Lab 1 Database Design using Normalization

Objective: Design a normalized database schema for an event management system.

Description: You have been tasked with designing the database schema for an event management system. The system will be used to manage events and registrations. Each event can have multiple registrations.

Requirements:

1. Create a normalized database schema consisting of two tables: Events and Registrations.
2. Define appropriate primary keys, foreign keys, and relationships between the tables.
3. Include necessary attributes in each table to store relevant information.
4. Ensure that the schema is normalized to eliminate redundancy and maintain data integrity.
5. Document your database schema, explaining the purpose of each table and its attributes.

Submission Guidelines:

- Submit a document containing the database schema design and explanations for each table and its attributes.
- Include SQL scripts to create the tables with constraints (e.g., primary keys, foreign keys).

Note: You can use tools like Microsoft SQL Server Management Studio or MySQL Workbench to design and generate SQL scripts for your database schema.

Database Schema Design for Event Registration System

This project has two tables which are the following:

1. Events
2. Registrations

The purpose of each table is listed below:

1. Events Table purpose: This table will store information regarding each specific event.
2. Registrations Table purpose: This table will store information regarding each event registration (information about the attendee).

The fields of the Events table are the following:

- EventID as the Primary Key, uniquely identifying each event. This field will be auto implemented.
- Name: This field will contain the name of the event. Does not allow null values.
- Date: This field will contain the date of the event. Does not allow null values.
- Location: This field will the location of the event. Does not allow null values.
- Description: This field will contain the description of the event. The data type is set to TEXT to allow as much information as possible.

The fields of the Registrations table are the following:

- RegistrationID as the Primary Key, uniquely identifying each registration record. This field will also be auto implemented.
- EventID as the Foreign Key with references to the EventID of the Event table.
- AttendeeFirstName: First name of the attendee who registered. Does not allow null values.
- AttendeeLastName: Last name of the attendee who registered. Does not allow null values.
- Email: Email address of the attendee. Does not allow null values.
- RegistrationDate: Date and time when the registration was made. Does not allow null values.

The relationship between the tables is the following:

- The Events table has a one-to-many relationship with the Registrations table because each event can have multiple registrations, but one registration can only belong to one event.

Database Schema

The following schema is normalized, eliminating redundancy. Both the Events and Registrations tables contain information that only corresponds to each model.

```sql
-- Create the Event Registration System database
CREATE DATABASE EventRegistrationSystem

-- Create Events table
CREATE TABLE Events (
    EventID INT PRIMARY KEY IDENTITY(1, 1),
    Name VARCHAR(50) NOT NULL,
    Date DATE NOT NULL,
    Location VARCHAR(50) NOT NULL,
    Description TEXT
);

-- Create Registrations table
CREATE TABLE Registrations (
    RegistrationID INT PRIMARY KEY IDENTITY(1, 1),
    EventID INT REFERENCES Events(EventID), -- FK
    AttendeeFirstName VARCHAR(50) NOT NULL,
    AttendeeLastName VARCHAR(50) NOT NULL,
    Email VARCHAR(50) NOT NULL,
    RegistrationDate DATETIME NOT NULL,
);
```