

-- 17) Average Salary by Geography

```
SELECT CONTINENT, COUNTRY, AVG(SALARY) AS AVG_SALARY
FROM emp_record_table
GROUP BY CONTINENT, COUNTRY
ORDER BY CONTINENT, COUNTRY;
```

d |   Filter Rows: | Export:  | Wrap Cell Content: 

CONTINENT	COUNTRY	AVG_SALARY
ASIA	CHINA	6500.0000
	INDIA	6166.6667
EUROPE	FRANCE	9000.0000
EUROPE	GERMANY	7600.0000
North America	CANADA	7000.0000
North America	USA	9440.0000
South America	COLOMBIA	5600.0000

```
D, FIRST_NAME, LAST_NAME, GENDER, DEPT
```

```
ord_table;
```


Rows: | Edit:    | Export/Import:   | Wrap Cell Content: 

LAST_NAME	GENDER	DEPT
Black	M	ALL
Hoffman	M	FINANCE
Butler	M	AUTOMOTIVE
Wilson	F	HEALTHCARE
Wilson	F	HEALTHCARE
Voltz	M	HEALTHCARE
Grove	F	FINANCE
Nowak	F	AUTOMOTIVE
Zhen	M	RETAIL
Collins	M	RETAIL
Hoffman	M	FINANCE
Allen	M	AUTOMOTIVE
Smith	M	RETAIL
Wilson	M	HEALTHCARE
Brennan	F	AUTOMOTIVE

```
-- 5)
SELECT CONCAT(FIRST_NAME, ' ', LAST_NAME) AS NAME
FROM emp_record_table
WHERE DEPT = 'Finance';
```

```
-- 6)
SELECT e.EMP_ID, e.FIRST_NAME, e.LAST_NAME, e.MANAGER_ID,
COUNT(r.EMP_ID) AS REPORTER_COUNT
FROM emp_record_table e
LEFT JOIN emp_record_table r ON e.EMP_ID = r.MANAGER_ID
WHERE e.MANAGER_ID IS NOT NULL
GROUP BY e.EMP_ID, e.FIRST_NAME, e.LAST_NAME, e.MANAGER_ID;
```

```
-- 7)
```


 Filter Rows:
 | Export: 
 | Wrap Cell Content: 

_ID	FIRST_NAME	LAST_NAME	MANAGER_ID	REPORTER_COUNT
	Eric	Hoffman	E103	0
	William	Butler	E428	0
	Dianna	Wilson	E083	0
	Dorothy	Wilson	E083	0
	Patrick	Voltz	E001	3
	Emily	Grove	E001	2
	Karene	Nowak	E428	0
	Nian	Zhen	E583	0
	Roy	Collins	E583	0
	Steve	Hoffman	E103	0

Result 6 ×

-- 4)

-- less than 2

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

-- Greater than 4

- ```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING
FROM emp_record_table
where EMP_RATING > 4;
```

-- Between 2 and 4

- ```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING
FROM emp_record_table
where EMP_RATING BETWEEN 2 AND 4;
```

Grid |   Filter Rows:  | Edit:    | Export/Import:   | Wrap Cell Content: 

| EMP_ID | FIRST_NAME | LAST_NAME | GENDER | DEPT       | EMP_RATING |
|--------|------------|-----------|--------|------------|------------|
| 105    | Eric       | Hoffman   | M      | FINANCE    | 3          |
| 110    | William    | Butler    | M      | AUTOMOTIVE | 2          |
| 103    | Emily      | Grove     | F      | FINANCE    | 4          |
| 145    | Nian       | Zhen      | M      | RETAIL     | 2          |
| 160    | Roy        | Collins   | M      | RETAIL     | 3          |
| 103    | Steve      | Hoffman   | M      | FINANCE    | 3          |
| 128    | Pete       | Allen     | M      | AUTOMOTIVE | 4          |
| 178    | David      | Smith     | M      | RETAIL     | 4          |
| 105    | Chad       | Wilson    | M      | HEALTHCARE | 2          |
| 183    | Janet      | Hale      | F      | RETAIL     | 2          |

emp\_record\_table 2    emp\_record\_table 3    emp\_record\_table 4 ×



```

40 -- 7)
41 • SELECT EMP_ID, FIRST_NAME, LAST_NAME, DEPT
42 FROM emp_record_table
43 WHERE DEPT = 'Healthcare'
44 UNION
45 SELECT EMP_ID, FIRST_NAME, LAST_NAME, DEPT
46 FROM emp_record_table
47 WHERE dept = 'Finance' ;
48
49 -- 8) |
50 • SELECT DEPT, EMP_ID, FIRST_NAME, LAST_NAME, ROLE, EMP_RATING,
51 MAX(EMP_RATING) OVER (PARTITION BY DEPT) AS MAX_RATING
52 FROM emp_record_table
53 GROUP BY DEPT, EMP_ID,
54 FIRST_NAME, LAST_NAME, ROLE, EMP_RATING;
55

