

# Explanation of Relationships in the Relational Database Diagram

## Tables and Relationships

### 1. Member Table

- **Primary Key:** MemberID
- Stores basic information about members, such as name, gender, age, contact details, and role.
- **Relationships:**
  - **One-to-One** with Membership (A member can have one or more memberships over time).
  - **One-to-Many** with Attendance (A member can attend multiple classes).
  - **One-to-Many** with Payment (A member can make multiple payments).
  - **One-to-Many** with Purchase (A member can make multiple purchases).
  - **One-to-One** with UserAuth (Each member has one authentication record).

### 2. Trainer Table

- **Primary Key:** TrainerID
- Stores information about trainers, including their area of expertise.
- **Relationships:**
  - **One-to-Many** with Class (A trainer can conduct multiple classes).

### 3. Membership Table

- **Primary Key:** MembershipID
- Tracks details about membership type, start and end dates, and price.
- **Relationships:**
  - **Many-to-One** with Member (A membership belongs to one member).

### 4. Class Table

- **Primary Key:** ClassID
- Contains information about fitness classes, such as name, schedule, and type.
- **Relationships:**
  - **Many-to-One** with Trainer (A class is conducted by one trainer).
  - **One-to-Many** with Attendance (A class can have multiple attendees).

### 5. Attendance Table

- **Primary Key:** AttendanceID
- Tracks members' attendance in classes.
- **Relationships:**
  - **Many-to-One** with Member (An attendance record is associated with one member).
  - **Many-to-One** with Class (An attendance record is associated with one class).

## 6. Payment Table

- **Primary Key:** PaymentID
- Tracks payment details of members.
- **Relationships:**
  - **Many-to-One** with Member (A payment is associated with one member).

## 7. UserAuth Table

- **Primary Key:** UserID
- Stores authentication details of members.
- **Relationships:**
  - **One-to-One** with Member (Each authentication record belongs to one member).

## 8. Product Table

- **Primary Key:** ProductID
- Stores details about purchasable products.
- **Relationships:**
  - **One-to-Many** with Purchase (A product can be purchased multiple times).

## 9. Purchase Table

- **Primary Key:** PurchaseID
- Tracks members' product purchases.
- **Relationships:**
  - **Many-to-One** with Member (A purchase belongs to one member).
  - **Many-to-One** with Product (A purchase involves one product).

## 10. Equipment Table

- **Primary Key:** EquipmentID
- Tracks the quantity and status of gym equipment.
- Currently, it has no explicit relationships with other tables in the schema.

## Relationship Types

1. **One-to-Many (1\*\* : N\*\*\*\*)\*\*:**
  - Member with Membership (A member can have multiple memberships over time).
  - Member with Attendance (A member can attend multiple classes).
  - Member with Payment (A member can make multiple payments).
  - Member with Purchase (A member can make multiple purchases).
  - Trainer with Class (A trainer can conduct multiple classes).
  - Class with Attendance (A class can have multiple attendees).
  - Product with Purchase (A product can be purchased multiple times).
2. **Many-to-One (N:1):**
  - Membership with Member (A membership belongs to one member).
  - Attendance with Member (An attendance record is associated with one member).

- Attendance with Class (An attendance record is associated with one class).
- Payment with Member (A payment is associated with one member).
- Purchase with Member (A purchase belongs to one member).
- Purchase with Product (A purchase involves one product).
- Class with Trainer (A class is conducted by one trainer).

3. **One-to-One (1:1):**

- Member with UserAuth (Each member has one authentication record).

4. **Unrelated Tables:**

- The Equipment table currently has no relationships with other tables but independently tracks equipment details.