

Name :- Siddhant Suresh Tondare

Project 1 : ScienceQtech Employee Performance Mapping.

The Task 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.
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
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.
6. Write a query to list only those employees who have someone reporting to them. Also, show the number of reporters (including the President).
6. Write a query to list down all the employees from the healthcare and finance departments using union. Take data from the employee record table.
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.
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.
10. Write a query to assign ranks to each employee based on their experience. Take data from the employee record table.

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.

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

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'.

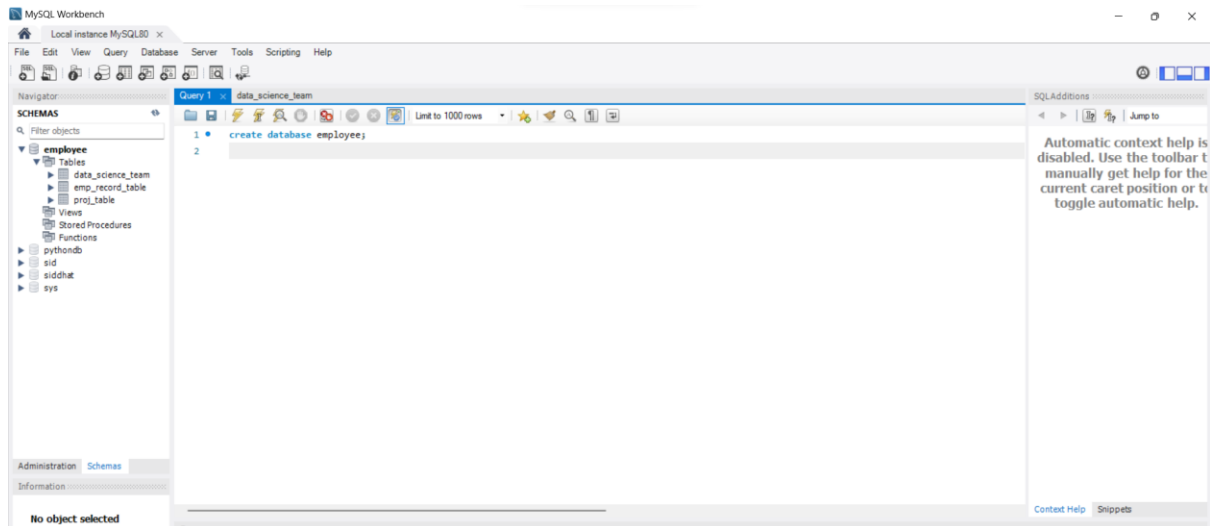
15. Create an index to improve the cost and performance of the query to find the employee whose FIRST_NAME is 'Eric' in the employee table after checking the execution plan.

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).

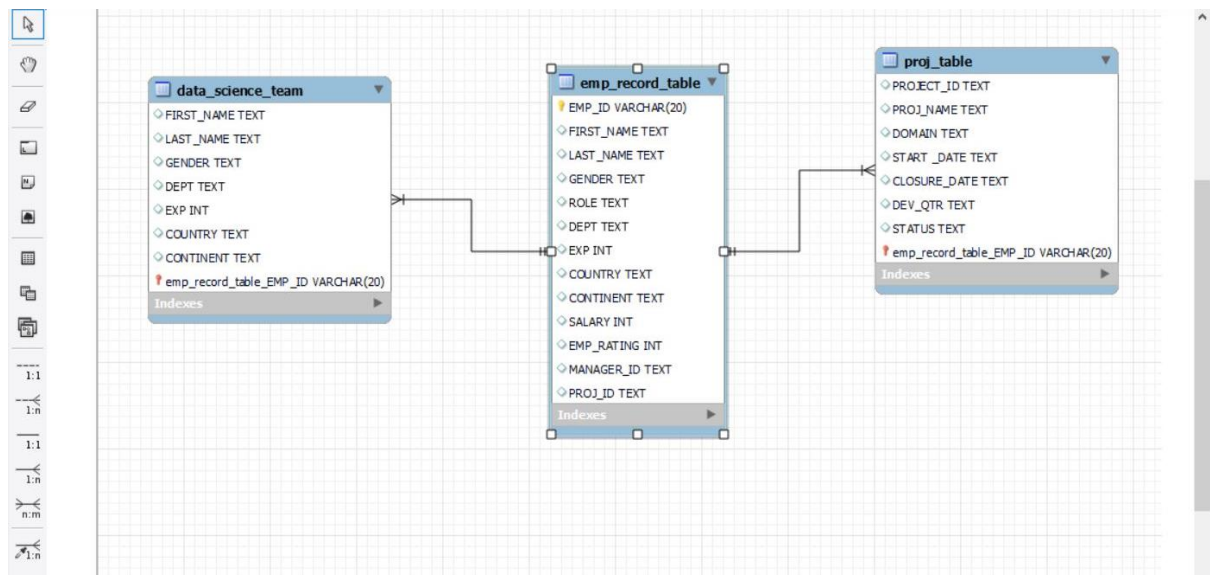
17. Write a query to calculate the average salary distribution based on the continent and country. Take data from the employee record table.

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.

