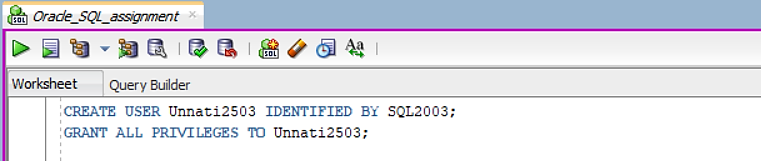
Oracle SQL Assignment Solutions

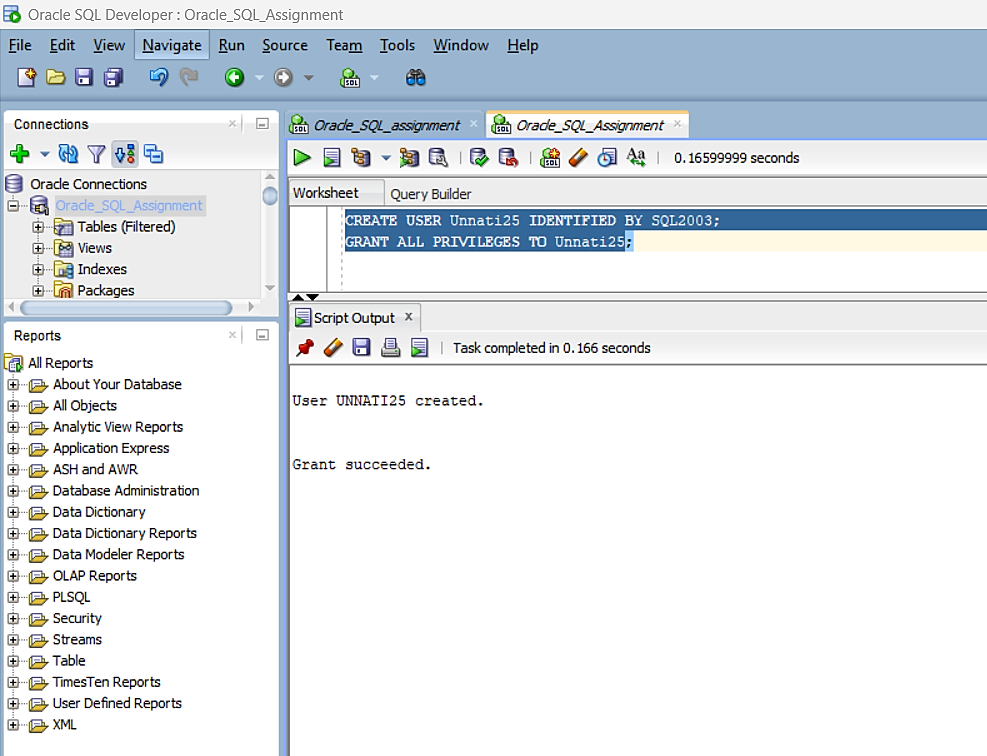
**Queries have been performed on Oracle SQL Developer**

**Step 1: Create User and Grant privileges.**

CREATE USER Unnati2503 IDENTIFIED BY SQL2003;

GRANT ALL PRIVILEGES TO Unnati2503;

******

******

**Step 2: Create department and employee table.**

CREATE TABLE dept (

id INT PRIMARY KEY,

name VARCHAR(50)

);

CREATE TABLE emp (

id INT PRIMARY KEY,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

salary INT,

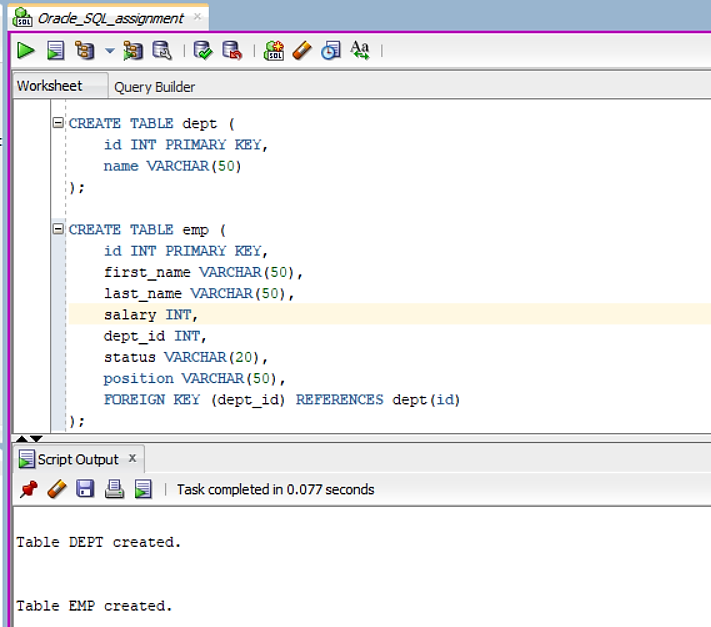
dept\_id INT,

status VARCHAR(20),

position VARCHAR(50),

FOREIGN KEY (dept\_id) REFERENCES dept(id)

);

******

**Step 3: Insert data into Department table and employee table.**

INSERT INTO dept (id, name) VALUES (1, 'Sales');

INSERT INTO dept (id, name) VALUES (2, 'Marketing');

INSERT INTO dept (id, name) VALUES (3, 'HR');

INSERT INTO dept (id, name) VALUES (4, 'IT');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (101, 'Jaggu', 'Singh', 45000, 1, 'Active', 'Sales Manager');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (102, 'Jiya', 'Sharma', 52000, 2, 'Active', 'Marketing Executive');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (103, 'Sam', 'Singh', 67000, 3, 'Active', 'HR Coordinator');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (104, 'Anjali', 'Verma', 95000, 4, 'Active', 'Software Developer');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (105, 'Mona', 'Patel', 58000, 1, 'Active', 'Sales Associate');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (106, 'Priya', 'Desai', 41000, NULL, 'Inactive', 'HR Assistant');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (107, 'Madhav', 'Reddy', 55000, NULL, 'Active', 'Marketing Analyst');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (108, 'Jagdish', 'Iyer', 49000, 1, 'Active', 'Sales Representative');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (109, 'Jasmin', 'Kumar', 65000, 2, 'Active', 'Marketing Manager');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

