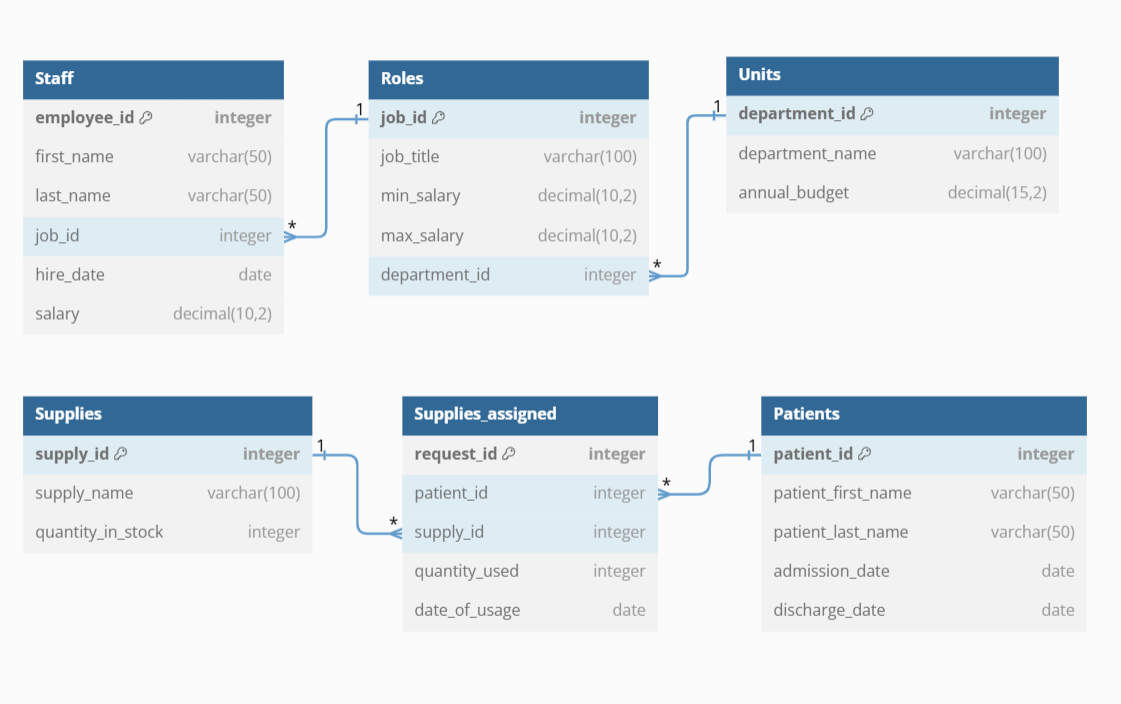
**Database for Hospital Management**

1. **Description of the project**

This project analyzes employee data to manage staffing costs, examines departmental budgets to enhance resource allocation, and evaluates patient care costs to streamline services. Additionally, it tracks medical supply usage to improve inventory efficiency. By analyzing data across these areas, the project identifies cost drivers, reduces waste, and improves profitability. Using past data for financial forecasting, the project provides tools for better planning and smarter decision-making, supporting both efficiency and high-quality patient care.

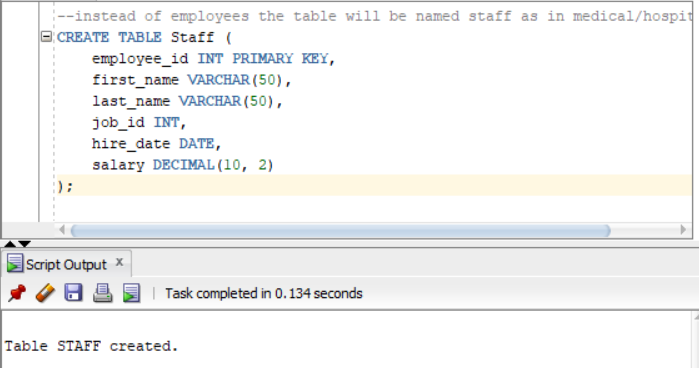
1. **Database schema**
2. **Constructing the database**

--creating the tables

CREATE TABLE Staff (

employee\_id INT PRIMARY KEY,

first\_name VARCHAR(50),

 last\_name VARCHAR(50),

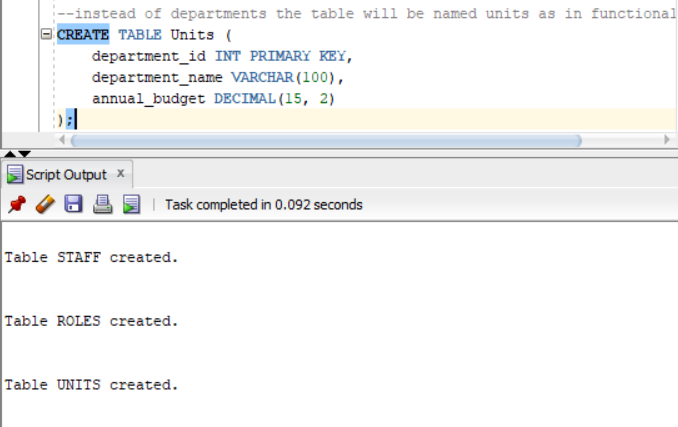
job\_id INT,

hire\_date DATE,

salary DECIMAL(10, 2)

);

