primary Key, Foreign Key referencing emp_record_table.EMP_ID)

- FIRST_NAME

- LAST_NAME

- GENDER

- ROLE

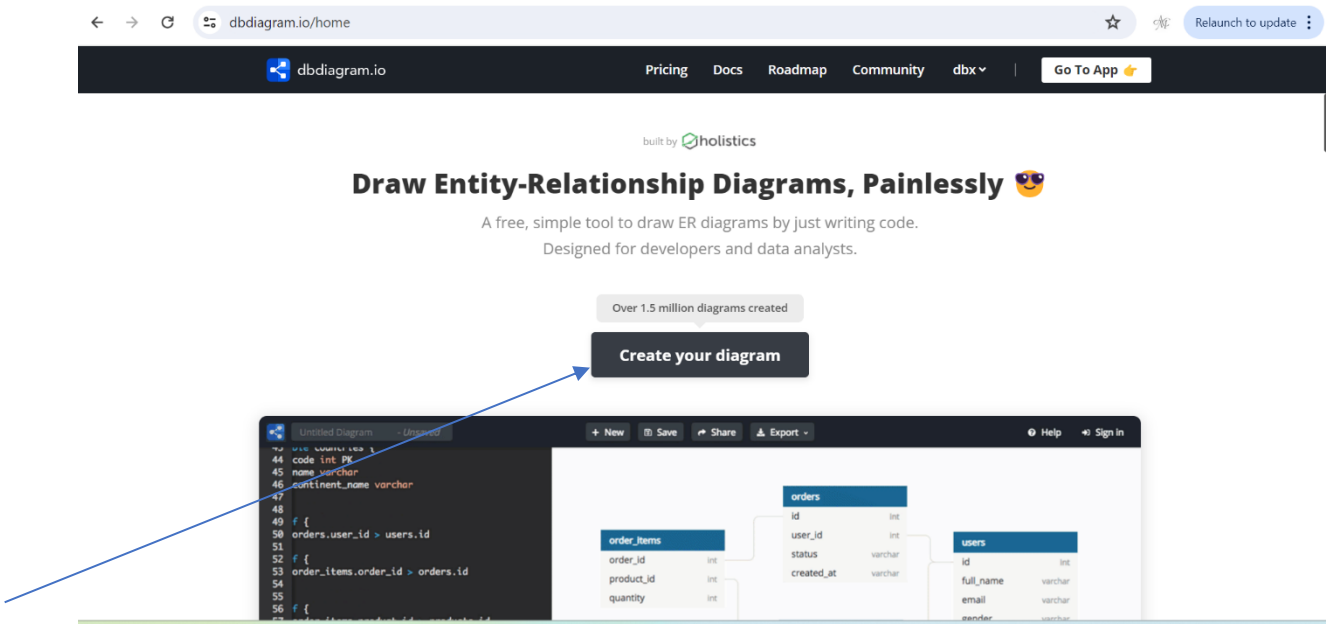
- DEPT

- EXP

- COUNTRY

- CONTINENT

I used **dbDiagram.io** to create an ER diagram, and here is the link https://dbdiagram.io.



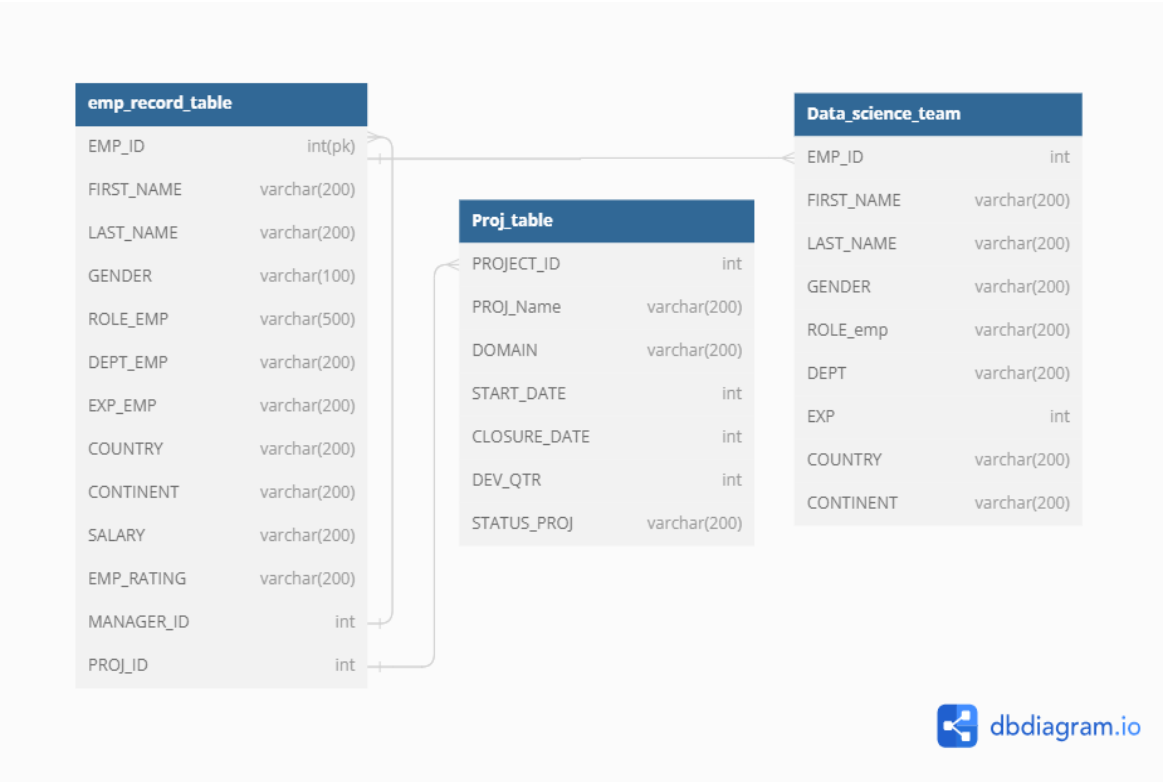
Relationships:

-One-to-Many relationship between emp_record_table and Proj_table (one employee can work on many projects, one project can have many employees).