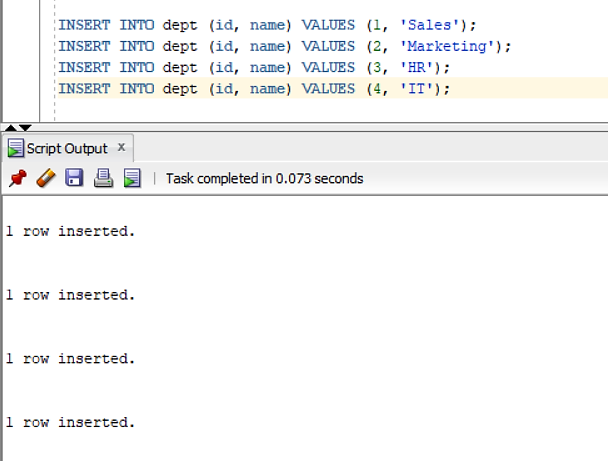
VALUES (110, 'Krishna', 'Mehta', 57000, 3, 'Inactive', 'HR Manager');

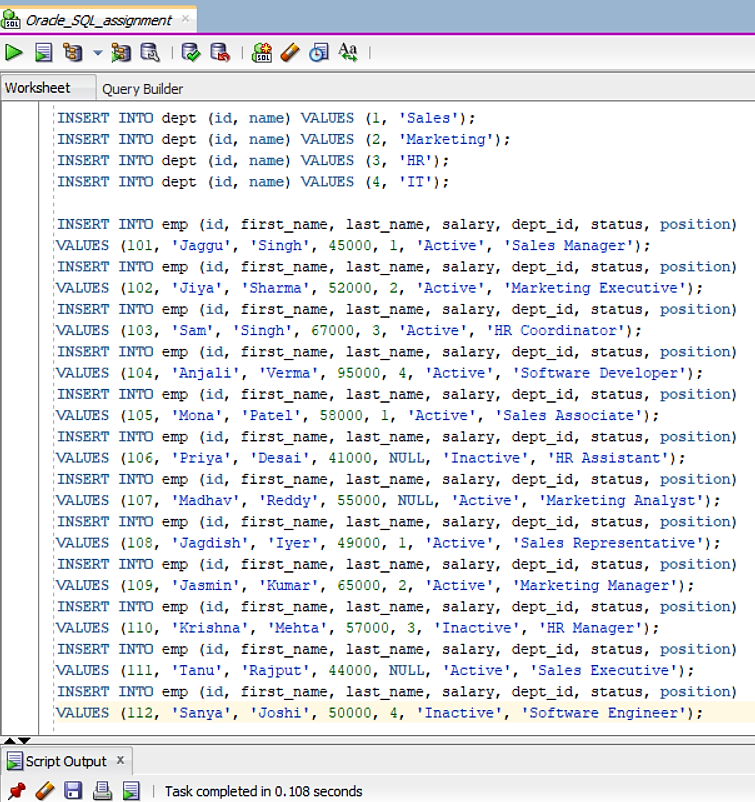
INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (111, 'Tanu', 'Rajput', 44000, NULL, 'Active', 'Sales Executive');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (112, 'Sanya', 'Joshi', 50000, 4, 'Inactive', 'Software Engineer');

******

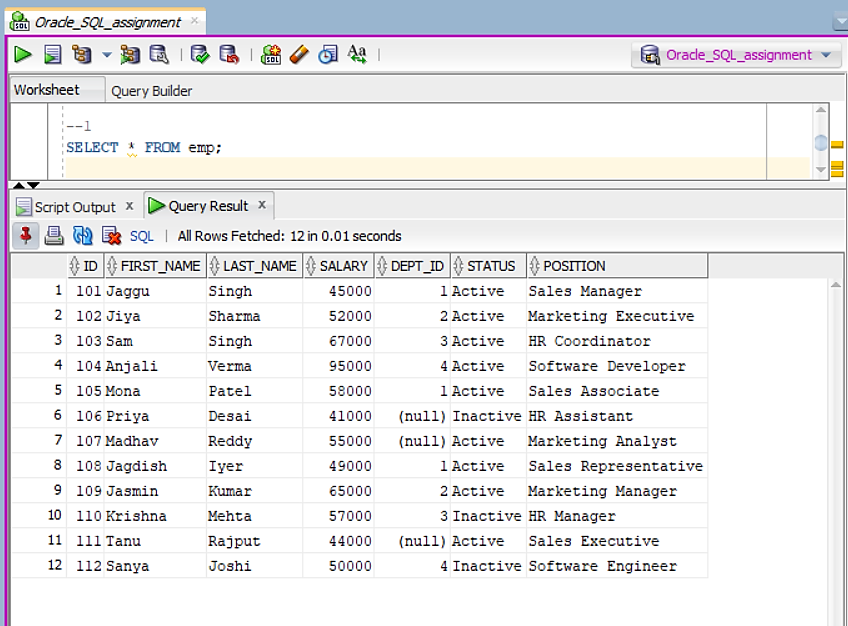
******

**### 1. \*\*Basic SELECT Query\*\***

- Write an SQL query to retrieve all columns from the `employees` table.

--1

SELECT \* FROM emp;



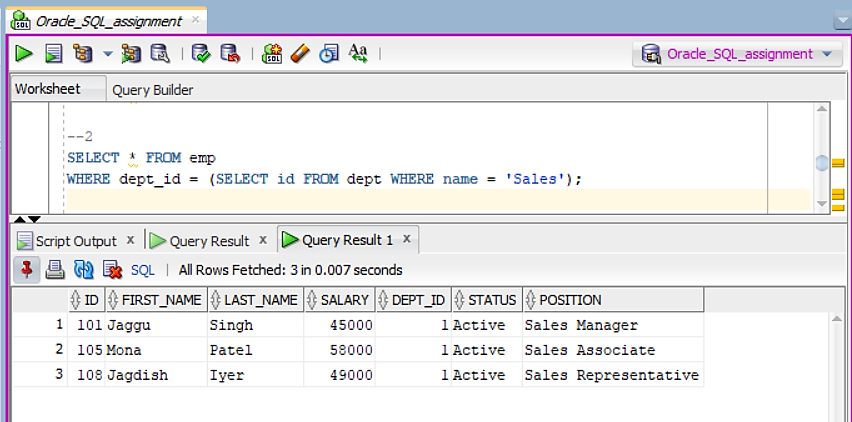
**### 2. \*\*Filtering Data\*\***

- Write an SQL query to find all employees who are working in the "Sales" department.

--2

SELECT \* FROM emp

WHERE dept\_id = (SELECT id FROM dept WHERE name = 'Sales');



**### 3. \*\*Sorting Data\*\***

- Write an SQL query to get the names and salaries of employees in the "Marketing" department, sorted by their salaries in descending order.

