\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**EXERCISE - 3**

DROP TABLE Employees PURGE;

create table **Employees**(

employee\_id int primary key,

first\_name varchar2(50) ,

last\_name varchar2(50),

hire\_date date ,

salary number(10,2)

);

===================== insert data in employee table ======================

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (1,'John','Doe','01-Jul-2022',60000.00);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (2,'Jane','Smith',null,55000.00);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (3,'Michael','Johnson','15-mar-2023',null);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (4,'Emily','Williams',null,null);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (5,'Daniel','Brown','05-sep-2022',48000.00);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (6,'Laura','Davis','20-apr-2023',52000.00);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (7,'Matthew','Wilson','10-dec-2021',54000.00);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (8,'Emma','Brown',null,null);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (9,'Olivia','Jones','30-may-2023',59000.00);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (10,'Noah','Smith','08-aug-2022',null);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (11,'William','Johnson','25-jun-2021',45000.00);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (12,'Ava','Williams',null,null);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (13,'Sophia','Davis','18-feb-2023',57000.00);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (14,'James','Wilson',null,null);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (15,'Oliver','Miller','12-oct-2022',51000.00);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (16,'Amelia','Brown',null,null);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (17,'Benjamin','Davis','02-dec-2021',53000.00);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (18,'Elijah','Wilson','09-mar-2023',null);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (19,'Lucas','Brown','22-jun-2022',56000.00);

insert into Employees(employee\_id,first\_name,last\_name,hire\_date,salary)

VALUES (20,'Mia','Johnson',null,null);

Recode inserted.

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

2 Jane Smith 55000

3 Michael Johnson 15-03-23

4 Emily Williams

5 Daniel Brown 05-09-22 48000

6 Laura Davis 20-04-23 52000

7 Matthew Wilson 10-12-21 54000

8 Emma Brown

9 Olivia Jones 30-05-23 59000

10 Noah Smith 08-08-22

11 William Johnson 25-06-21 45000

12 Ava Williams

13 Sophia Davis 18-02-23 57000

14 James Wilson

15 Oliver Miller 12-10-22 51000

16 Amelia Brown

17 Benjamin Davis 02-12-21 53000

18 Elijah Wilson 09-03-23

19 Lucas Brown 22-06-22 56000

20 Mia Johnson

*----------------------- Basic SELECT Queries: ------------------------*

1. Display the details of all employees.

SELECT \* from Employees;

2. Show the first and last names of employees.

SQL> SELECT first\_name , last\_name FROM Employees;

FIRST\_NAME LAST\_NAME

\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

John Doe

Jane Smith

Michael Johnson

Emily Williams

Daniel Brown

Laura Davis

Matthew Wilson

Emma Brown

Olivia Jones

Noah Smith

William Johnson

Ava Williams

Sophia Davis

James Wilson

Oliver Miller

Amelia Brown

Benjamin Davis

Elijah Wilson

Lucas Brown

Mia Johnson

3. List the employee IDs and hire dates.

SQL> SELECT employee\_id, hire\_date FROM Employees;

EMPLOYEE\_ID HIRE\_DATE

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

1 01-07-22

2

3 15-03-23

4

5 05-09-22

6 20-04-23

7 10-12-21

8

9 30-05-23

10 08-08-22

11 25-06-21

12

13 18-02-23

14

15 12-10-22

16

17 02-12-21

18 09-03-23

19 22-06-22

20

4. Retrieve the first name, last name, and salary of employees.

SQL> SELECT first\_name ,last\_name, salary FROM Employees;

FIRST\_NAME LAST\_NAME SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

John Doe 60000

Jane Smith 55000

Michael Johnson

Emily Williams

Daniel Brown 48000

Laura Davis 52000

Matthew Wilson 54000

Emma Brown

Olivia Jones 59000

Noah Smith

William Johnson 45000

Ava Williams

Sophia Davis 57000

James Wilson

Oliver Miller 51000

Amelia Brown

Benjamin Davis 53000

Elijah Wilson

Lucas Brown 56000

Mia Johnson

5. Display all columns for employees hired after January 1, 2022.

SQL> SELECT \* FROM Employees WHERE hire\_date > TO\_DATE('01-jan-2022', 'DD-MM-YYYY');