1A. CREATE DATABASE **employee**;



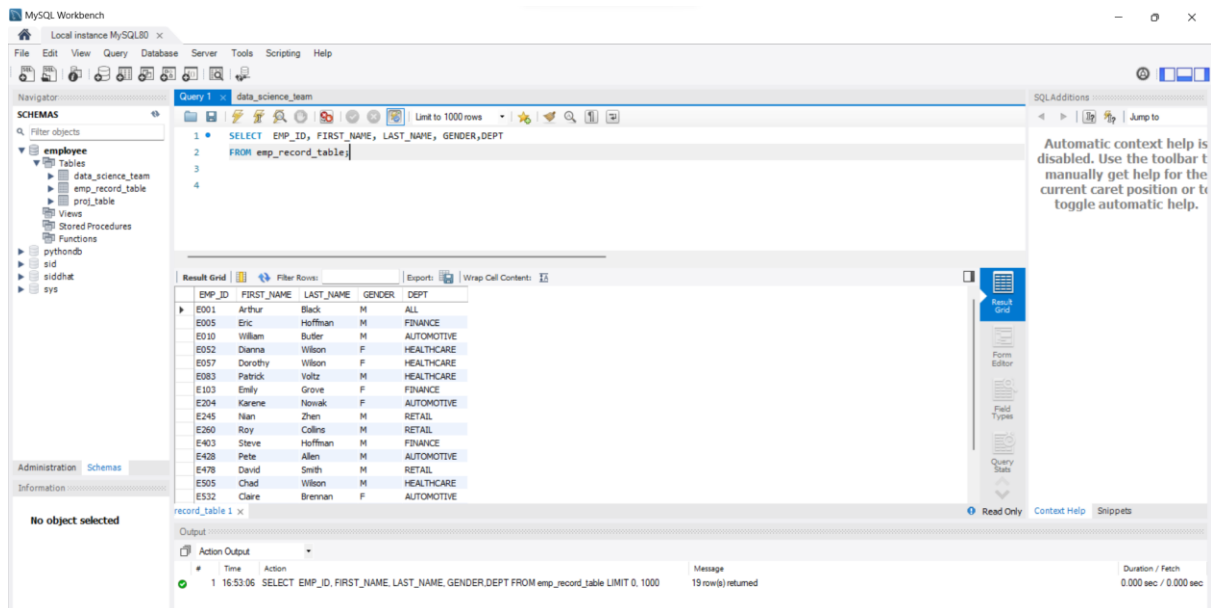
2. Create an ER diagram for the given **employee** database.



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.

3A.

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



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

4A. less than two

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

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

employee

Tables

data_science_team

emp_record_table

emp_table

Views

Stored Procedures

Functions

pythondb

sid

sidchat

sys

Administration Schemas

Information

No object selected

Query 1: data_science_team

Limit to 1000 rows

```
1 SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER,DEPT,EMP_RATING FROM emp_record_table
2 WHERE EMP_RATING<4;
3
4
5
```

Result Grid

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

emp_record_table 2

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	16.53.06	SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER,DEPT FROM emp_record_table LIMIT 0, 1000	19 row(s) returned	0.000 sec / 0.000 sec
2	16.55.25	SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER,DEPT,EMP_RATING FROM emp_record_table W...	3 row(s) returned	0.000 sec / 0.000 sec

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Read Only Context Help Snippets

4B.

greater than four

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

WHERE EMP_RATING>4;

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

employee

Tables

data_science_team

emp_record_table

emp_table

Views

Stored Procedures

Functions

pythondb

sid

sidchat

sys

Administration Schemas

Information

No object selected

Query 1: data_science_team

Limit to 1000 rows

```
1 SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER,DEPT,EMP_RATING FROM emp_record_table
2 WHERE EMP_RATING>4;
3
4
5
6
```

Result Grid

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 3

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	16.53.06	SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER,DEPT FROM emp_record_table LIMIT 0, 1000	19 row(s) returned	0.000 sec / 0.000 sec
2	16.55.25	SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER,DEPT,EMP_RATING FROM emp_record_table W...	3 row(s) returned	0.000 sec / 0.000 sec
3	16.57.37	SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER,DEPT,EMP_RATING FROM emp_record_table W...	4 row(s) returned	0.000 sec / 0.000 sec

