

# Gradable Project - ScienceQtech Employee Performance Mapping

## Course 4 - SQL Training

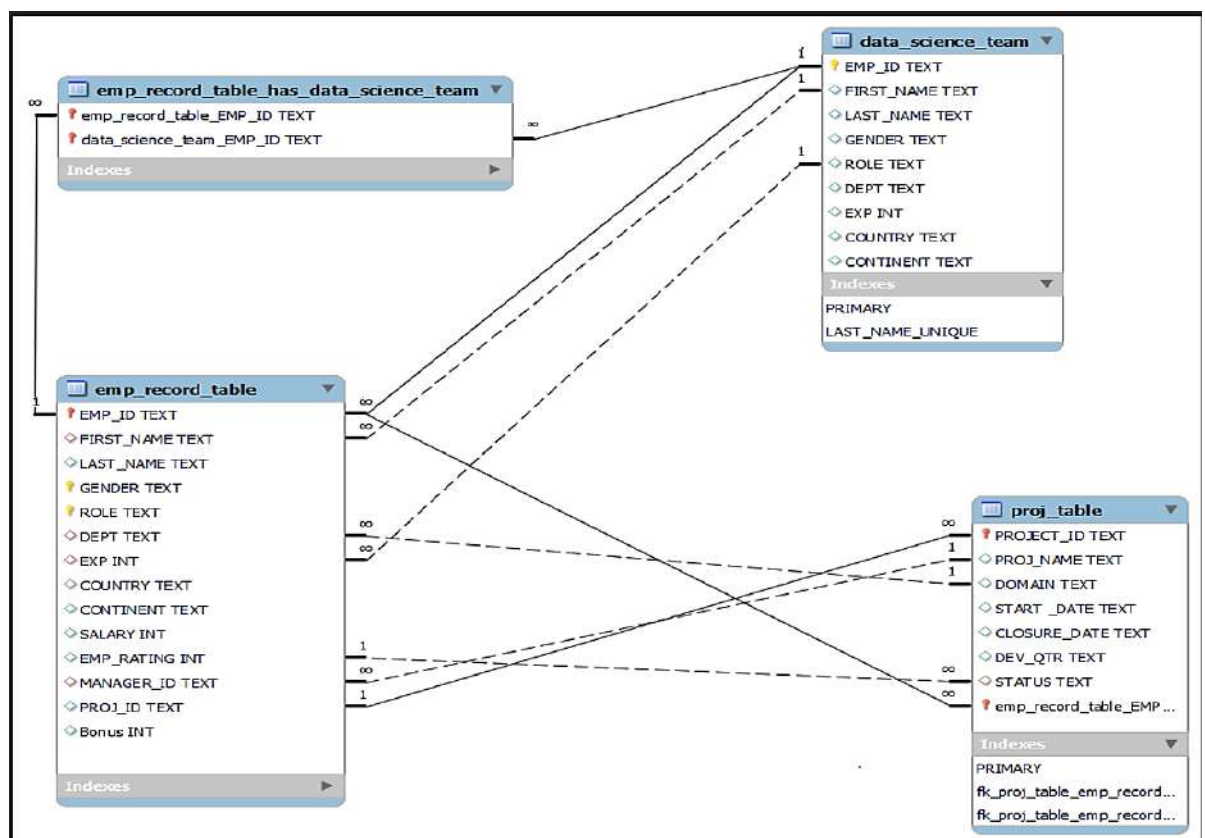
### TASK

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.

The screenshot shows a database management tool interface. On the left, the 'SCHEMAS' pane displays the 'employee' database structure, including tables like 'data\_science\_team', 'emp\_record\_table', and 'proj\_table'. The main pane shows a SQL query: `SELECT * FROM employee.data_science_team;` and its result grid.

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT
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
E057	Dorothy	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	9	USA	NORTH AMERICA
E204	Karen	Nowak	F	SENIOR DATA SCIENTIST	AUTOMOTIVE	8	GERMANY	EUROPE
E245	Nian	Zhen	M	SENIOR DATA SCIENTIST	RETAIL	6	CHINA	ASIA
E260	Roy	Collins	M	SENIOR DATA SCIENTIST	RETAIL	7	INDIA	ASIA
E403	Steve	Hoffman	M	ASSOCIATE DATA SCIENTIST	FINANCE	4	USA	NORTH AMERICA
E478	David	Smith	M	ASSOCIATE DATA SCIENTIST	RETAIL	3	COLOMBIA	SOUTH AMERICA
E505	Chad	Wilson	M	ASSOCIATE DATA SCIENTIST	HEALTHCARE	5	CANADA	NORTH AMERICA
E532	Claire	Brennan	F	ASSOCIATE DATA SCIENTIST	AUTOMOTIVE	3	GERMANY	EUROPE
E620	Katrina	Allen	F	JUNIOR DATA SCIENTIST	RETAIL	2	INDIA	ASIA
E640	Jenifer	Jhones	F	JUNIOR DATA SCIENTIST	RETAIL	1	COLOMBIA	SOUTH AMERICA

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.