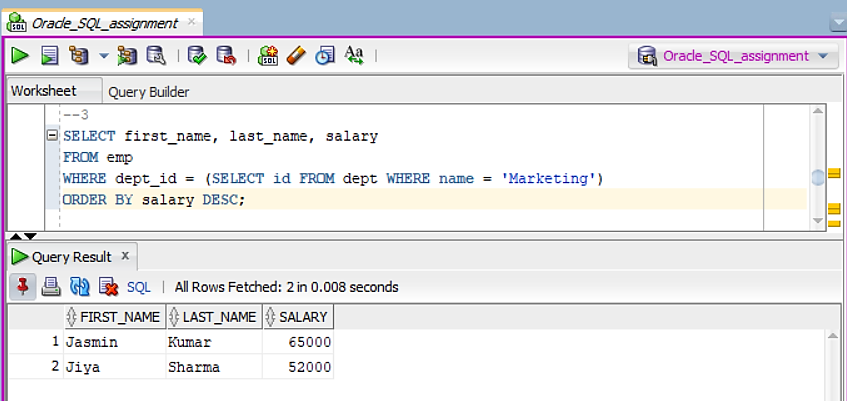
--3

SELECT first\_name, last\_name, salary

FROM emp

WHERE dept\_id = (SELECT id FROM dept WHERE name = 'Marketing')

ORDER BY salary DESC;



**### 4. \*\*Using Aggregate Functions\*\***

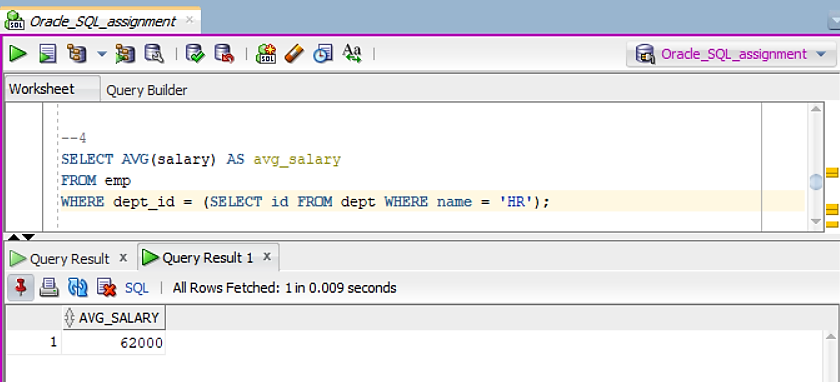
- Write an SQL query to calculate the average salary of employees in the "HR" department.

--4

SELECT AVG(salary) AS avg\_salary

FROM emp

WHERE dept\_id = (SELECT id FROM dept WHERE name = 'HR');



**### 5. \*\*Group By Clause\*\***

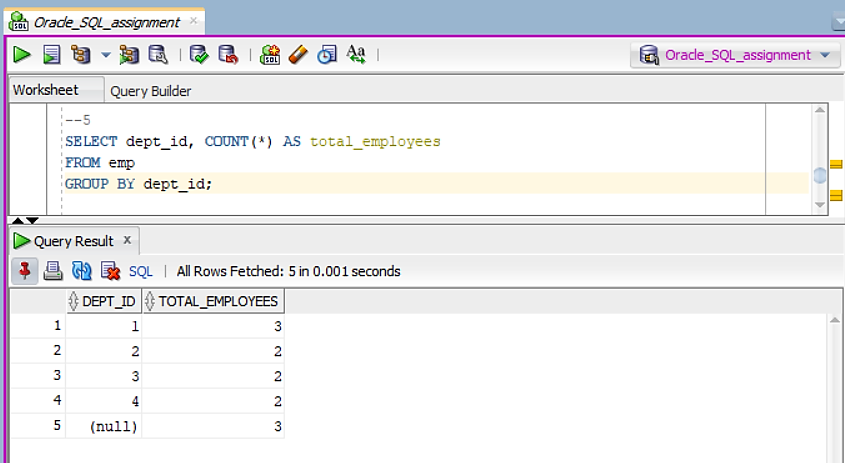
- Write an SQL query to find the total number of employees in each department.

--5

SELECT dept\_id, COUNT(\*) AS total\_employees

FROM emp

GROUP BY dept\_id;



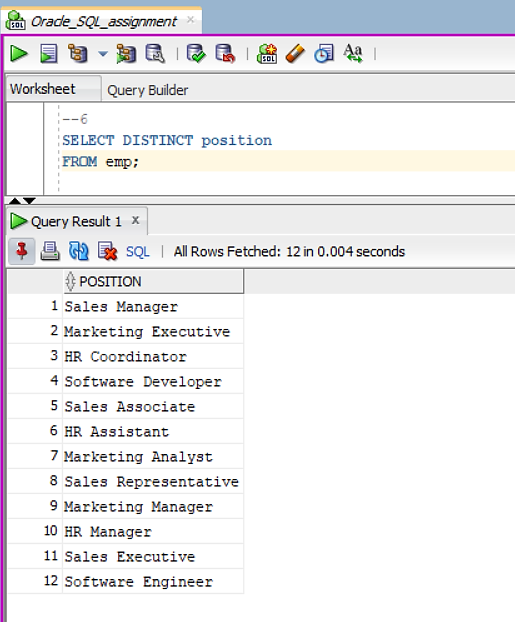
**### 6. \*\*Using DISTINCT\*\***

- Write an SQL query to list all unique job titles from the `employees` table.

--6

SELECT DISTINCT position

FROM emp;



**### 7. \*\*Using LIKE Operator\*\***

- Write an SQL query to retrieve all employees whose names start with the letter "J".

--7

SELECT \* FROM emp

WHERE first\_name LIKE 'J%';



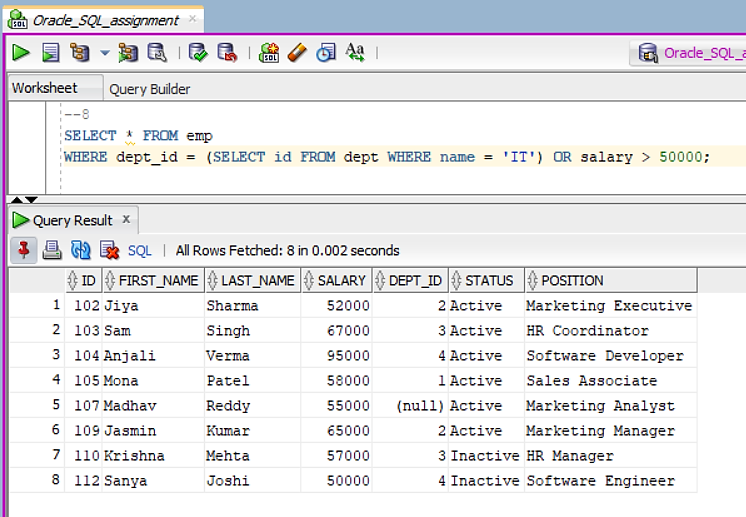
**### 8. \*\*Using AND/OR Conditions\*\***

- Write an SQL query to find employees who are either in the "IT" department or have a salary greater than $50,000.