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

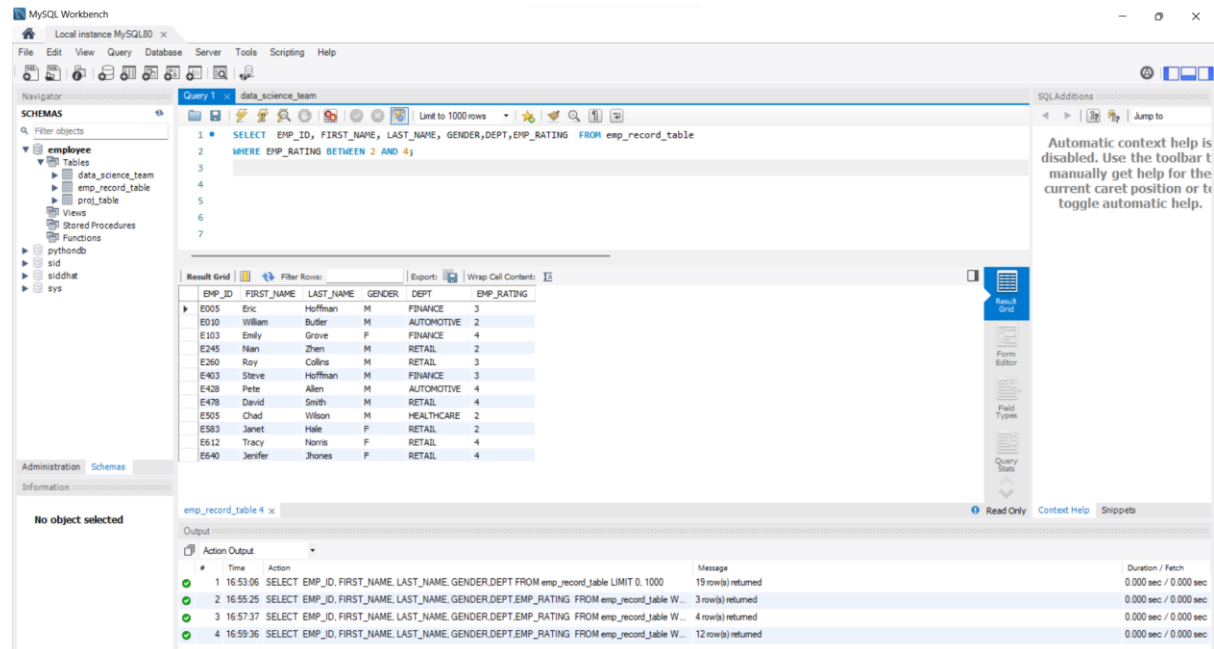
Read Only Context Help Snippets

4C.

between two and four

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

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



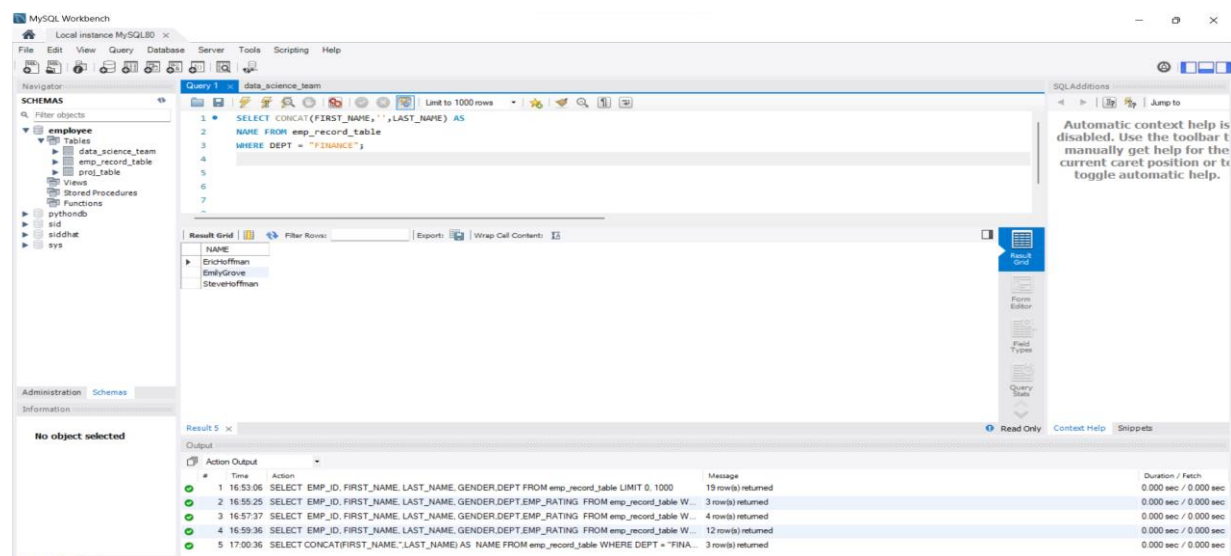
5. Write a query to concatenate the FIRST_NAME and the LAST_NAME of employees in the Finance department from the employee table and then give the resultant column alias as NAME.

5A.

```
SELECT 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).

6A.

```
SELECT m.EMP_ID,m.FIRST_NAME,m.LAST_NAME,m.ROLE,  
  
m.EXP,COUNT(e.EMP_ID) as "EMP_COUNT"  
  
FROM emp_record_table m  
  
INNER JOIN emp_record_table e  
  
ON m.EMP_ID = e.MANAGER_ID  
  
GROUP BY m.EMP_ID  
  
ORDER BY m.EMP_ID;
```

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

```
2 m.EXP,COUNT(e.EMP_ID) as "EMP_COUNT"  
3 FROM emp_record_table m  
4 INNER JOIN emp_record_table e  
5 ON m.EMP_ID = e.MANAGER_ID  
6 GROUP BY m.EMP_ID  
7 ORDER BY m.EMP_ID;  
8
```

The Results grid displays the following data:

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
E612	Tracy	Norris	MANAGER	13	2

The bottom panel shows the Output window with the following message:

```
1 16:53:06 SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT FROM emp_record_table LIMIT 0, 1000 19 row(s) returned 0.000 sec / 0.000 sec  
2 16:55:25 SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING FROM emp_record_table W... 3 row(s) returned 0.000 sec / 0.000 sec  
3 16:57:37 SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING FROM emp_record_table W... 4 row(s) returned 0.000 sec / 0.000 sec  
4 16:59:36 SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING FROM emp_record_table W... 12 row(s) returned 0.000 sec / 0.000 sec  
5 17:00:36 SELECT CONCAT(FIRST_NAME, LAST_NAME) AS NAME FROM emp_record_table WHERE DEPT = "FINA... 3 row(s) returned 0.000 sec / 0.000 sec  
6 17:04:43 SELECT m.EMP_ID,m.FIRST_NAME,m.LAST_NAME,m.ROLE,m.EXP,COUNT(e.EMP_ID) as "EMP_COUNT" ... 6 row(s) returned 0.000 sec / 0.000 sec
```

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.

7A.

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

```
WHERE DEPT = "HEALTHCARE"
```

```
UNION
```

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

```
WHERE DEPT = "FINANCE"
```

```
ORDER BY DEPT, EMP_ID;
```

The screenshot shows the MySQL Workbench interface. The 'Query' tab is active, displaying the following SQL query:

```
1 SELECT EMP_ID, FIRST_NAME, LAST_NAME, DEPT FROM emp_record_table
2 WHERE DEPT = "HEALTHCARE"
3 UNION
4 SELECT EMP_ID, FIRST_NAME, LAST_NAME, DEPT FROM emp_record_table
5 WHERE DEPT = "FINANCE"
6 ORDER BY DEPT, EMP_ID;
```

The 'Result Grid' shows the output of the query, which is a table with 4 columns: EMP_ID, FIRST_NAME, LAST_NAME, and DEPT. The data is as follows:

EMP_ID	FIRST_NAME	LAST_NAME	DEPT
E005	Eric	Hoffman	FINANCE
E103	Emily	Grove	FINANCE
E403	Steve	Hoffman	FINANCE
E052	Danna	Wilson	HEALTHCARE
E057	Dorothy	Wilson	HEALTHCARE
E083	Patrick	Voltz	HEALTHCARE
E505	Chad	Wilson	HEALTHCARE

The 'Output' tab at the bottom shows the execution log, including the time taken for each query step and the number of rows returned.

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.

8A.

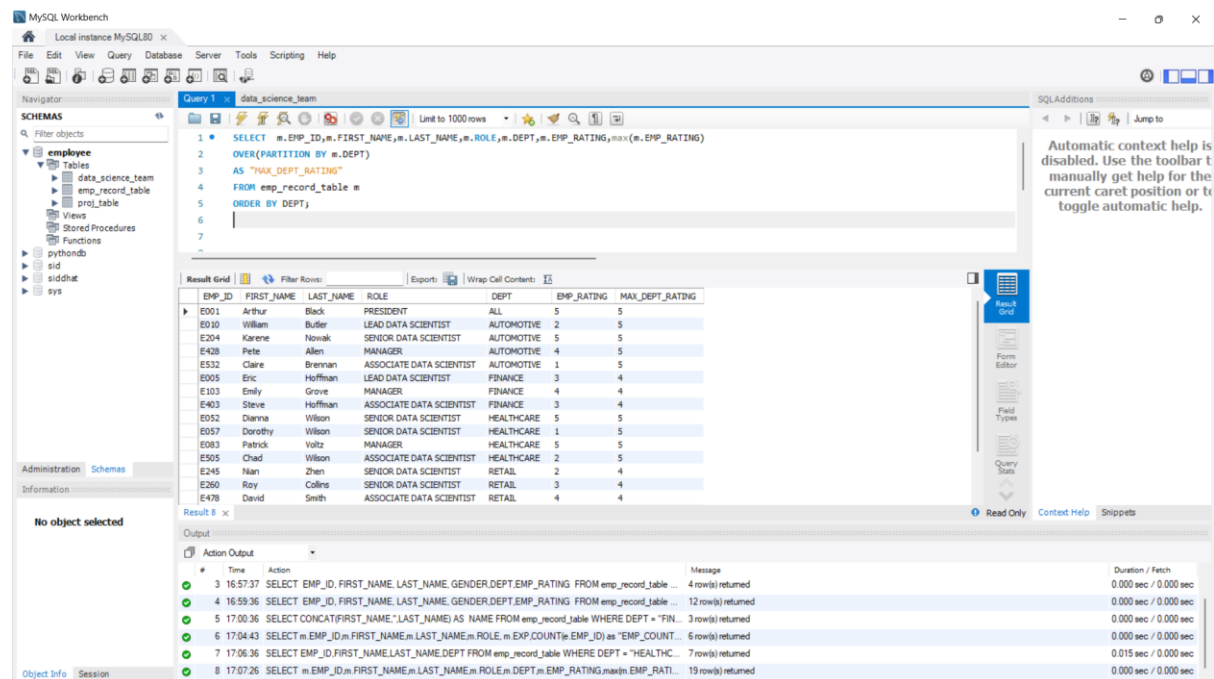
```
SELECT
m.EMP_ID,m.FIRST_NAME,m.LAST_NAME,m.ROLE,m.DEPT,m.EMP_RATING,max(m.EMP_
RATING)

OVER(PARTITION BY m.DEPT)

