

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.

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.

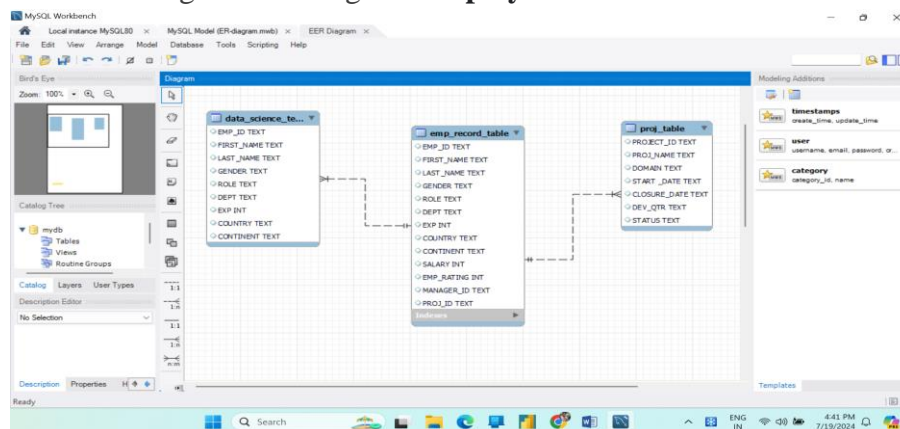
- EMP_ID – ID of the employee
- FIRST_NAME – First name of the employee
- LAST_NAME – Last name of the employee
- GENDER – Gender of the employee
- ROLE – Post of the employee
- DEPT – Field of the employee
- EXP – Years of experience the employee has
- COUNTRY – Country in which the employee is presently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

The task to be performed:

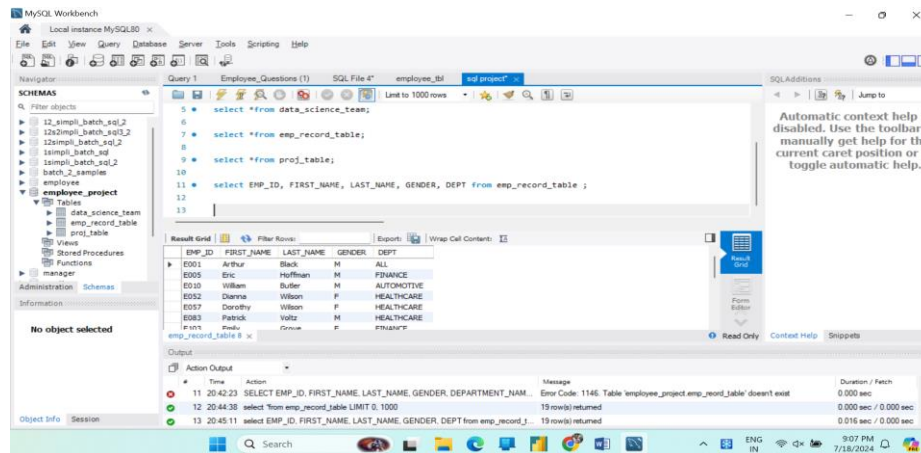
1. Create a database named *employee*, then import **data_science_team.csv** **proj_table.csv** and **emp_record_table.csv** into the **employee** database from the given resources.

- Right click on the Table Tab
- Click on Table Data import Wizard
- Select file Path
- Import Data