-One-to-Many self-relationship in emp_record_table for MANAGER_ID (one manager can manage many employees).

-One-to-One relationship between emp_record_table and Data_science_team (each data science team member is a unique employee).

Here is the source code for the ER diagram https://dbdiagram.io/d/ER_DIAGRAM_1-667edf579939893dae8d8fd7.



Here is the textual representation of the ER diagram:

[emp_record_table]

EMP_ID (PK)

FIRST_NAME

LAST_NAME

GENDER

ROLE_EMP

DEPT_EMP

EXP_EMP

COUNTRY

CONTINENT

SALARY

EMP_RATING

MANAGER_ID (FK -> emp_record_table.EMP_ID)

PROJ_ID (FK -> Proj_table.PROJECT_ID)

| 1

|

|

| *

[Proj_table]

PROJECT_ID (PK)

PROJ_Name

DOMAIN

START_DATE

CLOSURE_DATE

DEV_QTR

STATUS

| 1

|

|

|

| 1

[Data_science_team]

EMP_ID (PK, FK -> emp_record_table.EMP_ID)

FIRST_NAME

LAST_NAME

GENDER

ROLE_emp

DEPT

EXP

COUNTRY

CONTINENT

Q 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.

Solution:

```
SELECT

EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT

FROM

emp_record_table;
```

(PRESS CONTROL + ENTER)

Query 1 data_science_team emp_record_table

Limit to 1000 rows

```
SELECT
    EMP_ID,
    FIRST_NAME,
    LAST_NAME,
    GENDER,
    DEPT
FROM
    emp_record_table;
```

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

Q4. Write a query to fetch EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPARTMENT, and EMP_RATING if the EMP_RATING is:

1. less than two:

Solution:

SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING FROM emp_record_table WHERE EMP_RATING < 2;

Query 1 data_science_team emp_record_table

Limit to 1000 rows

```
SELECT
    EMP_ID, FIRST_NAME, LAST_NAME,
    GENDER,
    DEPT,
    EMP_RATING
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

2. greater than four:

Solution:

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

Query 1 | data_science_team | emp_record_table

Limit to 1000 rows

```
16        emp_record_table
17    WHERE
18        EMP_RATING < 2;
19        SELECT
20        EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING
21    FROM
22        emp_record_table
23    WHERE
24        EMP_RATING > 4;
```

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

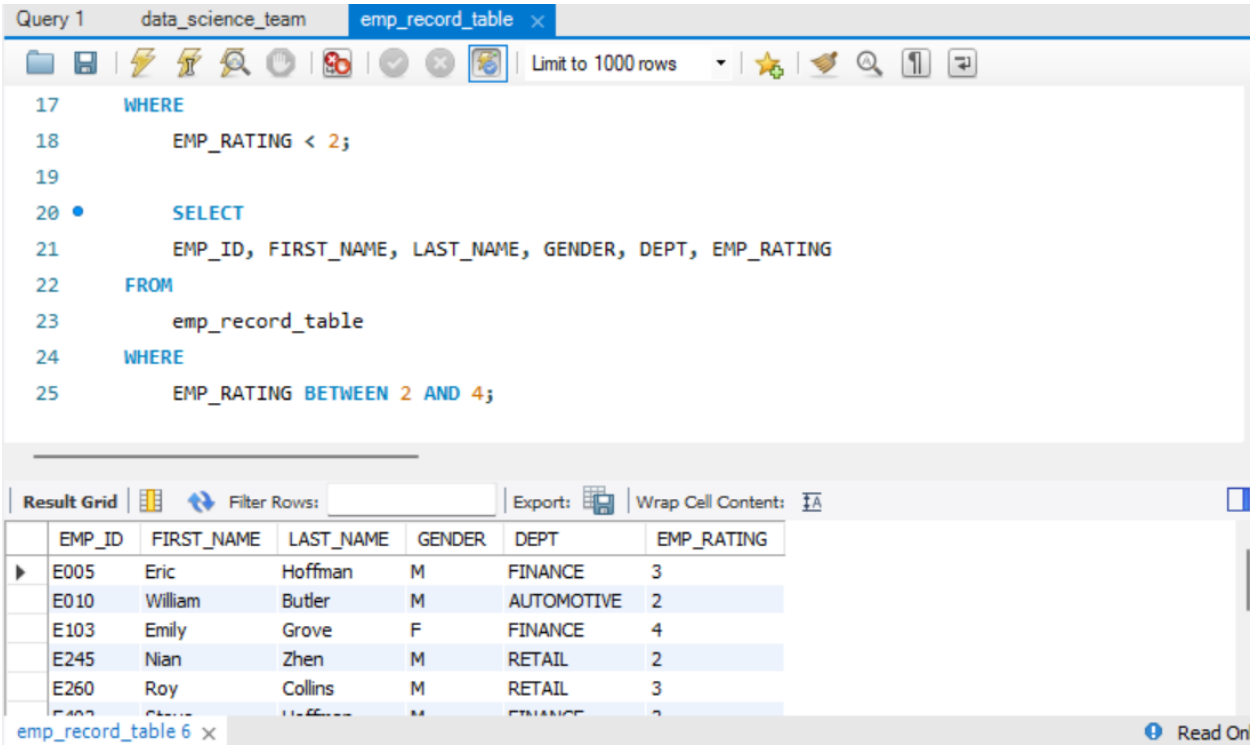
EMP_ID	FIRST_NAME	LAST_NAME	GENDER	DEPT	EMP_RATING
E001	Arthur	Black	M	ALL	5
E052	Dianna	Wilson	F	HEALTHCARE	5
E083	Patrick	Voltz	M	HEALTHCARE	5
E204	Karene	Nowak	F	AUTOMOTIVE	5

3. between two and four:

Solution:

```
SELECT
    EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING
FROM
```

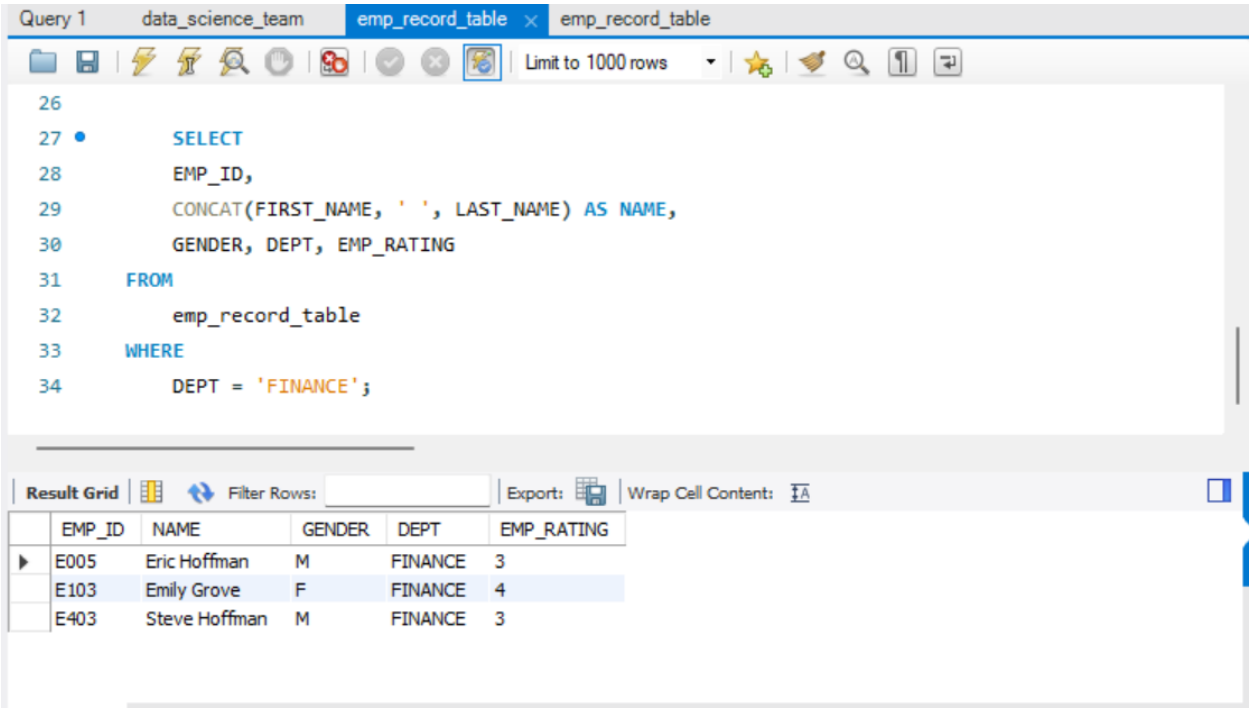
```
    emp_record_table
WHERE
    EMP_RATING BETWEEN 2 AND 4;
```



Q 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.

Solution:

```
SELECT
    EMP_ID,
    CONCAT(FIRST_NAME, ' ', LAST_NAME) AS NAME,
    GENDER, DEPT, EMP_RATING
FROM
    emp_record_table
WHERE DEPT = 'FINANCE';
```



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

Solution:

```
SELECT
    EMP_ID, FIRST_NAME, LAST_NAME, ROLE, DEPT, EMP_RATING,
    COUNT(EMP_ID) AS NUM_REPORTERS
FROM
    emp_record_table
GROUP BY
   EMP_ID, FIRST_NAME, LAST_NAME, ROLE, DEPT, EMP_RATING
ORDER BY
    NUM_REPORTERS DESC;
```

Query 1 | data_science_team | emp_record_table | emp_record_table

Limit to 1000 rows

