Table Schemas

Table: Attends

```
sqlite> .schema Attends
CREATE TABLE Attends (
    memberId INTEGER,
    classId INTEGER,
    attendanceDate DATE NOT NULL,
    FOREIGN KEY (memberId) REFERENCES Member(memberId), -- Foreign key to the Member table
    FOREIGN KEY (classId) REFERENCES Class(classId) -- Foreign key to the Class table
);
```

Table: Equipment

```
sqlite> .schema Equipment
CREATE TABLE Equipment (
    equipmentID INTEGER PRIMARY KEY AUTOINCREMENT, -- Unique ID
    name varchar(50) NOT NULL,
    classType varchar(30) CHECK(classType IN ('Cardio', 'Strength', 'Flexibility', 'Recovery')), -- Type of equipment
    quantity INTEGER check (quantity > 0), -- Quantity of the equipment available in the gym
    gymId INTEGER,
    FOREIGN KEY (gymId) REFERENCES GymFacility(gymId) -- Foreign key to the GymFacility table
);
```

Table: Instructor

```
sqlite> .schema Instructor
CREATE TABLE Instructor (
    instructorID INTEGER PRIMARY KEY AUTOINCREMENT, -- Unique ID
    name varchar(50) NOT NULL,
    specialty varchar(50),
    phone varchar(15),
    email varchar(100) NOT NULL
);
```

Table: MembershipPlan

```
sqlite> .schema MembershipPlan
CREATE TABLE MembershipPlan (
    planId INTEGER PRIMARY KEY AUTOINCREMENT, -- Unique ID
    planType varchar(50) CHECK(planType IN('Monthly', 'Annual')) NOT NULL, -- Type of plan
    cost NUMERIC NOT NULL
);
```

Table: Class

```
sqlite> .schema Class
CREATE TABLE Class (
    classID INTEGER PRIMARY KEY AUTOINCREMENT, -- Unique ID
    className varchar(50) NOT NULL,
    classType varchar(50) NOT NULL CHECK(classType IN ('Yoga', 'Zumba', 'HIIT', 'Weights')), -- Type of class
    duration INTEGER NOT NULL,
    classCapacity INTEGER NOT NULL, -- Max number of participants for the class
    instructorID INTEGER,
    gymID INTEGER,
    FOREIGN KEY (instructorID) REFERENCES Instructor(instructorID), -- Foreign key to the Instructor table
    FOREIGN KEY (gymID) REFERENCES GymFacility(gymID) -- Foreign key to the GymFacility table
);
```

Table: GymFacility

```
sqlite> .schema GymFacility
CREATE TABLE GymFacility (
    gymID INTEGER PRIMARY KEY AUTOINCREMENT, -- Unique ID
    location varchar(100) NOT NULL,
    phone varchar(50),
    manager varchar(50)
);
```

Table: Member

```
sqlite> .schema Member
CREATE TABLE Member (
    memberID INTEGER PRIMARY KEY AUTOINCREMENT, -- Unique ID
    name varchar(50) NOT NULL,
    email varchar(50) NOT NULL UNIQUE,
    phone varchar(15),
    address varchar(100),
    age INTEGER CHECK (age >= 15), -- Member's age must be 15 or older
    membershipStartDate TEXT NOT NULL,
    membershipEndDate TEXT NOT NULL CHECK(membershipEndDate > membershipStartDate)
);
```

Table: Payment

```
sqlite> .schema Payment
CREATE TABLE Payment (
    paymentId INTEGER PRIMARY KEY AUTOINCREMENT, -- Unique ID
    memberID INTEGER,
    planId INTEGER,
    amountPaid REAL NOT NULL,
    paymentDate DATE NOT NULL,
    FOREIGN KEY (memberId) REFERENCES Member(memberId), -- Foreign key to the Member table
    FOREIGN KEY (planId) REFERENCES MembershipPlan(planId) -- Foreign key to the MembershipPlan table
);
```