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

3 Michael Johnson 15-03-23

5 Daniel Brown 05-09-22 48000

6 Laura Davis 20-04-23 52000

9 Olivia Jones 30-05-23 59000

10 Noah Smith 08-08-22

13 Sophia Davis 18-02-23 57000

15 Oliver Miller 12-10-22 51000

18 Elijah Wilson 09-03-23

19 Lucas Brown 22-06-22 56000

6. Display the employee IDs and last names of employees.

SQL> SELECT employee\_id, last\_name FROM Employees;

EMPLOYEE\_ID LAST\_NAME

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

1 Doe

2 Smith

3 Johnson

4 Williams

5 Brown

6 Davis

7 Wilson

8 Brown

9 Jones

10 Smith

11 Johnson

12 Williams

13 Davis

14 Wilson

15 Miller

16 Brown

17 Davis

18 Wilson

19 Brown

20 Johnson

7. List the hire dates and salaries of all employees.

SQL> SELECT hire\_date, salary FROM Employees;

HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

01-07-22 60000

55000

15-03-23

05-09-22 48000

20-04-23 52000

10-12-21 54000

30-05-23 59000

08-08-22

25-06-21 45000

18-02-23 57000

12-10-22 51000

02-12-21 53000

09-03-23

22-06-22 56000

8. Display all columns for employees with a salary greater than $50000.00.

SQL> SELECT \* FROM Employees WHERE salary >50000.00;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

2 Jane Smith 55000

6 Laura Davis 20-04-23 52000

7 Matthew Wilson 10-12-21 54000

9 Olivia Jones 30-05-23 59000

13 Sophia Davis 18-02-23 57000

15 Oliver Miller 12-10-22 51000

17 Benjamin Davis 02-12-21 53000

19 Lucas Brown 22-06-22 56000

9. Show the first name, last name, and hire date of employees hired before July 1, 2023.

SQL> SELECT first\_name,last\_name,hire\_date FROM Employees WHERE hire\_date < TO\_DATE('01-jul-2023', 'DD-MM-YYYY');

FIRST\_NAME LAST\_NAME HIRE\_DATE

\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

John Doe 01-07-22

Michael Johnson 15-03-23

Daniel Brown 05-09-22

Laura Davis 20-04-23

Matthew Wilson 10-12-21

Olivia Jones 30-05-23

Noah Smith 08-08-22

William Johnson 25-06-21

Sophia Davis 18-02-23

Oliver Miller 12-10-22

Benjamin Davis 02-12-21

Elijah Wilson 09-03-23

Lucas Brown 22-06-22

*-------------------------- Using WHERE Clause: -------------------------*

1. List employees hired in the year 2023.

SQL> SELECT \* FROM Employees WHERE EXTRACT(YEAR FROM hire\_date) = 2023;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

3 Michael Johnson 15-03-23

6 Laura Davis 20-04-23 52000

9 Olivia Jones 30-05-23 59000

13 Sophia Davis 18-02-23 57000

18 Elijah Wilson 09-03-23

2. Show employees with salaries greater than $55000.00.

SQL> SELECT \* FROM Employees WHERE salary > 55000.00;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

9 Olivia Jones 30-05-23 59000

13 Sophia Davis 18-02-23 57000

19 Lucas Brown 22-06-22 56000

3. Display employees with null hire dates.

SQL> SELECT \* FROM Employees WHERE hire\_date IS NULL;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

2 Jane Smith 55000

4 Emily Williams

8 Emma Brown

12 Ava Williams

14 James Wilson

16 Amelia Brown

20 Mia Johnson

4. List employees with first names starting with 'J'.

SQL> SELECT \* FROM Employees WHERE first\_name LIKE 'J%';

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

2 Jane Smith 55000

14 James Wilson

5. Retrieve employees with a salary between $50000.00 and $60000.00.

SQL> SELECT \* FROM employees WHERE salary >= 50000.00 AND salary <= 60000.00;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