--8

SELECT \* FROM emp

WHERE dept\_id = (SELECT id FROM dept WHERE name = 'IT') OR salary > 50000;



**### 9. \*\*Joining Tables (Inner Join)\*\***

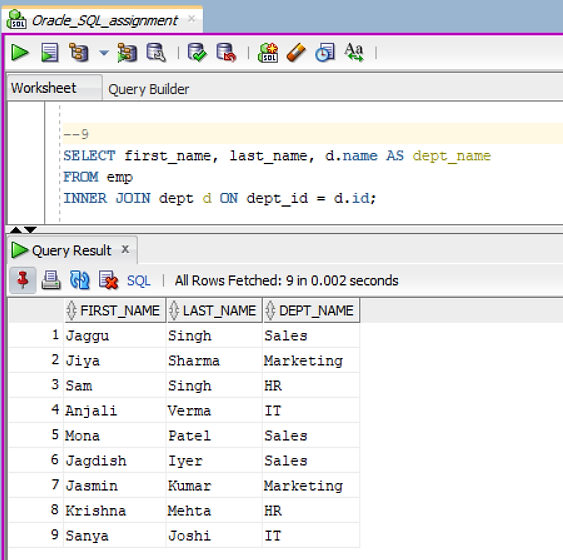
- Write an SQL query to display employee names along with their department names by joining the `employees` and `departments` tables.

--9

SELECT first\_name, last\_name, d.name AS dept\_name

FROM emp

INNER JOIN dept d ON dept\_id = d.id;



**### 10. \*\*Joining Tables (Left Join)\*\***

- Write an SQL query to display all employees and their department names, including those employees who are not assigned to any department.

--10

SELECT e.first\_name, e.last\_name, d.name AS dept\_name

FROM emp e

LEFT JOIN dept d ON e.dept\_id = d.id;



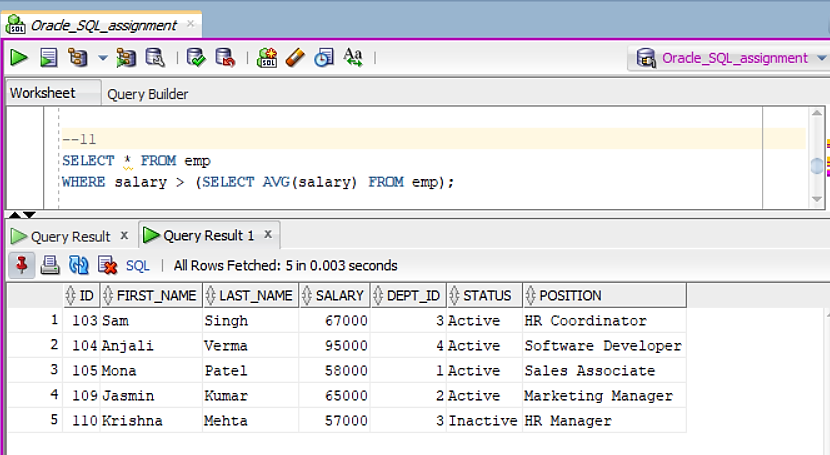
**### 11. \*\*Subqueries\*\***

- Write an SQL query to find employees whose salary is greater than the average salary in the `employees` table .

--11

SELECT \* FROM emp

WHERE salary > (SELECT AVG(salary) FROM emp);



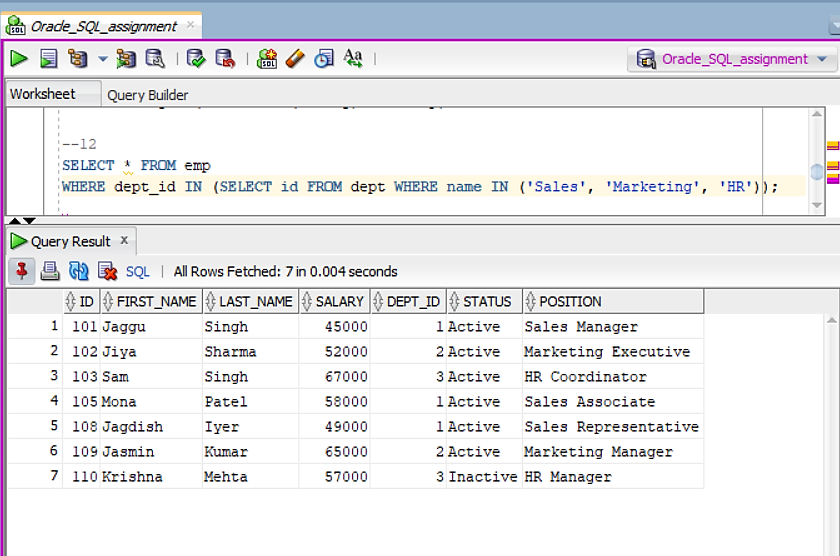
### 12. \*\*Using IN Operator\*\*

- Write an SQL query to list all employees who belong to the departments "Sales", "Marketing", or "HR".

--12

SELECT \* FROM emp

WHERE dept\_id IN (SELECT id FROM dept WHERE name IN ('Sales', 'Marketing', 'HR'));



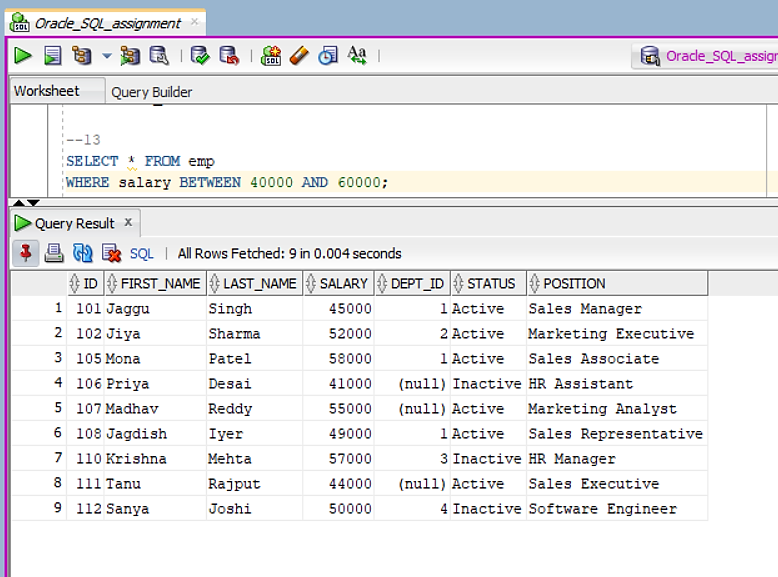
**### 13. \*\*Using BETWEEN Operator\*\***

- Write an SQL query to find employees whose salaries are between $40,000 and $60,000.

