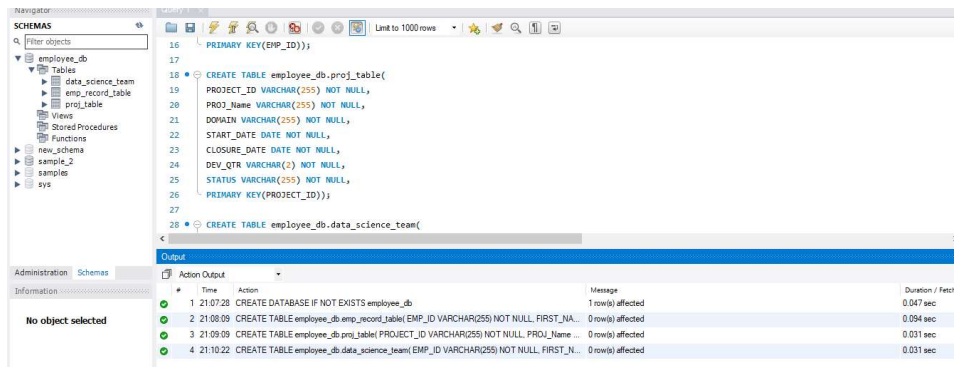
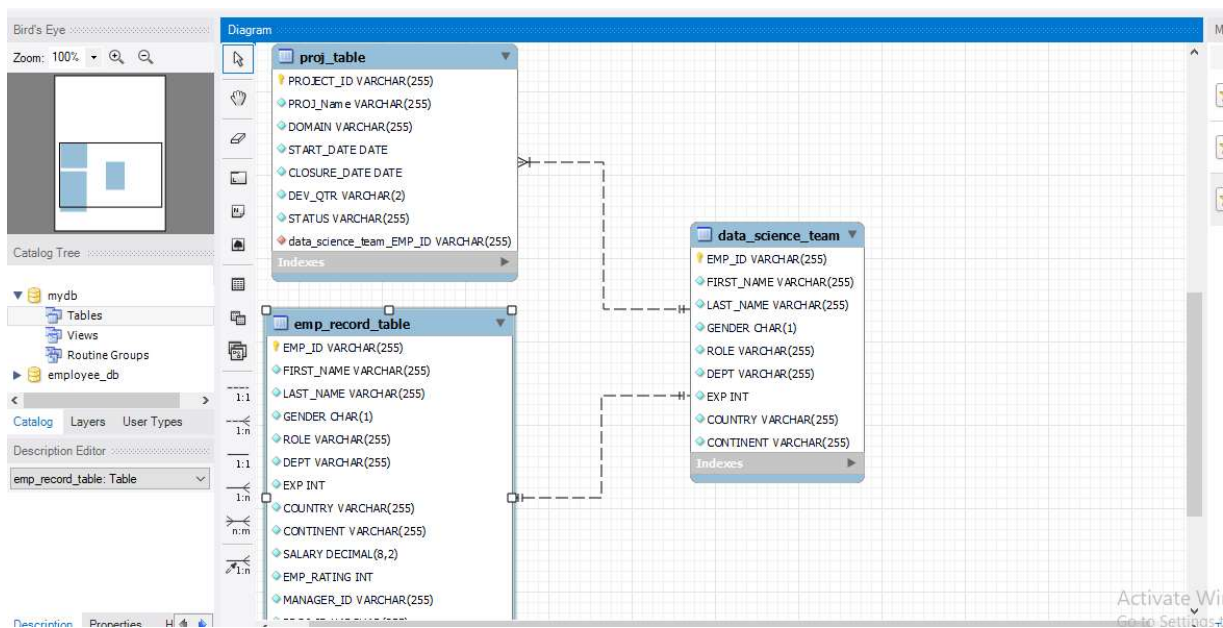


1. Create a database named **employee**, then import **data_science_team.csv**, **proj_table.csv** and **emp_record_table.csv** into the **employee** database from the given resources.



