**Faculty of Computing**



##### Database labLab-5

Query

**Name:**

Ayesha Tassawar

**Sap Id:**

55421

**Lab Instructor:**

Fareeha Ashraf.

**Task 1:** Retrieve all columns from the "Users" table.  
  
CREATE DATABASE lab5;  
USE lab5;

CREATE TABLE user (

id INT,

username VARCHAR(10),

email VARCHAR(10),

password VARCHAR(10)

);

INSERT INTO user (id, username, email, password)

VALUES

(1, "ali", "ali@123", "ali\*123"),

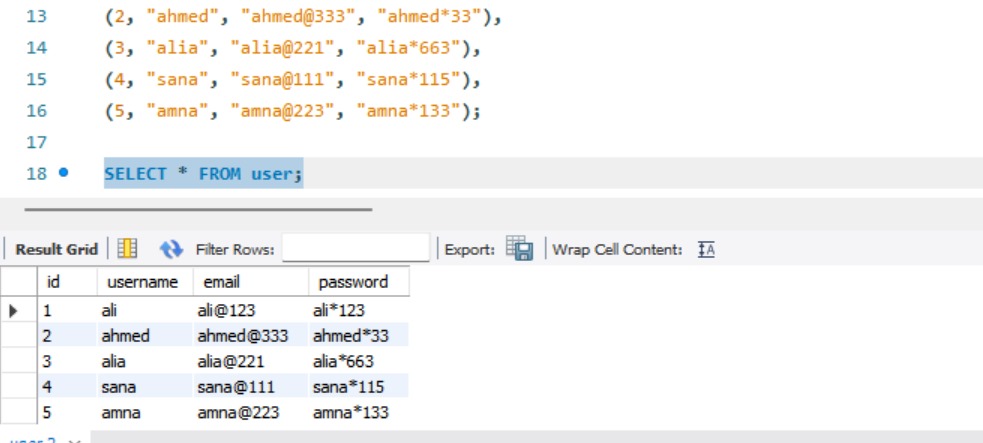
(2, "ahmed", "ahmed@333", "ahmed\*33"),

(3, "alia", "alia@221", "alia\*663"),

(4, "sana", "sana@111", "sana\*115"),

(5, "amna", "amna@223", "amna\*133");

SELECT \* FROM user;



**Task 2:** Retrieve all users whose usernames start with the letter 'A'.

CREATE DATABASE lab5;  
USE lab5;

CREATE TABLE user (

id INT,

username VARCHAR(10),

email VARCHAR(10),

password VARCHAR(10)

);

INSERT INTO user (id, username, email, password)

VALUES

(1, "ali", "ali@123", "ali\*123"),

(2, "ahmed", "ahmed@333", "ahmed\*33"),

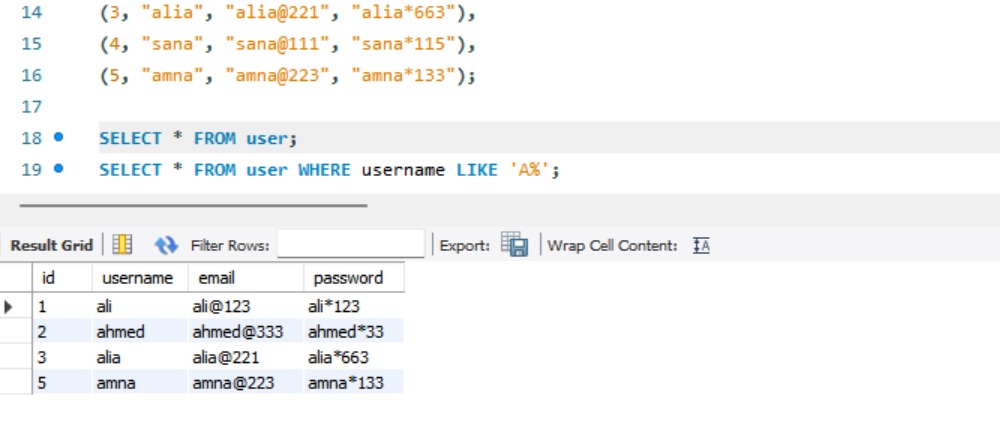
(3, "alia", "alia@221", "alia\*663"),

(4, "sana", "sana@111", "sana\*115"),

(5, "amna", "amna@223", "amna\*133");

SELECT \* FROM user;

SELECT \* FROM user WHERE username LIKE 'A%';



**Task 3:** Retrieve all products priced between $10 and $50 and whose names contain 'phone' or 'tablet'.

CREATE DATABASE shop;

USE shop;

CREATE TABLE products (

id INT,

name VARCHAR(50),

description VARCHAR(100),

price INT

);

INSERT INTO products (name, description, price)