--13

SELECT \* FROM emp

WHERE salary BETWEEN 40000 AND 60000;



**### 14. \*\*Using EXISTS\*\***

- Write an SQL query to find departments that have at least one employee with a salary greater than $70,000.

--14

SELECT name FROM dept

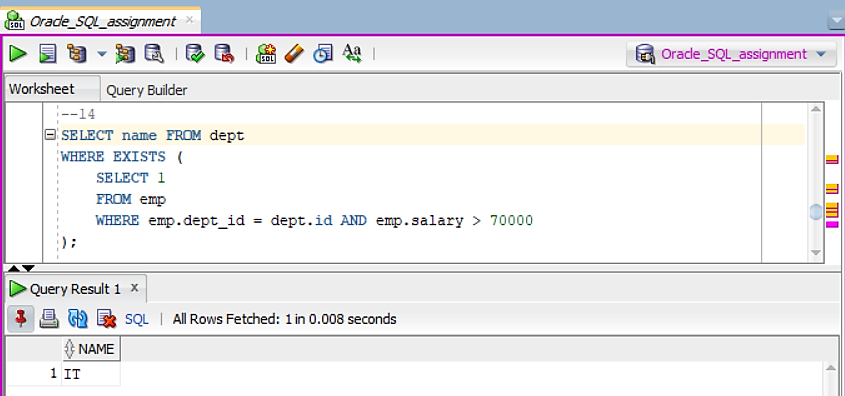
WHERE EXISTS (

SELECT 1

FROM emp

WHERE emp.dept\_id = dept.id AND emp.salary > 70000

);



**### 15. \*\*Date Functions\*\***

- Write an SQL query to find all employees who joined after January 1, 2020.

--15

ALTER TABLE emp

ADD hire\_date DATE;

UPDATE emp SET hire\_date = TO\_DATE('2019-04-15', 'YYYY-MM-DD') WHERE id = 101;

UPDATE emp SET hire\_date = TO\_DATE('2020-06-20', 'YYYY-MM-DD') WHERE id = 102;

UPDATE emp SET hire\_date = TO\_DATE('2018-05-30', 'YYYY-MM-DD') WHERE id = 103;

UPDATE emp SET hire\_date = TO\_DATE('2021-03-25', 'YYYY-MM-DD') WHERE id = 104;

UPDATE emp SET hire\_date = TO\_DATE('2021-07-10', 'YYYY-MM-DD') WHERE id = 105;

UPDATE emp SET hire\_date = TO\_DATE('2022-01-05', 'YYYY-MM-DD') WHERE id = 106;

UPDATE emp SET hire\_date = TO\_DATE('2021-11-15', 'YYYY-MM-DD') WHERE id = 107;

UPDATE emp SET hire\_date = TO\_DATE('2020-02-20', 'YYYY-MM-DD') WHERE id = 108;

UPDATE emp SET hire\_date = TO\_DATE('2019-09-12', 'YYYY-MM-DD') WHERE id = 109;

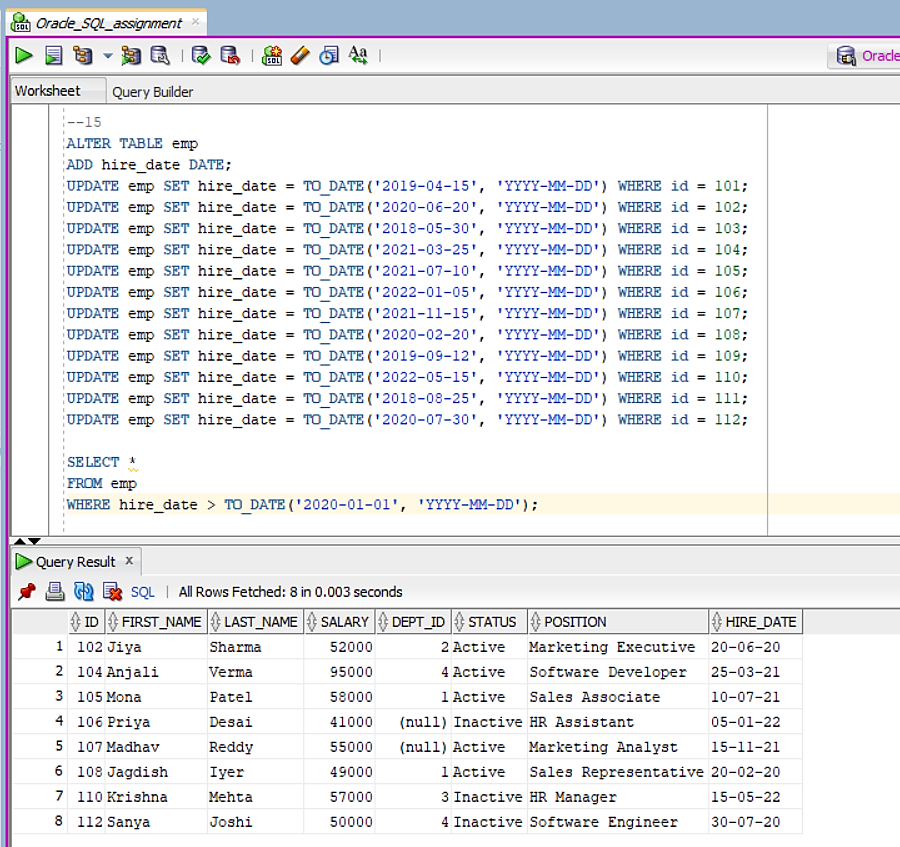
UPDATE emp SET hire\_date = TO\_DATE('2022-05-15', 'YYYY-MM-DD') WHERE id = 110;

UPDATE emp SET hire\_date = TO\_DATE('2018-08-25', 'YYYY-MM-DD') WHERE id = 111;

UPDATE emp SET hire\_date = TO\_DATE('2020-07-30', 'YYYY-MM-DD') WHERE id = 112;

SELECT \* FROM emp

WHERE hire\_date > TO\_DATE('2020-01-01', 'YYYY-MM-DD');



**### 16. \*\*Updating Data\*\***

- Write an SQL query to increase the salary of all employees in the "IT" department by 10%.