```
185 •       SELECT
186         EMP_ID, FIRST_NAME, LAST_NAME, ROLE, DEPT, EMP_RATING,
187         COUNT(EMP_ID) AS NUM_REPORTERS
188     FROM
189         emp_record_table
190
191     GROUP BY
192        EMP_ID, FIRST_NAME, LAST_NAME, ROLE, DEPT, EMP_RATING
193     ORDER BY
194         NUM_REPORTERS DESC;
```

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

EMP_ID	FIRST_NAME	LAST_NAME	ROLE	DEPT	EMP_RATING	NUM_REPORTERS
E001	Arthur	Black	PRESIDENT	ALL	5	1
E005	Eric	Hoffman	LEAD DATA SCIENTIST	FINANCE	3	1
E010	William	Butler	LEAD DATA SCIENTIST	AUTOMOTIVE	2	1
E052	Dianna	Wilson	SENIOR DATA SCIENTIST	HEALTHCARE	5	1
E057	Dorothy	Wilson	SENIOR DATA SCIENTIST	HEALTHCARE	1	1

Q 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.

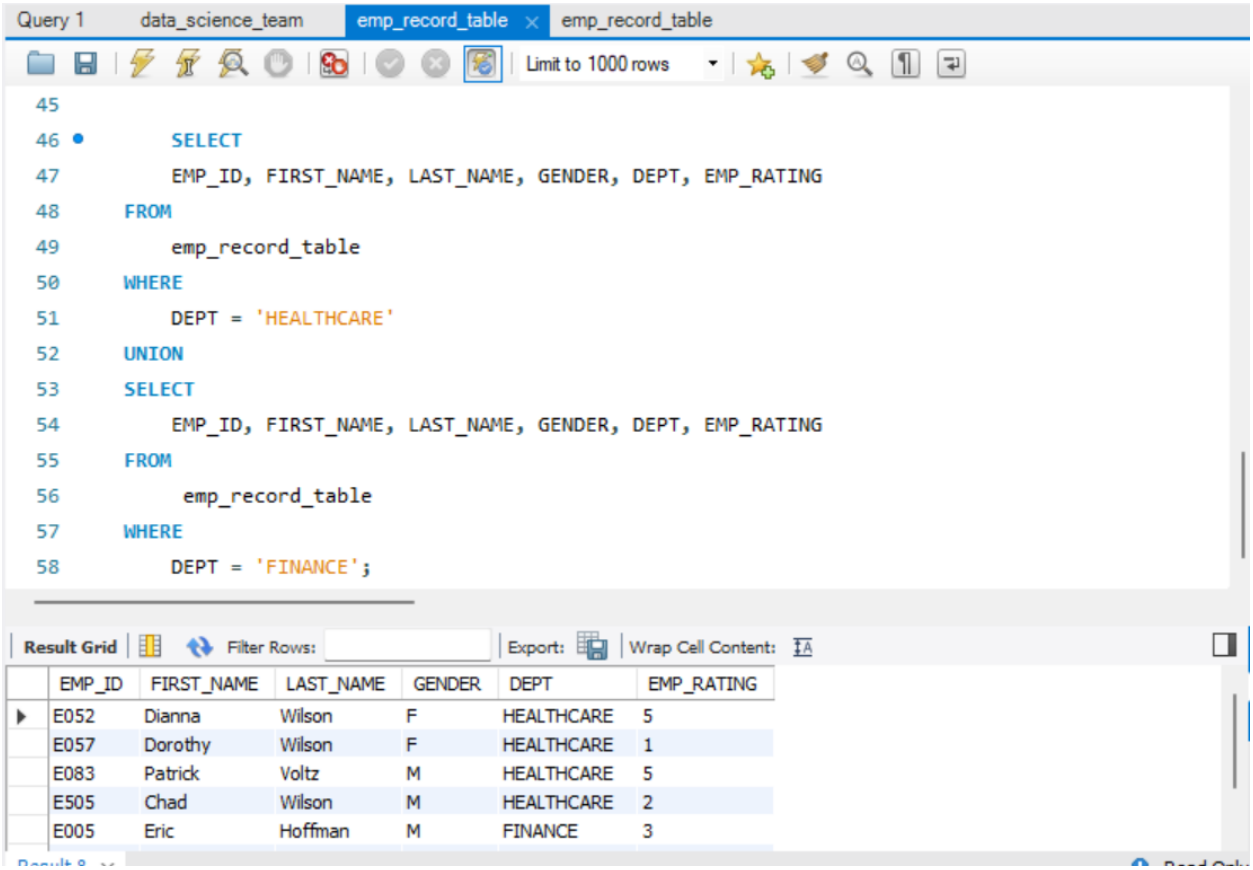
Solution:

```
SELECT
    EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING
FROM
    emp_record_table
WHERE
    DEPT = 'HEALTHCARE'
UNION
SELECT
    EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING
FROM
```

emp_record_table

WHERE

DEPT = 'FINANCE';



Q 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.

Solution:

SELECT

EMP_ID, FIRST_NAME, LAST_NAME, ROLE, DEPT, EMP_RATING,

max(EMP_RATING) over (partition by DEPT) AS MAX_DEPT_RATING

FROM

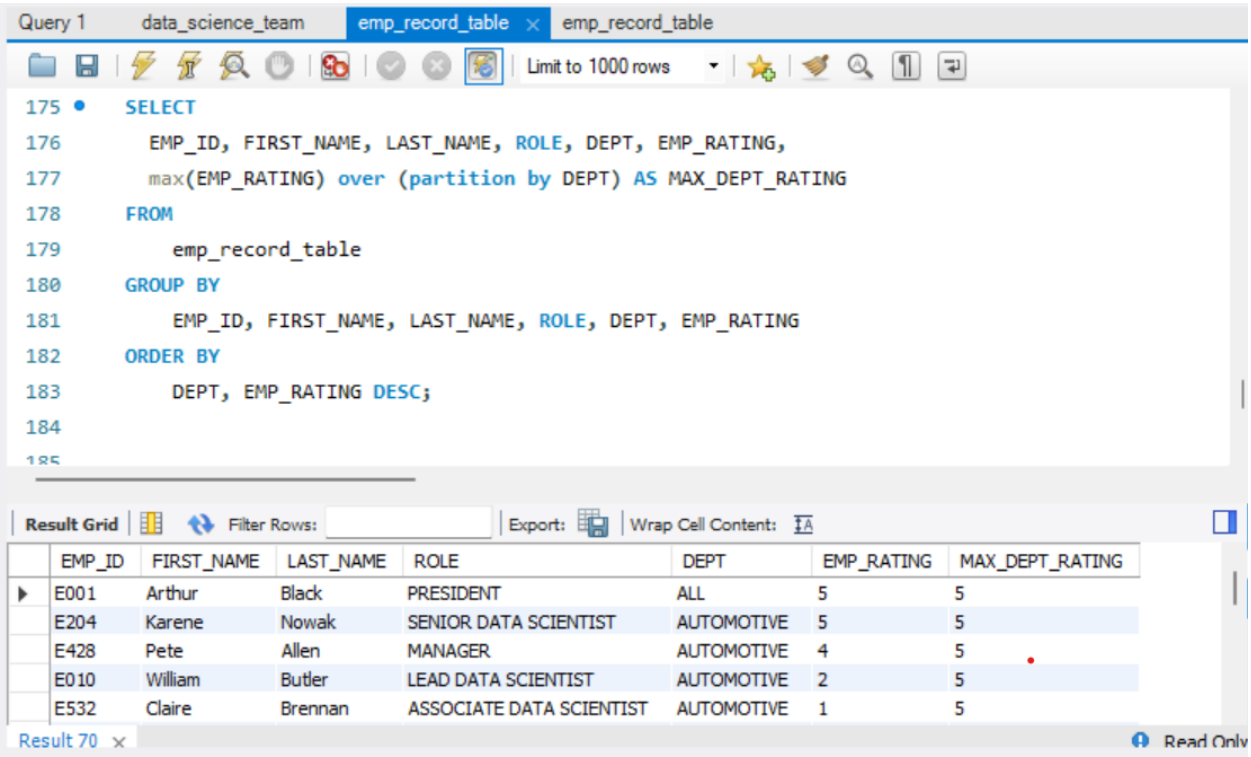
emp_record_table

GROUP BY

EMP_ID, FIRST_NAME, LAST_NAME, ROLE, DEPT, EMP_RATING

ORDER BY

DEPT, EMP_RATING DESC;



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

Q 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.

Solution:

SELECT

ROLE,

MIN(SALARY) AS MIN_SALARY,

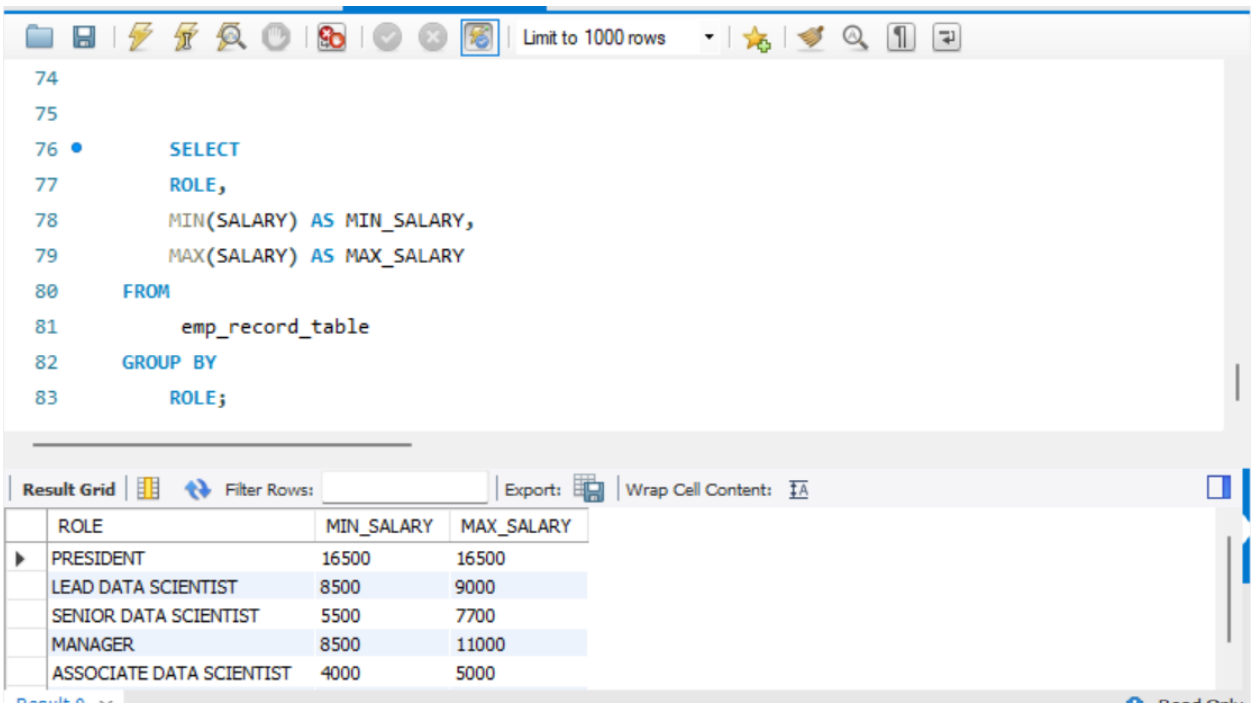
MAX(SALARY) AS MAX_SALARY

FROM

emp_record_table

GROUP BY

ROLE;



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

Solution:

SELECT

EMP_ID, FIRST_NAME, LAST_NAME,

ROLE,

DEPT, EXP AS EXPERIENCE,

RANK() OVER (ORDER BY EXP DESC) AS EXPERIENCE_RANK

FROM

emp_record_table;

Query 1 | data_science_team | emp_record_table | emp_record_table

Limit to 1000 rows

```
        emp_record_table
    GROUP BY
        ROLE;
        SELECT
        EMP_ID, FIRST_NAME, LAST_NAME,
        ROLE,
        DEPT, EXP AS EXPERIENCE,
        RANK() OVER (ORDER BY EXP DESC) AS EXPERIENCE_RANK
    FROM
         emp_record_table;
```

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

EMP_ID	FIRST_NAME	LAST_NAME	ROLE	DEPT	EXPERIENCE	EXPERIENCE_RANK
E001	Arthur	Black	PRESIDENT	ALL	20	1
E083	Patrick	Voltz	MANAGER	HEALTHCARE	15	2
E103	Emily	Grove	MANAGER	FINANCE	14	3
E428	Pete	Allen	MANAGER	AUTOMOTIVE	14	3
E583	Janet	Hale	MANAGER	RETAIL	14	3

Result 10 | Read Only

Q 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.

Solution:

```
CREATE VIEW high_salary_employees_by_country AS

SELECT  FIRST_NAME, LAST_NAME, SALARY, COUNTRY

FROM emp_record_table

WHERE  SALARY > 6000;

SELECT * FROM high_salary_employees_by_country;
```

Query 1 | data_science_team | emp_record_table | emp_record_table

Limit to 1000 rows

```
        DEPT, EXP AS EXPERIENCE,
        RANK() OVER (ORDER BY EXP DESC) AS EXPERIENCE_RANK
    FROM
          emp_record_table;