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.

```
39 • SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER,DEPT
40 FROM employee_db.emp_record_table;
41
```

Result Grid

| EMP_ID | FIRST_NAME | LAST_NAME | GENDER | DEPT |
|--------|------------|-----------|--------|------------|
| E005 | Eric | Hoffman | M | FINANCE |
| E010 | William | Butler | M | AUTOMOTIVE |
| E052 | Dianna | Wilson | F | HEALTHCARE |
| E057 | Dorothy | Wilson | F | HEALTHCARE |
| E204 | Karene | Nowak | F | AUTOMOTIVE |
| E245 | Nian | Zhen | M | RETAIL |
| E260 | Roy | Collins | M | RETAIL |
| E403 | Steve | Hoffman | M | FINANCE |

cord_table 1 x

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

```
44 • SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER,DEPT,EMP_RATING FROM employee_db.emp_record_table
45 WHERE EMP_RATING < 2;
46
47
```

Result Grid

| 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 |
| NULL | NULL | NULL | NULL | NULL | NULL |

```
47 • SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER,DEPT,EMP_RATING FROM employee_db.emp_record_table
48 WHERE EMP_RATING > 4;
49
```

Result Grid

| EMP_ID | FIRST_NAME | LAST_NAME | GENDER | DEPT | EMP_RATING |
|--------|------------|-----------|--------|------------|------------|
| E052 | Dianna | Wilson | F | HEALTHCARE | 5 |
| E204 | Karene | Nowak | F | AUTOMOTIVE | 5 |
| NULL | NULL | NULL | NULL | NULL | NULL |

```

50 • SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER,DEPT,EMP_RATING FROM employee_db.emp_record_table
51 WHERE EMP_RATING between 2 and 4 ORDER BY EMP_RATING;
52

```

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

| | EMP_ID | FIRST_NAME | LAST_NAME | GENDER | DEPT | EMP_RATING |
|--|--------|------------|-----------|--------|------------|------------|
| | E010 | William | Butler | M | AUTOMOTIVE | 2 |
| | E245 | Nian | Zhen | M | RETAIL | 2 |
| | E505 | Chad | Wilson | M | HEALTHCARE | 2 |
| | E005 | Eric | Hoffman | M | FINANCE | 3 |
| | E260 | Roy | Collins | M | RETAIL | 3 |
| | E403 | Steve | Hoffman | M | FINANCE | 3 |
| | E478 | David | Smith | M | 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.

```

53 • SELECT CONCAT_WS(' ', FIRST_NAME, LAST_NAME) AS 'FULL NAME'
54 FROM employee_db.emp_record_table
55 WHERE DEPT = "FINANCE";

```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

| FULL NAME |
|----------------|
| Eric, Hoffman |
| 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).

| Navigator | Query 1 | | | | | | |
|--|--|----------|---------------|---------------|---|----------------|---|
| SCHEMAS Filter objects employee_db Tables Views Stored Procedures Functions new_schema sample_2 samples sys | <pre> 59 COUNT(DISTINCT emp_record_table.EMP_ID) AS num_reporters 60 FROM 61 employee_db.emp_record_table 62 JOIN 63 emp_record_table AS reporters ON emp_record_table.EMP_ID = reporters.MANAGER_ID 64 OR (emp_record_table.MANAGER_ID IS NULL AND emp_record_table.EMP_ID = reporters.MANAGER_ID) 65 GROUP BY 66 emp_record_table.EMP_ID, emp_record_table.FIRST_NAME, emp_record_table.LAST_NAME 67 HAVING 68 COUNT(DISTINCT emp_record_table.EMP_ID) > 0 </pre> | | | | | | |
| Result Grid Filter Rows: Export: Wrap Cell Content: | <table> <tr> <th>emp_name</th><th>num_reporters</th></tr> <tr> <td>Eric, Hoffman</td><td>1</td></tr> <tr> <td>Steve, Hoffman</td><td>1</td></tr> </table> | emp_name | num_reporters | Eric, Hoffman | 1 | Steve, Hoffman | 1 |
| emp_name | num_reporters | | | | | | |
| Eric, Hoffman | 1 | | | | | | |
| Steve, Hoffman | 1 | | | | | | |

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.

```

71
72 • SELECT * FROM employee_db.emp_record_table WHERE DEPT="Finance"
73 UNION
74 SELECT * FROM employee_db.emp_record_table WHERE DEPT="Healthcare";
75
76
77

```

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 | 8500.00 | 3 | E103 | P105 |
| E403 | Steve | Hoffman | M | ASSOCIATE DATA SCIENTIST | FINANCE | 4 | USA | NORTH AMERICA | 5000.00 | 3 | E103 | P105 |
| E052 | Dianna | Wilson | F | SENIOR DATA SCIENTIST | HEALTHCARE | 6 | CANADA | NORTH AMERICA | 5500.00 | 5 | E083 | P103 |
| E057 | Dorothy | Wilson | F | SENIOR DATA SCIENTIST | HEALTHCARE | 9 | USA | NORTH AMERICA | 7700.00 | 1 | E083 | P302 |
| E505 | Chad | Wilson | M | ASSOCIATE DATA SCIENTIST | HEALTHCARE | 5 | CANADA | NORTH AMERICA | 5000.00 | 2 | E083 | P103 |