CREATE TABLE roles (

 job\_id INT PRIMARY KEY,

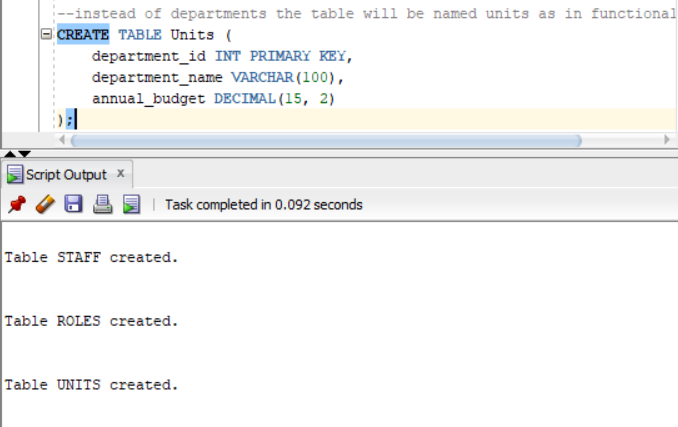
job\_title VARCHAR(100),

min\_salary DECIMAL(10, 2),

max\_salary DECIMAL(10, 2),

department\_id INT

);



CREATE TABLE units (

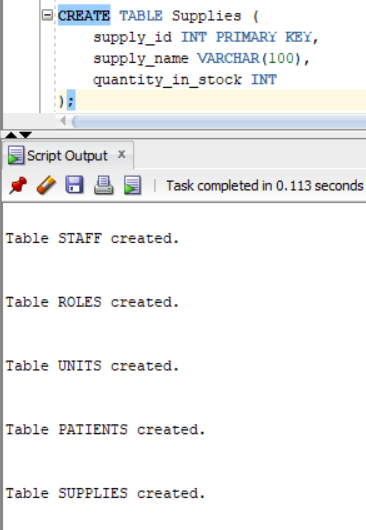
department\_id INT PRIMARY KEY,

department\_name VARCHAR(100),

annual\_budget DECIMAL(15, 2)

);

CREATE TABLE Patients (

 patient\_id INT PRIMARY KEY,

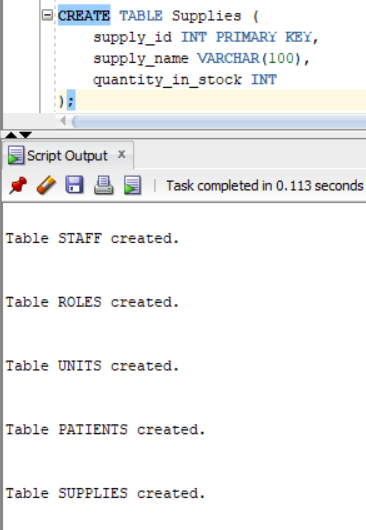
patient\_first\_name VARCHAR(50),

patient\_last\_name VARCHAR(50),

admission\_date DATE,

discharge\_date DATE

);

CREATE TABLE Supplies(

supply\_id INT PRIMARY KEY,

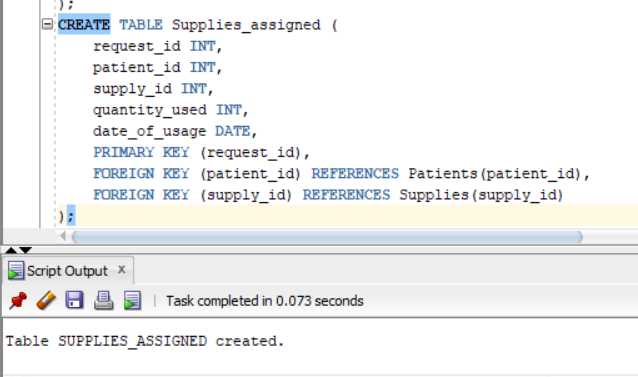
supply\_name VARCHAR(100),

quantity\_in\_stock INT

);

CREATE TABLE Supplies\_assigned (

request\_id INT,

 patient\_id INT,

supply\_id INT,

quantity\_used INT,

date\_of\_usage DATE,

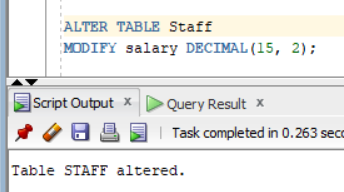
PRIMARY KEY (request\_id),

FOREIGN KEY (patient\_id) REFERENCES Patients(patient\_id),

FOREIGN KEY (supply\_id) REFERENCES Supplies(supply\_id)

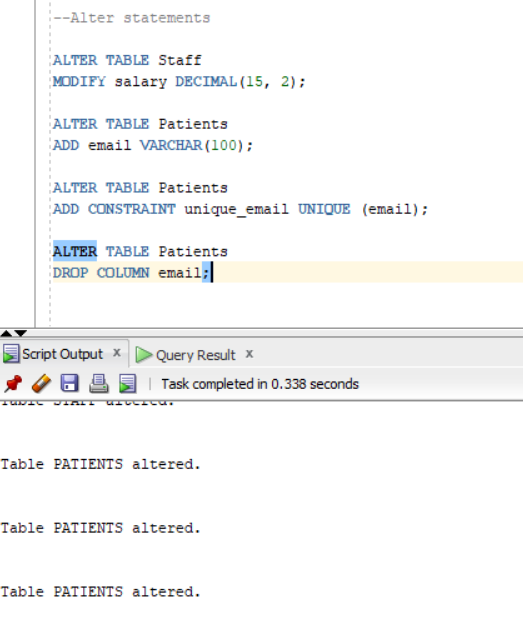
);

--

ALTER TABLE Staff

MODIFY salary DECIMAL(15, 2);

--



ALTER TABLE Patients

ADD email VARCHAR(100);

--

ALTER TABLE Patients

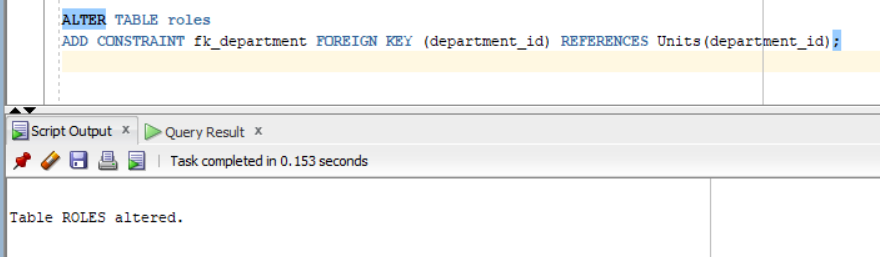
ADD CONSTRAINT unique\_email UNIQUE (email);