```

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

| DEPT       | EMP_ID | FIRST_NAME | LAST_NAME | ROLE                     | EMP_RATING | MAX_RATING |
|------------|--------|------------|-----------|--------------------------|------------|------------|
| ALL        | E001   | Arthur     | Black     | PRESIDENT                | 5          | 5          |
| AUTOMOTIVE | E010   | William    | Butler    | LEAD DATA SCIENTIST      | 2          | 5          |
| AUTOMOTIVE | E204   | Karene     | Nowak     | SENIOR DATA SCIENTIST    | 5          | 5          |
| AUTOMOTIVE | E428   | Pete       | Allen     | MANAGER                  | 4          | 5          |
| AUTOMOTIVE | E532   | Claire     | Brennan   | ASSOCIATE DATA SCIENTIST | 1          | 5          |
| FINANCE    | E005   | Eric       | Hoffman   | LEAD DATA SCIENTIST      | 3          | 4          |
| FINANCE    | E103   | Emily      | Grove     | MANAGER                  | 4          | 4          |
| FINANCE    | E403   | Steve      | Hoffman   | ASSOCIATE DATA SCIENTIST | 3          | 4          |
| HEALTHCARE | E052   | Dianna     | Wilson    | SENIOR DATA SCIENTIST    | 5          | 5          |
| HEALTHCARE | E057   | Dorothy    | Wilson    | SENIOR DATA SCIENTIST    | 1          | 5          |
| HEALTHCARE | E083   | Patrick    | Voltz     | MANAGER                  | 5          | 5          |

result 7   Result 8 ×

```

56 -- 9)
57 • SELECT ROLE,
58 MIN(SALARY) AS MIN_SALARY,
59 MAX(SALARY) AS MAX_SALARY
60 FROM emp_record_table
61 GROUP BY ROLE;
62
63 -- 10) Using Rank() Function
64 • SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP,
65 RANK() OVER (ORDER BY EXP DESC) AS EXPERIENCE_RANK
66 FROM emp_record_table;
67

```

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

| EMP_ID | FIRST_NAME | LAST_NAME | EXP | EXPERIENCE_RANK |
|--------|------------|-----------|-----|-----------------|
| E001   | Arthur     | Black     | 20  | 1               |
| E083   | Patrick    | Voltz     | 15  | 2               |
| E103   | Emily      | Grove     | 14  | 3               |
| E428   | Pete       | Allen     | 14  | 3               |
| E583   | Janet      | Hale      | 14  | 3               |
| E612   | Tracy      | Norris    | 13  | 6               |
| E010   | William    | Butler    | 12  | 7               |
| E005   | Eric       | Hoffman   | 11  | 8               |
| E057   | Dorothy    | Wilson    | 9   | 9               |
| E204   | Karene     | Nowak     | 8   | 10              |
| E260   | Roy        | Collins   | 7   | 11              |
| E052   | Dianna     | Wilson    | 6   | 12              |
| E245   | Nian       | Zhen      | 6   | 12              |
| E505   | Chad       | Wilson    | 5   | 14              |
| E403   | Steve      | Hoffman   | 4   | 15              |
| E478   | David      | Smith     | 3   | 16              |

Result 9    Result 10 ×

```

68 -- 11) Using View Creation for virtual table
69 • CREATE VIEW high_salary_employees
70 AS
71 SELECT EMP_ID, FIRST_NAME, LAST_NAME, COUNTRY, SALARY
72 FROM emp_record_table
73 WHERE SALARY > 6000;
74
75 • select * from high_salary_employees;
76

```

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

| 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   |
| 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   |

```

17 -- 12) Nasted Query
18 ● SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP
19 FROM emp_record_table
20 WHERE EXP > (SELECT AVG(EXP) + 10 FROM emp_record_table);
21
22 -- 13) Stored Procedure
23 DELIMITER //
24 ● CREATE PROCEDURE
25 GetExperiencedEmployees()
26 BEGIN
27 SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP
28 FROM emp_record_table
29 WHERE EXP > 3;
30 END //
31 DELIMITER ;
32 ● -- Call Procedure
33 CALL GetExperiencedEmployees();

