```sql
CREATE DATABASE lab_5;
--DROP DATABASE lab_5;
USE lab_5;

-- TASK 1
CREATE TABLE Client(
	ClientID INT PRIMARY KEY IDENTITY(1,1),
	ClientName VARCHAR(100) NOT NULL,
	Address VARCHAR(200),
	Email VARCHAR(30) UNIQUE,
	Phone INT,
	Business VARCHAR(100) NOT NULL,
);

CREATE TABLE Project(
	ProjectID INT PRIMARY KEY IDENTITY(1,1),
	Description VARCHAR(200) NOT NULL,
	StartDate DATE,
	PlannedEndDate DATE,
	ActualEndDate DATE,
	Budget INT,
	ClientID INT,

	CHECK (ActualEndDate > PlannedEndDate),
	CHECK (Budget > 0),
	FOREIGN KEY(ClientID) REFERENCES Client(ClientID)
);

CREATE TABLE Department(
	DepartmentNo INT PRIMARY KEY IDENTITY(1,1),
	DepartmentName VARCHAR(100) NOT NULL

);
CREATE TABLE Employee(
	EmployeeNo INT PRIMARY KEY IDENTITY(1,1),
	EmployeeName VARCHAR(100) NOT NULL,
	Job VARCHAR(100),
	Salary INT,
	DepartmentNo INT,
	CHECK (Salary > 1700),
	FOREIGN KEY (DepartmentNo) REFERENCES Department(DepartmentNo)
);

CREATE TABLE EmployeeProjectTask(
	ProjectID INT,
	EmployeeNo INT,
	StartDate DATE,
	EndDate DATE,
	Task VARCHAR(100),
	Status VARCHAR(100),
	FOREIGN KEY(ProjectID) REFERENCES Project(ProjectID),
	FOREIGN KEY(EmployeeNo) REFERENCES Employee(EmployeeNo),
	PRIMARY KEY(ProjectID,EmployeeNo)

);

INSERT INTO Client(ClientName,Address,Email,Phone,Business) VALUES
```

```
      ('John Doe', '123 Main Street, Cityville', 'johndoe@example.com', 555-1234,
'Manufacturer'),
      ('Jane Smith', '456 Elm Avenue, Townsville', 'janesmith@example.com', 555-5678,
'Reseller');

INSERT INTO Project(Description,StartDate,PlannedEndDate,ActualEndDate,Budget,ClientID)
VALUES
      ('Accounting', '2023-01-01', '2023-03-31', '2023-04-10', 5000, 1),
      ('Payroll', '2023-02-15', '2023-04-30', '2023-05-15', 8000, 2);


INSERT INTO Department(DepartmentName) VALUES
      ('Accounting'),
      ('Sales');

INSERT INTO Employee(EmployeeName,Job,Salary,DepartmentNo) VALUES
      ('Michael Johnson', 'Sales Manager', 2500, 1),
      ('Emily Thompson', 'Software Engineer', 1900, 2);


INSERT INTO EmployeeProjectTask VALUES
      (1, 1, '2023-01-15', '2023-02-28', 'Designing', 'In Progress'),
      (2, 2, '2023-03-10', '2023-04-15', 'Coding', 'Complete');


-- TASK 2
SELECT employeeName
FROM employee
WHERE employeeName LIKE 'M%'


-- TASK 3
SELECT EmployeeNo,EmployeeName
FROM employee
WHERE LEN(employeeName) = (
    SELECT MAX(LEN(employeeName))
    FROM employee
);


-- TASK 4
SELECT d.DepartmentName,e.EmployeeName,e.Salary
FROM Employee e
INNER JOIN Department d
ON e.DepartmentNo = d.DepartmentNo
ORDER BY e.Salary DESC;

-- TASK 5
SELECT d.DepartmentNo, d.DepartmentName, COUNT(e.EmployeeNo) AS NumberOfEmployees
FROM Department d
LEFT JOIN Employee e
ON d.DepartmentNo = e.DepartmentNo
GROUP BY d.DepartmentNo, d.DepartmentName;

-- TASK 6

INSERT INTO Employee(EmployeeName,Job,Salary,DepartmentNo) VALUES
('Sarah Johnson', 'Marketing Specialist', 2200, 1),
```

```
('Robert Williams', 'Accountant', 1950, 2);

SELECT * FROM Employee;

SELECT d.DepartmentNo, d.DepartmentName,SUM(e.Salary) AS SumSalary
FROM Department d
INNER JOIN Employee e
ON d.DepartmentNo = e.DepartmentNo
GROUP BY d.DepartmentNo, d.DepartmentName

HAVING SUM(e.Salary) = (
    SELECT MAX(SumSalary)
    FROM (
        SELECT SUM(Salary) AS SumSalary
        FROM Employee
        GROUP BY DepartmentNo
    ) AS DepartmentSum
);
```