AS "MAX_DEPT_RATING"

FROM emp_record_table m

ORDER BY DEPT;
```



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

```
1 SELECT m.EMP_ID,m.FIRST_NAME,m.LAST_NAME,m.ROLE,m.DEPT,m.EMP_RATING,max(m.EMP_RATING)
2 OVER(PARTITION BY m.DEPT)
3 AS "MAX_DEPT_RATING"
4 FROM emp_record_table m
5 ORDER BY DEPT;
```

The Results Grid displays the following data:

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
E005	Eric	Hoffman	LEAD DATA SCIENTIST	FINANCE	3	4
E103	Emily	Grove	MANAGER	FINANCE	4	4
E403	Steve	Huffman	ASSOCIATE DATA SCIENTIST	FINANCE	3	4
E052	Dianna	Wilson	SENIOR DATA SCIENTIST	HEALTHCARE	5	5
E057	Dorothy	Wilson	SENIOR DATA SCIENTIST	HEALTHCARE	1	5
E083	Patrick	Voltz	MANAGER	HEALTHCARE	5	5
E505	Chad	Wilson	ASSOCIATE DATA SCIENTIST	HEALTHCARE	2	5
E245	Nan	Zhen	SENIOR DATA SCIENTIST	RETAIL	2	4
E260	Roy	Collins	SENIOR DATA SCIENTIST	RETAIL	3	4
E478	David	Smith	ASSOCIATE DATA SCIENTIST	RETAIL	4	4

The Action Output pane shows the execution of the query, indicating that 19 rows were returned.

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.

9A.

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, ROLE, MAX(SALARY), MIN(SALARY)

FROM emp_record_table

WHERE ROLE IN("PRESIDENT","LEAD DATA SCIENTIST","SENIOR DATA
SCIENTIST","MANAGER","ASSOCIATE DATA SCIENTIST","JUNIOR DATA SCIENTIST")

GROUP BY ROLE;
```

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHMAS

Filter objects

Tables

- data_science_team
- emp_record_table
- proj_table
- Views
- Stored Procedures
- Functions
- pythondb
- sid
- sidhat
- sys

Administration Schemas

Information

No object selected

Query 1: data_science_team

```

1 SELECT EMP_ID, FIRST_NAME, LAST_NAME, ROLE, MAX(SALARY), MIN(SALARY)
2 -- Execute the selected portion of the script or everything, if there is no selection
3 WHERE ROLE IN('PRESIDENT','LEAD DATA SCIENTIST','SENIOR DATA SCIENTIST','MANAGER','ASSOCIATE DATA SCIENTIST','JUNIOR DATA SCIENTIST')
4 GROUP BY ROLE;
5
6
7

```

Result Grid

EMP_ID	FIRST_NAME	LAST_NAME	ROLE	MAX(SALARY)	MIN(SALARY)
E001	Arthur	Black	PRESIDENT	16500	16500
E005	Eric	Hoffman	LEAD DATA SCIENTIST	9000	8500
E052	Dianna	Wilson	SENIOR DATA SCIENTIST	7700	5500
E083	Patrick	Voltz	MANAGER	11000	8500
E403	Steve	Hoffman	ASSOCIATE DATA SCIENTIST	5000	4000
E620	Katrina	Allen	JUNIOR DATA SCIENTIST	3000	2800

Output

#	Time	Action	Message	Duration / Fetch
4	16:59:36	SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING FROM emp_record_table	12 row(s) returned	0.000 sec / 0.000 sec
5	17:00:36	SELECT CONCAT(FIRST_NAME, LAST_NAME) AS NAME FROM emp_record_table WHERE DEPT = 'FIN...	3 row(s) returned	0.000 sec / 0.000 sec
6	17:04:43	SELECT m.EMP_ID, m.FIRST_NAME, m.LAST_NAME, m.ROLE, m.EXP COUNT(m.EMP_ID) as 'EMP_COUNT...	6 row(s) returned	0.000 sec / 0.000 sec
7	17:06:36	SELECT EMP_ID, FIRST_NAME, LAST_NAME, DEPT FROM emp_record_table WHERE DEPT = 'HEALTHC...	7 row(s) returned	0.015 sec / 0.000 sec
8	17:07:26	SELECT m.EMP_ID, m.FIRST_NAME, m.LAST_NAME, m.ROLE, m.DEPT, m.EMP_RATING, max(m.EMP_RATI...	13 row(s) returned	0.000 sec / 0.000 sec
9	17:08:08	SELECT EMP_ID, FIRST_NAME, LAST_NAME, ROLE, MAX(SALARY), MIN(SALARY) FROM emp_record_ta...	6 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

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

10A.

SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP,

RANK() OVER(ORDER BY EXP) EXP_RANK

FROM emp_record_table;

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHMAS

Filter objects

Tables

- data_science_team
- emp_record_table
- proj_table
- Views
- Stored Procedures
- Functions
- pythondb
- sid
- sidhat
- sys

Administration Schemas

Information

No object selected

Query 1: data_science_team

```

1 SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP,
2 RANK() OVER(ORDER BY EXP) EXP_RANK
3 FROM emp_record_table;
4
5
6
7

```

Result Grid

EMP_ID	FIRST_NAME	LAST_NAME	EXP	EXP_RANK
E640	Jennifer	Jones	1	1
E620	Katrina	Allen	2	2
E478	David	Smith	3	3
E532	Claire	Brennan	3	3
E403	Steve	Hoffman	4	5
E505	Chad	Wilson	5	6
E052	Dianna	Wilson	6	7
E245	Nian	Zhen	6	7
E260	Roy	Collins	7	9
E204	Karen	Nowak	8	10
E057	Dorothy	Wilson	9	11
E005	Eric	Hoffman	11	12
E010	William	Butler	12	13
E612	Tracy	Norris	13	14
E103	Emily	Grove	14	15

Output

#	Time	Action	Message	Duration / Fetch
5	17:00:36	SELECT CONCAT(FIRST_NAME, LAST_NAME) AS NAME FROM emp_record_table WHERE DEPT = 'FIN...	3 row(s) returned	0.000 sec / 0.000 sec
6	17:04:43	SELECT m.EMP_ID, m.FIRST_NAME, m.LAST_NAME, m.ROLE, m.EXP COUNT(m.EMP_ID) as 'EMP_COUNT...	6 row(s) returned	0.000 sec / 0.000 sec
7	17:06:36	SELECT EMP_ID, FIRST_NAME, LAST_NAME, DEPT FROM emp_record_table WHERE DEPT = 'HEALTHC...	7 row(s) returned	0.015 sec / 0.000 sec
8	17:07:26	SELECT m.EMP_ID, m.FIRST_NAME, m.LAST_NAME, m.ROLE, m.DEPT, m.EMP_RATING, max(m.EMP_RATI...	13 row(s) returned	0.000 sec / 0.000 sec
9	17:08:08	SELECT EMP_ID, FIRST_NAME, LAST_NAME, ROLE, MAX(SALARY), MIN(SALARY) FROM emp_record_ta...	6 row(s) returned	0.000 sec / 0.000 sec
10	17:09:01	SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP, RANK() OVER(ORDER BY EXP) EXP_RANK FROM em...	13 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

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.

11A.

```
CREATE VIEW employees_in_various_countries AS
```

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, COUNTRY, SALARY
```

```
FROM emp_record_table
```

```
WHERE SALARY > 6000;
```

```
SELECT * FROM employees_in_various_countries;
```

The screenshot shows a database query editor with a query window and a result grid. The query window contains the following SQL code:

```
1 • CREATE VIEW employees_in_various_countries AS
2   SELECT EMP_ID, FIRST_NAME, LAST_NAME, COUNTRY, SALARY
3   FROM emp_record_table
4   WHERE SALARY > 6000;
5
6 • SELECT * FROM employees_in_various_countries;
7
```

The result grid displays the data returned by the second query. It has columns for EMP_ID, FIRST_NAME, LAST_NAME, COUNTRY, and SALARY. The data is as follows:

EMP_ID	FIRST_NAME	LAST_NAME	COUNTRY	SALARY
E001	Arthur	Blad Black	USA	16500
E005	Eric	Hoffman	USA	8500
E010	William	Butler	FRANCE	9000
E057	Dorothy	Wilson	USA	7700
E083	Patrick	Voltz	USA	9500
E103	Emily	Grove	CANADA	10500
E204	Karene	Nowak	GERMANY	7500
E245	Nian	Zhen	CHINA	6500
E260	Roy	Collins	INDIA	7000
E428	Pete	Allen	GERMANY	11000
E583	Janet	Hale	COLOMBIA	10000
E612	Tracy	Norris	INDIA	8500

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

12A.

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP FROM emp_record_table
```

```
WHERE EMP_ID IN (SELECT manager_id FROM emp_record_table);
```

The screenshot shows the SQL Developer interface. On the left, the 'SCHEMAS' pane shows the 'employee' schema with tables 'data_science_team', 'emp_record_table', and 'proj_table'. The main 'Query Editor' contains the following SQL query:

```

1 SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP FROM emp_record_table
2 WHERE EMP_ID IN (SELECT manager_id FROM emp_record_table);
3
4
5
6
7

```

The 'Result Grid' at the bottom displays the results of the query:

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
E612	Tracy	Norris	13

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.

13A.

DELIMITER &&

CREATE PROCEDURE get_experience_details()

BEGIN

SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP FROM emp_record_table WHERE EXP > 3;

END &&

CALL get_experience_details();

The screenshot shows the SQL Developer interface. On the left, the 'SCHEMAS' pane shows the 'employee' schema with tables 'data_science_team', 'emp_record_table', and 'proj_table'. The main 'Query Editor' contains the following SQL query:

```

1 DELIMITER &&
2 CREATE PROCEDURE get_experience_details()
3 BEGIN
4 SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP FROM emp_record_table WHERE EXP > 3;
5 END &&
6 CALL get_experience_details();
7

```

The 'Result Grid' at the bottom displays the results of the query:

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
E083	Patrick	Voltz	15
E103	Emily	Grove	14
E204	Karene	Nowak	8
E245	Nian	Zhen	6
E260	Roy	Collins	7
E403	Steve	Hoffman	4
E428	Pete	Allen	14
E505	Chad	Wilson	5
E583	Janet	Hale	14
E612	Tracy	Norris	13

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'.

