Assignment: Employee and Department Management System

Duration: ~1 hour
Skills Covered: SQLite, Python, SQL Queries, SQLAlchemy, Foreign Keys, CRUD Operations, Pandas, Visualization (matplotlib)

Objective

You will build a Python-based Employee and Department Management System using SQLite. This system should allow users to:

Add new departments and employees.

View all employees with their department names.

Update employee salary.

Delete an employee.

Display a department-wise employee count (with a visualization).

Provided Tables

Your SQLite database should have the following tables:

1. Departments Table

id (PK)	name (TEXT, UNIQUE)
1	HR
2	IT
3	Finance
4	Marketing

2. Employees Table

id (PK)	name (TEXT)	age (INTEGER)	department_id (FK)	salary (REAL)
1	Alice	30	1	50000
2	Bob	25	2	60000
3	Charlie	35	3	70000
4	David	28	4	55000

Assignment Tasks

1. Set Up the Database (10 min)

Connect to an SQLite database company.db.
Create the departments table.
Create the employees table with a foreign key reference to departments.id.

2. Implement CRUD Operations (30 min)

Add New Department

Allow the user to add a new department (if it doesn't already exist).

Add New Employee

Allow the user to enter:
Name, Age, Department (as text, map to department_id), Salary
Store the employee in the database.

Update Employee Salary

Prompt the user to select an employee by ID and enter a new salary.
Update the salary in the database.

Delete an Employee

Prompt the user for an employee ID and delete the employee from the database.

View All Employees with Department Names

Retrieve and display all employees with department names (using JOIN).

3. Perform Data Analysis & Visualization (20 min)

Count Employees by Department

Query how many employees are in each department.
Display the count in a table format.

Plot a Bar Chart

Use Matplotlib to generate a bar chart of department-wise employee count.

Example User Flow

Welcome to Employee Management System
1. Add New Department
2. Add New Employee
3. Update Employee Salary
4. Delete Employee
5. View All Employees
6. Show Employee Count by Department
7. Exit

Enter your choice: 2
Enter Employee Name: Emma
Enter Age: 27
Enter Department: IT
Enter Salary: 62000
Employee added successfully!

Enter your choice: 5
(1, 'Alice', 30, 'HR', 50000)
(2, 'Bob', 25, 'IT', 60000)
(3, 'Charlie', 35, 'Finance', 70000)
(4, 'David', 28, 'Marketing', 55000)
(5, 'Emma', 27, 'IT', 62000)

Enter your choice: 6
Department | Employee Count
HR | 1
IT | 2
Finance | 1
Marketing | 1
Bar Chart Generated!

Submission Requirements

Submit a single Python script (assignment.py).
Ensure the script creates the tables if they don’t exist.
The script should allow user interaction via input.
The script should include at least one JOIN query.
The bar chart should be generated dynamically from the database.

Bonus Challenges

Add a search feature: Allow users to search employees by name.
Export the employee data to a CSV file.
Use SQLAlchemy ORM instead of sqlite3.