**Question:**

1. Create a database named **company_db**.
2. Create a table named **employees** with the following attributes and suitable data types:
    - emp_id (auto increment, primary key)
    - emp_name
    - department
    - salary
    - manager_id
    - city
3. Insert the following records into the Employee table **(do not insert emp_id values explicitly)**:
    i. Noor, HR, 50000, 3, Beirut
    ii. Hassan, IT, 60000, 4, Baalback
    iii. Hady, HR, 70000, NULL, Beirut
    iv. Ali, IT, 80000, NULL, Tripoli
    v. Mouhammad, Sales, 45000, 6, Saida
    vi. Zayn, Sales, 65000, NULL, Beqaa

**Write SQL queries for the following:**

a. Display all **distinct departments** from the employees table.

b. Display employee names and salaries where salary is greater than 50,000 **AND** the employee belongs to either the IT or HR department.

c. Display the names of employees whose names **start with the letter 'H'**.

d. Display the **top 3 highest paid employees**.

e. Display the **total salary paid department-wise**.

f. Display the **total salary paid department-wise along with the grand total**.

g. Display employees working in **Beirut** and employees working in **Beqaa** using the **UNION** operator.

h. Using a **SELF JOIN**, display employee names along with their respective manager names.

i. Display the names of employees whose salary is **greater than the average salary** of all employees.

---

j. Create a **VIEW** named *IT_Employees* that displays the employee name, department, and salary of employees working in the IT department.

k. Write an SQL query to display all records from the **IT_Employees** view.

l. Create a **stored procedure** named *GetAllEmployees* that displays all records from the Employee table.
*(Procedure without parameters)*

m. Create a **stored procedure** named *GetEmployeesByDepartment* that accepts a department name as input and displays employees belonging to that department.
*(Procedure with one input parameter)*

n. Create a **stored procedure** named *GetEmployeesBySalary* that accepts a salary amount as input and displays employees whose salary is greater than the given value.
*(Procedure with one input parameter)*

---