--

ALTER TABLE Patients

DROP COLUMN email;

--

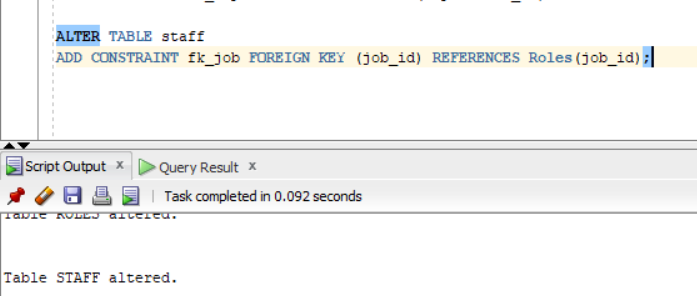
ALTER TABLE roles

ADD CONSTRAINT fk\_department

FOREIGN KEY (department\_id)

REFERENCES Units(department\_id);

--

ALTER TABLE staff

ADD CONSTRAINT fk\_job

FOREIGN KEY (job\_id)

REFERENCES Roles(job\_id);

1. **DML statements: INSERT, UPDATE, DELETE, and MERGE**

--Inserting values into Staff table

INSERT INTO staff(employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (1, 'John', 'Doe', 2, TO\_DATE ('2022-01-15', 'YYYY-MM-DD'), 35000.00);

INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (2, 'Mark', 'Smith', 3, TO\_DATE ('2022-05-23', 'YYYY-MM-DD'), 63000);

INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (3, 'Carry', 'Donovan', 1, TO\_DATE ('2023-10-06', 'YYYY-MM-DD'), 77000);

INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (4, 'Lucas', 'Taylor', 4, TO\_DATE ('2021-02-11', 'YYYY-MM-DD'), 50000);

INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (5, 'Chloe', 'Perez', 2, TO\_DATE ('2022-03-20', 'YYYY-MM-DD'), 45000);

INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (6, 'Noah', 'Adams', 5, TO\_DATE ('2020-09-01', 'YYYY-MM-DD'), 49000);

INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (7, 'Ava', 'Nelson', 3, TO\_DATE ('2022-08-15', 'YYYY-MM-DD'),60000);

INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (8, 'Elijah', 'Carter', 6, TO\_DATE ('2021-11-22', 'YYYY-MM-DD'), 47000);

INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (9, 'Grace', 'Mitchell', 1, TO\_DATE ('2023-04-30', 'YYYY-MM-DD'), 87000);

INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (10, 'Logan', 'Gonzalez', 4, TO\_DATE ('2020-07-13', 'YYYY-MM-DD'), 54000);

INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (11, 'Zoe', 'Rodriguez', 2, TO\_DATE ('2021-10-10', 'YYYY-MM-DD'), 44000);

INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (12, 'Daniel', 'Hall', 6, TO\_DATE ('2023-06-07', 'YYYY-MM-DD'), 37000);

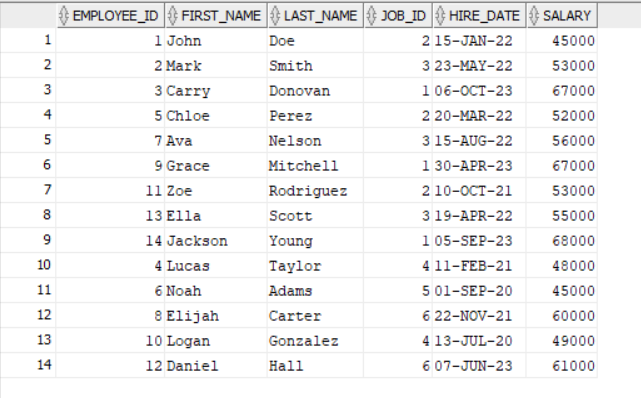
INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (13, 'Ella', 'Scott', 3, TO\_DATE ('2022-04-19', 'YYYY-MM-DD'), 60000);

INSERT INTO staff (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (14, 'Jackson', 'Young', 1, TO\_DATE ('2023-09-05', 'YYYY-MM-DD'), 85000);

SELECT \* FROM staff;



--Inserting values into Roles table

INSERT INTO roles (job\_id, job\_title, min\_salary, max\_salary, department\_id)

VALUES (2, 'Nurse', 25000, 50000,2);

INSERT INTO roles (job\_id, job\_title, min\_salary, max\_salary, department\_id)

VALUES (3, 'Doctor', 45000, 65000,3);

INSERT INTO roles (job\_id, job\_title, min\_salary, max\_salary, department\_id)

VALUES (1, 'Surgeon', 50000, 90000,1);

INSERT INTO roles (job\_id, job\_title, min\_salary, max\_salary, department\_id)

VALUES (4, 'Pharmacist', 40000, 70000, 7);

INSERT INTO roles (job\_id, job\_title, min\_salary, max\_salary, department\_id)

VALUES (5, 'Radiologist', 45000, 75000, 1);

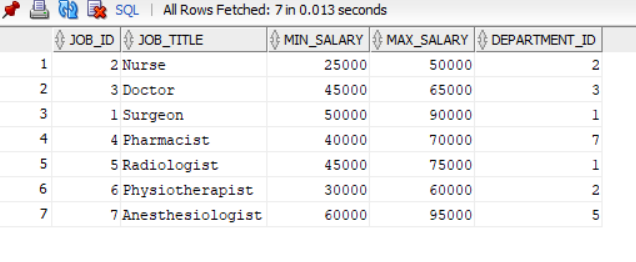
INSERT INTO roles (job\_id, job\_title, min\_salary, max\_salary, department\_id)