Result 2 x Read Only

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.

```

42 • SELECT * FROM employee_db.emp_record_table;

```

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 | 8500.00 | 3 | E103 | P105 |
| E1010 | William | Butler | M | LEAD DATA SCIENTIST | AUTOMOTIVE | 12 | FRANCE | EUROPE | 9000.00 | 2 | E428 | P204 |
| E052 | Dianna | Wilson | F | SENIOR DATA SCIENTIST | HEALTHCARE | 6 | CANADA | NORTH AMERICA | 5500.00 | 5 | E083 | P103 |
| E057 | Dorothy | Wilson | F | SENIOR DATA SCIENTIST | HEALTHCARE | 9 | USA | NORTH AMERICA | 7700.00 | 1 | E083 | P302 |
| E204 | Karene | Nowak | F | SENIOR DATA SCIENTIST | AUTOMOTIVE | 8 | GERMANY | EUROPE | 7500.00 | 5 | E428 | P204 |
| E245 | Nian | Zhen | M | SENIOR DATA SCIENTIST | RETAIL | 6 | CHINA | ASIA | 6500.00 | 2 | E583 | P109 |
| E260 | Roy | Collins | M | SENIOR DATA SCIENTIST | RETAIL | 7 | INDIA | ASIA | 7000.00 | 3 | E583 | NA |
| E403 | Steve | Hoffman | M | ASSOCIATE DATA SCIENTIST | FINANCE | 4 | USA | NORTH AMERICA | 5000.00 | 3 | E103 | P105 |

Result 2 x Apply Revert

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.

```

72 • SELECT ROLE, MAX(SALARY) AS max_salary, MIN(SALARY) AS min_salary
73 FROM employee_db.emp_record_table
74 GROUP BY ROLE
75 ORDER BY ROLE;
76
77
78

```

Result Grid

| ROLE | max_salary | min_salary |
|--------------------------|------------|------------|
| ASSOCIATE DATA SCIENTIST | 5000.00 | 4000.00 |
| JUNIOR DATA SCIENTIST | 3000.00 | 2800.00 |
| LEAD DATA SCIENTIST | 9000.00 | 8500.00 |
| SENIOR DATA SCIENTIST | 7700.00 | 5500.00 |

Result 3 x Read On

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

```

76 • SELECT *,
77 RANK() OVER (order by EXP) as 'rank'
78 FROM employee_db.emp_record_table;
79
80
81
82
83
84

```

| EMP_ID | FIRST_NAME | LAST_NAME | GENDER | ROLE | DEPT | EXP | COUNTRY | CONTINENT | SALARY | EMP_RATING | MANAGER_ID | PROJ_ID | rank |
|--------|------------|-----------|--------|--------------------------|------------|-----|----------|---------------|---------|------------|------------|---------|------|
| E640 | Jenifer | Jhones | F | JUNIOR DATA SCIENTIST | RETAIL | 1 | COLOMBIA | SOUTH AMERICA | 2800.00 | 4 | E612 | P406 | 1 |
| E620 | Katrina | Allen | F | JUNIOR DATA SCIENTIST | RETAIL | 2 | INDIA | ASIA | 3000.00 | 1 | E612 | P406 | 2 |
| E478 | David | Smith | M | ASSOCIATE DATA SCIENTIST | RETAIL | 3 | COLOMBIA | SOUTH AMERICA | 4000.00 | 4 | E583 | P109 | 3 |
| E532 | Claire | Brennan | F | ASSOCIATE DATA SCIENTIST | AUTOMOTIVE | 3 | GERMANY | EUROPE | 4300.00 | 1 | E428 | P204 | 3 |
| E403 | Steve | Hoffman | M | ASSOCIATE DATA SCIENTIST | FINANCE | 4 | USA | NORTH AMERICA | 5000.00 | 3 | E103 | P105 | 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.