The screenshot shows the SQL Developer interface. The Navigator pane on the left displays the 'employee' schema with tables 'data\_science\_team', 'emp\_record\_table', and 'proj\_table'. The main query window contains the following SQL code:

```
1 select emp_id, first_name, last_name, gender, dept
2 from emp_record_table
3
```

The Result Grid displays the following data:

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	Karene	Nowak	F	AUTOMOTIVE
E245	Nian	Zhen	M	RETAIL
E260	Roy	Collins	M	RETAIL
E403	Steve	Hoffman	M	FINANCE
E428	Pete	Allen	M	AUTOMOTIVE
E478	David	Smith	M	RETAIL
E505	Chad	Wilson	M	HEALTHCARE
E532	Claire	Brennan	F	AUTOMOTIVE

The bottom pane shows the table structure for 'emp\_record\_table':

Column	Text
EMP_ID	text
FIRST_NAME	text
LAST_NAME	text
GENDER	text

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

The screenshot shows the SQL Developer interface. The Navigator pane on the left displays the 'employee' schema with tables 'data\_science\_team', 'emp\_record\_table', and 'proj\_table'. The main query window contains the following SQL code:

```
1 select * from emp_record_table
2 where emp_rating < 2;
```

The Result Grid displays the following data:

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT	SALARY	EMP_RATING	MANAGER_ID	PROJ_ID
E057	Dorothy	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	9	USA	NORTH AMERICA	7700	1	E083	P302
E532	Claire	Brennan	F	ASSOCIATE DATA SCIENTIST	AUTOMOTIVE	3	GERMANY	EUROPE	4300	1	E428	P204
E620	Katrina	Allen	F	JUNIOR DATA SCIENTIST	RETAIL	2	INDIA	ASIA	3000	1	E612	P406

The bottom pane shows the table structure for 'emp\_record\_table':

Column	Text
EMP_ID	text
FIRST_NAME	text
LAST_NAME	text
GENDER	text

- greater than four

The screenshot shows the SQL Developer interface. The Navigator pane on the left displays the schema structure, including the 'employee' table. The central SQL Editor contains the following query:

```
1 select * from emp_record_table
2 where emp_rating > 4;
```

The 'Result Grid' at the bottom displays the query results for the 'emp\_record\_table'.

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
E052	Diana	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	6	CANADA	NORTH AMERICA	5500	5	E083	P103
E083	Patrick	Voltz	M	MANAGER	HEALTHCARE	15	USA	NORTH AMERICA	9500	5	E001	NULL
E204	Karen	Nowak	F	SENIOR DATA SCIENTIST	AUTOMOTIVE	8	GERMANY	EUROPE	7500	5	E428	P204

- between two and four

The screenshot shows the SQL Developer interface. The Navigator pane on the left displays the schema structure, including the 'employee' table. The central SQL Editor contains the following query:

```
1 select * from emp_record_table
2 where emp_rating between 2 and 4;
```

The 'Result Grid' at the bottom displays the query results for the 'emp\_record\_table'.

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
E010	William	Butler	M	LEAD DATA SCIENTIST	AUTOMOTIVE	12	FRANCE	EUROPE	9000	2	E428	P204
E103	Emily	Grove	F	MANAGER	FINANCE	14	CANADA	NORTH AMERICA	10500	4	E001	NULL
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
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
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
E640	Jennifer	Jones	F	JUNIOR DATA SCIENTIST	RETAIL	1	COLOMBIA	SOUTH AMERICA	2800	4	E612	P406



- 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.