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

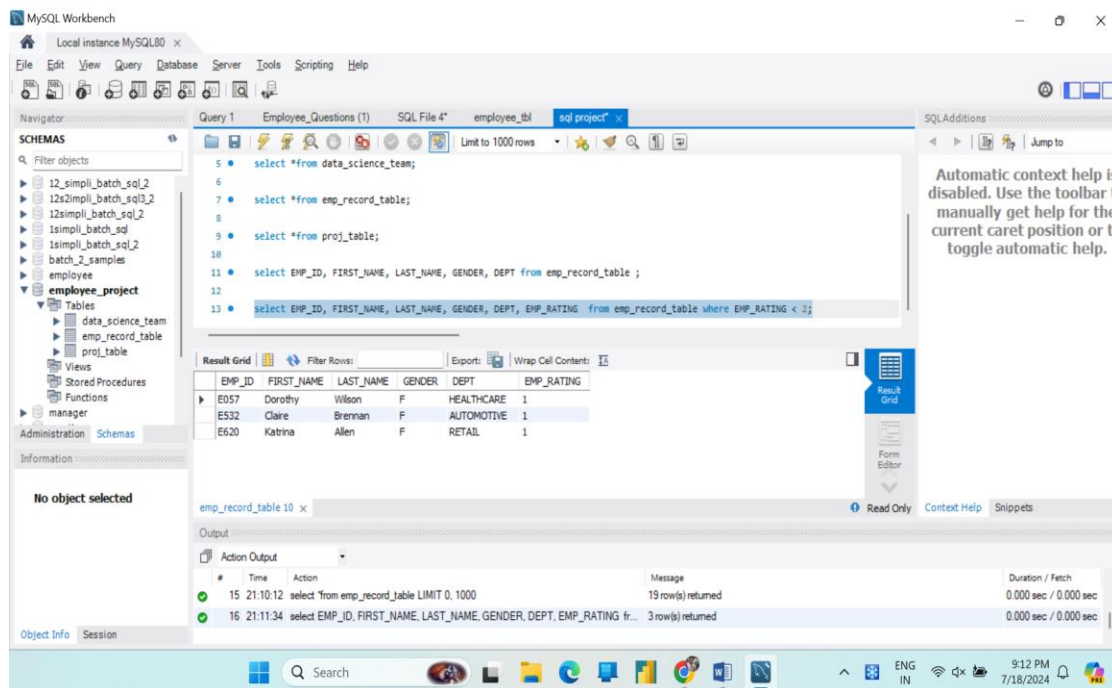


- Write a query to fetch EMP_ID, FIRST_NAME, LAST_NAME, GENDER, and DEPARTMENT from the employee record table, and make a list of employees and details of their department.

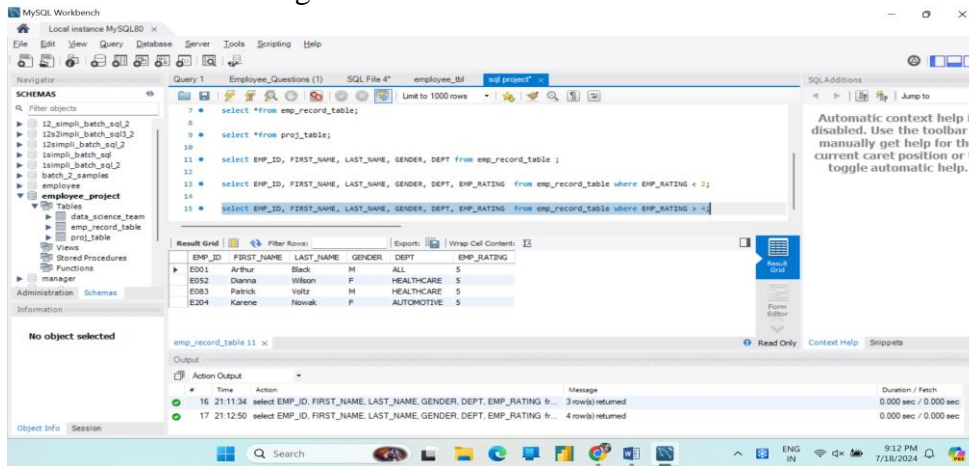


- Write a query to fetch EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPARTMENT, and EMP_RATING if the EMP_RATING is:

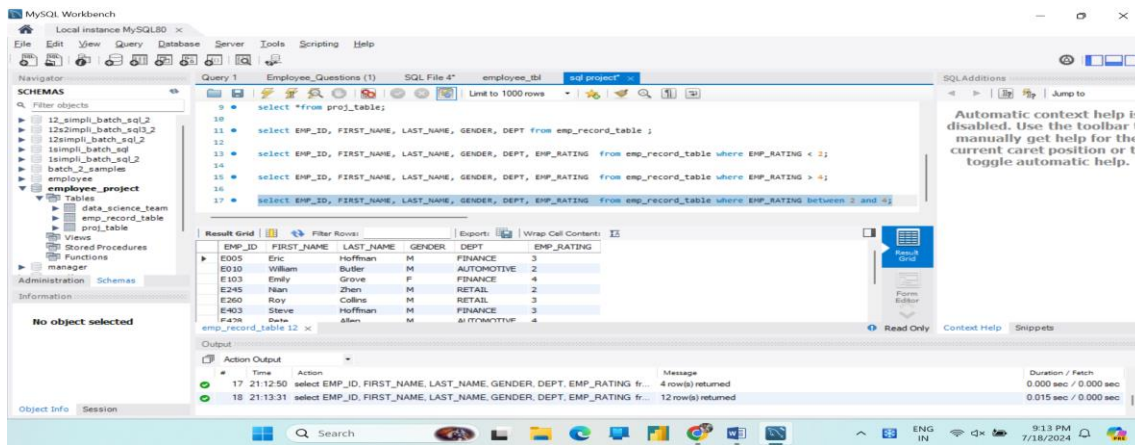
- less than two



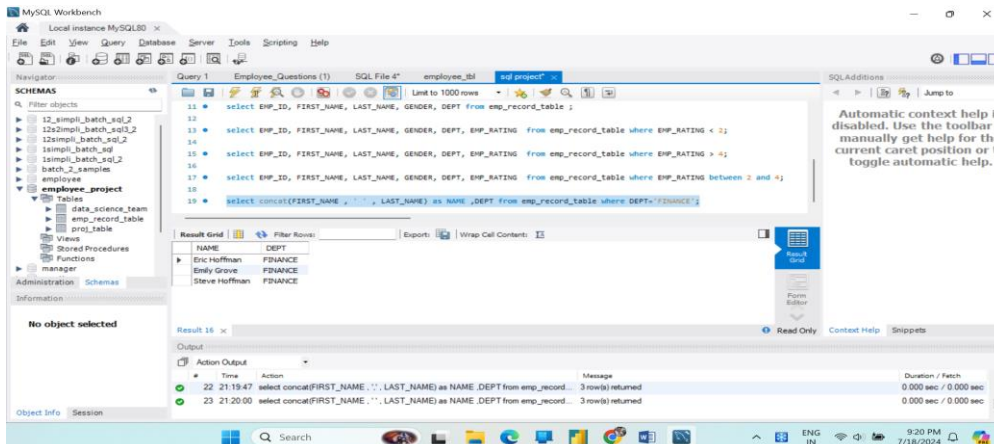
- greater than four



- between two and four



- 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.



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

The screenshot shows the MySQL Workbench interface. The SQL editor contains a query that selects employee details and the number of reporters for each employee. The query is as follows:

```
13 select EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING from emp_record_table where EMP_RATING < 2;
14
15 select EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING from emp_record_table where EMP_RATING > 4;
16
17 select EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING from emp_record_table where EMP_RATING between 2 and 4;
18
19 select concat(FIRST_NAME, ' ', LAST_NAME) as NAME, DEPT from emp_record_table where DEPT='FINANCE';
20
21 select employee.emp_id, concat(employee.first_name, ' ', employee.last_name) as employee_name, manager.manager_id,
22 concat(manager.first_name, ' ', manager.last_name) as Manager_Name, manager.role as role from emp_record_table employee
23 join emp_record_table manager on employee.manager_id = manager.emp_id;
```

The Result Grid shows the following data:

emp_id	employee_name	manager_id	Manager_name	role
E612	Tracy Norris	0000	Arthur Black	PRESIDENT
E583	Janet Hale	0000	Arthur Black	PRESIDENT
E428	Pete Allen	0000	Arthur Black	PRESIDENT
E103	Emily Grove	0000	Arthur Black	PRESIDENT
E083	Patrick Voltz	0000	Arthur Black	PRESIDENT
E505	Chad Wilson	E001	Patrick Voltz	MANAGER
E057	Dorothy Wilson	E001	Patrick Voltz	MANAGER
E052	Dianna Wilson	E001	Patrick Voltz	MANAGER
E403	Steve Hoffman	E001	Patrick Voltz	MANAGER

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.