VALUES (6, 'Physiotherapist', 30000, 60000, 2);

INSERT INTO roles (job\_id, job\_title, min\_salary, max\_salary, department\_id)

VALUES (7, 'Anesthesiologist', 60000, 95000, 5);

SELECT \* FROM roles;



--Inserting values into Units table

INSERT INTO units (department\_id, department\_name, annual\_budget)

VALUES (1, 'Emergency', 2200000.00);

INSERT INTO units (department\_id, department\_name, annual\_budget)

VALUES (2, 'Orthopedics', 2000000);

INSERT INTO units (department\_id, department\_name, annual\_budget)

VALUES (3, 'Pediatrics', 2500000);

INSERT INTO units (department\_id, department\_name, annual\_budget)

VALUES (4, 'Radiology', 1800000);

INSERT INTO units (department\_id, department\_name, annual\_budget)

VALUES (5, 'Cardiology', 3000000);

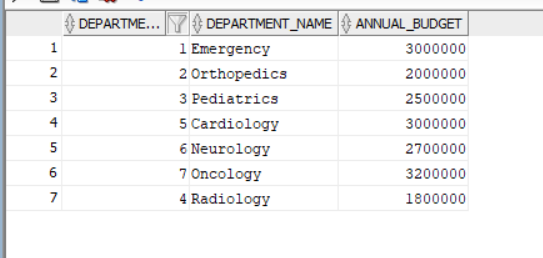
INSERT INTO units (department\_id, department\_name, annual\_budget)

VALUES (6, 'Neurology', 2700000);

INSERT INTO units (department\_id, department\_name, annual\_budget)

VALUES (7, 'Oncology', 3200000);

SELECT \* FROM units;



--Inserting values into Patients table

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

VALUES (1, 'Alice', 'Smith', TO\_DATE ('2023-03-11', 'YYYY-MM-DD'), TO\_DATE ('2023-03-15', 'YYYY-MM-DD'));

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

VALUES (2, 'Malcom', 'Davis', TO\_DATE ('2024-01-16', 'YYYY-MM-DD'), TO\_DATE ('2024-02-01', 'YYYY-MM-DD'));

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

VALUES (3, 'Ioana Alexandra', 'Ilinca', TO\_DATE ('2025-01-16', 'YYYY-MM-DD'), TO\_DATE ('2025-01-17', 'YYYY-MM-DD'));

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

VALUES (4, 'John', 'Raye', TO\_DATE ('2022-12-20', 'YYYY-MM-DD'), TO\_DATE ('2022-12-28', 'YYYY-MM-DD'));

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

VALUES (5, 'Emma', 'Johnson', TO\_DATE ('2023-04-01', 'YYYY-MM-DD'), TO\_DATE ('2023-04-10', 'YYYY-MM-DD'));

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

VALUES (6, 'Liam', 'Brown', TO\_DATE ('2023-06-15', 'YYYY-MM-DD'), TO\_DATE ('2023-06-20', 'YYYY-MM-DD'));

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

VALUES (7, 'Olivia', 'Garcia', TO\_DATE ('2023-07-10', 'YYYY-MM-DD'), TO\_DATE ('2023-07-18', 'YYYY-MM-DD'));

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

VALUES (8, 'Ethan', 'Martinez', TO\_DATE ('2023-08-22', 'YYYY-MM-DD'), TO\_DATE ('2023-08-25', 'YYYY-MM-DD'));

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

VALUES (9, 'Sophia', 'Lee', TO\_DATE ('2023-09-12', 'YYYY-MM-DD'), TO\_DATE ('2023-09-18', 'YYYY-MM-DD'));

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

VALUES (10, 'James', 'Hernandez', TO\_DATE ('2023-10-03', 'YYYY-MM-DD'), TO\_DATE ('2023-10-11', 'YYYY-MM-DD'));

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

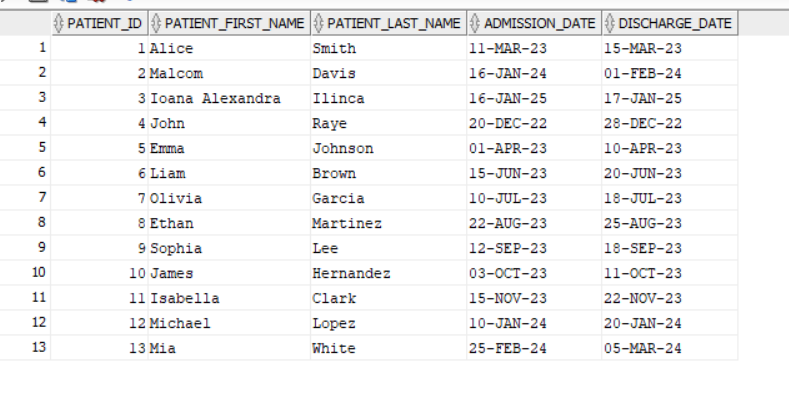
VALUES (11, 'Isabella', 'Clark', TO\_DATE ('2023-11-15', 'YYYY-MM-DD'), TO\_DATE ('2023-11-22', 'YYYY-MM-DD'));

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

VALUES (12, 'Michael', 'Lopez', TO\_DATE ('2024-01-10', 'YYYY-MM-DD'), TO\_DATE ('2024-01-20', 'YYYY-MM-DD'));

INSERT INTO patients (patient\_id, patient\_first\_name, patient\_last\_name, admission\_date, discharge\_date)

VALUES (13, 'Mia', 'White', TO\_DATE ('2024-02-25', 'YYYY-MM-DD'), TO\_DATE ('2024-03-05', 'YYYY-MM-DD'));