14A.

DELIMITER &&

```
CREATE FUNCTION Employee_ROLE(
```

```
EXP int
```

```
)
```

```
RETURNS VARCHAR(40)
```

```
DETERMINISTIC
```

```
BEGIN
```

```
DECLARE Employee_ROLE VARCHAR(40);
```

```
IF EXP>12 AND 16 THEN
```

```
SET Employee_ROLE="MANAGER";
```

```
ELSEIF EXP>10 AND 12 THEN
```

```
SET Employee_ROLE ="LEAD DATA SCIENTIST";
```

```
ELSEIF EXP>5 AND 10 THEN
```

```
SET Employee_ROLE ="SENIOR DATA SCIENTIST";
```

```
ELSEIF EXP>2 AND 5 THEN
```

```
SET Employee_ROLE ="ASSOCIATE DATA SCIENTIST";
```

```
ELSEIF EXP<=2 THEN
```

```
SET Employee_ROLE ="JUNIOR DATA SCIENTIST";
```

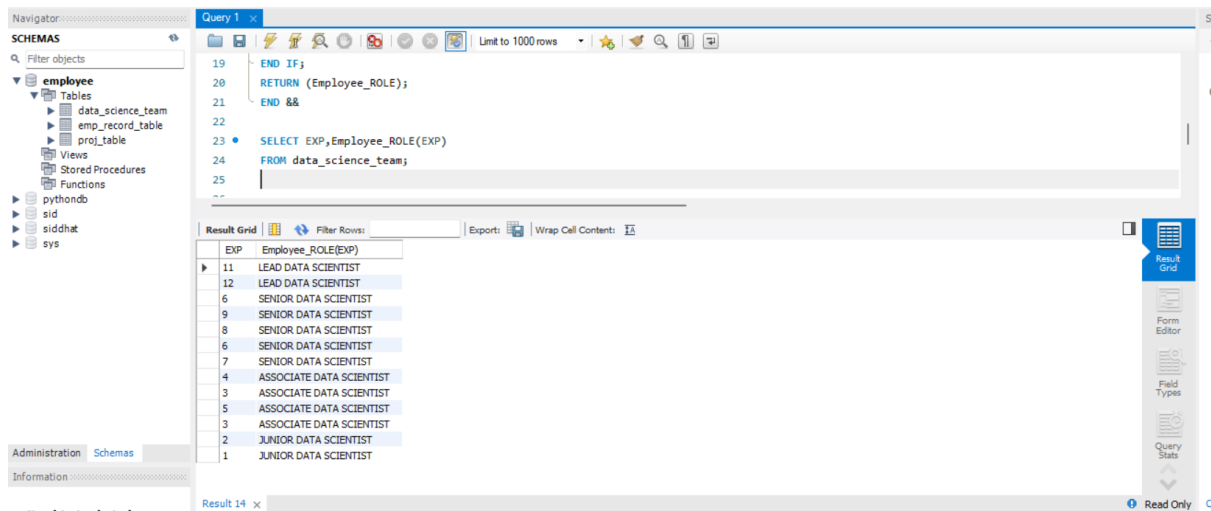
```
END IF;
```

```
RETURN (Employee_ROLE);
```

```
END &&
```

```
SELECT EXP,Employee_ROLE(EXP)
```

```
FROM data_science_team;
```



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.

15A.

```
CREATE INDEX idx_first_name
```

```
ON emp_record_table(FIRST_NAME(20));
```

```
SELECT * FROM emp_record_table
```

```
WHERE FIRST_NAME='Eric';
```

Query 1:
 1 CREATE INDEX idx_first_name
 2 ON emp_record_table(FIRST_NAME(20));
 3 SELECT * FROM emp_record_table
 4 WHERE FIRST_NAME='Eric';
 5
 6
 7

Visual Explain | Display Info | Read = Eval cost | Overview | View Source

Query cost: 0.35
 Query_block #1
 0.35 1 row
 Non-Unique Key Lookup
 emp_record_table
 idx_first_name

emp_record_table
 Access Type: ref
 Non-Unique Key Lookup
 Cost Hint: Low-medium - Low if number of matching rows is small, higher as the number of rows increases.
 Used Columns: EMP_ID, FIRST_NAME, LAST_NAME, GENDER, ROLE, DEPT, DIV, COUNTRY, CONTINENT, SALARY, EMP_RATING, MANAGER_ID, PROJ_ID
 Key/Index: idx_first_name
 Ref.: const
 Used Key Parts: FIRST_NAME
 Possible Keys: idx_first_name
 Attached Condition: (emp_record_table.FIRST_NAME = 'Eric')
 Rows Examined per Scan: 1
 Rows Produced per Join: 1
 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: 0.10
 Prefix: 0.35
 Data Read: 120

Output
 Action Output
 # Time Action
 15 17:15:52 CALL get_expense_details()
 16 17:17:12 CREATE FUNCTION Employee_ROLE(EMP INT) RETURNS VARCHAR(40) DETERMINISTIC BE
 17 17:17:12 SELECT EXP_Employee_ROLE(EMP) FROM data_science_team LIMIT 0, 1000 ;
 18 17:18:11 CREATE INDEX idx_first_name ON emp_record_table(FIRST_NAME(20))
 19 17:18:11 SELECT * FROM emp_record_table WHERE FIRST_NAME='Eric' LIMIT 0, 1000
 20 17:18:31 EXPLAIN SELECT * FROM emp_record_table WHERE FIRST_NAME='Eric'

Object Info Session

SQLAdditions
 Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Duration / Fetch
 0.015 sec / 0.000 sec
 0.015 sec
 0.000 sec / 0.000 sec
 0.063 sec
 0.000 sec / 0.000 sec
 0.000 sec

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).