2 Jane Smith 55000

6 Laura Davis 20-04-23 52000

7 Matthew Wilson 10-12-21 54000

9 Olivia Jones 30-05-23 59000

13 Sophia Davis 18-02-23 57000

15 Oliver Miller 12-10-22 51000

17 Benjamin Davis 02-12-21 53000

19 Lucas Brown 22-06-22 56000

6. List employees with last names containing 'son'.

SQL> SELECT \* FROM employees WHERE last\_name LIKE '%son';

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

3 Michael Johnson 15-03-23

7 Matthew Wilson 10-12-21 54000

11 William Johnson 25-06-21 45000

14 James Wilson

18 Elijah Wilson 09-03-23

20 Mia Johnson

7. Show employees with salaries less than $60000.00 and hired in 2022.

SQL> SELECT \* FROM Employees WHERE salary < 60000.00 AND EXTRACT(YEAR FROM hire\_date) = 2022;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

5 Daniel Brown 05-09-22 48000

15 Oliver Miller 12-10-22 51000

19 Lucas Brown 22-06-22 56000

8. Display employees with hire dates after January 1, 2021, and salaries less than $55000.00.

SQL> SELECT \* FROM Employees WHERE hire\_date > TO\_DATE('01-jan-2021', 'DD-MM-YYYY') AND salary < 55000.00;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

5 Daniel Brown 05-09-22 48000

6 Laura Davis 20-04-23 52000

7 Matthew Wilson 10-12-21 54000

11 William Johnson 25-06-21 45000

15 Oliver Miller 12-10-22 51000

17 Benjamin Davis 02-12-21 53000

9. List employees with first names starting with 'M' or last names ending with 'son'.

SQL> SELECT \* FROM Employees WHERE first\_name LIKE 'M%' AND last\_name LIKE '%son';

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

3 Michael Johnson 15-03-23

7 Matthew Wilson 10-12-21 54000

20 Mia Johnson

10. Show employees with salaries greater than $50000.00 and not hired in 2023.

SQL> SELECT \* FROM Employees WHERE salary > 50000.00 AND EXTRACT(YEAR FROM hire\_date) != 2023;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

7 Matthew Wilson 10-12-21 54000

15 Oliver Miller 12-10-22 51000

17 Benjamin Davis 02-12-21 53000

19 Lucas Brown 22-06-22 56000

*--------------------- Using NVL() Function: --------------------*

1. Show the employee IDs and salaries. Replace null salaries with 0.

SQL> SELECT Employee\_id, NVL(salary, 0) AS salary FROM employees;

EMPLOYEE\_ID SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 60000

2 55000

3 0

4 0

5 48000

6 52000

7 54000

8 0

9 59000

10 0

11 45000

12 0

13 57000

14 0

15 51000

16 0

17 53000

18 0

19 56000

20 0

2. Display the first name and last name of employees. Replace null first names with 'Unknown'.

SQL> SELECT NVL(first\_name, 'Unknown') AS first\_name, last\_name FROM employees;

FIRST\_NAME LAST\_NAME

\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

John Doe

Jane Smith

Michael Johnson

Emily Williams

Daniel Brown

Laura Davis

Matthew Wilson

Emma Brown

Olivia Jones

Noah Smith

William Johnson

Ava Williams

Sophia Davis

James Wilson

Oliver Miller

Amelia Brown

Benjamin Davis

Elijah Wilson

Lucas Brown

Mia Johnson

3. Show the employee IDs and salaries. Replace null salaries with the average salary.

SQL> SELECT Employee\_id, NVL(salary, **AVG**(salary) OVER ()) AS salary FROM Employees;

EMPLOYEE\_ID SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1 60000

2 55000

3 53636.3636363636363636363636363636363636

4 53636.3636363636363636363636363636363636

5 48000

6 52000

7 54000

8 53636.3636363636363636363636363636363636

9 59000

10 53636.3636363636363636363636363636363636

11 45000

12 53636.3636363636363636363636363636363636

13 57000

14 53636.3636363636363636363636363636363636

15 51000

16 53636.3636363636363636363636363636363636

17 53000

18 53636.3636363636363636363636363636363636

19 56000

20 53636.3636363636363636363636363636363636

*----------------- Using Concatenate Operator: ---------------------*

1. List the full names (first name and last name) of employees.

SQL> SELECT first\_name || ' ' || last\_name As full\_name FROM Employees;