SELECT \* FROM patients;

--Inserting values into Supplies table

INSERT INTO supplies (supply\_id, supply\_name, quantity\_in\_stock)

VALUES (1, 'Syringe', 500);

INSERT INTO supplies (supply\_id, supply\_name, quantity\_in\_stock)

VALUES (2, 'IV', 630);

INSERT INTO supplies (supply\_id, supply\_name, quantity\_in\_stock)

VALUES (3, 'Scalpels', 350);

INSERT INTO supplies (supply\_id, supply\_name, quantity\_in\_stock)

VALUES (4, 'Bandages', 1200);

INSERT INTO supplies (supply\_id, supply\_name, quantity\_in\_stock)

VALUES (5, 'Gloves', 2000);

INSERT INTO supplies (supply\_id, supply\_name, quantity\_in\_stock)

VALUES (6, 'Face Masks', 1500);

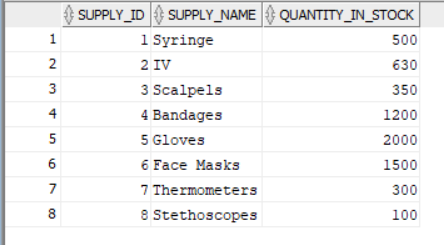
INSERT INTO supplies (supply\_id, supply\_name, quantity\_in\_stock)

VALUES (7, 'Thermometers', 300);

INSERT INTO supplies (supply\_id, supply\_name, quantity\_in\_stock)

VALUES (8, 'Stethoscopes', 100);

SELECT \* FROM supplies;



--Insert values into Supplies\_assigned values

INSERT INTO supplies\_assigned (request\_id, patient\_id, supply\_id, quantity\_used, date\_of\_usage)

VALUES (171,1, 2, 1, TO\_DATE('2023-11-12', 'YYYY-MM-DD'));

INSERT INTO supplies\_assigned (request\_id, patient\_id, supply\_id, quantity\_used, date\_of\_usage)

VALUES (1432,2, 3, 1, TO\_DATE ('2024-01-16', 'YYYY-MM-DD'));

INSERT INTO supplies\_assigned (request\_id, patient\_id, supply\_id, quantity\_used, date\_of\_usage)

VALUES (1123, 3, 1, 2, TO\_DATE('2025-01-16', 'YYYY-MM-DD'));

INSERT INTO supplies\_assigned (request\_id, patient\_id, supply\_id, quantity\_used, date\_of\_usage)

VALUES (1204, 1, 4, 3, TO\_DATE('2023-03-11', 'YYYY-MM-DD'));

INSERT INTO supplies\_assigned (request\_id, patient\_id, supply\_id, quantity\_used, date\_of\_usage)

VALUES (1547, 2, 5, 2, TO\_DATE('2024-01-20', 'YYYY-MM-DD'));

INSERT INTO supplies\_assigned (request\_id, patient\_id, supply\_id, quantity\_used, date\_of\_usage)

VALUES (1629, 3, 6, 1, TO\_DATE('2025-01-17', 'YYYY-MM-DD'));

INSERT INTO supplies\_assigned (request\_id, patient\_id, supply\_id, quantity\_used, date\_of\_usage)

VALUES (1780, 1, 3, 1, TO\_DATE('2023-03-13', 'YYYY-MM-DD'));

INSERT INTO supplies\_assigned (request\_id, patient\_id, supply\_id, quantity\_used, date\_of\_usage)

VALUES (1902, 2, 8, 1, TO\_DATE('2024-02-01', 'YYYY-MM-DD'));

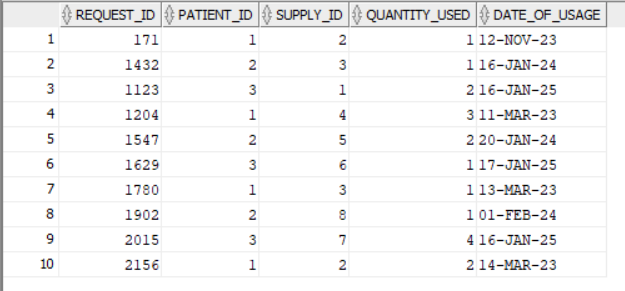
INSERT INTO supplies\_assigned (request\_id, patient\_id, supply\_id, quantity\_used, date\_of\_usage)

VALUES (2015, 3, 7, 4, TO\_DATE('2025-01-16', 'YYYY-MM-DD'));

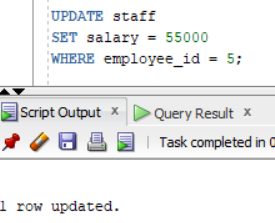
INSERT INTO supplies\_assigned (request\_id, patient\_id, supply\_id, quantity\_used, date\_of\_usage)

VALUES (2156, 1, 2, 2, TO\_DATE('2023-03-14', 'YYYY-MM-DD'));

SELECT \* FROM supplies\_assigned;



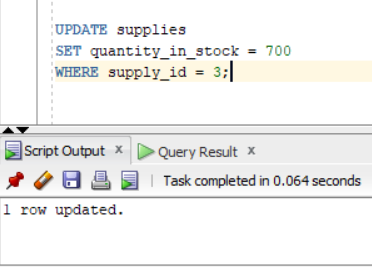
--

UPDATE staff

SET salary = 55000

WHERE employee\_id = 5;

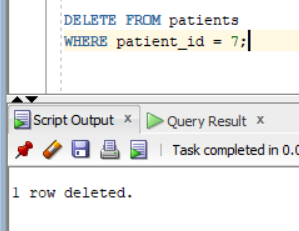
--

UPDATE supplies

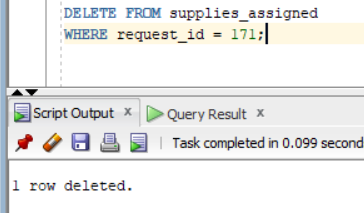
SET quantity\_in\_stock = 700

WHERE supply\_id = 3;