16A.

update emp_record_table set salary=(select salary +(select salary*.05*EMP_RATING))

SELECT *FROM emp_record_table;

Query 1:
 1 SELECT *FROM emp_record_table;
 2 update emp_record_table set salary=(select salary +(select salary*.05*EMP_RATING));
 3
 4
 5
 6
 7

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

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	E103	P105
E005	Eric	Huffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA	9000	2	E428	P204
E010	William	Butler	M	LEAD DATA SCIENTIST	AUTOMOTIVE	12	FRANCE	EUROPE	8500	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	P103
E103	Emily	Grove	F	MANAGER	FINANCE	14	CANADA	NORTH AMERICA	10500	4	E001	P103
E204	Karen	Novak	F	SENIOR DATA SCIENTIST	AUTOMOTIVE	8	GERMANY	EUROPE	7500	5	E428	P204
E245	Nan	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	Huffman	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	P103
E478	David	Smith	M	ASSOCIATE DATA SCIENTIST	RETAIL	3	COLOMBIA	SOUTH AMERICA	4000	4	E583	P109
E505	Chad	Wilson	M	ASSOCIATE DATA SCIENTIST	HEALTHCARE	5	CANADA	NORTH AMERICA	5000	2	E083	P103
E532	Clare	Brennan	F	ASSOCIATE DATA SCIENTIST	AUTOMOTIVE	3	GERMANY	EUROPE	4300	1	E428	P204

Output
 Action Output
 # Time Action Message
 19 17:18:11 SELECT * FROM emp_record_table WHERE FIRST_NAME='Eric' LIMIT 0, 1000 1 row(s) returned
 20 17:18:31 EXPLAIN SELECT * FROM emp_record_table WHERE FIRST_NAME='Eric' OK
 21 17:18:31 EXPLAIN FORMAT=JSON SELECT * FROM emp_record_table WHERE FIRST_NAME='Eric' OK
 22 17:20:12 update emp_record_table set salary=(select salary +(select salary*.05*EMP_RATING)) SELECT * FROM emp_record_table Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'SELECT * FROM emp_record_table' at line 1
 23 17:21:29 update emp_record_table set salary=(select salary +(select salary*.05*EMP_RATING)) Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that uses 'EMP_ID'
 24 17:21:37 SELECT * FROM emp_record_table LIMIT 0, 1000 13 row(s) returned

Object Info Session

SQLAdditions
 Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Duration / Fetch
 0.000 sec / 0.000 sec
 0.000 sec
 0.000 sec
 0.000 sec
 0.000 sec
 0.000 sec

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

17A.

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, SALARY, COUNTRY, CONTINENT,  
AVG(salary) OVER (PARTITION BY COUNTRY) AVG_salary_IN_COUNTRY,  
AVG(salary) OVER (PARTITION BY CONTINENT) AVG_salary_IN_CONTINENT,  
COUNT(*) OVER (PARTITION BY COUNTRY) COUNT_IN_COUNTRY,  
COUNT(*) OVER (PARTITION BY CONTINENT) COUNT_IN_CONTINENT  
FROM emp_record_table;
```

The screenshot shows the MySQL Workbench interface. The 'Query Editor' window contains the following SQL query:

```
1 SELECT EMP_ID, FIRST_NAME, LAST_NAME, SALARY, COUNTRY, CONTINENT,  
2 AVG(salary) OVER (PARTITION BY COUNTRY) AVG_salary_IN_COUNTRY,  
3 AVG(salary) OVER (PARTITION BY CONTINENT) AVG_salary_IN_CONTINENT,  
4 COUNT(*) OVER (PARTITION BY COUNTRY) COUNT_IN_COUNTRY,  
5 COUNT(*) OVER (PARTITION BY CONTINENT) COUNT_IN_CONTINENT  
6 FROM emp_record_table;  
7
```

The 'Result Grid' window displays the results of the query. The columns are: EMP_ID, FIRST_NAME, LAST_NAME, SALARY, COUNTRY, CONTINENT, AVG_salary_IN_COUNTRY, AVG_salary_IN_CONTINENT, COUNT_IN_COUNTRY, and COUNT_IN_CONTINENT. The results show data for various employees, including Nan Zhen, Roy Collins, Tracy Norris, Katrina Allen, William Butler, Karene Nemak, Pete Allen, Claire Brennan, Dianna Wilson, Emily Grove, Chad Wilson, Arthur Black, Eric Huffman, Dorothy Wilson, and Patrick Voltz.

The 'Output' window shows the execution of the query. The first two rows show the execution of the query and the results. The third row shows an error message: 'Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL version. You are using safe update mode and you tried to update a table without a WHERE that uses...'. The fourth row shows the execution of the query and the results.

--- END ---