FULL\_NAME

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

John Doe

Jane Smith

Michael Johnson

Emily Williams

Daniel Brown

Laura Davis

Matthew Wilson

Emma Brown

Olivia Jones

Noah Smith

William Johnson

Ava Williams

Sophia Davis

James Wilson

Oliver Miller

Amelia Brown

Benjamin Davis

Elijah Wilson

Lucas Brown

Mia Johnson

2. Show employee IDs and concatenated first and last names.

SQL> SELECT employee\_id, first\_name || ' ' || last\_name As full\_name FROM Employees;

EMPLOYEE\_ID FULL\_NAME

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1 John Doe

2 Jane Smith

3 Michael Johnson

4 Emily Williams

5 Daniel Brown

6 Laura Davis

7 Matthew Wilson

8 Emma Brown

9 Olivia Jones

10 Noah Smith

11 William Johnson

12 Ava Williams

13 Sophia Davis

14 James Wilson

15 Oliver Miller

16 Amelia Brown

17 Benjamin Davis

18 Elijah Wilson

19 Lucas Brown

20 Mia Johnson

3. Display a message: "Employee [employee\_id] [first\_name] [last\_name] has a salary of [salary]".

SQL> SELECT 'Employee ' || employee\_id || ' ' || first\_name || ' ' || last\_name || ' has a salary of ' || NVL (salary,0) || '.' AS message

2 FROM employees;

MESSAGE

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Employee 1 John Doe has a salary of 60000.

Employee 2 Jane Smith has a salary of 55000.

Employee 3 Michael Johnson has a salary of 0.

Employee 4 Emily Williams has a salary of 0.

Employee 5 Daniel Brown has a salary of 48000.

Employee 6 Laura Davis has a salary of 52000.

Employee 7 Matthew Wilson has a salary of 54000.

Employee 8 Emma Brown has a salary of 0.

Employee 9 Olivia Jones has a salary of 59000.

Employee 10 Noah Smith has a salary of 0.

Employee 11 William Johnson has a salary of 45000.

Employee 12 Ava Williams has a salary of 0.

Employee 13 Sophia Davis has a salary of 57000.

Employee 14 James Wilson has a salary of 0.

Employee 15 Oliver Miller has a salary of 51000.

Employee 16 Amelia Brown has a salary of 0.

Employee 17 Benjamin Davis has a salary of 53000.

Employee 18 Elijah Wilson has a salary of 0.

Employee 19 Lucas Brown has a salary of 56000.

Employee 20 Mia Johnson has a salary of 0

4. List the full names (last name, comma, first name) of employees.

SQL> SELECT last\_name || ',' || first\_name AS full\_name FROM employees;

FULL\_NAME

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Doe,John

Smith,Jane

Johnson,Michael

Williams,Emily

Brown,Daniel

Davis,Laura

Wilson,Matthew

Brown,Emma

Jones,Olivia

Smith,Noah

Johnson,William

Williams,Ava

Davis,Sophia

Wilson,James

Miller,Oliver

Brown,Amelia

Davis,Benjamin

Wilson,Elijah

Brown,Lucas

Johnson,Mia

5. Show employee IDs and concatenated last name, space, and first name.

SQL> SELECT employee\_id, last\_name || ' ' || first\_name As full\_name FROM Employees;

EMPLOYEE\_ID FULL\_NAME

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1 Doe John

2 Smith Jane

3 Johnson Michael

4 Williams Emily

5 Brown Daniel

6 Davis Laura

7 Wilson Matthew

8 Brown Emma

9 Jones Olivia

10 Smith Noah

11 Johnson William

12 Williams Ava

13 Davis Sophia

14 Wilson James

15 Miller Oliver

16 Brown Amelia

17 Davis Benjamin

18 Wilson Elijah

19 Brown Lucas

20 Johnson Mia

6. Display a message: "Employee [last\_name], [first\_name] has a salary of [salary]."

SQL> SELECT 'Employee' || ' ' || last\_name || ' ' || first\_name || ' ' || 'has a salary of' || salary || ' ' || '.' AS message FROM

Employees;

MESSAGE

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Employee Doe John has a salary of60000 .