The screenshot shows the MySQL Workbench interface. The SQL editor contains a query that uses a union to list employees from the healthcare and finance departments. The query is as follows:

```
19 select concat(FIRST_NAME, ' ', LAST_NAME) as NAME, DEPT from emp_record_table where DEPT='FINANCE';
20
21 select employee.emp_id, concat(employee.first_name, ' ', employee.last_name) as employee_name, manager.manager_id,
22 concat(manager.first_name, ' ', manager.last_name) as Manager_Name, manager.role as role from emp_record_table employee
23 join emp_record_table manager on employee.manager_id = manager.emp_id;
24
25 select EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT as Department from emp_record_table where dept = 'HEALTHCARE'
26 union
27 select EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT as Department from emp_record_table where dept = 'FINANCE';
```

The Result Grid shows the following data:

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	Department
E052	Dianna	Wilson	F	HEALTHCARE
E057	Dorothy	Wilson	F	HEALTHCARE
E083	Patrick	Voltz	M	HEALTHCARE
E505	Chad	Wilson	M	HEALTHCARE
E005	Eric	Hoffman	M	FINANCE
E103	Emily	Grove	F	FINANCE
E403	Steve	Hoffman	M	FINANCE

The Action Output shows the following results:

#	Time	Action	Message	Duration / Fetch
25	21:38:45	Select employee.emp_id, concat(employee.first_name, ' ', employee.last_name) ...	18 row(s) returned	0.000 sec / 0.000 sec
26	21:44:43	select EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT as Department from ...	7 row(s) returned	0.000 sec / 0.000 sec

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.

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

```

25 select EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT as Department from emp_record_table where dept = 'HEALTHCARE'
26 union
27 select EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT as Department from emp_record_table where dept = 'FINANCE';
28
29 select emp_id, first_name, last_name, role, dept, emp_rating, MAX(emp_rating) AS MAX_EMP_RATING
30 FROM emp_record_table
31 GROUP BY dept, emp_id, first_name, last_name, role, emp_rating;
32
33
34

```

The Result Grid shows the following data:

emp_id	first_name	last_name	role	dept	emp_rating	MAX_EMP_RATING
E001	Arthur	Black	PRESIDENT	ALL	5	5
E005	Eric	Hoffman	LEAD DATA SCIENTIST	FINANCE	3	3
E010	William	Butler	LEAD DATA SCIENTIST	AUTOMOTIVE	2	2
E052	Dianne	Wilson	SENIOR DATA SCIENTIST	HEALTHCARE	5	5
E057	Dorothy	Wilson	SENIOR DATA SCIENTIST	HEALTHCARE	1	1
E083	Patrick	Voltz	MANAGER	HEALTHCARE	5	5
E103	Emily	Grove	MANAGER	FINANCE	4	4

The Output pane shows the following message:

```

12 12:04:43 SET GLOBAL sql_mode=(SELECT REPLACE(@@sql_mode,'ONLY_FULL_GROU... 0 row(s) affected
0.000 sec

```

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.

