

ScienceQtech Employee Performance Mapping;

Problem scenario:

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

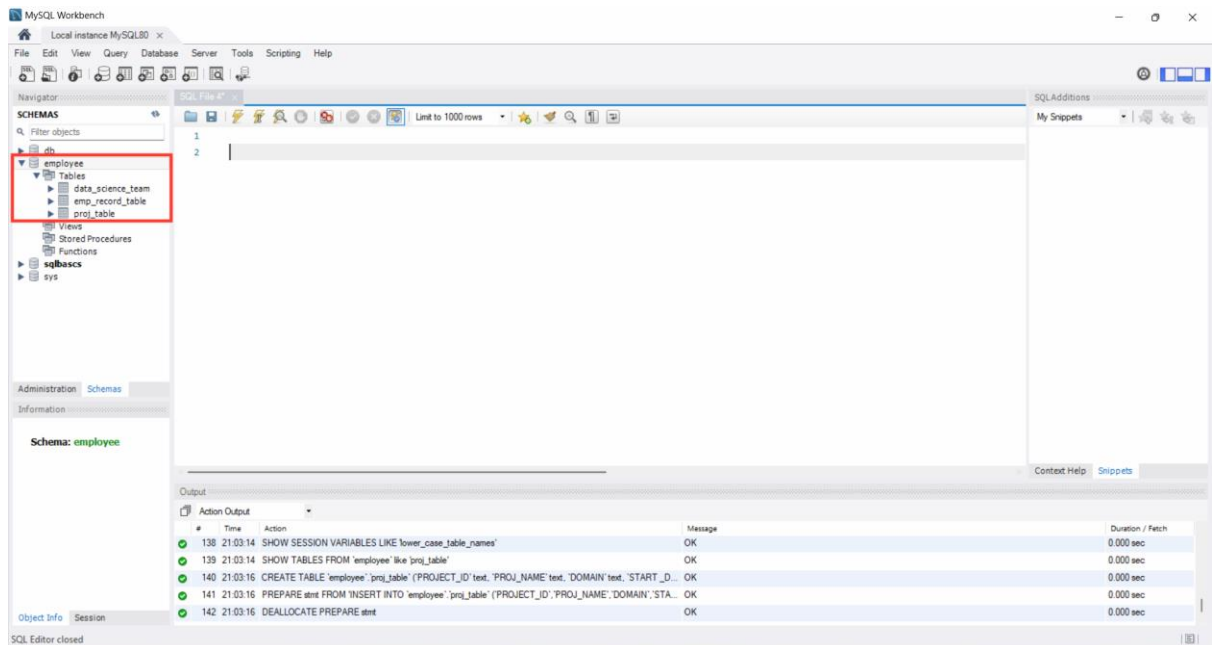
- PROJ_Name – Name of the project
- DOMAIN – Field of the project
- START_DATE – Day the project began
- CLOSURE_DATE – Day the project was or will be completed
- DEV_QTR – Quarter in which the project was scheduled
- STATUS – Status of the project currently

Data_science_team: It contains information about all the employees in the Data Science team.

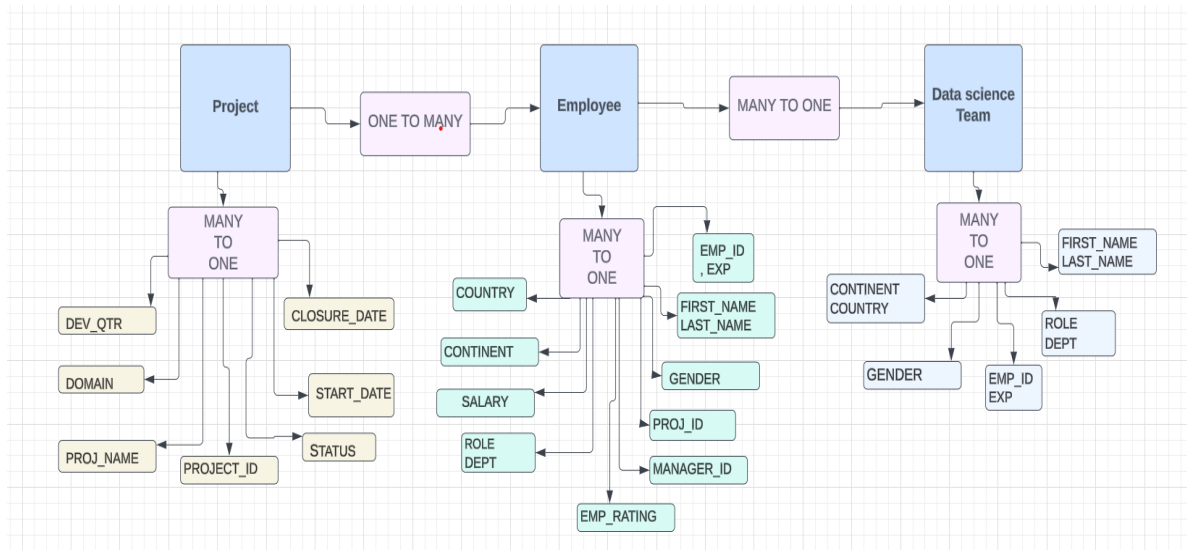
- EMP_ID – ID of the employee
- FIRST_NAME – First name of the employee
- LAST_NAME – Last name of the employee
- GENDER – Gender of the employee
- ROLE – Post of the employee
- DEPT – Field of the employee
- EXP – Years of experience the employee has
- COUNTRY – Country in which the employee is presently 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.



