**Case Study Fitness Center Management System**

**Description:**

A Fitness Center Management System is a comprehensive software solution designed to optimize the operations of gyms or fitness centers. This system covers a wide range of functionalities, including member management, class scheduling, and equipment maintenance. It aims to enhance the overall efficiency of fitness facilities by automating manual processes, providing accurate member data, and ensuring smooth coordination of classes and equipment maintenance.

**Relevance:**

**1. Trainer Management:**

It includes personal information, qualifications, and schedules of the trainer. This helps in assigning trainers to specific classes and monitoring their availability.

**2. Equipment Inventory:**

Enables systematic tracking of fitness equipment, ensuring that the center has the necessary tools for different classes. It aids in maintenance planning and prevents disruptions due to equipment shortages or malfunctions.

**3. Room Allocation:**

This table facilitates the allocation of rooms for fitness activities. It ensures that classes have appropriate facilities and avoids scheduling conflicts. This is vital for providing a smooth and organized experience for both trainers and members.

**4. Course Information:**

Stores details about different fitness courses offered by the center, aiding in effective course management. It includes information on course names, descriptions, and fees, helping in planning and marketing courses.

**5. Member Management:**

This table stores comprehensive details about each member, facilitating member registration, communication, and engagement. It's essential for maintaining an updated and accurate member database.

**6. Membership Management:**

Tracks various membership types, durations, and rates. This table is crucial for managing membership subscriptions, renewals, and fee structures. It ensures that the center can efficiently handle different membership scenarios.

**7. Member Registration:**

Captures details about member registrations, including associated memberships, courses, and start dates. This supports the onboarding process for new members and ensures that they are enrolled in the appropriate programs**.**

**8. Member Attendance Tracking:**

Records member attendance, connecting it to specific registrations. This is crucial for monitoring member participation in classes and assessing overall member engagement. It provides insights into class popularity and helps in adjusting schedules based on attendance patterns.

**Table Structure**

Actors

* Trainer
* Equipment
* Rooms
* Courses
* Member
* Membership

Other Tables

* EquipmentRoom
* CourseTrainer
* MemberRegistration
* MemberAttendance

**1. Trainer Table:**

**Attributes:**

TrainerID (Primary Key): Unique identifier for each trainer.

TrainerName: Full name of the trainer.

DateOfBirth: Date of birth of the trainer.

ContactNumber: Contact number of the trainer.

Email: Email address of the trainer.

**Data Types and Constraints:**

TrainerID: Integer (Primary Key, Autoincrement).

TrainerName: VARCHAR.

DateOfBirth: Date.

ContactNumber: VARCHAR with length constraint.

Email: VARCHAR.

**2. Equipment Table:**

**Attributes:**

EquipmentID (Primary Key): Unique identifier for each piece of equipment.

EquipmentName: Name of the equipment.

Quantity: Quantity of the equipment available.

Description: Description of the equipment.

**Data Types and Constraints:**

EquipmentID: Integer (Primary Key, Autoincrement).

EquipmentName: VARCHAR.

Quantity: Integer.

Description: VARCHAR.

**3. Rooms Table:**

**Attributes:**

RoomID (Primary Key): Unique identifier for each room.

RoomNumber: Room number.

**Data Types and Constraints:**

RoomID: Integer (Primary Key, Autoincrement).

RoomNumber: VARCHAR.

**4. Courses Table:**

**Attributes:**

CourseID (Primary Key): Unique identifier for each course.

CourseName: Name of the fitness course.

Description: Description of the fitness course.

Rate: Fee of the course.

**Data Types and Constraints:**

CourseID: Integer (Primary Key, Autoincrement).

CourseName: VARCHAR.

Description: VARCHAR.

Rate: Decimal.

**5. Member Table:**

**Attributes:**

MemberID (Primary Key): Unique identifier for each member.

MemberName: Full name of the member.

DateOfBirth: Date of birth of the member.

ContactNumber: Contact number of the member.

Email: Email address of the member.

**Data Types and Constraints:**

MemberID: Integer (Primary Key, Autoincrement).

MemberName: VARCHAR.

DateOfBirth: Date.

ContactNumber: VARCHAR with length constraint.

Email: VARCHAR.

**6. Membership Table:**

**Attributes:**

MembershipID (Primary Key): Unique identifier for each membership.

MembershipType: Type of the membership.

Duration: Duration of the membership.

Rate: Rate of the membership.

**Data Types and Constraints:**

MembershipID: Integer (Primary Key, Autoincrement).

MembershipType: VARCHAR.

Duration: Integer.

Rate: Decimal.

**7. EquipmentRoom Table:**

**Attributes:**

EquipmentRoomID (Primary Key): Unique identifier for each equipmentroom association.

RoomID: Foreign key referencing Rooms table.

EquipmentID: Foreign key referencing Equipment table.