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: Filter objects

SCHEMAS

- employee_db
 - Tables
 - data_science_team
 - emp_record_table
 - proj_table
 - Views
 - employee_view
 - Stored Procedures
 - Functions
- new_schema
- sample_2
- samples
- sys

Query 1 x

```

73 • CREATE VIEW Employee_view AS
74 SELECT EMP_ID, FIRST_NAME, LAST_NAME, COUNTRY, DEPT, SALARY
75 FROM employee_db.emp_record_table
76 WHERE SALARY > 6000;
77
78 • select * from Employee_view;

```

| EMP_ID | FIRST_NAME | LAST_NAME | COUNTRY | DEPT | SALARY |
|--------|------------|-----------|---------|------------|---------|
| E005 | Eric | Hoffman | USA | FINANCE | 8500.00 |
| E010 | William | Butler | FRANCE | AUTOMOTIVE | 9000.00 |
| E057 | Dorothy | Wilson | USA | HEALTHCARE | 7700.00 |
| E204 | Karene | Nowak | GERMANY | AUTOMOTIVE | 7500.00 |
| E245 | Nian | Zhen | CHINA | RETAIL | 6500.00 |

Employee view 5 x

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

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: Filter objects

SCHEMAS

- employee_db
 - Tables
 - data_science_team
 - emp_record_table
 - proj_table
 - Views
 - employee_view
 - Stored Procedures
 - Functions
- new_schema
- sample_2
- samples
- sys

Query 1 x

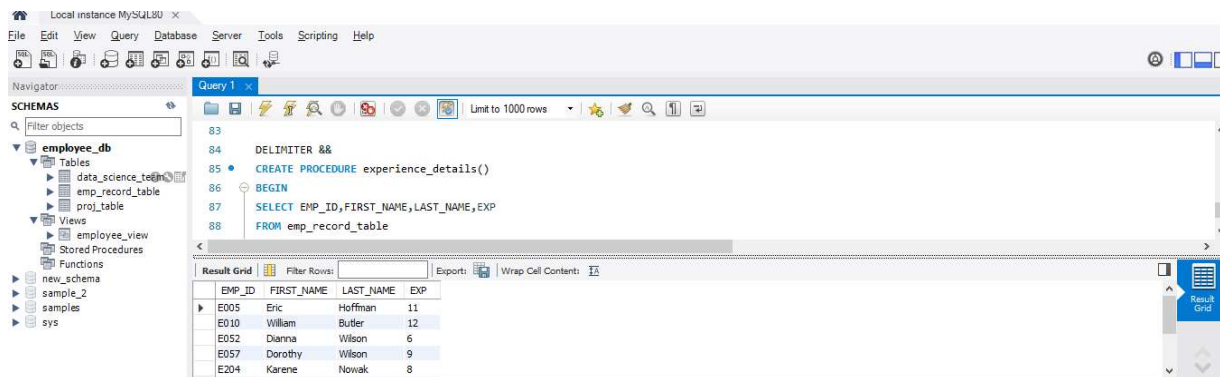
```

79
80 • SELECT *
81 FROM employee_db.emp_record_table
82 WHERE EXP > 10;
83
84

```

| 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 | 8500.00 | 3 | E103 | P105 |
| E010 | William | Butler | M | LEAD DATA SCIENTIST | AUTOMOTIVE | 12 | FRANCE | EUROPE | 9000.00 | 2 | E428 | P204 |

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.



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:

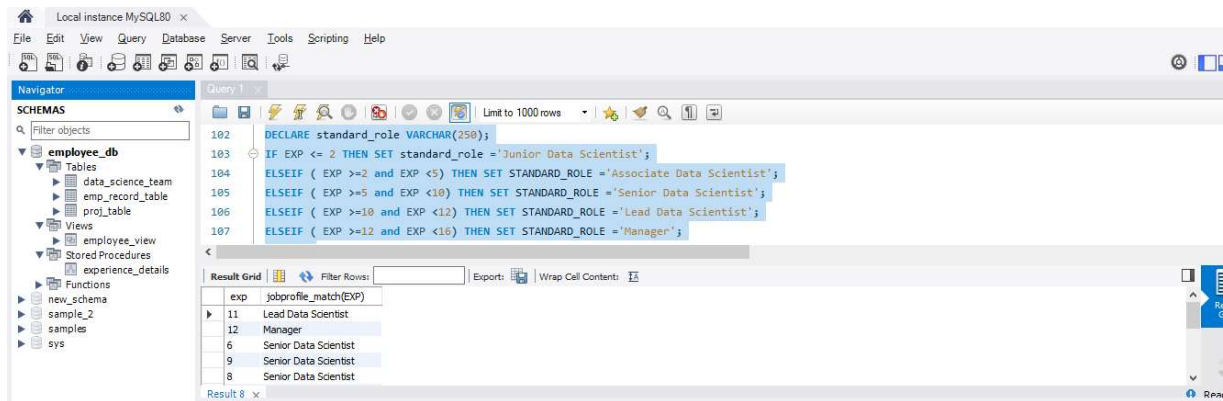
For an employee with experience less than or equal to 2 years assign 'JUNIOR DATA SCIENTIST',