Employee Smith Jane has a salary of55000 .

Employee Johnson Michael has a salary of .

Employee Williams Emily has a salary of .

Employee Brown Daniel has a salary of48000 .

Employee Davis Laura has a salary of52000 .

Employee Wilson Matthew has a salary of54000 .

Employee Brown Emma has a salary of .

Employee Jones Olivia has a salary of59000 .

Employee Smith Noah has a salary of .

Employee Johnson William has a salary of45000 .

Employee Williams Ava has a salary of .

Employee Davis Sophia has a salary of57000 .

Employee Wilson James has a salary of .

Employee Miller Oliver has a salary of51000 .

Employee Brown Amelia has a salary of .

Employee Davis Benjamin has a salary of53000 .

Employee Wilson Elijah has a salary of .

Employee Brown Lucas has a salary of56000 .

Employee Johnson Mia has a salary of .

*---------------------------- Using ANY(), SOME(), and ALL(): -------------------------*

1. List employees with salaries greater than ANY other employees salary.

SQL> SELECT \* From Employees WHERE salary > ANY (SELECT salary FROM Employees);

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

9 Olivia Jones 30-05-23 59000

13 Sophia Davis 18-02-23 57000

19 Lucas Brown 22-06-22 56000

2 Jane Smith 55000

7 Matthew Wilson 10-12-21 54000

17 Benjamin Davis 02-12-21 53000

6 Laura Davis 20-04-23 52000

15 Oliver Miller 12-10-22 51000

5 Daniel Brown 05-09-22 48000

2. Show employees with salaries greater than the minimum salary in the company.

SQL> SELECT \* FROM Employees WHERE salary > (SELECT **MIN**(salary) FROM Employees);

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

2 Jane Smith 55000

5 Daniel Brown 05-09-22 48000

6 Laura Davis 20-04-23 52000

7 Matthew Wilson 10-12-21 54000

9 Olivia Jones 30-05-23 59000

13 Sophia Davis 18-02-23 57000

15 Oliver Miller 12-10-22 51000

17 Benjamin Davis 02-12-21 53000

19 Lucas Brown 22-06-22 56000

3. Display employees with salaries less than ALL other employees salaries.

SQL> SELECT \* FROM Employees WHERE salary < ALL (SELECT salary FROM Employees);

no rows selected

4. List employees with salaries greater than ANY other employees salary.

SQL> SELECT \* FROM Employees WHERE salary > ANY (SELECT salary FROM Employees);

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

9 Olivia Jones 30-05-23 59000

13 Sophia Davis 18-02-23 57000

19 Lucas Brown 22-06-22 56000

2 Jane Smith 55000

7 Matthew Wilson 10-12-21 54000

17 Benjamin Davis 02-12-21 53000

6 Laura Davis 20-04-23 52000

15 Oliver Miller 12-10-22 51000

5 Daniel Brown 05-09-22 48000

5. Show employees with salaries greater than SOME of the salaries.

SQL> SELECT \* FROM Employees WHERE salary > SOME (SELECT salary FROM Employees);

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

9 Olivia Jones 30-05-23 59000

13 Sophia Davis 18-02-23 57000

19 Lucas Brown 22-06-22 56000

2 Jane Smith 55000

7 Matthew Wilson 10-12-21 54000

17 Benjamin Davis 02-12-21 53000

6 Laura Davis 20-04-23 52000

15 Oliver Miller 12-10-22 51000

5 Daniel Brown 05-09-22 48000

6. Display employees with salaries less than ALL employees salaries.

SQL> SELECT \* FROM Employees WHERE salary < ALL (SELECT salary FROM Employees);

no rows selected

*--------------------------- Combining Functions and Operators: ---------*

1. Display the employee IDs and names for employees hired in 2022.

SQL> SELECT employee\_id, first\_name, last\_name FROM Employees WHERE EXTRACT(YEAR FROM hire\_date) = 2022;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

1 John Doe

5 Daniel Brown

10 Noah Smith

15 Oliver Miller

19 Lucas Brown

2. List the first name and salary of employees with salaries greater than $55000.00 and hired in 2021.

SQL> SELECT first\_name, salary FROM Employees WHERE salary > 55000 AND EXTRACT(YEAR FROM hire\_date) = 2021;

no rows selected

3. Display the employee IDs and last names for employees hired before January 1, 2022, or with a salary greater than $55000.00.