**Data Types and Constraints:**

EquipmentRoomID: Integer (Primary Key, Autoincrement).

RoomID: Integer (Foreign Key).

EquipmentID: Integer (Foreign Key).

**8. CourseTrainer Table:**

**Attributes:**

CourseTrainerID (Primary Key): Unique identifier for each coursetrainer association.

CourseID: Foreign key referencing Courses table.

TrainerID: Foreign key referencing Trainer table.

RoomID: Foreign key referencing Rooms table.

**Data Types and Constraints:**

CourseTrainerID: Integer (Primary Key, Autoincrement).

CourseID: Integer (Foreign Key).

TrainerID: Integer (Foreign Key).

RoomID: Integer (Foreign Key).

**9. MemberRegistration Table:**

**Attributes:**

MemberRegistrationID (Primary Key): Unique identifier for each member registration.

MemberID: Foreign key referencing Member table.

MembershipID: Foreign key referencing Membership table.

CourseTrainerID: Foreign key referencing CourseTrainer table.

StartDate: Start date of the membership.

**Data Types and Constraints:**

MemberRegistrationID: Integer (Primary Key, Autoincrement).

MemberID: Integer (Foreign Key).

MembershipID: Integer (Foreign Key).

CourseTrainerID: Integer (Foreign Key).

StartDate: Date.

**10. MemberAttendance Table:**

**Attributes:**

MemberAttendanceID (Primary Key): Unique identifier for each member attendance.

MemberRegistrationID: Foreign key referencing MemberRegistration table.

TimeIn: Time when the member arrived.

TimeOut: Time when the member left.

**Data Types and Constraints:**

MemberAttendanceID: Integer (Primary Key, Autoincrement).

MemberRegistrationID: Integer (Foreign Key).

TimeIn: DateTime.

TimeOut: DateTime.

**CREATE TABLE Queries**create database fitness\_db;

use fitness\_db;

CREATE TABLE Trainer (

TrainerID INT PRIMARY KEY AUTO\_INCREMENT,

TrainerName VARCHAR(25),

DateOfBirth DATE,

ContactNumber VARCHAR(10),

Email VARCHAR(30)

);

CREATE TABLE Equipment (

EquipmentID INT PRIMARY KEY AUTO\_INCREMENT,

EquipmentName VARCHAR(25),

Quantity INT,

Description VARCHAR(50)

);

CREATE TABLE Rooms (

RoomID INT PRIMARY KEY AUTO\_INCREMENT,

RoomNumber VARCHAR(10)

);

CREATE TABLE Courses (

CourseID INT PRIMARY KEY AUTO\_INCREMENT,

CourseName VARCHAR(25),

Description VARCHAR(50),

Rate DECIMAL(10, 2)

);

CREATE TABLE Member (

MemberID INT PRIMARY KEY AUTO\_INCREMENT,

MemberName VARCHAR(25),

DateOfBirth DATE,

ContactNumber VARCHAR(10),

Email VARCHAR(30)

);

CREATE TABLE Membership (

MembershipID INT PRIMARY KEY AUTO\_INCREMENT,

MembershipType VARCHAR(25),

Duration INT,

Rate DECIMAL(10, 2)

);

CREATE TABLE EquipmentRoom (

EquipmentRoomID INT PRIMARY KEY AUTO\_INCREMENT,

RoomID INT,

EquipmentID INT,

FOREIGN KEY (RoomID) REFERENCES Rooms(RoomID),

FOREIGN KEY (EquipmentID) REFERENCES Equipment(EquipmentID)

);

CREATE TABLE CourseTrainer (

CourseTrainerID INT PRIMARY KEY AUTO\_INCREMENT,

CourseID INT,

TrainerID INT,

RoomID INT,

FOREIGN KEY (CourseID) REFERENCES Courses(CourseID),

FOREIGN KEY (TrainerID) REFERENCES Trainer(TrainerID),

FOREIGN KEY (RoomID) REFERENCES Rooms(RoomID)

);

CREATE TABLE MemberRegistration (

MemberRegistrationID INT PRIMARY KEY AUTO\_INCREMENT,

MemberID INT,

MembershipID INT,

CourseTrainerID INT,

StartDate DATE,

FOREIGN KEY (MemberID) REFERENCES Member(MemberID),

FOREIGN KEY (MembershipID) REFERENCES Membership(MembershipID),

FOREIGN KEY (CourseTrainerID) REFERENCES CourseTrainer(CourseTrainerID)

);

CREATE TABLE MemberAttendance (

MemberAttendanceID INT PRIMARY KEY AUTO\_INCREMENT,

MemberRegistrationID INT,

TimeIn DATETIME,

TimeOut DATETIME,

FOREIGN KEY (MemberRegistrationID) REFERENCES MemberRegistration(MemberRegistrationID)

);

**ER Diagram**