For an employee with the experience of 2 to 5 years assign 'ASSOCIATE DATA SCIENTIST',

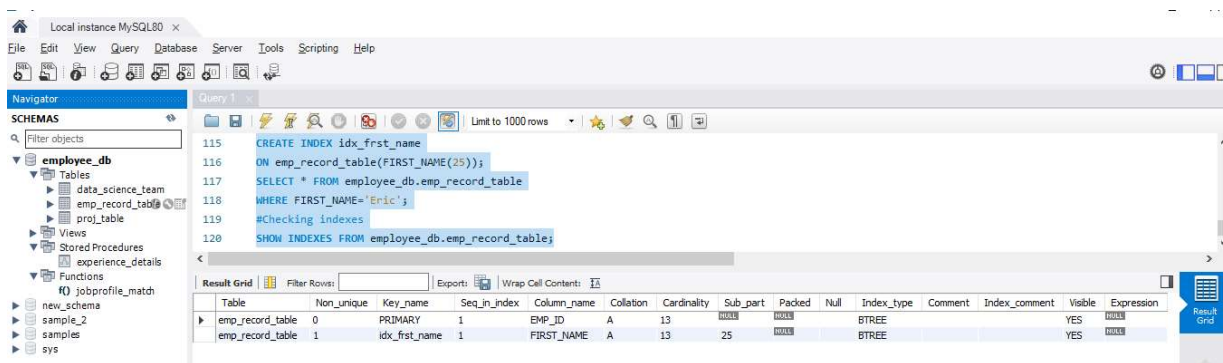
For an employee with the experience of 5 to 10 years assign 'SENIOR DATA SCIENTIST',

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'.



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.



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).

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

employee_db

- Tables
 - data_science_team
 - emp_record_table
 - proj_table
- Views
- Stored Procedures
- experience_details
- Functions
 - jobprofile_match
- new_schema
- sample_2
- samples
- sys

Query 1

```

122
123
124 SELECT EMP_ID, FIRST_NAME, LAST_NAME, SALARY, EMP_RATING, round((0.05 * SALARY * EMP_RATING),0) AS Bonus,
125 (SALARY + round((0.05 * SALARY * EMP_RATING),0)) AS Total_salary
126 FROM employee_db.emp_record_table
127 ORDER BY Total_salary asc;

```

Result Grid

| EMP_ID | FIRST_NAME | LAST_NAME | SALARY | EMP_RATING | Bonus | Total_salary |
|--------|------------|-----------|---------|------------|-------|--------------|
| E620 | Katrina | Allen | 3000.00 | 1 | 150 | 3150.00 |
| E640 | Jenifer | Jhones | 2800.00 | 4 | 560 | 3360.00 |
| E532 | Clare | Brennan | 4300.00 | 1 | 215 | 4515.00 |
| E478 | David | Smith | 4000.00 | 4 | 800 | 4800.00 |
| E505 | Chad | Wilson | 5000.00 | 2 | 500 | 5500.00 |

Result 11 x

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

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

employee_db

- Tables
 - data_science_team
 - emp_record_table
 - proj_table
- Views
- Stored Procedures
- experience_details
- Functions
 - jobprofile_match
- new_schema
- sample_2
- samples
- sys

Query 1

```

127 ORDER BY Total_salary asc;
128
129 SELECT COUNTRY,CONTINENT, ROUND(AVG(SALARY), 2)
130 FROM employee_db.emp_record_table
131 GROUP BY COUNTRY,CONTINENT;
132

```

Result Grid

| COUNTRY | CONTINENT | ROUND(AVG(SALARY), 2) |
|---------|---------------|-----------------------|
| USA | NORTH AMERICA | 7066.67 |
| FRANCE | EUROPE | 9000.00 |
| CANADA | NORTH AMERICA | 5250.00 |
| GERMANY | EUROPE | 5900.00 |

Result 12 x