Domain Driven Design

Based on the project or application the contents and the table is created.

Example- Educational Application

Student, Teacher, Course, Department, Subject

SQL queries

-- create

CREATE TABLE Students (

Student\_id INT PRIMARY KEY,

name varchar(100),

course varchar(100),

join\_date DATE

);

-- insert

INSERT INTO Students VALUES (1, 'Clark', 'Data Engineer','2025-09-02');

INSERT INTO Students VALUES (2, 'Dave', 'AI Engineer','2025-09-03');

INSERT INTO Students VALUES (3, 'Ava', 'Sales Intern','2025-10-15');

-- fetch

SELECT \* FROM Students;

SELECT name, course FROM students;

SELECT \* from Students WHERE course = "Data Engineer";

SELECT \* from students WHERE join\_date > '2025-09-03';

SELECT \* from students

WHERE course = "Data Engineer" AND join\_date>'2025-09-03';

SELECT \* from students

WHERE course in ('AI Engineer','Sales Intern')

SELECT \* from Students

WHERE join\_date BETWEEN '2025-09-02' AND '2025-10-15';

SELECT \* from students WHERE name LIKE 'A%';

SELECT \* FROM students WHERE name like '%a';

SELECT \* from students where name like '%it%';

UPDATE students

SET Course = "Advanced Data Engineering"

WHERE Student\_id = 1;

UPDATE students

SET join\_date = '2025-09-15'

WHERE name = 'Ava';

DELETE From Students where join\_date <'2025-10-15';

DELETE FROM Students where Student\_id =2;

Assignment Answers

CREATE TABLE Products (

product\_id INT PRIMARY KEY,

product\_name VARCHAR(100),

category VARCHAR(50),

price DECIMAL(10,2),

stock\_quantity INT,

added\_date DATE

);

INSERT INTO Products (product\_id, product\_name, category, price, stock\_quantity, added\_date)

VALUES

(1, 'Smartphone', 'Electronics', 1499.99, 25, '2024-12-15'),

(2, 'Sofa Set', 'Furniture', 1899.00, 8, '2023-06-20'),

(3, 'Study Table', 'Furniture', 850.00, 0, '2023-02-01'),

(4, 'Smartwatch', 'Electronics', 1999.00, 12, '2025-01-10'),

(5, 'Shoes Rack', 'Furniture', 499.00, 15, '2022-10-30');

SELECT \* FROM Products;

SELECT product\_name, price FROM Products;

SELECT \* FROM Products WHERE stock\_quantity < 10;

SELECT \* FROM Products WHERE price BETWEEN 500 AND 2000;

SELECT \* FROM Products WHERE added\_date > '2023-01-01';

SELECT \* FROM Products WHERE product\_name LIKE 'S%';

SELECT \* FROM Products WHERE category IN ('Electronics', 'Furniture');

UPDATE Products SET price = 1599.99 WHERE product\_id = 1;

UPDATE Products SET stock\_quantity = stock\_quantity + 5 WHERE category = 'Furniture';

DELETE FROM Products WHERE product\_id = 5;

DELETE FROM Products WHERE stock\_quantity = 0;

# Assignment

Create table departments(

dept\_id INT PRIMARY key,

dept\_name VARCHAR (100)

)

Insert Into departments values

(1,'Human Resources'),

(2,'Engineering'),

(3,'Marketing');

Create table employees(

emp\_id INT PRIMARY key,

emp\_name Varchar(100),

dept\_id INT,

salary INT,

foreign key (dept\_id) references departments(dept\_id)

)

Insert into employees

values

(101,'Amit Sharma',1,30000),

(102,'Neha Reddy',2,45000),

(103,'Faizan Ali',2,40000),

(104,'Divya Mehta',3,35000),

(105,'Ravi Verma',Null,20000);

select e.emp\_name, d.dept\_name from employees e

left join departments d on e.dept\_id = d.dept\_id;

select e.emp\_name from employees e where e.dept\_id Is Null;

select d.dept\_name, count(e.emp\_id) as Employee\_count from departments d

left join employees e on e.dept\_id = d.dept\_id

group by dept\_name;

SELECT d.dept\_id, d.dept\_name

FROM departments d

LEFT JOIN employees e ON d.dept\_id = e.dept\_id

WHERE e.emp\_id IS NULL;

SELECT e.emp\_name, d.dept\_name, e.salary

FROM employees e

JOIN departments d ON e.dept\_id = d.dept\_id

WHERE e.salary > 40000;