The screenshot shows the SQL Developer interface with a query window open. The query is as follows:

```

1 select *, concat(first_name, ' ', last_name) as Name
2 from emp_record_table
3 where dept = 'finance'

```

The result grid displays the following data:

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT	SALARY	EMP_RATING	MANAGER_ID	PROJ_ID	Name
E005	Eric	Hoffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA	8500	3	E103	P105	Eric Hoffman
E103	Emily	Grove	F	MANAGER	FINANCE	14	CANADA	NORTH AMERICA	10500	4	E001	NALE	Emily Grove
E403	Steve	Hoffman	M	ASSOCIATE DATA SCIENTIST	FINANCE	4	USA	NORTH AMERICA	5000	3	E103	P105	Steve Hoffman

The left sidebar shows the schema structure with the 'emp\_record\_table' selected. The bottom left pane shows the table and column details for 'emp\_record\_table'.

- 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 SQL Developer interface with a query window open. The query is as follows:

```

1 select manager_id, concat(first_name, ' ', last_name), role, count(emp_id) over (partition by manager_id ) cnt from emp_record_table
2 order by cnt

```

The result grid displays the following data:

manager_id	concat(first_name, ' ', last_name)	role	cnt
NALE	Arthur Black	PRESIDENT	1
E103	Eric Hoffman	LEAD DATA SCIENTIST	2
E103	Steve Hoffman	ASSOCIATE DATA SCIENTIST	2
E612	Kabrina Allen	JUNIOR DATA SCIENTIST	2
E612	Jenifer Jhones	JUNIOR DATA SCIENTIST	2
E083	Dianna Wilson	SENIOR DATA SCIENTIST	3
E083	Dorothy Wilson	SENIOR DATA SCIENTIST	3
E083	Chad Wilson	ASSOCIATE DATA SCIENTIST	3
E428	William Butler	LEAD DATA SCIENTIST	3
E428	Karene Nowak	SENIOR DATA SCIENTIST	3
E428	Claire Brennan	ASSOCIATE DATA SCIENTIST	3
E583	Nian Zhen	SENIOR DATA SCIENTIST	3
E583	Roy Collins	SENIOR DATA SCIENTIST	3
E583	David Smith	ASSOCIATE DATA SCIENTIST	3
E001	Patrick Voltz	MANAGER	5
E001	Emily Grove	MANAGER	5
E001	Pete Allen	MANAGER	5
E001	Janet Hale	MANAGER	5
E001	Tracy Norris	MANAGER	5

The left sidebar shows the schema structure with the 'emp\_record\_table' selected. The bottom left pane shows the table and column details for 'emp\_record\_table'.

- 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 SQL Developer interface with a query window titled 'emp\_record\_table'. The query is as follows:

```

1 SELECT * FROM emp_record_table WHERE DEPT="FINANCE"
2 union
3 SELECT * FROM emp_record_table WHERE DEPT="HEALTHCARE"
4

```

The result grid 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	P105
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.

The screenshot shows the SQL Developer interface with a query window titled 'emp\_record\_table'. The query is as follows:

```

1 SELECT emp_id, first_name, last_name, role, dept, emp_rating, max(emp_rating) as Max_emp_rating
2 FROM emp_record_table
3 group by dept;
4

```

The result grid displays 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	4
E010	William	Butler	LEAD DATA SCIENTIST	AUTOMOTIVE	2	5
E052	Dianna	Wilson	SENIOR DATA SCIENTIST	HEALTHCARE	5	5
E245	Nian	Zhen	SENIOR DATA SCIENTIST	RETAIL	2	4

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 SQL Developer interface with a query window titled 'SQL File 5\*' containing the following SQL query:

```
1 • select emp_id, first_name, role, min(salary), max(salary)
2   from emp_record_table
3  group by role
```

The 'Result Grid' tab is active, displaying the results of the query. The table has 6 columns: emp\_id, first\_name, role, min(salary), and max(salary). The data is grouped by role.

emp_id	first_name	role	min(salary)	max(salary)
E001	Arthur	PRESIDENT	16500	16500
E005	Eric	LEAD DATA SCIENTIST	8500	9000
E052	Dianna	SENIOR DATA SCIENTIST	5500	7700
E083	Patrick	MANAGER	8500	11000
E403	Steve	ASSOCIATE DATA SCIENTIST	4000	5000
E620	Katrina	JUNIOR DATA SCIENTIST	2800	3000

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 SQL Developer interface with a query window titled 'SQL File 5\*' containing the following SQL query:

```
1 • SELECT *, rank() over( order by EXP) as 'rank' FROM
2   emp_record_table;
```

The 'Result Grid' tab is active, displaying the results of the query. The table has 15 columns: EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, ROLE, DEPT, EXP, COUNTRY, CONTINENT, SALARY, EMP\_RATING, MANAGER\_ID, PROJ\_ID, and rank. The data is sorted by experience (EXP) in ascending order.