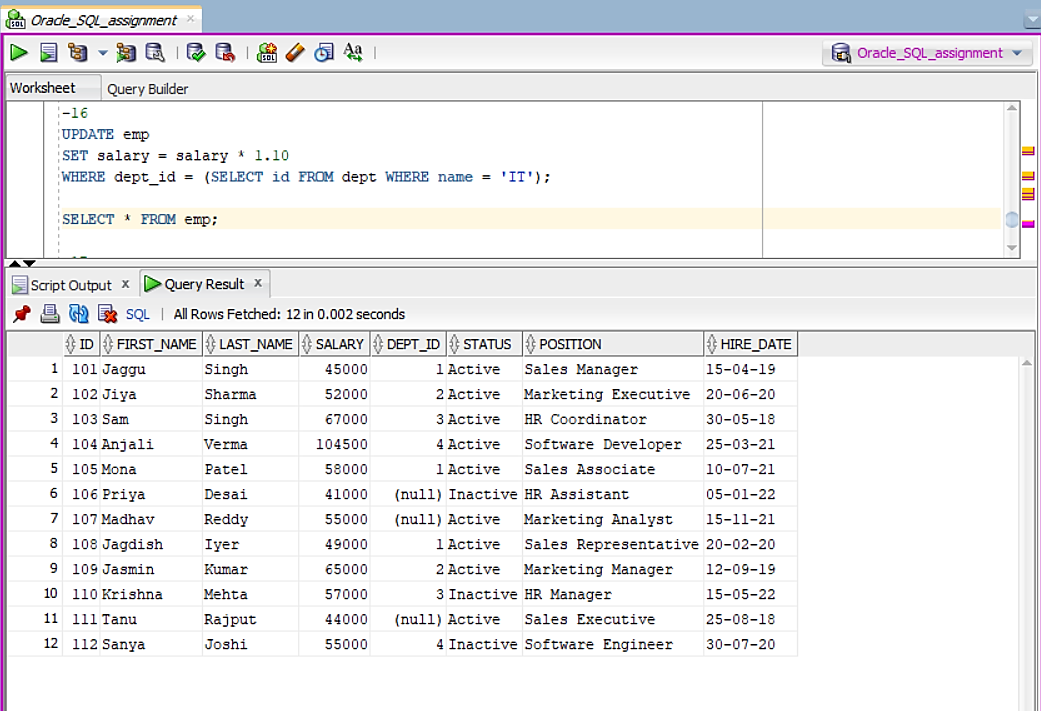
--16

UPDATE emp

SET salary = salary \* 1.10

WHERE dept\_id = (SELECT id FROM dept WHERE name = 'IT');

SELECT \* FROM emp;



**### 17. \*\*Deleting Data\*\***

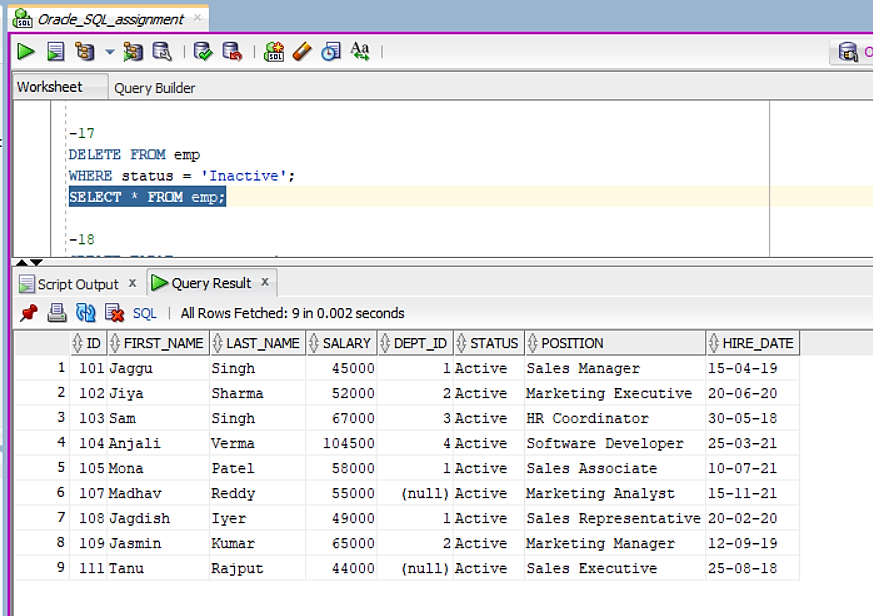
- Write an SQL query to delete all employees who are no longer with the company.

--17

DELETE FROM emp

WHERE status = 'Inactive';

SELECT \* FROM emp;



**### 18. \*\*Creating a Table\*\***

- Write an SQL query to create a table called `customers` with columns `customer\_id`, `first\_name`, `last\_name`, `email`, and `phone\_number`.

--18

CREATE TABLE customers (

customer\_id INT PRIMARY KEY,

first\_name VARCHAR(50),

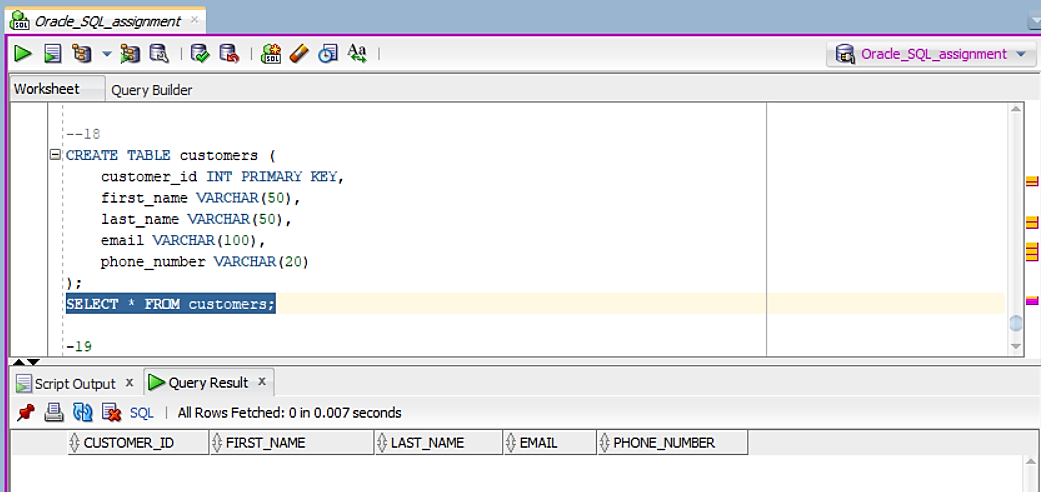
last\_name VARCHAR(50),

email VARCHAR(100),

phone\_number VARCHAR(20)

);

SELECT \* FROM customers;

****

**### 19. \*\*Modifying a Table (ALTER)\*\***

- Write an SQL query to add a new column `hire\_date` to the `employees` table.