VALUES

('Smartphone', 'Latest model smartphone', 45),

('Tablet X', 'High-resolution display tablet', 50),

('Headphones', 'Noise-canceling headphones', 30),

('Gaming Laptop', 'High-performance laptop', 1200),

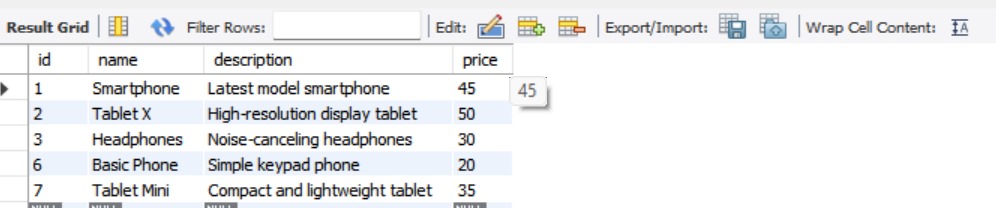
('Smartwatch', 'Fitness tracking smartwatch', 70),

('Basic Phone', 'Simple keypad phone', 20),

('Tablet Mini', 'Compact and lightweight tablet', 35);

SELECT \* FROM products

WHERE price BETWEEN 10 AND 50

AND (name LIKE '%phone%' OR name LIKE '%tablet%');  


**Task 4:** Retrieve all orders placed by user\_id 3 ordered by their creation date in descending order.CREATE DATABASE online\_store;

USE online\_store;

CREATE TABLE orders (

id INT AUTO\_INCREMENT PRIMARY KEY,

user\_id INT,

created\_at DATETIME,

total\_price INT

);

INSERT INTO orders (user\_id, created\_at, total\_price)

VALUES

(3, '2024-03-10 14:30:00', 500),

(2, '2024-02-15 09:45:00', 300),

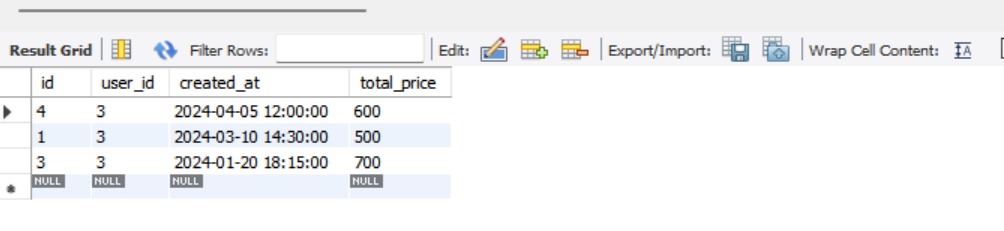
(3, '2024-01-20 18:15:00', 700),

(3, '2024-04-05 12:00:00', 600),

(1, '2024-05-01 10:00:00', 450);

SELECT \* FROM orders

WHERE user\_id = 3

ORDER BY created\_at DESC;  
  


**Task 5:** Find the minimum and maximum price of products in the "Products" table.

CREATE DATABASE inventory1;

USE inventory1;

CREATE TABLE products (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(50),

description VARCHAR(100),

price INT

);

INSERT INTO products (name, description, price)

VALUES

('Laptop', 'High-performance laptop', 1200),

('Smartphone', 'Latest model smartphone', 45),

('Tablet', 'Compact and lightweight tablet', 35),

('Headphones', 'Noise-canceling headphones', 30),

('Smartwatch', 'Fitness tracking smartwatch', 70),

('Basic Phone', 'Simple keypad phone', 20);

SELECT MIN(price) AS SmallestPrice

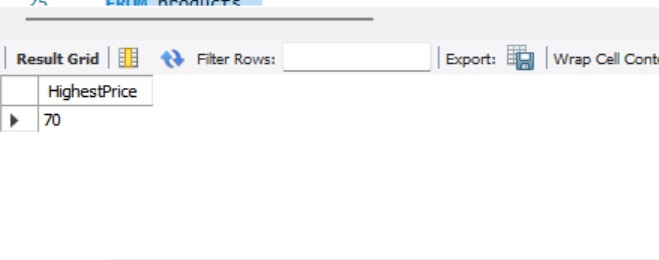
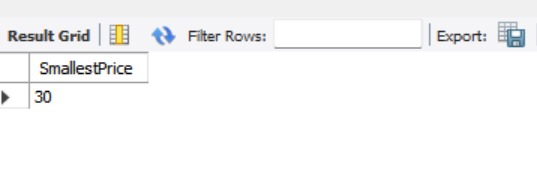
FROM products

WHERE price > 25;

SELECT MAX(price) AS HighestPrice

FROM products

WHERE price < 1000;

****

**Task 6:** Retrieve the first 10 orders placed.

CREATE DATABASE order\_management;

