Database Systems LAB – BSDSF23
(Morning/Afternoon)
Lab 06 – 11-04-2025

Copy the portable DB Browser for SQLite from server (\\printsrv) and start it.

Emp Table:

ENO	Ename	Title
Filter	Filter	Filter
101	Alice	Manager
102	Bob	Engineer
103	Charlie	Analyst
104	David	Engineer
105	Eva	Manager
106	Frank	Developer
107	Grace	HR Specialist
108	Henry	Data Scientist
109	Ivy	Analyst
110	Jack	Developer

ASG Table:

ENO	PNO	RESP	DUR
Filter	Filter	Filter	Filter
101	1	Lead Research	12
102	2	Backend Developer	8
103	3	Data Analysis	10
104	2	UI Developer	6
105	4	HR Automation	5
106	5	Security Engineer	9
107	4	HR Policy Developer	7
108	6	Cloud Architect	11
109	3	Blockchain Specialist	8
110	2	Frontend Developer	6

Project Table:

PNO	PNAME	BUDGET
Filter	Filter	Filter
1	AI Research	50000
2	Web Development	30000
3	Finance App	40000
4	HR System	20000
5	Cybersecurity	60000
6	Cloud Migration	70000

Pay Table:

TITLE	SAL
Filter	Filter
Manager	8000
Engineer	6000
Analyst	5000
Developer	6500
HR Specialist	5500
Data Scientist	7500

5. List all project numbers (PNO) along with the total number of employees assigned, and the average duration spent on each project.

Only include those projects that:

1. Have more than one employee assigned
2. Have an average assignment duration greater than 7 month

6. Show each job title along with:

1. The number of employees having that title
2. The maximum salary for that title

Only include titles that:

3. Have at least 2 employees
4. Have a maximum salary greater than 6000

4. Display the names and titles of employees who are working on projects where the number of assigned employees is greater than 1.

3. List the job titles and the total number of employees for each title, only if the average salary of that title is greater than the overall average salary of all employees working on projects with budget less than 60000.

2. List titles and the number of employees holding each title, but only show those titles for which the total salary of employees is greater than the average salary of titles that don't contain the word 'Engineer'.

1. Display project names and the number of employees assigned to each project, only for those projects where the total assigned duration is greater than the average duration of projects that have a budget greater than or equal to 40000 and project name does not start with 'C'.

7. Create a view named Project_Employee_Stats that displays the following details for each project:

1. Project Name (PNAME)
2. Total Number of Employees Assigned (using COUNT)
3. Average Duration of Assignments

Conditions:

1. Only include projects with more than 1 employee assigned.
2. The average duration of assignments must be greater than 6 months.
3. After creating the view, you should select from the view and display the results:
4. Order the results first by Total_Employees in descending order.
5. Then by Avg_Duration in ascending order.

8. Create a view named High_Value_Assignments that stores the details of employees assigned to high-budget projects (projects with budget greater than the average project budget).

The view should include:

1. E.ENO, E.Ename, E.Title
2. P.PNAME, P.BUDGET
3. A.RESP, A.DUR

The filter condition for selecting projects should be based on a subquery that calculates the average budget of all projects.

-- Create the Students table
CREATE TABLE Students (
    StudentID INTEGER PRIMARY KEY,
    FirstName TEXT,
    LastName TEXT,
    Age INTEGER,
    Grade TEXT
 );
	INSERT INTO Students (StudentID, FirstName, LastName, Age, Grade) VALUES
	(1, 'Ali', 'Hasan', 20, 'A'),
	(2, 'Sara', 'Khan', 21, 'B'),
	(3, 'Ahmed', 'Raza', 19, 'C'),
	(4, 'Fatima', 'Noor', 22, 'A'),
	(5, 'Usman', 'Malik', 20, 'B');

ALTER TABLE Tasks: (use above table script)
1: Add a column Email of type TEXT
2: Rename column Age to StudentAge
3: rename students table to EnrolledStudents
4: Delete all data from the table
5: Remove the Students table

Database Systems LAB – BSDSF23
(Morning/Afternoon)
Lab 06 – 11-04-2025

CREATE TABLE Employees (
    EmpID INTEGER PRIMARY KEY,
    Name TEXT NOT NULL,
    Position TEXT,
    ManagerID INTEGER,
    FOREIGN KEY (ManagerID)
REFERENCES Employees(EmpID)
);
	INSERT INTO Employees (EmpID, Name, Position, ManagerID) VALUES
	(1, 'Sara Khan', 'CTO', NULL), -- Top-level manager
	(2, 'Ahmed Raza', 'CFO', NULL), -- Top-level manager
	(3, 'Fatima Noor', 'Dev Manager', 1), -- Reports to CTO
	(4, 'Usman Malik', 'Developer', 3), -- Reports to Dev Manager
	(5, 'Hassan Iqbal', 'Accountant', 2); -- Reports to CFO

Use above Table Script :

Write query to list all employees with their manager names using a self join.