--19

ALTER TABLE emp

ADD hire\_date DATE;

UPDATE emp SET hire\_date = TO\_DATE('2019-04-15', 'YYYY-MM-DD') WHERE id = 101;

UPDATE emp SET hire\_date = TO\_DATE('2020-06-20', 'YYYY-MM-DD') WHERE id = 102;

UPDATE emp SET hire\_date = TO\_DATE('2018-05-30', 'YYYY-MM-DD') WHERE id = 103;

UPDATE emp SET hire\_date = TO\_DATE('2021-03-25', 'YYYY-MM-DD') WHERE id = 104;

UPDATE emp SET hire\_date = TO\_DATE('2021-07-10', 'YYYY-MM-DD') WHERE id = 105;

UPDATE emp SET hire\_date = TO\_DATE('2022-01-05', 'YYYY-MM-DD') WHERE id = 106;

UPDATE emp SET hire\_date = TO\_DATE('2021-11-15', 'YYYY-MM-DD') WHERE id = 107;

UPDATE emp SET hire\_date = TO\_DATE('2020-02-20', 'YYYY-MM-DD') WHERE id = 108;

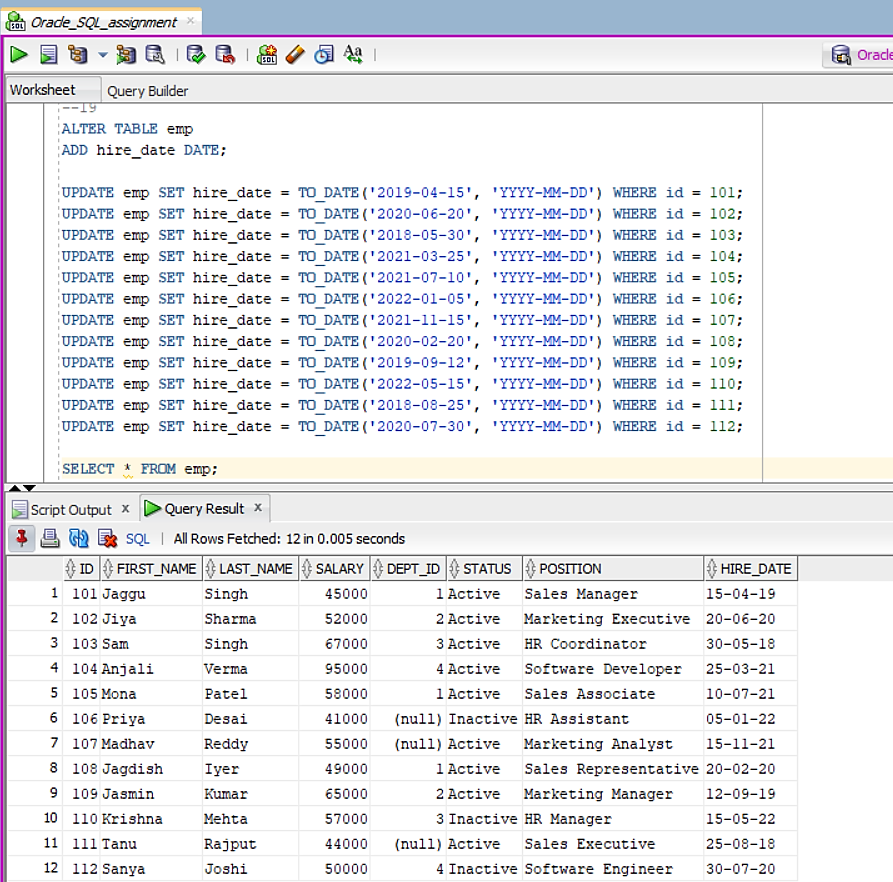
UPDATE emp SET hire\_date = TO\_DATE('2019-09-12', 'YYYY-MM-DD') WHERE id = 109;

UPDATE emp SET hire\_date = TO\_DATE('2022-05-15', 'YYYY-MM-DD') WHERE id = 110;

UPDATE emp SET hire\_date = TO\_DATE('2018-08-25', 'YYYY-MM-DD') WHERE id = 111;

UPDATE emp SET hire\_date = TO\_DATE('2020-07-30', 'YYYY-MM-DD') WHERE id = 112;

SELECT \* FROM emp;



**### 20. \*\*Dropping a Table\*\***

- Write an SQL query to drop the `temporary\_employees` table if it exists.

--20

CREATE TABLE temporary\_employees (

emp\_id INT PRIMARY KEY,

first\_name VARCHAR2(50),

last\_name VARCHAR2(50),

salary NUMBER(10, 2),

department\_id INT,

status VARCHAR2(20),

position VARCHAR2(50)

);

DROP TABLE temporary\_employees;



**Complete SQL Script:**

CREATE USER Unnati2503 IDENTIFIED BY SQL2003;

GRANT ALL PRIVILEGES TO Unnati2503;

CREATE TABLE dept (

id INT PRIMARY KEY,

name VARCHAR(50)

);

CREATE TABLE emp (

id INT PRIMARY KEY,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

salary INT,

dept\_id INT,

status VARCHAR(20),

position VARCHAR(50),

FOREIGN KEY (dept\_id) REFERENCES dept(id)

);

INSERT INTO dept (id, name) VALUES (1, 'Sales');

INSERT INTO dept (id, name) VALUES (2, 'Marketing');

INSERT INTO dept (id, name) VALUES (3, 'HR');

INSERT INTO dept (id, name) VALUES (4, 'IT');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (101, 'Jaggu', 'Singh', 45000, 1, 'Active', 'Sales Manager');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (102, 'Jiya', 'Sharma', 52000, 2, 'Active', 'Marketing Executive');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (103, 'Sam', 'Singh', 67000, 3, 'Active', 'HR Coordinator');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (104, 'Anjali', 'Verma', 95000, 4, 'Active', 'Software Developer');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (105, 'Mona', 'Patel', 58000, 1, 'Active', 'Sales Associate');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (106, 'Priya', 'Desai', 41000, NULL, 'Inactive', 'HR Assistant');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (107, 'Madhav', 'Reddy', 55000, NULL, 'Active', 'Marketing Analyst');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (108, 'Jagdish', 'Iyer', 49000, 1, 'Active', 'Sales Representative');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (109, 'Jasmin', 'Kumar', 65000, 2, 'Active', 'Marketing Manager');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (110, 'Krishna', 'Mehta', 57000, 3, 'Inactive', 'HR Manager');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (111, 'Tanu', 'Rajput', 44000, NULL, 'Active', 'Sales Executive');