SQL> SELECT employee\_id,last\_name FROM Employees WHERE hire\_date < TO\_DATE('01-jan-2022', 'DD-MM-YYYY') AND salary > 55000.00;

no rows selected

4. List the first name and salary of employees hired in 2022 with salaries greater than $50000.00.

SQL> SELECT first\_name, salary FROM Employees WHERE EXTRACT(YEAR FROM hire\_date) = 2022 AND salary > 50000;

FIRST\_NAME SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

John 60000

Oliver 51000

Lucas 56000

*------------------- Using Logical Operators: -----------------------------------------*

1. Display employees hired in 2022 with salaries less than $55000.00.

SELECT \* FROM Employees WHERE EXTRACT(YEAR FROM hire\_date) = 2022 AND salary < 55000.00;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

5 Daniel Brown 05-09-22 48000

15 Oliver Miller 12-10-22 51000

2. List employees with non-null hire dates and salaries.

SQL> SELECT \* FROM Employees WHERE hire\_date IS NOT NULL AND salary IS NOT NULL;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

5 Daniel Brown 05-09-22 48000

6 Laura Davis 20-04-23 52000

7 Matthew Wilson 10-12-21 54000

9 Olivia Jones 30-05-23 59000

11 William Johnson 25-06-21 45000

13 Sophia Davis 18-02-23 57000

15 Oliver Miller 12-10-22 51000

17 Benjamin Davis 02-12-21 53000

19 Lucas Brown 22-06-22 56000

3. Show employees with salaries less than $50000.00 or hired in 2023.

SQL> SELECT \* FROM Employees WHERE salary < 50000.00 OR EXTRACT(YEAR FROM hire\_date) = 2023;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

3 Michael Johnson 15-03-23

5 Daniel Brown 05-09-22 48000

6 Laura Davis 20-04-23 52000

9 Olivia Jones 30-05-23 59000

11 William Johnson 25-06-21 45000

13 Sophia Davis 18-02-23 57000

18 Elijah Wilson 09-03-23

4. Display employees with last names containing 'Brown' or hired in 2023 with salaries less than $60000.00.

SQL> SELECT \* FROM Employees WHERE (last\_name LIKE '%Brown%' OR EXTRACT(YEAR FROM hire\_date) = 2023) AND salary < 60000.00;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

5 Daniel Brown 05-09-22 48000

6 Laura Davis 20-04-23 52000

9 Olivia Jones 30-05-23 59000

13 Sophia Davis 18-02-23 57000

19 Lucas Brown 22-06-22 56000

5. List employees with non-null hire dates and salaries.

SQL> SELECT \* FROM Employees WHERE hire\_date IS NOT NULL AND salary IS NOT NULL;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

1 John Doe 01-07-22 60000

5 Daniel Brown 05-09-22 48000

6 Laura Davis 20-04-23 52000

7 Matthew Wilson 10-12-21 54000

9 Olivia Jones 30-05-23 59000

11 William Johnson 25-06-21 45000

13 Sophia Davis 18-02-23 57000

15 Oliver Miller 12-10-22 51000

17 Benjamin Davis 02-12-21 53000

19 Lucas Brown 22-06-22 56000

6. Show employees with salaries less than $50000.00 and hire dates either before January 1, 2022, or null.

SQL> SELECT \* FROM Employees WHERE salary < 50000.00 AND (hire\_date < DATE '2022-01-01' OR hire\_date IS NULL);

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME HIRE\_DATE SALARY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

11 William Johnson 25-06-21 45000