```

Result Grid | Filter Rows:

Export:

Wrap Cell Content:

| 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   |

Result 13 X



```

97 ● CREATE FUNCTION CheckJobProfile(p_emp_id VARCHAR(50))
98 RETURNS VARCHAR(50)
99 DETERMINISTIC
100 READS SQL DATA
101 BEGIN
102 DECLARE v_years INT DEFAULT 0;
103 DECLARE v_profile VARCHAR(50);
104 SELECT IFNULL(EXP, 0) INTO v_years FROM data_science_team WHERE EMP_ID = p_emp_id LIMIT 1;
105 IF v_years <= 2 THEN SET v_profile = 'JUNIOR DATA SCIENTIST';
106 ELSEIF v_years <= 5 THEN SET v_profile = 'ASSOCIATE DATA SCIENTIST';
107 ELSEIF v_years <= 10 THEN SET v_profile = 'SENIOR DATA SCIENTIST';
108 ELSEIF v_years <= 12 THEN SET v_profile = 'LEAD DATA SCIENTIST';
109 ELSEIF v_years <= 16 THEN SET v_profile = 'MANAGER';
110 ELSE SET v_profile = 'NOT DEFINED';
111 END IF;
112 RETURN v_profile;
113 END //
114 DELIMITER ;
115 ● -- Use Function
116 SELECT EMP_ID, FIRST_NAME, LAST_NAME, EXP, CheckJobProfile(EMP_ID) AS JOB_PROFILE
117 FROM data_science_team ORDER BY EXP;

```

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

|   | EMP_ID | FIRST_NAME | LAST_NAME | EXP | JOB_PROFILE              |
|---|--------|------------|-----------|-----|--------------------------|
| ▶ | E640   | Jenifer    | Jhones    | 1   | JUNIOR DATA SCIENTIST    |
|   | E620   | Katrina    | Allen     | 2   | JUNIOR DATA SCIENTIST    |
|   | E478   | David      | Smith     | 3   | ASSOCIATE DATA SCIENTIST |
|   | E532   | Claire     | Brennan   | 3   | ASSOCIATE DATA SCIENTIST |
|   | E403   | Steve      | Hoffman   | 4   | ASSOCIATE DATA SCIENTIST |
|   | E505   | Chad       | Wilson    | 5   | ASSOCIATE DATA SCIENTIST |
|   | E052   | Dianna     | Wilson    | 6   | SENIOR DATA SCIENTIST    |
|   | E245   | Nian       | Zhen      | 6   | SENIOR DATA SCIENTIST    |
|   | E260   | Roy        | Collins   | 7   | SENIOR DATA SCIENTIST    |

-- 15) Index Creation

```
CREATE INDEX idx_first_name ON emp_record_table(FIRST_NAME);
```

-- Check Execution Plan

```
EXPLAIN SELECT * FROM
emp_record_table WHERE FIRST_NAME = 'Eric';
```

-- 16) Bonus Calculation

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, SALARY, EMP_RATING, (SALARY * EMP_RATING * 0.05) AS B
FROM emp_record_table;
```

rid |   Filter Rows:  | Export:  | Wrap Cell Content: 

| _ID | FIRST_NAME | LAST_NAME | SALARY | EMP_RATING | BONUS   |
|-----|------------|-----------|--------|------------|---------|
|     | Arthur     | Black     | 16500  | 5          | 4125.00 |
|     | Eric       | Hoffman   | 8500   | 3          | 1275.00 |
|     | William    | Butler    | 9000   | 2          | 900.00  |
|     | Dianna     | Wilson    | 5500   | 5          | 1375.00 |
|     | Dorothy    | Wilson    | 7700   | 1          | 385.00  |
|     | Patrick    | Voltz     | 9500   | 5          | 2375.00 |
|     | Emily      | Grove     | 10500  | 4          | 2100.00 |
|     | Karene     | Nowak     | 7500   | 5          | 1875.00 |
|     | Nian       | Zhen      | 6500   | 2          | 650.00  |
|     | Roy        | Collins   | 7000   | 3          | 1050.00 |
|     | Steve      | Hoffman   | 5000   | 3          | 750.00  |
|     | Pete       | Allen     | 11000  | 4          | 2200.00 |
|     | David      | Smith     | 4000   | 4          | 800.00  |
|     | Chad       | Wilson    | 5000   | 2          | 500.00  |
|     | Claire     | Brennan   | 4300   | 1          | 215.00  |

Result 19 ×