INSERT INTO emp (id, first\_name, last\_name, salary, dept\_id, status, position)

VALUES (112, 'Sanya', 'Joshi', 50000, 4, 'Inactive', 'Software Engineer');

--1

SELECT \* FROM emp;

--2

SELECT \* FROM emp

WHERE dept\_id = (SELECT id FROM dept WHERE name = 'Sales');

--3

SELECT first\_name, last\_name, salary

FROM emp

WHERE dept\_id = (SELECT id FROM dept WHERE name = 'Marketing')

ORDER BY salary DESC;

--4

SELECT AVG(salary) AS avg\_salary

FROM emp

WHERE dept\_id = (SELECT id FROM dept WHERE name = 'HR');

--5

SELECT dept\_id, COUNT(\*) AS total\_employees

FROM emp

GROUP BY dept\_id;

--6

SELECT DISTINCT position

FROM emp;

--7

SELECT \* FROM emp

WHERE first\_name LIKE 'J%';

--8

SELECT \* FROM emp

WHERE dept\_id = (SELECT id FROM dept WHERE name = 'IT') OR salary > 50000;

--9

SELECT first\_name, last\_name, d.name AS dept\_name

FROM emp

INNER JOIN dept d ON dept\_id = d.id;

--10

SELECT e.first\_name, e.last\_name, d.name AS dept\_name

FROM emp e

LEFT JOIN dept d ON e.dept\_id = d.id;

--11

SELECT \* FROM emp

WHERE salary > (SELECT AVG(salary) FROM emp);

--12

SELECT \* FROM emp

WHERE dept\_id IN (SELECT id FROM dept WHERE name IN ('Sales', 'Marketing', 'HR'));

--13

SELECT \* FROM emp

WHERE salary BETWEEN 40000 AND 60000;

--14

SELECT name FROM dept

WHERE EXISTS (

SELECT 1

FROM emp

WHERE emp.dept\_id = dept.id AND emp.salary > 70000

);

--15

ALTER TABLE emp

ADD hire\_date DATE;

UPDATE emp SET hire\_date = TO\_DATE('2019-04-15', 'YYYY-MM-DD') WHERE id = 101;

UPDATE emp SET hire\_date = TO\_DATE('2020-06-20', 'YYYY-MM-DD') WHERE id = 102;

UPDATE emp SET hire\_date = TO\_DATE('2018-05-30', 'YYYY-MM-DD') WHERE id = 103;

UPDATE emp SET hire\_date = TO\_DATE('2021-03-25', 'YYYY-MM-DD') WHERE id = 104;

UPDATE emp SET hire\_date = TO\_DATE('2021-07-10', 'YYYY-MM-DD') WHERE id = 105;

UPDATE emp SET hire\_date = TO\_DATE('2022-01-05', 'YYYY-MM-DD') WHERE id = 106;

UPDATE emp SET hire\_date = TO\_DATE('2021-11-15', 'YYYY-MM-DD') WHERE id = 107;

UPDATE emp SET hire\_date = TO\_DATE('2020-02-20', 'YYYY-MM-DD') WHERE id = 108;

UPDATE emp SET hire\_date = TO\_DATE('2019-09-12', 'YYYY-MM-DD') WHERE id = 109;

UPDATE emp SET hire\_date = TO\_DATE('2022-05-15', 'YYYY-MM-DD') WHERE id = 110;

UPDATE emp SET hire\_date = TO\_DATE('2018-08-25', 'YYYY-MM-DD') WHERE id = 111;

UPDATE emp SET hire\_date = TO\_DATE('2020-07-30', 'YYYY-MM-DD') WHERE id = 112;

FROM emp

WHERE hire\_date > TO\_DATE('2020-01-01', 'YYYY-MM-DD');

-16

UPDATE emp

SET salary = salary \* 1.10

WHERE dept\_id = (SELECT id FROM dept WHERE name = 'IT');

SELECT \* FROM emp;

-17

DELETE FROM emp

WHERE status = 'Inactive';

SELECT \* FROM emp;

--18

CREATE TABLE customers (

customer\_id INT PRIMARY KEY,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

email VARCHAR(100),

phone\_number VARCHAR(20)

);

SELECT \* FROM customers;

--19

ALTER TABLE emp

ADD hire\_date DATE;

UPDATE emp SET hire\_date = TO\_DATE('2019-04-15', 'YYYY-MM-DD') WHERE id = 101;

UPDATE emp SET hire\_date = TO\_DATE('2020-06-20', 'YYYY-MM-DD') WHERE id = 102;

UPDATE emp SET hire\_date = TO\_DATE('2018-05-30', 'YYYY-MM-DD') WHERE id = 103;

UPDATE emp SET hire\_date = TO\_DATE('2021-03-25', 'YYYY-MM-DD') WHERE id = 104;

UPDATE emp SET hire\_date = TO\_DATE('2021-07-10', 'YYYY-MM-DD') WHERE id = 105;

UPDATE emp SET hire\_date = TO\_DATE('2022-01-05', 'YYYY-MM-DD') WHERE id = 106;

UPDATE emp SET hire\_date = TO\_DATE('2021-11-15', 'YYYY-MM-DD') WHERE id = 107;

UPDATE emp SET hire\_date = TO\_DATE('2020-02-20', 'YYYY-MM-DD') WHERE id = 108;

UPDATE emp SET hire\_date = TO\_DATE('2019-09-12', 'YYYY-MM-DD') WHERE id = 109;

UPDATE emp SET hire\_date = TO\_DATE('2022-05-15', 'YYYY-MM-DD') WHERE id = 110;

UPDATE emp SET hire\_date = TO\_DATE('2018-08-25', 'YYYY-MM-DD') WHERE id = 111;

UPDATE emp SET hire\_date = TO\_DATE('2020-07-30', 'YYYY-MM-DD') WHERE id = 112;

SELECT \* FROM emp;

--20

CREATE TABLE temporary\_employees (

emp\_id INT PRIMARY KEY,

first\_name VARCHAR2(50),

last\_name VARCHAR2(50),

salary NUMBER(10, 2),

department\_id INT,

status VARCHAR2(20),

position VARCHAR2(50)

);

DROP TABLE temporary\_employees;