--

DELETE FROM patients

WHERE patient\_id = 7;

--

DELETE FROM supplies\_assigned

WHERE request\_id = 171;

--Insert a new employee into the staff table with job\_id = 7 if no such job already exists in the staff table, using max\_salary from the roles table

MERGE INTO staff target

USING (

SELECT job\_id, max\_salary

 FROM roles

WHERE job\_id = 7

) source

ON (target.job\_id = source.job\_id)

WHEN NOT MATCHED THEN

INSERT (employee\_id, first\_name, last\_name, job\_id, hire\_date, salary)

VALUES (15, 'Alice', 'Johnson', source.job\_id, DATE '2024-02-20', source.max\_salary);

**5. Diverse and relevant SELECT statements for the project theme**

**-- 1.Employees with salaries greater than 50,000**

SELECT \* FROM staff

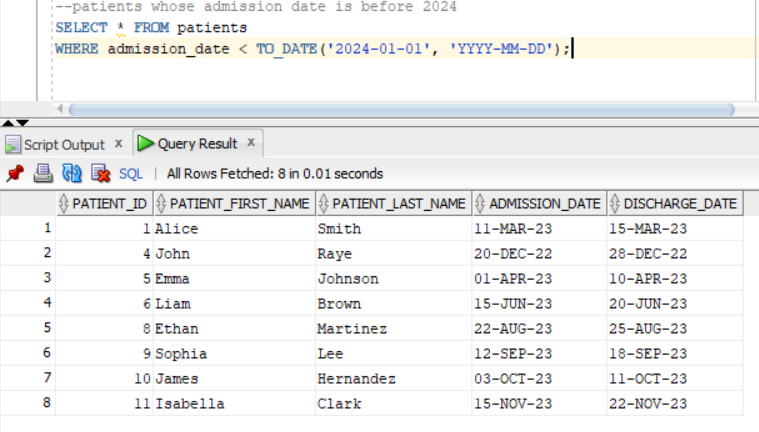
WHERE salary > 50000;



**--2.Patients whose admission date is before 2024**

SELECT \* FROM patients

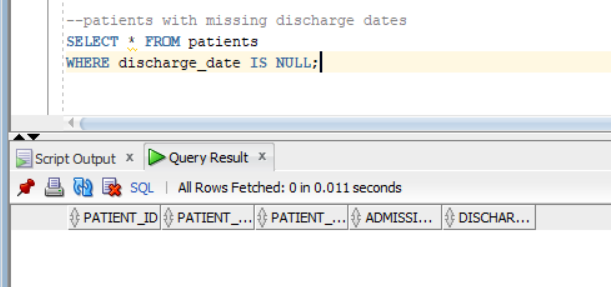
WHERE admission\_date < TO\_DATE('2024-01-01', 'YYYY-MM-DD');



**--3.Patients with missing discharge dates**

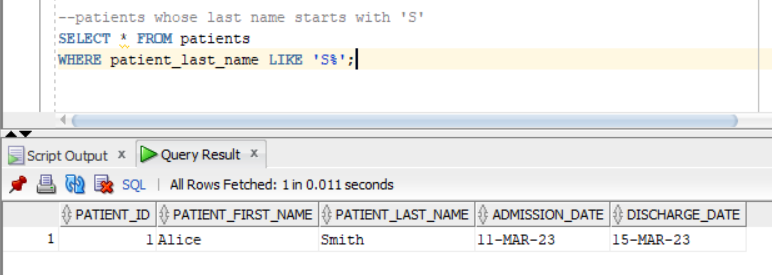
SELECT \* FROM patients

WHERE discharge\_date IS NULL;



**--4.Patients whose last name starts with 'S'**

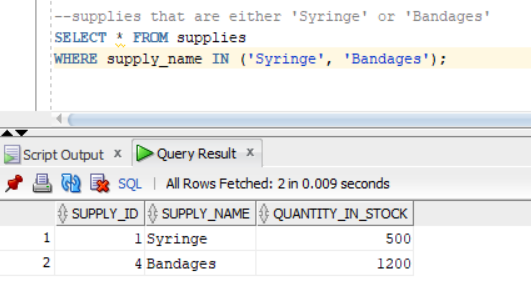
SELECT \* FROM patients

WHERE patient\_last\_name LIKE 'S%';

**--5.Supplies that are either 'Syringe' or 'Bandages'**

SELECT \* FROM supplies

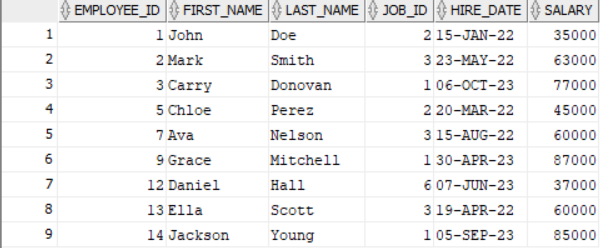
WHERE supply\_name IN ('Syringe', 'Bandages');



**--6.Employees who were hired between 2022 and 2023**

SELECT \* FROM staff

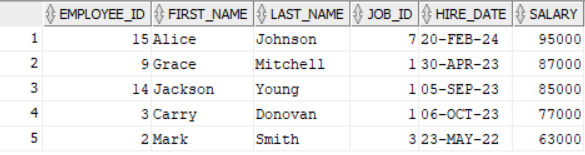
WHERE hire\_date BETWEEN TO\_DATE('2022-01-01', 'YYYY-MM-DD')

AND TO\_DATE('2023-12-31', 'YYYY-MM-DD');