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.
select distinct EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT from emp_record_table;

The screenshot shows the MySQL Workbench interface. The SQL editor contains the query: `select distinct EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT from emp_record_table;`. The Results grid displays 19 rows of data with columns: EMP_ID, FIRST_NAME, LAST_NAME, GENDER, and DEPT. The Output pane shows the execution log with four entries, the last of which corresponds to the executed query, returning 19 rows in 0.015 seconds.

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
E103	Emily	Grove	F	FINANCE
E204	Karen	Nowak	F	AUTOMOTIVE
E245	Nian	Zhen	M	RETAIL
E260	Roy	Collins	M	RETAIL
E403	Steve	Hoffman	M	FINANCE
E428	Pete	Allen	M	AUTOMOTIVE

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;
select distinct EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING from emp_record_table where EMP_RATING < 2;

The screenshot shows the MySQL Workbench interface. The SQL editor contains the query: `select distinct EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING from emp_record_table where EMP_RATING < 2;`. The Results grid is empty. The Output pane shows an error message: "Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL version; near '< 2;' at line 2".

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	DEPT	EMP_RATING
--------	------------	-----------	--------	------	------------

Greater than four

select distinct EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING from emp_record_table where EMP_RATING > 4;

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 select distinct EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING from emp_record_table
2 WHERE EMP_RATING >4;
```

The Results Grid displays the following data:

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

Between two and four

select distinct EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING from emp_record_table where between 2 and 4;

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

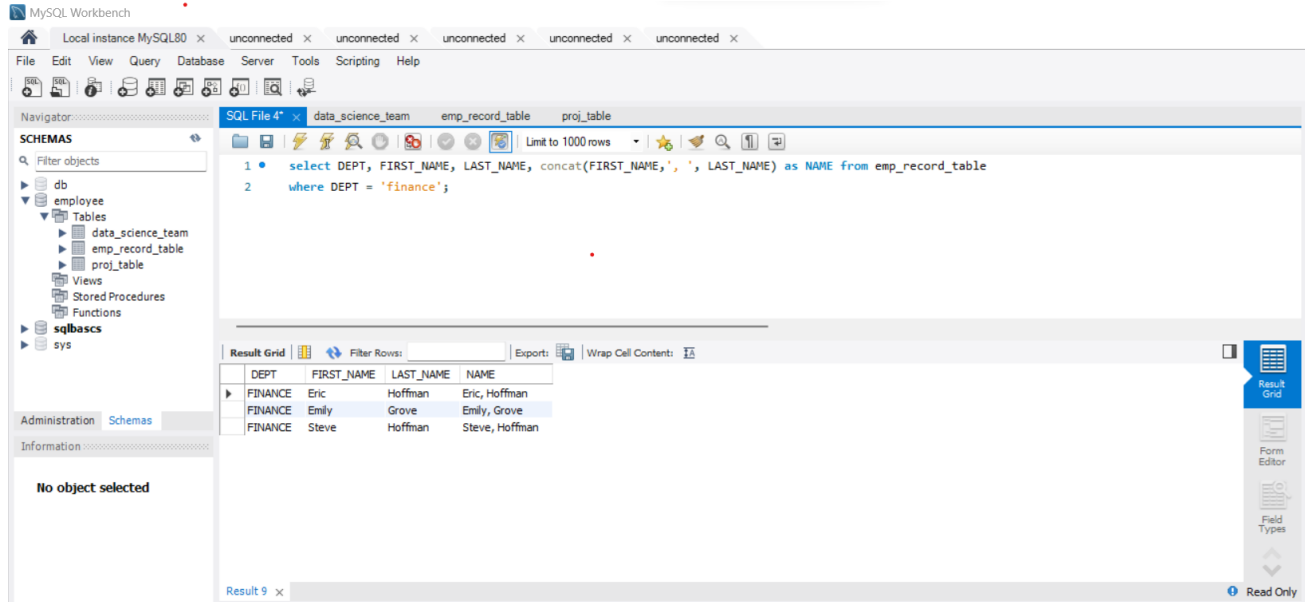
```
1 select distinct EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING from emp_record_table
2 WHERE EMP_RATING between 2 and 4;
```

The Results Grid displays the following data:

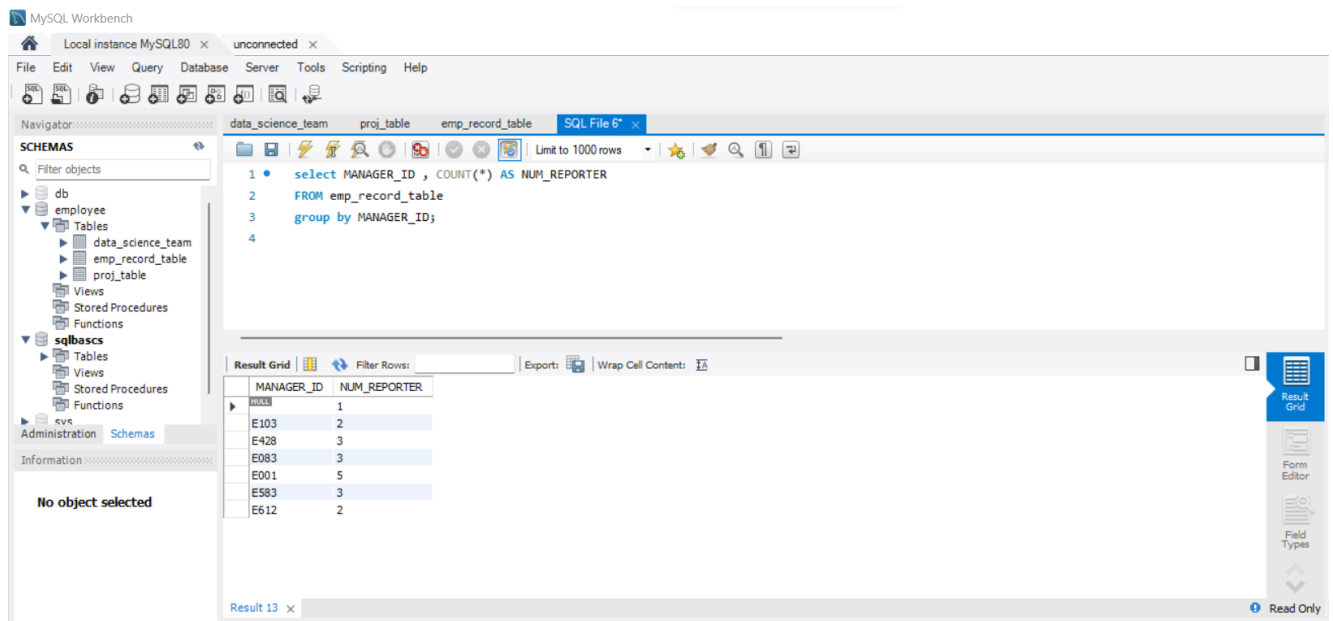
EMP_ID	FIRST_NAME	LAST_NAME	GENDER	DEPT	EMP_RATING
E005	Eric	Hoffman	M	FINANCE	3
E010	William	Butler	M	AUTOMOTIVE	2
E103	Emily	Grove	F	FINANCE	4
E245	Nian	Zhen	M	RETAIL	2
E260	Roy	Collins	M	RETAIL	3
E403	Steve	Hoffman	M	FINANCE	3
E428	Pete	Allen	M	AUTOMOTIVE	4
E478	David	Smith	M	RETAIL	4
E505	Chad	Wilson	M	HEALTHCARE	2
E583	Janet	Hale	F	RETAIL	2
E612	Tracy	Norris	F	RETAIL	4
E640	Jennifer	Jhones	F	RETAIL	4

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.