          CREATE VIEW high_salary_employees_by_country AS
    SELECT  FIRST_NAME, LAST_NAME, SALARY, COUNTRY
    FROM emp_record_table
    WHERE  SALARY > 6000;
    SELECT * FROM high_salary_employees_by_country;
```

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

FIRST_NAME	LAST_NAME	SALARY	COUNTRY
Arthur	Black	16500	USA
Eric	Hoffman	8500	USA
William	Butler	9000	FRANCE
Dorothy	Wilson	7700	USA
Patrick	Voltz	9500	USA

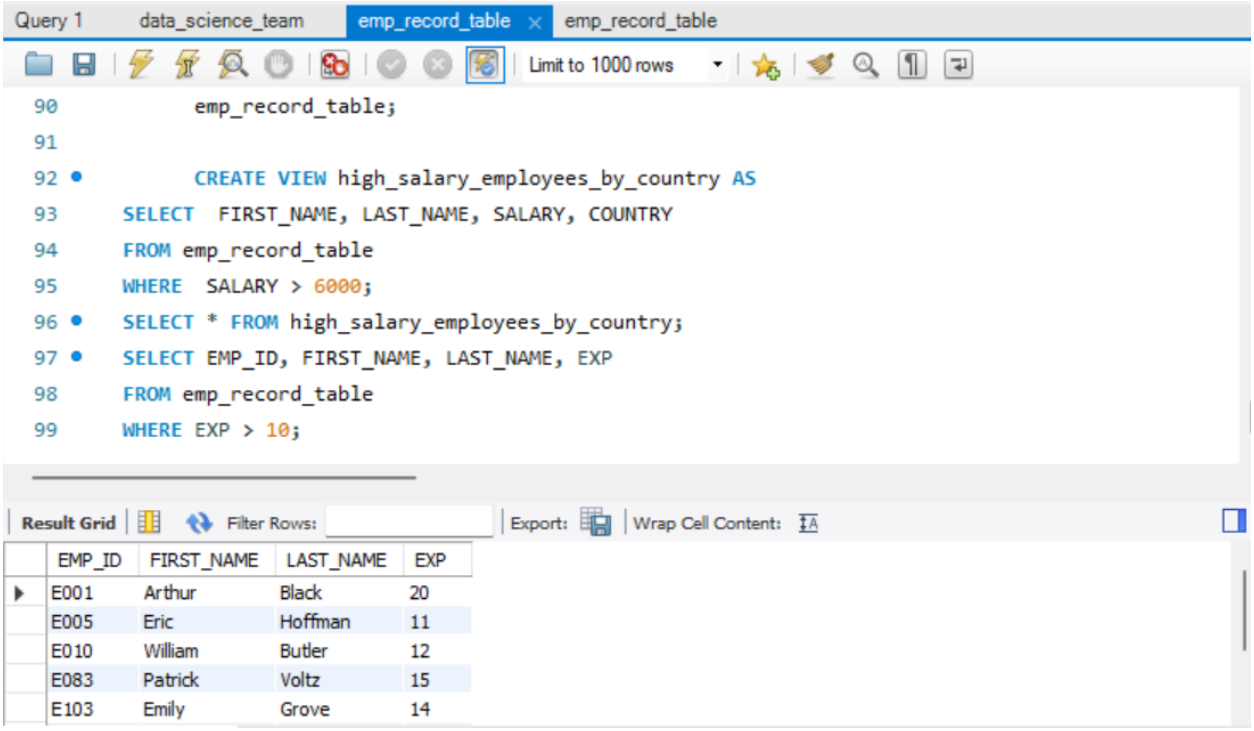
Q 12. Write a nested query to find employees with experience of more than ten years. Take data from the employee record table.

Solution:

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP

FROM emp_record_table

WHERE EXP > 10;
```



Q 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.

Solution:

```
CREATE PROCEDURE GetEmployeesWithExperience()

BEGIN

   SELECT

      EMP_ID, FIRST_NAME, LAST_NAME, GENDER, ROLE, DEPT, EXP, COUNTRY, CONTINENT, SALARY, EMP_RATING, MANAGER_ID, PROJ_ID

   FROM

      emp_record_table

   WHERE

      EXP > 3;

 END // DELIMITER;

CALL GetEmployeesWithExperience();
```

Query 1 | data_science_team | emp_record_table | emp_record_table

Limit to 1000 rows

```
CREATE PROCEDURE GetEmployeesWithExperience()
BEGIN
    SELECT
        EMP_ID, FIRST_NAME, LAST_NAME, GENDER, ROLE, DEPT, EXP, COUNTRY, CONTINENT,SALARY,
        EMP_RATING, MANAGER_ID, PROJ_ID
    FROM
        emp_record_table
    WHERE
        EXP > 3;
END //

DELIMITER ;
```

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

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT
E001	Arthur	Black	M	PRESIDENT	ALL	20	USA	NORTH AMERICA
E005	Eric	Hoffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA
E010	William	Butler	M	LEAD DATA SCIENTIST	AUTOMOTIVE	12	FRANCE	EUROPE
E052	Dianna	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	6	CANADA	NORTH AMERICA

Result 50 | Read Only

Output

Action Output

#	Time	Action	Message
106	11:05:31	CREATE PROCEDURE GetEmployeesWithExperience() BEGIN SELECT ...	0 row(s) affected

Q 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'.

Solution:

```
CREATE FUNCTION CheckJobProfile(exp INT, role VARCHAR(50))

RETURNS VARCHAR(50)
```

```
DETERMINISTIC
BEGIN
    DECLARE expected_role VARCHAR(50);
    IF exp <= 2 THEN
        SET expected_role = 'JUNIOR DATA SCIENTIST';
    ELSEIF exp > 2 AND exp <= 5 THEN
        SET expected_role = 'ASSOCIATE DATA SCIENTIST';
    ELSEIF exp > 5 AND exp <= 10 THEN
        SET expected_role = 'SENIOR DATA SCIENTIST';
    ELSEIF exp > 10 AND exp <= 12 THEN
        SET expected_role = 'LEAD DATA SCIENTIST';
    ELSEIF exp > 12 AND exp <= 16 THEN
        SET expected_role = 'MANAGER';
    ELSE
        SET expected_role = 'UNKNOWN';  -- For experience greater than 16 years
    END IF;
    IF role = expected_role THEN
        RETURN 'MATCHES';
    ELSE
        RETURN 'DOES NOT MATCH';
    END IF;
END //
```

DELIMITER ;