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	4	E612	P406	1
E620	Katrina	Allen	F	JUNIOR DATA SCIENTIST	RETAIL	2	INDIA	ASIA	3000	1	E612	P406	2
E478	David	Smith	M	ASSOCIATE DATA SCIENTIST	RETAIL	3	COLOMBIA	SOUTH AMERICA	4000	4	E583	P109	3
E532	Claire	Brennan	F	ASSOCIATE DATA SCIENTIST	AUTOMOTIVE	3	GERMANY	EUROPE	4300	1	E428	P204	3
E403	Steve	Hoffman	M	ASSOCIATE DATA SCIENTIST	FINANCE	4	USA	NORTH AMERICA	5000	3	E103	P105	4
E505	Chad	Wilson	M	ASSOCIATE DATA SCIENTIST	HEALTHCARE	5	CANADA	NORTH AMERICA	5000	2	E083	P103	5
E052	Dianna	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	6	CANADA	NORTH AMERICA	5500	5	E083	P103	6
E245	Nian	Zhen	M	SENIOR DATA SCIENTIST	RETAIL	6	CHINA	ASIA	6500	2	E583	P109	6
E260	Roy	Collins	M	SENIOR DATA SCIENTIST	RETAIL	7	INDIA	ASIA	7000	3	E583	NA	7
E204	Karene	Nowak	F	SENIOR DATA SCIENTIST	AUTOMOTIVE	8	GERMANY	EUROPE	7500	5	E428	P204	8
E057	Dorothy	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	9	USA	NORTH AMERICA	7700	1	E083	P302	9
E005	Eric	Hoffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA	8500	3	E103	P105	10
E010	William	Butler	M	LEAD DATA SCIENTIST	AUTOMOTIVE	12	FRANCE	EUROPE	9000	2	E428	P204	11
E612	Tracy	Norris	F	MANAGER	RETAIL	13	INDIA	ASIA	8500	4	E001	NA	12
E103	Emily	Grove	F	MANAGER	FINANCE	14	CANADA	NORTH AMERICA	10500	4	E001	NA	13
E428	Pete	Allen	M	MANAGER	AUTOMOTIVE	14	GERMANY	EUROPE	11000	4	E001	NA	13
E583	Janet	Hale	F	MANAGER	RETAIL	14	COLOMBIA	SOUTH AMERICA	10000	2	E001	NA	13
E083	Patrick	Voltz	M	MANAGER	HEALTHCARE	15	USA	NORTH AMERICA	9500	5	E001	NA	14
E001	Arthur	Black	M	PRESIDENT	ALL	20	USA	NORTH AMERICA	16500	5		NA	15



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 SQL Developer interface with a query window titled 'emp\_record\_table'. The query is as follows:

```
1 • SELECT * FROM employee.emp_record_table WHERE SALARY>6000
2 order by salary Desc
3
```

The result grid displays the following 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
E428	Pete	Allen	M	MANAGER	AUTOMOTIVE	14	GERMANY	EUROPE	11000	4	E001	NULL
E103	Emily	Grove	F	MANAGER	FINANCE	14	CANADA	NORTH AMERICA	10500	4	E001	NULL
E583	Janet	Hale	F	MANAGER	RETAIL	14	COLOMBIA	SOUTH AMERICA	10000	2	E001	NULL
E083	Patrick	Voltz	M	MANAGER	HEALTHCARE	15	USA	NORTH AMERICA	9500	5	E001	NULL
E010	William	Butler	M	LEAD DATA SCIENTIST	AUTOMOTIVE	12	FRANCE	EUROPE	9000	2	E428	P204
E005	Eric	Hoffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA	8500	3	E103	P105
E612	Tracy	Norris	F	MANAGER	RETAIL	13	INDIA	ASIA	8500	4	E001	NULL
E057	Dorothy	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	9	USA	NORTH AMERICA	7700	1	E083	P302
E204	Karen	Nowak	F	SENIOR DATA SCIENTIST	AUTOMOTIVE	8	GERMANY	EUROPE	7500	5	E428	P204
E260	Roy	Collins	M	SENIOR DATA SCIENTIST	RETAIL	7	INDIA	ASIA	7000	3	E583	NA
E245	Nan	Zhen	M	SENIOR DATA SCIENTIST	RETAIL	6	CHINA	ASIA	6500	2	E583	P109

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 the SQL Developer interface with a query window titled 'emp\_record\_table'. The query is as follows:

```
1 • SELECT emp_id, concat(first_name, ' ', last_name), exp FROM emp_record_table WHERE EXP>10
2 order by exp;
```