```
select DEPT, FIRST_NAME, LAST_NAME, concat(FIRST_NAME, ' ', LAST_NAME) as NAME from  
emp_record_table  
where DEPT = 'finance';
```



6. Write a query to list only those employees who have someone reporting to them. Also, show the number of reporters (including the President).
Select MANAGER_ID, COUNT(*) AS NUM_REPORTER
FROM emp_record_table
group by MANAGER_ID;



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

```
select distinct * from emp_record_table
where DEPT in ('FINANCE')
UNION
select distinct * from emp_record_table
where DEPT in ('HEALTHCARE');
```

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 select distinct * from emp_record_table
2 where DEPT in ('FINANCE')
3 UNION
4 select distinct * from emp_record_table
5 where DEPT in ('HEALTHCARE');
```

The Results window displays the following data:

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT	SALARY	EMP_RATING	MANAGER_ID	PROJ_ID
E005	Eric	Hoffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA	8500	3	E103	P105
E103	Emily	Grove	F	MANAGER	FINANCE	14	CANADA	NORTH AMERICA	10500	4	E001	P105
E403	Steve	Hoffman	M	ASSOCIATE DATA SCIENTIST	FINANCE	4	USA	NORTH AMERICA	5000	3	E103	P105
E052	Dianna	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	6	CANADA	NORTH AMERICA	5500	5	E083	P103
E057	Dorothy	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	9	USA	NORTH AMERICA	7700	1	E083	P302
E083	Patrick	Voltz	M	MANAGER	HEALTHCARE	15	USA	NORTH AMERICA	9500	5	E001	P103
E505	Chad	Wilson	M	ASSOCIATE DATA SCIENTIST	HEALTHCARE	5	CANADA	NORTH AMERICA	5000	2	E083	P103

- 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.

```
select distinct EMP_ID, FIRST_NAME, LAST_NAME, ROLE, DEPT, max(EMP_RATING) as
MAX_EMP_RATING
from emp_record_table
WHERE EMP_RATING = (SELECT MAX(EMP_RATING) FROM emp_record_table)
group by DEPT,EMP_ID, FIRST_NAME, LAST_NAME, ROLE,EMP_RATING;
```

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 select distinct EMP_ID, FIRST_NAME, LAST_NAME, ROLE, DEPT, max(EMP_RATING) as MAX_EMP_RATING
2 from emp_record_table
3 WHERE EMP_RATING = (SELECT MAX(EMP_RATING) FROM emp_record_table)
4 group by DEPT,EMP_ID, FIRST_NAME, LAST_NAME, ROLE,EMP_RATING;
```

The Results window displays the following data:

EMP_ID	FIRST_NAME	LAST_NAME	ROLE	DEPT	MAX_EMP_RATING
E001	Arthur	Black	PRESIDENT	ALL	5
E052	Dianna	Wilson	SENIOR DATA SCIENTIST	HEALTHCARE	5
E083	Patrick	Voltz	MANAGER	HEALTHCARE	5
E204	Karene	Nowak	SENIOR DATA SCIENTIST	AUTOMOTIVE	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.

```
select ROLE, max(SALARY) as MAX_SALARY, min(SALARY) as MIN_SALARY
from emp_record_table
group by ROLE;
```

The screenshot shows the MySQL Workbench interface. The query editor contains the following SQL query:

```
1 select ROLE, max(SALARY) as MAX_SALARY, min(SALARY) as MIN_SALARY
2 from emp_record_table
3 group by ROLE;
```

The Results window displays the following data:

ROLE	MAX_SALARY	MIN_SALARY
PRESIDENT	16500	16500
LEAD DATA SCIENTIST	9000	8500
SENIOR DATA SCIENTIST	7700	5500
MANAGER	11000	8500
ASSOCIATE DATA SCIENTIST	5000	4000
JUNIOR DATA SCIENTIST	3000	2800

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

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, ROLE, EXP,
Rank() OVER ( order by exp desc )AS 'EXP_rank'
FROM emp_record_table;
```

The screenshot shows the MySQL Workbench interface. The query editor contains the following SQL query:

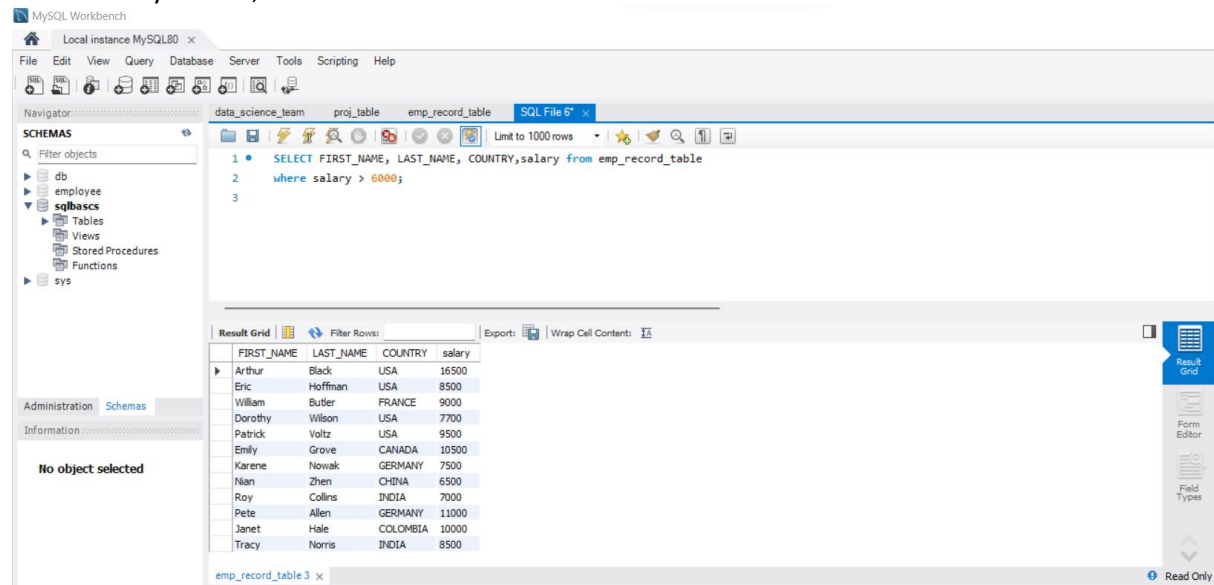
```
1 SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, ROLE, EXP,
2 Rank() OVER ( order by exp desc )AS 'EXP_rank'
3 FROM emp_record_table;
```

The Results window displays the following data:

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	DEPT	ROLE	EXP	EXP_rank
E001	Arthur	Black	M	ALL	PRESIDENT	20	1
E083	Patrick	Voltz	M	HEALTHCARE	MANAGER	15	2
E103	Emily	Grove	F	FINANCE	MANAGER	14	3
E428	Pete	Allen	M	AUTOMOTIVE	MANAGER	14	3
E583	Janet	Hale	F	RETAIL	MANAGER	14	3
E612	Tracy	Norris	F	RETAIL	MANAGER	13	6
E010	William	Butler	M	AUTOMOTIVE	LEAD DATA SCIENTIST	12	7
E005	Eric	Hoffman	M	FINANCE	LEAD DATA SCIENTIST	11	8
E057	Dorothy	Wilson	F	HEALTHCARE	SENIOR DATA SCIENTIST	9	9
E204	Karene	Nowak	F	AUTOMOTIVE	SENIOR DATA SCIENTIST	8	10
E260	Roy	Collins	M	RETAIL	SENIOR DATA SCIENTIST	7	11
FNK9	Nanana	Willen	F	HEALTHCARE	SENIOR DATA SCIENTIST	6	12

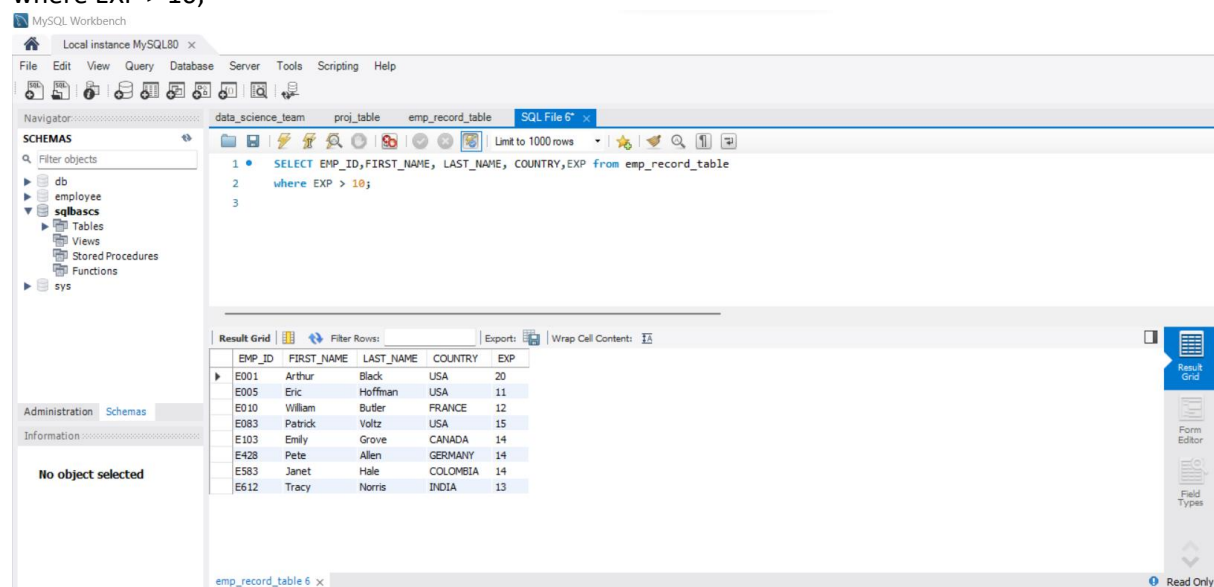
11. Write a query to create a view that displays employees in various countries whose salary is more than six thousand. Take data from the employee record table.

```
SELECT FIRST_NAME, LAST_NAME, COUNTRY,salary from emp_record_table  
where salary > 6000;
```