**-- 7.Employees with salaries greater than any of the given values**

SELECT \* FROM staff

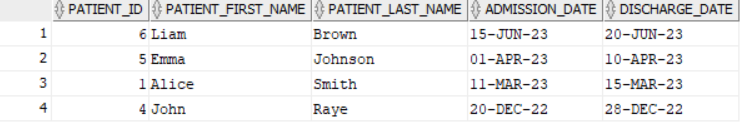
WHERE salary > ANY (SELECT salary FROM staff WHERE job\_id = 3);



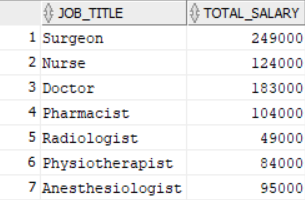
**-- 8.Patients with admission dates earlier than all patients with a specific id**

SELECT \* FROM patients

WHERE admission\_date < ALL (SELECT admission\_date FROM patients WHERE patient\_id = 8);



**--9.Calculate total salaries for each job title by summing salaries and grouping by job\_title**

SELECT r.job\_title,

SUM(s.salary) AS total\_salary

FROM roles r

JOIN staff s ON r.job\_id = s.job\_id

GROUP BY r.job\_title;

**--10.Employees and their roles**

SELECT s.first\_name, s.last\_name, r.job\_title

FROM staff s

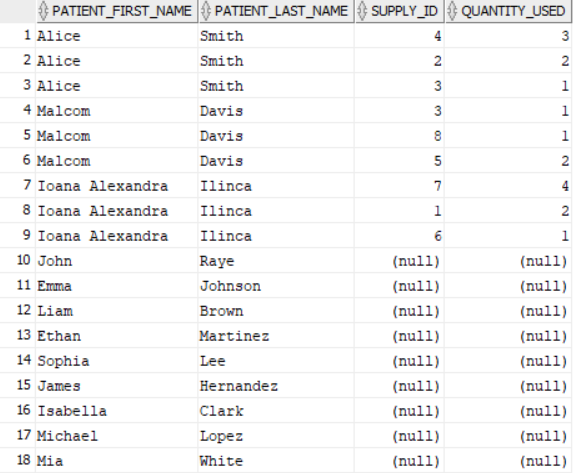
INNER JOIN roles r ON s.job\_id = r.job\_id;



**--11.All patients and their supply usage, including patients who did not use supplies**

SELECT p.patient\_first\_name, p.patient\_last\_name, sa.supply\_id, sa.quantity\_used

FROM patients p

LEFT JOIN supplies\_assigned sa ON p.patient\_id = sa.patient\_id;

**--12.Total quantity of syringes used**

SELECT SUM(quantity\_used) AS total\_syringes\_used

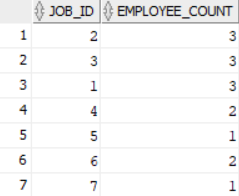
FROM supplies\_assigned

WHERE supply\_id = 1;

**--13.Number of employees in each role**

SELECT job\_id, COUNT(\*) AS employee\_count

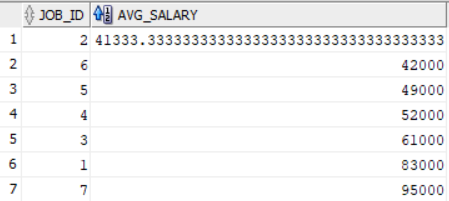
FROM staff

GROUP BY job\_id;

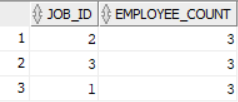
**--14.Average salary per role**

SELECT job\_id, AVG(salary) AS avg\_salary

FROM staff

GROUP BY job\_id;

**--15.Roles with more than 2employees**

SELECT job\_id, COUNT(\*) AS employee\_count

FROM staff

GROUP BY job\_id

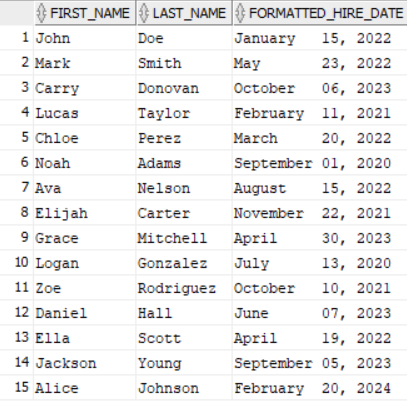
HAVING COUNT(\*) > 2;

**--16. Format the hire date of employees as 'Month Day, Year'**

SELECT first\_name, last\_name,

TO\_CHAR(hire\_date, 'Month DD, YYYY')

AS formatted\_hire\_date

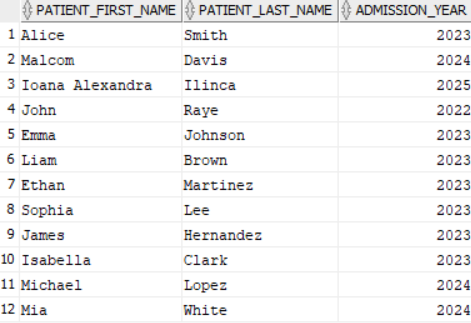
FROM staff;

**--17. Extract the year from the patient's admission date**

SELECT patient\_first\_name, patient\_last\_name,

EXTRACT(YEAR FROM admission\_date) AS admission\_year

FROM patients;

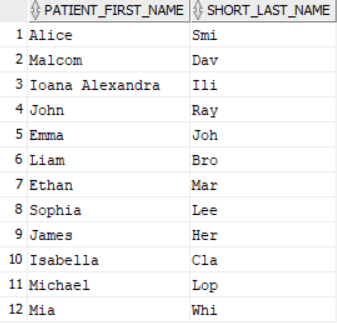


**--18. Get the first 3 characters of the patient's last name**

SELECT patient\_first\_name,

SUBSTR(patient\_last\_name, 1, 3)