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

```

26 union
27 select EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT as Department from emp_record_table where dept = 'FINANCE';
28
29 select emp_id, first_name, last_name, role, dept, emp_rating, MAX(emp_rating) AS MAX_EMP_RATING
30 FROM emp_record_table
31 GROUP BY dept, emp_id, first_name, last_name, role, emp_rating;
32
33 select role, max(salary) as max_sal, min(salary) as min_sal from emp_record_table GROUP BY role;
34
35

```

The Result Grid shows the following data:

role	max_sal	min_sal
PRESIDENT	16500	16500
LEAD DATA SCIENTIST	9000	8500
SENIOR DATA SCIENTIST	7700	5500
MANAGER	11000	8500
ASSOCIATE DATA SCIENTIST	5000	4000
SENIOR DATA SCIENTIST	7000	7000

The Output pane shows the following messages:

```

16 12:08:36 SELECT emp_id, first_name, last_name, role, dept, emp_rating, MAX(emp_rating) ... 19 row(s) returned
0.000 sec / 0.000 sec
17 12:10:52 SELECT emp_id, first_name, last_name, role, dept, emp_rating, MAX(emp_rating) ... Error Code: 1055. Expression #1 of SELECT list is not in GROUP BY clause and co... 0.000 sec
18 12:11:41 SELECT emp_id, first_name, last_name, role, dept, emp_rating, MAX(emp_rating) ... 19 row(s) returned
0.000 sec / 0.000 sec

```

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

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

```

27 select EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT as Department from emp_record_table where dept = 'FINANCE';
28
29 • SELECT emp_id, first_name, last_name, role, dept, emp_rating, MAX(emp_rating) AS MAX_EMP_RATING
30 FROM emp_record_table
31 GROUP BY dept, emp_id, first_name, last_name, role, emp_rating;
32
33 • SELECT role, max(salary) as max_sal, min(salary) as min_sal from emp_record_table GROUP BY role;
34
35 • select emp_id, first_name, last_name, role, dept, EXP, row_number() over (order by EXP DESC) as RANKING
36 from emp_record_table;
37
38

```

The Result Grid shows the following data:

emp_id	first_name	last_name	role	dept	EXP	RANKING
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	4
E583	Janet	Hale	MANAGER	RETAIL	14	5
E612	Tracy	Norris	MANAGER	RETAIL	13	6
E010	William	Butler	LEAD DATA SCIENTIST	AUTOMOTIVE	12	7

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.

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

```

34
35 • select emp_id, first_name, last_name, role, dept, EXP, row_number() over (order by EXP DESC) as RANKING
36 from emp_record_table;
37
38 • create view SalaryEmployees as select emp_id, first_name, last_name, country, salary from emp_record_table
39 where salary > 6000;
40 • select * from SalaryEmployees;
41
42

```

The Result Grid shows the following data:

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

The Action Output shows the following messages:

#	Time	Action	Message	Duration / Fetch
22	12:22:48	select from emp_record_table LIMIT 0, 1000	19 row(s) returned	0.000 sec / 0.000 sec
23	12:31:50	create view SalaryEmployees as select emp_id, first_name, last_name, country, sal...	0 row(s) affected	0.031 sec
24	12:32:47	select from SalaryEmployees LIMIT 0, 1000	12 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows MySQL Workbench with a query editor containing the following SQL code:

```

38 • create view SalaryEmployees as select emp_id, first_name, last_name, country, salary from emp_record_table
39 • where salary > 60000;
40 • select * from SalaryEmployees;
41
42 • select EMP_ID, FIRST_NAME, LAST_NAME, exp
43 • from (select * from emp_record_table
44 • where EXP > 10
45 • order by exp)
46 • as EXP_GREATER_THAN_10;
47

```

The Result Grid shows the following data:

EMP_ID	FIRST_NAME	LAST_NAME	EXP
E005	Eric	Hoffman	11
E010	William	Butler	12
E612	Tracy	Norris	13
E103	Emily	Grove	14
E428	Pete	Allen	14

The Output pane shows the execution of the query, indicating that 8 row(s) were returned.

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.