The result grid displays the following data:

emp_id	concat(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
E583	Janet Hale	14
E083	Patrick Voltz	15
E001	Arthur Black	20

The screenshot shows the SQL Developer interface with a query window titled 'emp\_record\_table'. The query is as follows:

```
1 • SELECT * FROM emp_record_table WHERE EXP>10
2 order by exp;
```

The result grid 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
E010	William	Butler	M	LEAD DATA SCIENTIST	AUTOMOTIVE	12	FRANCE	EUROPE	9000	2	E428	P204
E612	Tracy	Norris	F	MANAGER	RETAIL	13	INDIA	ASIA	8500	4	E001	NULL
E103	Emily	Grove	F	MANAGER	FINANCE	14	CANADA	NORTH AMERICA	10500	4	E001	NULL
E428	Pete	Allen	M	MANAGER	AUTOMOTIVE	14	GERMANY	EUROPE	11000	4	E001	NULL
E583	Janet	Hale	F	MANAGER	RETAIL	14	COLOMBIA	SOUTH AMERICA	10000	2	E001	NULL
E083	Patrick	Voltz	M	MANAGER	HEALTHCARE	15	USA	NORTH AMERICA	9500	5	E001	NULL
E001	Arthur	Black	M	PRESIDENT	ALL	20	USA	NORTH AMERICA	16500	5	NULL	NULL

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 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 'emp\_record\_table' is selected, and its columns are listed: EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, ROLE, DEPT, EXP, COUNTRY, and CONTINENT. The main editor shows the following SQL code:

```
1 Delimiter //
```

```
2 create procedure emp_procedure()
```

```
3 begin
```

```
4 select emp_id, first_name, exp from emp_record_table
```

```
5 where
```

```
6 exp > 3
```

```
7 order by exp, first_name, emp_id;
```

```
8 end//
```

The 'Result Grid' at the bottom shows the results of the query, with columns 'emp\_id', 'first\_name', and 'exp'. The results are as follows:

emp_id	first_name	exp
E001	Arthur	20
E005	Eric	11
E010	William	12
E052	Dianna	6
E057	Dorothy	9
E083	Patrick	15
E103	Emily	14
E204	Karene	8
E245	Nian	6
E260	Roy	7
E403	Steve	4
E428	Pete	14
E505	Chad	5
E583	Janet	14

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'.



File Edit View Query Database Server Tools Scripting Help

Navigator: SQL File 5\* x proj\_table emp\_record\_table

SCHEMAS

Filter objects

employee

Tables

data\_science\_team

emp\_record\_table

Columns

Indexes

Foreign Keys

Triggers

proj\_table

Columns

Indexes

Foreign Keys

Triggers

Views

Stored Procedures

Functions

Administration Schemas

Information

Table: emp\_record\_table

Columns:

EMP\_ID text

FIRST\_NAME text

LAST\_NAME text

GENDER text

ROLE text

DEPT text

EXP int

COUNTRY text

CONTINENT text

SALARY int

EMP\_RATING int

MANAGER\_ID text

PROJ\_ID text

```

1 select Case
2   when role = 'Junior Data Scientist' then exp <= 2
3   when role = 'Associate Data Scientist' then exp between 2 and 5
4   when role = 'Senior Data Scientist' then exp between 5 and 10
5   when role = 'Lead Data Scientist' then exp between 10 and 12
6   when role = 'Manager' then exp between 12 and 16
7   else 'Unfit'
8 End as Exp, Role, Emp_id from emp_record_table

```

Result Grid

Filter Rows:

Export:

Wrap Cell Content: F

Exp	Role	Emp_id
Unfit	PRESIDENT	E001
1	LEAD DATA SCIENTIST	E005
1	LEAD DATA SCIENTIST	E010
1	SENIOR DATA SCIENTIST	E052
1	SENIOR DATA SCIENTIST	E057
1	MANAGER	E083
1	MANAGER	E103
1	SENIOR DATA SCIENTIST	E204
1	SENIOR DATA SCIENTIST	E245
1	SENIOR DATA SCIENTIST	E260
1	ASSOCIATE DATA SCIENTIST	E403
1	MANAGER	E428
1	ASSOCIATE DATA SCIENTIST	E478
1	ASSOCIATE DATA SCIENTIST	E505
1	ASSOCIATE DATA SCIENTIST	E532
1	MANAGER	E583
1	MANAGER	E612
1	JUNIOR DATA SCIENTIST	E620
1	JUNIOR DATA SCIENTIST	E640

Result 10 x

Output

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.