USE order\_management;

CREATE TABLE orders (

id INT ,

user\_id INT,

created\_at DATETIME,

total\_price INT

);

INSERT INTO orders (user\_id, created\_at, total\_price)

VALUES

(1, '2024-03-01 10:00:00', 150),

(2, '2024-03-02 11:30:00', 200),

(3, '2024-03-03 14:45:00', 300),

(4, '2024-03-04 09:15:00', 400),

(5, '2024-03-05 16:20:00', 250),

(6, '2024-03-06 18:50:00', 500),

(7, '2024-03-07 12:10:00', 100),

(8, '2024-03-08 08:40:00', 350),

(9, '2024-03-09 20:25:00', 450),

(10, '2024-03-10 13:55:00', 600),

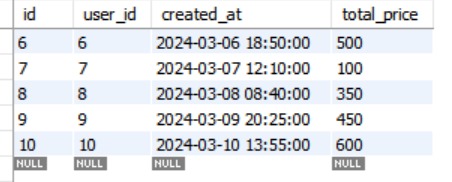
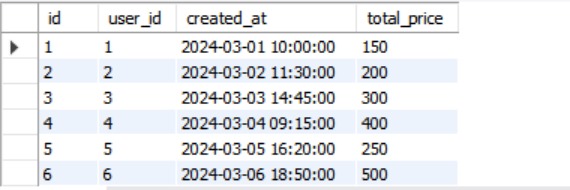
(11, '2024-03-11 15:30:00', 550),

(12, '2024-03-12 17:45:00', 700);

SELECT \* FROM orders

ORDER BY created\_at ASC

LIMIT 10;



**Task 7:** Count the total number of orders placed, calculate the average price of products, and sum up the total price of all orders.

CREATE DATABASE sales\_management;

USE sales\_management;

CREATE TABLE item (

id INT ,

product\_id INT,

created\_at DATETIME,

total\_price INT

);

INSERT INTO item (product\_id, created\_at, total\_price)

VALUES

(1, '2024-03-01 10:00:00', 150),

(2, '2024-03-02 11:30:00', 200),

(3, '2024-03-03 14:45:00', 300),

(4, '2024-03-04 09:15:00', 400),

(5, '2024-03-05 16:20:00', 250),

(6, '2024-03-06 18:50:00', 500),

(7, '2024-03-07 12:10:00', 100),

(8, '2024-03-08 08:40:00', 350),

(9, '2024-03-09 20:25:00', 450),

(10, '2024-03-10 13:55:00', 600);

SELECT COUNT(product\_id) AS Total\_Orders

FROM item

WHERE total\_price > 0;

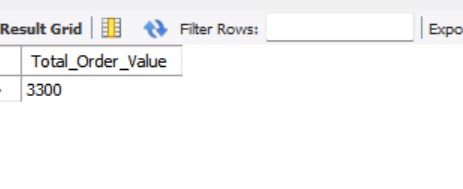
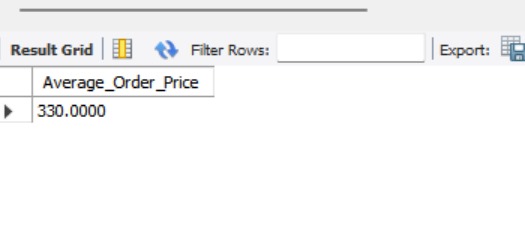
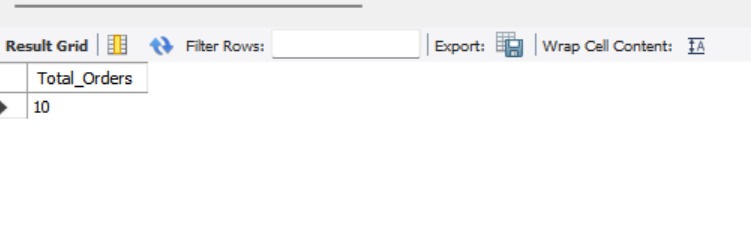
SELECT AVG(total\_price) AS Average\_Order\_Price

FROM item

WHERE total\_price > 0;

SELECT SUM(total\_price) AS Total\_Order\_Value

FROM item

WHERE total\_price > 0;  


**Task 8:** Retrieve all users whose emails end with '@example.com'.   
  
CREATE DATABASE user\_management;

USE user\_management;

CREATE TABLE user (

username VARCHAR(50),

email VARCHAR(100),

password VARCHAR(50)

);

INSERT INTO user (username, email, password)

VALUES

('Ali', 'ali@example.com', 'ali123'),

('Ahmed', 'ahmed@example.com', 'ahmed456'),

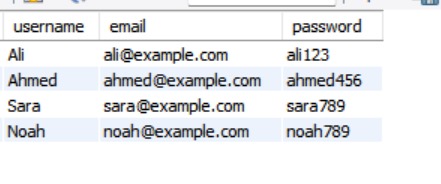
('Sara', 'sara@example.com', 'sara789'),