The screenshot shows MySQL Workbench with a query editor containing the following SQL code:

```

48 • USE `employee_project`;
49 • DROP procedure IF EXISTS `EMP_DETAILS`;
50
51 • USE `employee_project`;
52 • DROP procedure IF EXISTS `employee_project`.`EMP_DETAILS`;
53 •
54
55 DELIMITER $$
56 • USE `employee_project` $$
57 • CREATE DEFINER=`root`@`localhost` PROCEDURE `EMP_DETAILS`()
58 • BEGIN
59 • SELECT * FROM femp_record_table WHERE EXP > 3 order by EXP;
60 • END$$
61
62 DELIMITER ;
63 •
64
65 • CALL EMP_DETAILS;

```

The Result Grid shows the following data:

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT	Salary
E403	Steve	Hoffman	M	ASSOCIATE DATA SCIENTIST	FINANCE	4	USA	NORTH AMERICA	50
E505	Chad	Wilson	M	ASSOCIATE DATA SCIENTIST	HEALTHCARE	5	CANADA	NORTH AMERICA	50
E052	Dianna	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	6	CANADA	NORTH AMERICA	55
E245	Nian	Zhen	M	SENIOR DATA SCIENTIST	RETAIL	6	CHINA	ASIA	65
E628	Pete	Allen	M	SENIOR DATA SCIENTIST	RETAIL	7	INDIA	ASIA	70

The Output pane shows the execution of the stored procedure, indicating that 5 row(s) were returned.

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'.

The screenshot displays the MySQL Workbench interface. The 'Schemas' pane on the left shows the 'employee_project' database selected. The 'Query Editor' in the center contains a SQL script that defines a stored function 'check_role' and a query to use it. The 'Result Grid' at the bottom shows the output of the query, listing employees and their assigned roles based on their experience.

```
CREATE FUNCTION check_role(exp int) RETURNS VARCHAR(40)
DETERMINISTIC
BEGIN
    DECLARE check_role VARCHAR(40);
    IF exp <= 2 THEN SET check_role = "JUNIOR DATA SCIENTIST";
    elseif exp > 2 AND exp <= 5 THEN SET check_role = "ASSOCIATE DATA SCIENTIST";
    elseif exp > 5 AND exp <= 10 THEN SET check_role = "SENIOR DATA SCIENTIST";
    elseif exp > 10 AND exp <= 12 THEN SET check_role = "LEAD DATA SCIENTIST";
    elseif exp > 12 AND exp <= 16 THEN SET check_role = "MANAGER";
    end if;
    RETURN(check_role);
END //
delimiter ;

-- checking Data Science Team
select EMP_ID, FIRST_NAME, LAST_NAME, EXP, ROLE, check_role(exp)
from data_science_team WHERE ROLE = check_role(exp);
```

EMP_ID	FIRST_NAME	LAST_NAME	EXP	ROLE	check_role(exp)
E005	Eric	Hoffman	11	LEAD DATA SCIENTIST	LEAD DATA SCIENTIST
E010	William	Butler	12	LEAD DATA SCIENTIST	LEAD DATA SCIENTIST
E052	Dianna	Wilson	6	SENIOR DATA SCIENTIST	SENIOR DATA SCIENTIST
E057	Dorothy	Wilson	9	SENIOR DATA SCIENTIST	SENIOR DATA SCIENTIST
E204	Karen	Novak	8	SENIOR DATA SCIENTIST	SENIOR DATA SCIENTIST
E245	Nian	Zhen	6	SENIOR DATA SCIENTIST	SENIOR DATA SCIENTIST
E760	Ravi	Online	7	SENIOR DATA SCIENTIST	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.

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

```
68  elseif exp > 2 AND exp <= 5 THEN SET check_role = "ASSOCIATE DATA SCIENTIST";
69  elseif exp > 5 AND exp <= 10 THEN SET check_role = "SENIOR DATA SCIENTIST";
70  elseif exp > 10 AND exp <= 12 THEN SET check_role = "LEAD DATA SCIENTIST";
71  elseif exp > 12 AND exp <= 16 THEN SET check_role = "MANAGER";
72  end if;
73  RETURN(check_role);
74  END //
75  delimiter ;
76
77  -- checking Data Science Team
78  select EMP_ID, FIRST_NAME, LAST_NAME, EXP_ROLE, check_role(exp)
79  from data_science_team WHERE ROLE = check_role(exp);
80
81  create index idx_first_name on emp_record_table(FIRST_NAME(20));
82  explain select * from emp_record_table where FIRST_NAME = 'Eric';
83  show indexes from emp_record_table;
```

The Result Grid shows the execution plan for the query in line 82:

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	emp_record_table		ref	idx_first_name	idx_first_name	803	const	1	100.00	Using where

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).