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

```
SELECT EMP_ID,FIRST_NAME, LAST_NAME, COUNTRY,EXP from emp_record_table  
where EXP > 10;
```



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.

```
DROP PROCEDURE IF EXISTS EMP_GT_3;
```

```
DELIMITER &&
```

```
CREATE PROCEDURE EMP_GT_3 (IN exp_no int)
```

```
BEGIN
```

```
SELECT * FROM emp_record_table WHERE exp > exp_no;
```

```
END &&
```

```
DELIMITER ;
```

```
CALL EMP_GT_3(3);
```

The screenshot shows the MySQL Workbench interface. The SQL Editor window contains the following code:

```
1 DROP PROCEDURE IF EXISTS EMP_GT_3;
2 DELIMITER &&
3 CREATE PROCEDURE EMP_GT_3 (IN exp_no int)
4 BEGIN
5 SELECT * FROM emp_record_table WHERE exp > exp_no;
6 END &&
7 DELIMITER ;
8 CALL EMP_GT_3(3);
9
```

The Results window displays the output of the stored procedure call, showing a table with 14 columns: EMP_ID, FIRST_NAME, LAST_NAME, GENDER, ROLE, DEPT, EXP, COUNTRY, CONTINENT, SALARY, EMP_RATING, MANAGER_ID, and PROJ_ID. The table contains 14 rows of employee data.

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT	SALARY	EMP_RATING	MANAGER_ID	PROJ_ID
E001	Arthur	Black	M	PRESIDENT	ALL	20	USA	NORTH AMERICA	16500	5	NULL	NULL
E005	Eric	Hoffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA	8500	3	E103	P105
E010	William	Butler	M	LEAD DATA SCIENTIST	AUTOMOTIVE	12	FRANCE	EUROPE	9000	2	E428	P204
E052	Dianna	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	6	CANADA	NORTH AMERICA	5500	5	E083	P103
E057	Dorothy	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	9	USA	NORTH AMERICA	7700	1	E083	P302
E083	Patrick	Voltz	M	MANAGER	HEALTHCARE	15	USA	NORTH AMERICA	9500	5	E001	NULL
E103	Emily	Grove	F	MANAGER	FINANCE	14	CANADA	NORTH AMERICA	10500	4	E001	NULL
E204	Karene	Nowak	F	SENIOR DATA SCIENTIST	AUTOMOTIVE	8	GERMANY	EUROPE	7500	5	E428	P204
E245	Nian	Zhen	M	SENIOR DATA SCIENTIST	RETAIL	6	CHINA	ASIA	6500	2	E583	P109
E260	Roy	Collins	M	SENIOR DATA SCIENTIST	RETAIL	7	INDIA	ASIA	7000	3	E583	NA
E403	Steve	Hoffman	M	ASSOCIATE DATA SCIENTIST	FINANCE	4	USA	NORTH AMERICA	5000	3	E103	P105
E428	Pete	Allen	M	MANAGER	AUTOMOTIVE	14	GERMANY	EUROPE	11000	4	E001	NULL
E505	Chad	Wilson	M	ASSOCIATE DATA SCIENTIST	HEALTHCARE	5	CANADA	NORTH AMERICA	5000	2	E083	P103
E583	Janet	Hale	F	MANAGER	RETAIL	14	COLOMBIA	SOUTH AMERICA	10000	2	E001	NULL
E612	Tracy	Norris	F	MANAGER	RETAIL	13	INDIA	ASIA	8500	4	E001	NULL