Query 1 | data_science_team | emp_record_table | emp_record_table

Limit to 1000 rows

```
RETURNS VARCHAR(50)
DETERMINISTIC
BEGIN
    DECLARE expected_role VARCHAR(50);
    IF exp <= 2 THEN
        SET expected_role = 'JUNIOR DATA SCIENTIST';
    ELSEIF exp > 2 AND exp <= 5 THEN
        SET expected_role = 'ASSOCIATE DATA SCIENTIST';
    ELSEIF exp > 5 AND exp <= 10 THEN
        SET expected_role = 'SENIOR DATA SCIENTIST';
    ELSEIF exp > 10 AND exp <= 12 THEN
        SET expected_role = 'LEAD DATA SCIENTIST';
    ELSEIF exp > 12 AND exp <= 16 THEN
        SET expected_role = 'MANAGER';
    ELSE
        SET expected_role = 'UNKNOWN';  -- For experience greater than 16 years
    END IF;
    IF role = expected_role THEN
        RETURN 'MATCHES';
    ELSE
```

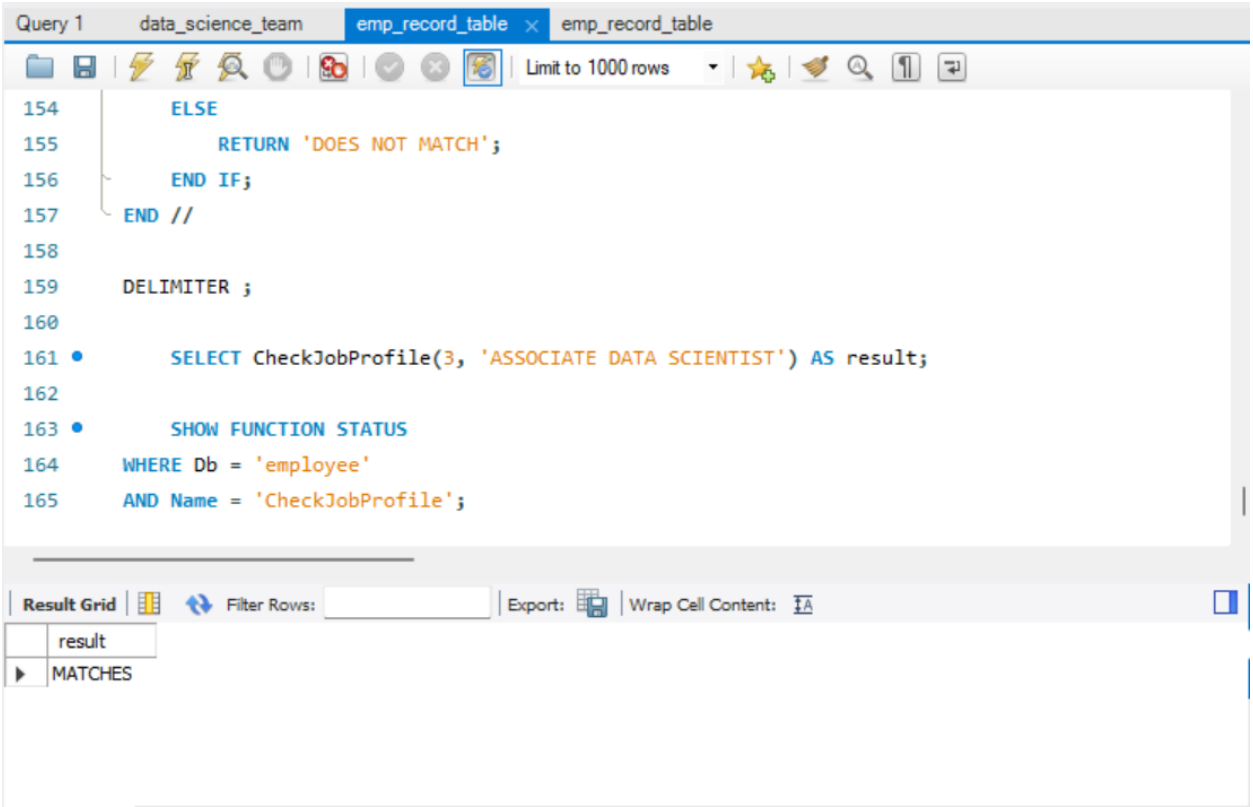
--For check function is created

SHOW FUNCTION STATUS

WHERE Db = 'employee'

AND Name = 'CheckJobProfile';

SELECT CheckJobProfile(3, 'ASSOCIATE DATA SCIENTIST') AS result;



Q 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.

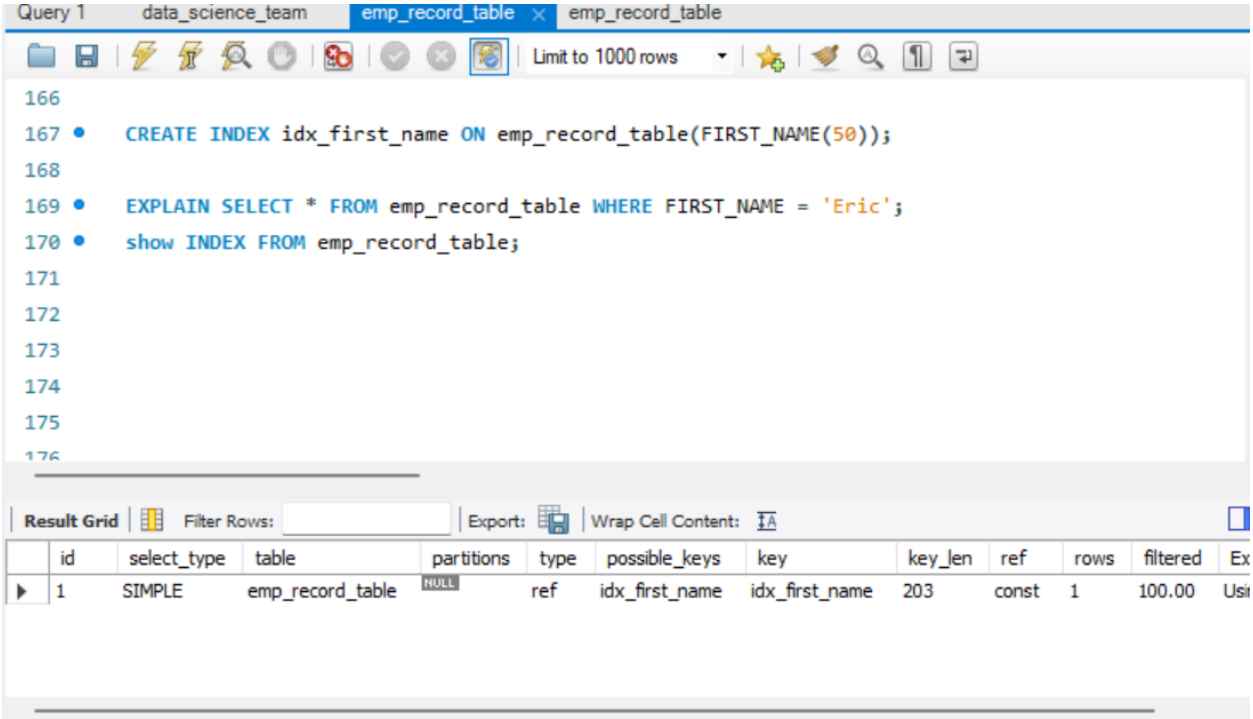
Solution:

```
CREATE INDEX idx_first_name ON emp_record_table(FIRST_NAME(50));

EXPLAIN SELECT * FROM emp_record_table WHERE FIRST_NAME = 'Eric';

SHOW INDEX FROM emp_record_table;


SELECT * FROM emp_record_table WHERE FIRST_NAME = 'Eric';
```