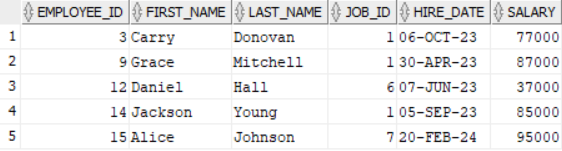
AS short\_last\_name

FROM patients;

**--19.Employees hired within the last 2 years**

SELECT \* FROM staff

WHERE hire\_date >= SYSDATE - 730;



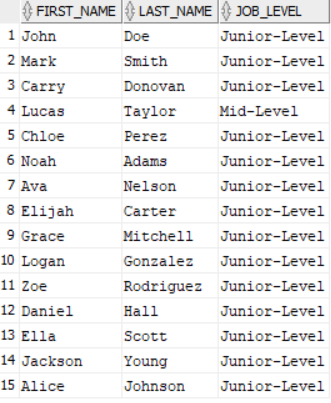
**-- 20.Display job title with custom labels based on salary**

SELECT first\_name, last\_name,

DECODE(salary, 50000, 'Mid-Level', 70000, 'Senior-Level', 'Junior-Level')

AS job\_level

FROM staff;



**--21. Display patient's status based on discharge date**

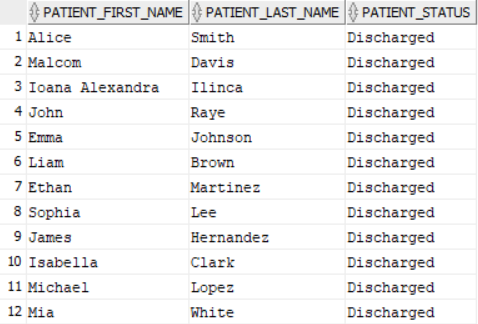
SELECT patient\_first\_name, patient\_last\_name,

CASE

WHEN discharge\_date IS NULL THEN 'Currently Admitted'

ELSE 'Discharged'

END AS patient\_status

FROM patients;

**-- 22.Show 'Assigned' if the supply is assigned to any patient**

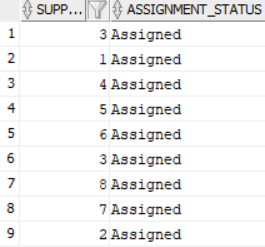
SELECT supply\_id,

CASE

WHEN quantity\_used IS NULL THEN 'Not Assigned'

ELSE 'Assigned'

END AS assignment\_status

FROM supplies\_assigned;

**--23.Unites employees with salaries over 70000 with**

**the supplies that have a stock over 100**

SELECT employee\_id, first\_name

FROM staff

WHERE salary > 70000

UNION

SELECT supply\_id, supply\_name

FROM supplies

WHERE quantity\_in\_stock > 100;

**--24.Employees who are earning a salary greater**

**than the average salary for their job role**

SELECT first\_name, last\_name, salary, job\_id

FROM staff

WHERE salary > (SELECT AVG(salary)

FROM staff WHERE job\_id = staff.job\_id);

**--25.An index on the patient's last name to improve query performance**

CREATE INDEX idx\_patient\_last\_name ON patients(patient\_last\_name);

****

**--26.A synonym for the patients table**

CREATE SYNONYM patient\_info FOR patients;



**--27.A sequence for generating unique request IDs**

CREATE SEQUENCE request\_id\_seq START WITH 1000 INCREMENT BY 1;



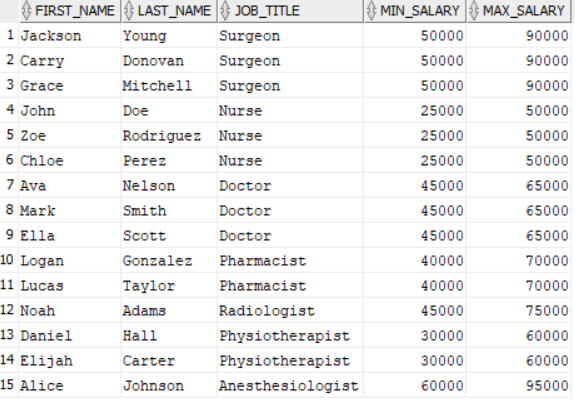
**--28.A view for employees and their role information**

CREATE VIEW employee\_roles AS

SELECT s.first\_name, s.last\_name, r.job\_title, r.min\_salary, r.max\_salary

FROM staff s

JOIN roles r ON s.job\_id = r.job\_id ;



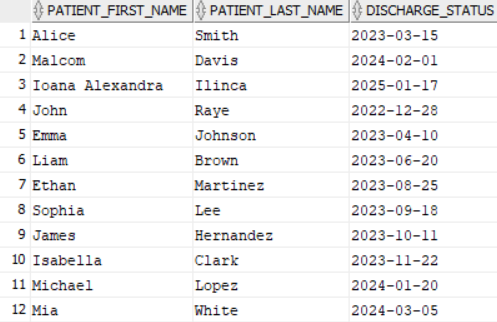
**-- 29.Retrieve patient names and discharge dates, displaying 'Still Admitted' if the discharge date is NULL**

SELECT patient\_first\_name,

patient\_last\_name,

NVL(TO\_CHAR(discharge\_date, 'YYYY-MM-DD'), 'Still Admitted')

AS discharge\_status

FROM patients;

**-- 30.The hierarchy of employees in the hospital by salary**

WITH ranked\_employees AS (

SELECT employee\_id, first\_name, last\_name, salary,

ROW\_NUMBER() OVER (ORDER BY salary DESC) AS rank

FROM staff

)

SELECT employee\_id, first\_name, last\_name, salary, LEVEL

FROM ranked\_employees

START WITH rank = 1

CONNECT BY PRIOR rank = rank - 1

ORDER BY LEVEL;