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',

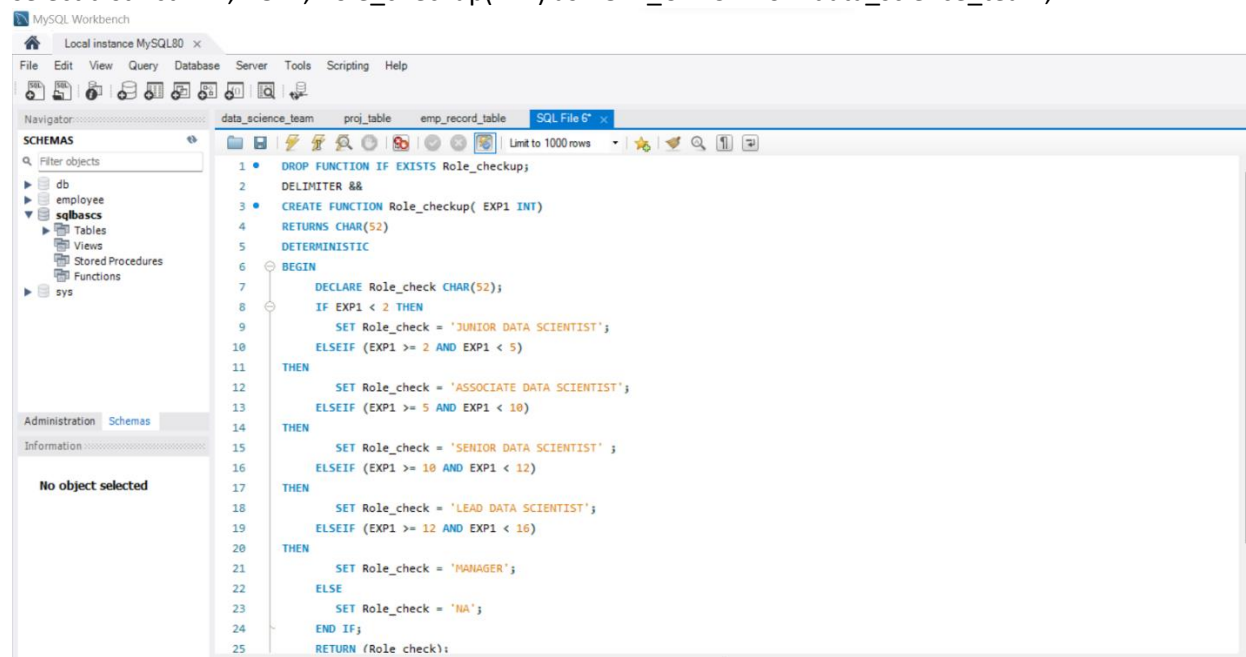
For an employee with the experience of 10 to 12 years assign 'LEAD DATA SCIENTIST',

For an employee with the experience of 12 to 16 years assign 'MANAGER'.

```

DROP FUNCTION IF EXISTS Role_checkup;
DELIMITER &&
CREATE FUNCTION Role_checkup( EXP1 INT)
RETURNS CHAR(52)
DETERMINISTIC
BEGIN
DECLARE Role_check CHAR(52);
IF EXP1 < 2 THEN
SET Role_check = 'JUNIOR DATA SCIENTIST';
ELSEIF (EXP1 >= 2 AND EXP1 < 5)
THEN
SET Role_check = 'ASSOCIATE DATA SCIENTIST';
ELSEIF (EXP1 >= 5 AND EXP1 < 10)
THEN
SET Role_check = 'SENIOR DATA SCIENTIST' ;
ELSEIF (EXP1 >= 10 AND EXP1 < 12)
THEN
SET Role_check = 'LEAD DATA SCIENTIST';
ELSEIF (EXP1 >= 12 AND EXP1 < 16)
THEN
SET Role_check = 'MANAGER';
ELSE
SET Role_check = 'NA';
END IF;
RETURN (Role_check);
END; &&
DELIMITER ;
select distinct EXP, ROLE, Role_checkup(EXP) as ROLE_CHECK from data_science_team;

```



MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: data_science_team proj_table emp_record_table SQL File 6 x

SCHEMAS

Filter objects

- db
- employee
- sqlbascs
 - Tables
 - Views
 - Stored Procedures
 - Functions
- sys

Administration Schemas

Information: No object selected

```

20 THEN
21     SET Role_check = 'MANAGER';
22 ELSE
23     SET Role_check = 'NA';
24 END IF;
25 RETURN (Role_check);
26 END; &&
27 DELIMITER ;
28 select distinct EXP, ROLE, Role_checkup(EXP) as ROLE_CHECK from data_science_team;

