General Assignment Description

Design a relational database based on **one of the proposed subject areas below**. The project should demonstrate your ability to analyze a domain, identify entities, relationships, and constraints, and represent them in conceptual (ER), logical (relational) and physical forms.

Subject Areas (choose one)

1. Hospital Management System

Information about patients, doctors, hospital departments, wards, and treatments must be stored in the database. Each patient can be treated by several doctors, and each doctor may work in one or more departments. Patients are assigned to wards during their hospitalization. Records should include the admission and discharge dates, diagnosis, and prescribed medicines. The system should also keep information about medical procedures, their cost, and the personnel responsible for them.

2. Cinema Ticket Booking System

A cinema company manages several theaters, each with a number of halls. Films are shown in specific halls at scheduled times. The database should record details about movies, showtimes, ticket sales, and visitors. Each visitor may purchase multiple tickets for different shows, and each film can be screened in various theaters. The system should also include information about genres, seating arrangements, and ticket prices.

3. Restaurant Reservation System

A restaurant keeps records of its customers, tables, reservations, and menu items. Customers can make reservations for specific dates and times and may order several dishes during their visit. The database should store data about the dishes offered, their prices, and categories. It should also include details about the employees who serve customers and process orders.

4. Music Festival Organization

A company organizes music festivals that involve multiple artists, venues, sponsors, and events. Each festival takes place over several days and includes performances on different stages. The database should contain information about artists, bands, the music genres they perform, the schedule of performances, and ticket sales. Sponsors may support one or several festivals, and each event has assigned staff members and technical equipment.

5. Car Rental Service

A car rental company stores information about customers, cars, rental branches, and rental contracts. Each car has its own model, registration number, category, and current status (available, rented, maintenance). Customers may rent several cars over time, and each rental has a start date, return date, and total payment. The database should also record details about branches, employees, and payments made for rentals.

6. Travel Agency

A travel agency organizes tours, excursions, and hotel stays. Each tour includes a set of destinations, one or more hotels, and optional excursions. The agency cooperates with guides and partners in different countries. Customers can book various tours and specify the number of travelers. The database should contain details about clients, tours, hotels, guides, and bookings, as well as information about prices and travel dates.

7. Fitness Club Management

A fitness club keeps records of its members, trainers, training programs, and sessions. Members can enroll in several programs, and trainers may lead multiple sessions per week. Each program has its own schedule, type (group or individual), and difficulty level. The database should store membership details, payments, attendance records, and trainer qualifications.

Requirements

Logical Data Modelling Requirements and Constraints

The logical schema must represent the blueprint of the database structure from a data-modeling point of view.

- At least 7 tables.
- The database must be designed with normalization up to at least the Third Normal Form (3NF).
- At least one many-to-many (M:N) relationship.
- Each table must have a primary key (PK).
- All relationships must have foreign keys (FK).
- Every table and field name should be descriptive and meaningful.
- Include hierarchical or recursive relationship (e.g. manager-employee) (if it makes sense in your design).
- Format: .pdf, .png, .jpeg

SQL Script Requirements

The SQL script defines the database schema, including the creation of tables, relationships, and constraints.

- Each table must have a primary key (PK).
- All relationships must have foreign keys (FK).
- Use the following constraints where appropriate: NOT NULL, UNIQUE, CHECK.
- Use DEFAULT where appropriate.
- Use appropriate data types for each attribute (numeric, text, date, boolean, etc.).
- Format: .sql or .txt

Your final SQL script must be clear, executable, and rerunnable.

- Executable
  - The script should run without errors in **PostgreSQL**.
  - It must successfully create all tables, relationships, and constraints in the correct order.
  - Foreign key dependencies should be respected.
- Rerunnable
  - The script should be safe to execute multiple times without causing errors.
  - The database can be rebuilt from scratch by simply re-executing the same script.
- Structured and Documented
  - Separate sections logically (e.g. -- Create tables, -- Add foreign keys, -- Constraints).
  - Use consistent indentation and naming conventions.

Example:

```
-- Table of students
CREATE TABLE student (
   student_id SERIAL PRIMARY KEY,
   name VARCHAR(100) NOT NULL,
   email VARCHAR(100) UNIQUE,
   phone VARCHAR(20)
);
```

Deliverables

Deliverables and Submission Guidelines

- All materials should be stored and submitted via **Git repository** (e.g. GitHub, GitLab).
- The repository should include all required artifacts:
  - Logical Schema – format: .pdf, .png, .jpeg
  - SQL Script – format: .sql or .txt

Submission in Teams

- Each student must submit a link to their Git repository containing all deliverables.
- The final version should be committed before the submission deadline.