Problem Statements for micra project of Database Engineering

Expected SQL Queries (for any chosen problem)

1. **Basic Query**: Insert/Update/Delete records.
2. **SELECT with WHERE**: Filter data (e.g., "List all patients aged > 30").
3. **Aggregate Function**: Compute averages, sums, or counts (e.g., "Total sales per product").
4. **Set Operation**: UNION/INTERSECT/EXCEPT (e.g., "Customers who bought both Product A and B").
5. **Natural Join**: Retrieve related data (e.g., "Show orders with customer details").

---

1. Hospital Management System

Design a database for a hospital to manage patients, doctors, appointments, and treatments. Include entities like **Patient, Doctor, Appointment, Prescription, and Department**.

2. Library Management System

Create a database for a library tracking books, members, loans, and publishers. Entities: **Book, Member, Loan, Publisher, Author**.

3. University Database

Model a university system with **Students, Courses, Professors, Enrollments, and Departments**. Track grades and course prerequisites.

4. E-Commerce Platform

Design a database for an online store with **Customers, Products, Orders, Payments, and Categories**.

5. Employee Payroll System

Develop a system for managing **Employees, Departments, Salaries, Attendance, and Projects**.

6. Airline Reservation System

Model an airline database with **Flights, Passengers, Tickets, Airports, and Crew**.

7. Hotel Booking System

Design a system for **Hotels, Rooms, Guests, Reservations, and Payments**.

8. Social Media Platform

Create a database for **Users, Posts, Comments, Likes, and Followers**.

9. Inventory Management

Track **Products, Suppliers, Orders, Warehouses, and Stock** for a retail business.

10. Banking System

Model a bank database with **Customers, Accounts, Transactions, Loans, and Branches**.

11. Movie Ticket Booking

Design a system for **Movies, Theaters, Shows, Bookings, and Customers**.

12. Gym Management System

Track **Members, Trainers, Workout Plans, Payments, and Equipment**.

13. School Management System

Model a school database with **Students, Teachers, Classes, Subjects, and Exams**.

14. Restaurant Management

Design a system for **Menu Items, Orders, Customers, Tables, and Chefs**.

15. Car Rental System

Track **Cars, Customers, Rentals, Payments, and Maintenance**.

16. Online Learning Platform

Model **Courses, Students, Instructors, Enrollments, and Quizzes**.

17. Pharmacy Management

Design a database for **Medicines, Customers, Sales, Suppliers, and Prescriptions**.

18. Event Management System

Track **Events, Attendees, Venues, Sponsors, and Tickets**.

19. Real Estate Agency

Model **Properties, Agents, Buyers, Sales, and Appointments**.

20. Courier Service System

Design a database for **Parcels, Customers, Deliveries, Branches, and Payments**.

21. Fitness Tracker App

Track **Users, Workouts, Meals, Goals, and Progress**.

22. Job Portal System

Model **Job Seekers, Employers, Job Listings, Applications, and Skills**.

23. Supermarket Billing System

Design a database for **Products, Customers, Bills, Suppliers, and Discounts**.

24. Music Streaming Service

Model **Songs, Artists, Albums, Users, and Playlists**.

25. Vehicle Service Center

Track **Customers, Vehicles, Services, Mechanics, and Invoices**.

Problem Statement No.	Roll Numbers Assigned
1	1,2,3
2	4,5,6
3	7,8,9
4	10,11,12
5	13,14,15
6	16,17,18
7	19,20,21
8	22,23,24
9	25,26,27,28
10	29,30,31
11	32,33,34
12	35,36,37
13	38,39,40
14	41,42,43
15	44,45,46
16	47,48,49
17	50,51,52
18	53,54,55
19	56,57,58
20	59,60,61
21	62,63,64
22	65,66,67
23	68,69,70
24	71,72,73
25	--

Database Engineering Micro-Project Report Guidelines

1. Title Page

- Project Title
- Course Name & Code
- Student Name(s) & ID(s)
- Instructor Name
- Submission Date

2. Introduction

- Brief description of the problem statement.
- Objectives of the project.
- Scope (e.g., entities, relationships, key functionalities).

3. ER Diagram

- **Visual representation** of the ER model (use tools like Lucidchart, draw.io, or hand-drawn scans).
- Clearly label:
  - **Entities** (e.g., Patient, Doctor)
  - **Attributes** (underline primary keys, italicize foreign keys)
  - **Relationships** (cardinality: 1:1, 1:N, M:N)
- **Assumptions** (if any) about the design.

4. Relational Schema

- List all tables derived from the ER diagram in **textual format**.
- For each table, include:
  - **Table name**
  - **Attributes** (with data types, e.g., `VARCHAR(50)`, `INT`)

- **Primary key (PK)** and **foreign keys (FK)**
- Example:

```
plaintext

Copy

PATIENT (PatientID [PK], Name, Age, Gender, DoctorID [FK])
```

5. SQL Queries

- Provide **5 SQL queries** with:
- **Query statement** (formatted for readability).
- **Purpose** (what the query achieves).
- **Sample output** (screenshot or table format).

Categories:

1. **Basic Query**: Insert/Update/Delete.
2. **SELECT with WHERE**: Filter data (e.g., "List doctors in Cardiology").
3. **Aggregate Function**: Use `COUNT`, `SUM`, `AVG`, etc. (e.g., "Average salary by department").
4. **Set Operation**: `UNION`, `INTERSECT`, or `MINUS` (e.g., "Customers who ordered both Pizza and Pasta").
5. **Natural Join**: Combine related tables (e.g., "Show students with their enrolled courses").

6. Screenshots & Results

- Screenshots of:
- Executed queries (with output).
- Database schema (tables created in MySQL/PostgreSQL/etc.).

7. Challenges & Learnings

- Difficulties faced during design/implementation.
- Key takeaways (e.g., normalization, query optimization).

8. Conclusion

- Summary of the project.
- Real-world applications of the database.

9. References

- Tools used (e.g., MySQL, pgAdmin).
- Any external resources (books, articles).

Formatting Rules

- Use **12pt Times New Roman/Arial**, 1.5 line spacing.
- Include **page numbers**.
- Label all figures/tables (e.g., "Figure 1: ER Diagram").
- Submit as **PDF** (code/output can be in appendices if lengthy).