```

Result Grid

EXP	ROLE	ROLE_CHECK
11	LEAD DATA SCIENTIST	LEAD DATA SCIENTIST
12	LEAD DATA SCIENTIST	MANAGER
6	SENIOR DATA SCIENTIST	SENIOR DATA SCIENTIST
9	SENIOR DATA SCIENTIST	SENIOR DATA SCIENTIST
8	SENIOR DATA SCIENTIST	SENIOR DATA SCIENTIST
7	SENIOR DATA SCIENTIST	SENIOR DATA SCIENTIST
4	ASSOCIATE DATA SCIENTIST	ASSOCIATE DATA SCIENTIST
3	ASSOCIATE DATA SCIENTIST	ASSOCIATE DATA SCIENTIST
5	ASSOCIATE DATA SCIENTIST	SENIOR DATA SCIENTIST
2	JUNIOR DATA SCIENTIST	ASSOCIATE DATA SCIENTIST
1	JUNIOR DATA SCIENTIST	JUNIOR DATA SCIENTIST

Result 10 x

Read Only

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.
- select distinct * from emp_record_table
where FIRST_NAME = 'Eric';

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: data_science_team proj_table emp_record_table SQL File 6 x project

SCHEMAS

Filter objects

- db
- employee
- sqlbascs
 - Tables
 - Views
 - Stored Procedures
 - Functions
- sys

Administration Schemas

Information: No object selected

```

1 select distinct * from emp_record_table
2 where FIRST_NAME = 'Eric';

```

Result Grid

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT	SALARY	EMP_RATING	MANAGER_ID	PROJ_ID
E005	Eric	Hoffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA	8500	3	E103	P105

emp_record_table 15 x

Read Only

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).

`select distinct *,(0.05*SALARY)*EMP_RATING as Bonus from emp_record_table;`

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator: data_science_team proj_table emp_record_table project* SQL File 7*

1 • `select distinct *,(0.05*SALARY)*EMP_RATING as Bonus from emp_record_table;`

Result Grid

FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT	SALARY	EMP_RATING	MANAGER_ID	PROJ_ID	Bonus
Arthur	Black	M	PRESIDENT	ALL	20	USA	NORTH AMERICA	16500	5	E009		4125.00
Eric	Hoffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA	8500	3	E103	P105	1275.00
William	Butler	M	LEAD DATA SCIENTIST	FINANCE	12	FRANCE	EUROPE	9000	2	E428	P204	900.00
Dianna	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	6	CANADA	NORTH AMERICA	5500	5	E083	P103	1375.00
Dorothy	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	9	USA	NORTH AMERICA	7700	1	E083	P302	385.00
Patrick	Voltz	M	MANAGER	HEALTHCARE	15	USA	NORTH AMERICA	9500	5	E001		2375.00
Emily	Grove	F	MANAGER	FINANCE	14	CANADA	NORTH AMERICA	10500	4	E001		2100.00
Karene	Nowak	F	SENIOR DATA SCIENTIST	HEALTHCARE	8	GERMANY	EUROPE	7500	5	E428	P204	1875.00
Nian	Zhen	M	SENIOR DATA SCIENTIST	RETAIL	6	CHINA	ASIA	6500	2	E583	P109	650.00
Roy	Collins	M	SENIOR DATA SCIENTIST	RETAIL	7	INDIA	ASIA	7000	3	E583	NA	1050.00
Steve	Hoffman	M	ASSOCIATE DATA SCIENTIST	FINANCE	4	USA	NORTH AMERICA	5000	3	E103	P105	750.00
Pete	Allen	M	MANAGER	HEALTHCARE	14	GERMANY	EUROPE	11000	4	E001		2200.00
David	Smith	M	ASSOCIATE DATA SCIENTIST	RETAIL	3	COLOMBIA	SOUTH AMERICA	4000	4	E583	P109	800.00
Chad	Wilson	M	ASSOCIATE DATA SCIENTIST	HEALTHCARE	5	CANADA	NORTH AMERICA	5000	2	E083	P103	500.00

Result 1 x

Read Only

17. Write a query to calculate the average salary distribution based on the continent and country. Take data from the employee record table.
- `select distinct country, CONTINENT, avg(SALARY)`
`from emp_record_table`
`group by country, CONTINENT;`

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator: data_science_team proj_table emp_record_table project* SQL File 7*

1 • `select distinct country, CONTINENT, avg(SALARY)`
2 • `from emp_record_table`
3 • `group by country, CONTINENT;`

Result Grid

country	CONTINENT	avg(SALARY)
USA	NORTH AMERICA	9440.0000
FRANCE	EUROPE	9000.0000
CANADA	NORTH AMERICA	7000.0000
GERMANY	EUROPE	7600.0000
CHINA	ASIA	6500.0000
INDIA	ASIA	6166.6667
COLOMBIA	SOUTH AMERICA	5600.0000

Result 2 x

Read Only