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

```
71  elseif exp > 12 AND exp <= 16 THEN SET check_role = "MANAGER";
72  end if;
73  RETURN(check_role);
74  END //
75  delimiter ;
76
77  -- checking Data Science Team
78  select EMP_ID, FIRST_NAME, LAST_NAME, EXP_ROLE, check_role(exp)
79  from data_science_team WHERE ROLE = check_role(exp);
80
81  create index idx_first_name on emp_record_table(FIRST_NAME(20));
82  explain select * from emp_record_table where FIRST_NAME = 'Eric';
83  show indexes from emp_record_table;
84
85  select EMP_ID, FIRST_NAME, LAST_NAME, role, DEPT, EXP, salary, EMP_RATING, SALARY, (SALARY*0.05)*EMP_RATING
86  as BONUS from emp_record_table;
```

The Result Grid shows the output of the query in line 85:

EMP_ID	FIRST_NAME	LAST_NAME	role	DEPT	EXP	salary	EMP_RATING	SALARY	BONUS
E001	Arthur	Black	PRESIDENT	ALL	20	16500	5	16500	4125.00
E005	Eric	Hoffman	LEAD DATA SCIENTIST	FINANCE	11	8500	3	8500	1275.00
E010	William	Butler	LEAD DATA SCIENTIST	AUTOMOTIVE	12	9000	2	9000	900.00
E052	Dianna	Wilson	SENIOR DATA SCIENTIST	HEALTHCARE	6	5500	5	5500	1375.00
E057	Dorothy	Wilson	SENIOR DATA SCIENTIST	HEALTHCARE	9	7700	1	7700	385.00
E083	Patrick	Voltz	MANAGER	HEALTHCARE	15	9500	5	9500	2375.00
E103	Emily	Grove	MANAGER	FINANCE	14	10500	4	10500	2100.00

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

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with the 'employee_project' database selected. The main editor window shows a SQL query (Query 1) with the following code:

```
79 from data_science_team WHERE ROLE = check_role(emp);
80
81 create index idx_first_name on emp_record_table(FIRST_NAME(200));
82 explain select * from emp_record_table where FIRST_NAME = "Eric";
83 show indexes from emp_record_table;
84
85 select EMP_ID, FIRST_NAME, LAST_NAME, role, DEPT, EXP, salary, EMP_RATING, SALARY, (SALARY*0.05)*EMP_RATING
86 as BONUS from emp_record_table;
87
88
89 select EMP_ID, FIRST_NAME, LAST_NAME, COUNTRY, CONTINENT, avg(SALARY) from emp_record_table
90 group by CONTINENT, COUNTRY, EMP_ID, FIRST_NAME, LAST_NAME
91 order by CONTINENT, COUNTRY;
92
93
```

The 'Result Grid' at the bottom displays the results of the query, showing columns: EMP_ID, FIRST_NAME, LAST_NAME, COUNTRY, CONTINENT, and avg(SALARY). The results are as follows:

EMP_ID	FIRST_NAME	LAST_NAME	COUNTRY	CONTINENT	avg(SALARY)
E245	Nian	Zhen	CHINA	ASIA	6500.0000
E260	Roy	Collins	INDIA	ASIA	7000.0000
E612	Tracy	Norris	INDIA	ASIA	8500.0000
E620	Kabrina	Allen	INDIA	ASIA	3000.0000
E010	William	Butler	FRANCE	EUROPE	9000.0000
E204	Karene	Nowak	GERMANY	EUROPE	7500.0000
E428	Pete	Allen	GERMANY	EUROPE	11000.0000

The right sidebar shows a message: 'Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.'