Q 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).

Solution:

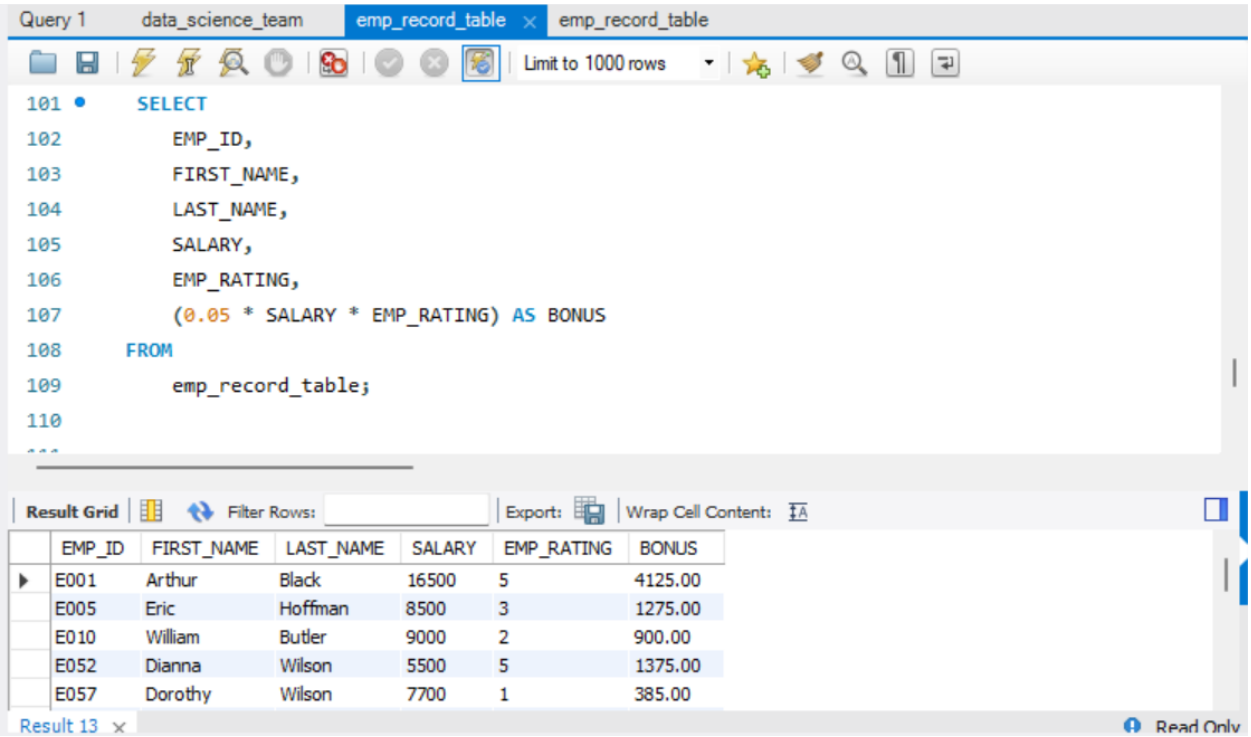
```
SELECT

EMP_ID, FIRST_NAME, LAST_NAME, SALARY, EMP_RATING,

   (0.05 * SALARY * EMP_RATING) AS BONUS

FROM

   emp_record_table;
```



EMP_ID	FIRST_NAME	LAST_NAME	SALARY	EMP_RATING	BONUS
E001	Arthur	Black	16500	5	4125.00
E005	Eric	Hoffman	8500	3	1275.00
E010	William	Butler	9000	2	900.00
E052	Dianna	Wilson	5500	5	1375.00
E057	Dorothy	Wilson	7700	1	385.00

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

Solution:

```
SELECT
    CONTINENT, COUNTRY,
    AVG(SALARY) AS AVERAGE_SALARY
FROM
    emp_record_table
GROUP BY
    CONTINENT, COUNTRY
ORDER BY
    CONTINENT, COUNTRY;
```

Query 1 | data_science_team | emp_record_table | emp_record_table

Limit to 1000 rows

```
        SELECT
        CONTINENT, COUNTRY,
        AVG(SALARY) AS AVERAGE_SALARY
    FROM
        emp_record_table
    GROUP BY
        CONTINENT, COUNTRY
    ORDER BY
        CONTINENT, COUNTRY;
```

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

CONTINENT	COUNTRY	AVERAGE_SALARY
ASIA	CHINA	6500.0000
ASIA	INDIA	6166.6667
EUROPE	FRANCE	9000.0000
EUROPE	GERMANY	7600.0000
NORTH AMERICA	CANADA	7000.0000