('John', 'john@sample.com', 'john123'),

('Emma', 'emma@demo.com', 'emma456'),

('Noah', 'noah@example.com', 'noah789');

SELECT \* FROM users

WHERE email LIKE ['%@example.com';](mailto:'%@example.com';)

**Task 9:** Retrieve all products with IDs 1, 5, and 9.

CREATE DATABASE product\_management;

USE product\_management;

CREATE TABLE product (

id INT ,

name VARCHAR(50),

description VARCHAR(100),

price INT

);

INSERT INTO product (id, name, description, price)

VALUES

(1, 'Laptop', 'High-performance laptop', 1200),

(2, 'Smartphone', 'Latest model smartphone', 800),

(3, 'Tablet', 'Compact and lightweight tablet', 500),

(4, 'Headphones', 'Noise-canceling headphones', 300),

(5, 'Smartwatch', 'Fitness tracking smartwatch', 200),

(6, 'Keyboard', 'Mechanical keyboard', 100),

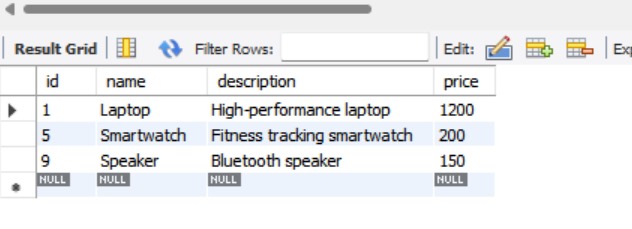
(7, 'Mouse', 'Wireless mouse', 50),

(8, 'Monitor', 'Full HD display', 250),

(9, 'Speaker', 'Bluetooth speaker', 150);

SELECT \* FROM products

WHERE id IN (1, 5, 9);



**Task 10**: Retrieve all orders placed between January 1, 2024, and February 1, 2024.

CREATE DATABASE order\_records;

USE order\_records;

CREATE TABLE order1 (

id INT ,

created\_at DATE,

total\_price INT

);

INSERT INTO order1 (id, created\_at, total\_price)

VALUES

(1, '2024-01-02', 150),

(2, '2024-01-10', 200),

(3, '2024-01-15', 300),

(4, '2024-01-20', 400),

(5, '2024-01-25', 250),

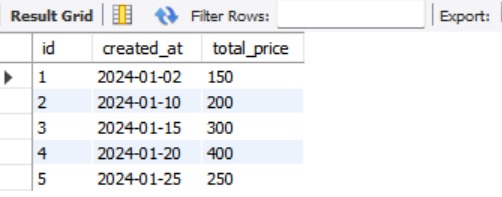
(6, '2024-02-02', 500),

(7, '2024-02-10', 100),

(8, '2024-02-15', 350);

SELECT \* FROM order1

WHERE created\_at BETWEEN '2024-01-01' AND '2024-02-01';



**Task 11**: Retrieve the total price of orders, aliased as "order\_total".

CREATE DATABASE order\_summary;

USE order\_summary;

CREATE TABLE orders (

id INT AUTO\_INCREMENT PRIMARY KEY,

user\_id INT,

created\_at DATE,

total\_price INT

);

INSERT INTO orders (user\_id, created\_at, total\_price)

VALUES

(1, '2024-03-01', 150),

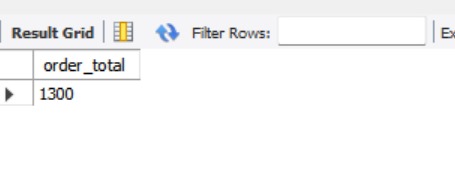
(2, '2024-03-02', 200),

(3, '2024-03-03', 300),

(4, '2024-03-04', 400),

(5, '2024-03-05', 250);

SELECT SUM(total\_price) AS order\_total FROM orders;



**Task 12:** Retrieve unique product names from the "Products" table.  
  
CREATE DATABASE product\_catalog;

USE product\_catalog;

CREATE TABLE products (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(50),

description VARCHAR(100),

price INT

);

INSERT INTO products (name, description, price)

VALUES

('Laptop', 'High-performance laptop', 1200),

('Smartphone', 'Latest model smartphone', 45),

('Tablet', 'Compact and lightweight tablet', 35),

('Headphones', 'Noise-canceling headphones', 30),

('Smartwatch', 'Fitness tracking smartwatch', 70),

('Smartphone', 'Latest model smartphone', 50),

('Tablet', 'Compact and lightweight tablet', 40);

SELECT DISTINCT name FROM products;