File Edit View Query Database Server Tools Scripting Help

Navigator: SQL File 5\* x proj\_table emp\_record\_table

SCHEMAS

Filter objects

employee

Tables

data\_science\_team

emp\_record\_table

Columns

Indexes

Foreign Keys

Triggers

proj\_table

Columns

Indexes

Foreign Keys

Triggers

Views

Stored Procedures

Functions

Administration Schemas

Information

Table: proj\_table

Columns:

PROJECT\_ID text

PROJ\_NAME text

DOMAIN text

START\_DATE text

CLOSURE\_DATE text

```

1 select * from Data_Science_team where first_name = 'Eric';

```

Visual Explain

Display Info: Read + Eval cost

Overview: View Source

Query cost: 2.30

query\_block#1

2.3 13 rows

Full Table Scan

Data\_Science\_team

Data\_Science\_team 3 x

Apply Revert

File Edit View Query Database Server Tools Scripting Help

SQL File 5\* x proj\_table emp\_record\_table

Limit to 1000 rows

```

1 • create index idx_first_name
2   on emp_record_table( last_name(30));
3 • select emp_id, last_name, role, emp_rating from emp_record_table
4   where first_name = 'Eric'

```

Visual Explain | Display Info: Read + Eval cost | Overview | View Source

Query cost: 2.15

query\_block#1

2.15 | 19 rows

Full Table Scan  
emp\_record\_table

emp\_record\_table 15 x | Read Only | Context Help | Snippets

Output

#	Time	Action	Message	Duration /
27	10:56:36	EXPLAIN Select emp_id, last_name, role, emp_rating from emp...	OK	0.000 sec
28	10:56:36	EXPLAIN FORMAT=JSON Select emp_id, last_name, role, em...	OK	0.000 sec

Automatic content disabled. Use the manually get the current caret position to toggle automatic content.

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).

File Edit View Query Database Server Tools Scripting Help

SQL File 5\* x emp\_record\_table

Limit to 1000 rows

```

1 • update emp_record_table
2   set bonus = (0.05*Salary)*emp_rating;
3 • select emp_id, first_name, last_name, role, bonus from emp_record_table;
4

```

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

emp_id	first_name	last_name	role	bonus
E001	Arthur	Black	PRESIDENT	4125
E005	Eric	Hoffman	LEAD DATA SCIENTIST	1275
E010	William	Butler	LEAD DATA SCIENTIST	900
E052	Dianna	Wilson	SENIOR DATA SCIENTIST	1375
E057	Dorothy	Wilson	SENIOR DATA SCIENTIST	385
E083	Patrick	Voltz	MANAGER	2375
E103	Emily	Grove	MANAGER	2100
E204	Karen	Nowak	SENIOR DATA SCIENTIST	1875
E245	Nian	Zhen	SENIOR DATA SCIENTIST	650
E260	Roy	Collins	SENIOR DATA SCIENTIST	1050
E403	Steve	Hoffman	ASSOCIATE DATA SCIENTIST	750
E428	Pete	Allen	MANAGER	2200
E478	David	Smith	ASSOCIATE DATA SCIENTIST	800
E505	Chad	Wilson	ASSOCIATE DATA SCIENTIST	500
E532	Claire	Brennan	ASSOCIATE DATA SCIENTIST	215
E583	Janet	Hale	MANAGER	1000
E612	Tracy	Norris	MANAGER	1700
E620	Katrina	Allen	JUNIOR DATA SCIENTIST	150
E640	Jennifer	Jones	JUNIOR DATA SCIENTIST	560

emp\_record\_table 16 x | Read Only

Output

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 a database management tool interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. The left sidebar contains a 'SCHEMAS' panel with a search filter and a tree view showing the 'employee' schema, including tables like 'data\_science\_team', 'emp\_record\_table', and 'proj\_table'. The main window displays an SQL query in a file named 'emp\_record\_table':

```
1 SELECT country,continent, AVG(Salary) FROM emp_record_table
2 GROUP BY Country, Continent;
```

Below the query editor, the 'Result Grid' shows the output of the query. The grid has three columns: 'country', 'continent', and 'AVG(Salary)'. The data is as follows:

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

The bottom left of the interface shows the 'Information' panel for the 'emp\_record\_table' table, listing columns: EMP\_ID (text), FIRST\_NAME (text), and LAST\_NAME (text).