COMP 3005 Final Project:

- Demonstrating Normal form (1NF, 2NF, 3NF) for all relations.

    - **1NF:** All attributes in all relations are atomic, therefore all relations are **in 1NF**

    - **2NF & 3NF:**

    1. **HealthMetric (MemberID, TimeStamp, Height, Weight, VO2Max, BodyFatPercentage)**

    - The following FDs are known:

        - (MemberID, TimeStamp) → Height
        - (MemberID, TimeStamp) → Weight
        - (MemberID, TimeStamp) → VO2Max
        - (MemberID, TimeStamp) → BodyFatPercentage

    - **PK**: {MemberID, TimeStamp}

    - **2NF:**

        - We can conclude this relation is **in 2NF** since all FDs are full functional dependencies (all non-prime attributes depend on the full primary key)

    - **3NF:**

        - This relation does not include any dependencies between non-prime attributes, therefore is by default **in 3NF** (no transient dependencies exist)

    2. **FitnessGoals (MemberID, TimeStamp, WeightGoal, VO2MaxGoal, BodyFatPercentageGoal)**

    - The following FDs are known:

        - (MemberID, TimeStamp) → WeightGoal
        - (MemberID, TimeStamp) → VO2MaxGoal
        - (MemberID, TimeStamp) → BodyFatPercentageGoal

- **PK**: {MemberID, TimeStamp}

- **2NF:**

    - We can conclude this relation is **in 2NF** since all FDs are full functional dependencies (all non-prime attributes depend on the full primary key)

- **3NF:**

    - This relation does not include any dependencies between non-prime attributes, therefore is by default **in 3NF** (no transient dependencies exist)

3. **Member (ID, fname, lname, DateOfBirth, Gender, Email, PhoneNumber)**

- The following FDs are known:

    - ID → fname
    - ID → lname
    - ID → DateOfBirth
    - ID → Gender
    - ID → Email
    - ID → PhoneNumber

- **Note:** Email or phone is not unique to a member (an email or phone number can be used by more than one member). Decision made to allow parents to register kids as members using their emails and phone numbers. This also permits shared emails etc. Identification of a member is done solely based on ID

- **PK**: {ID}

- **2NF:**

    - We can conclude this relation is **in 2NF** since all FDs are full functional dependencies (all non-prime attributes depend on the full primary key)

- **3NF:**

    - This relation does not include any dependencies between non-prime attributes, therefore is by default **in 3NF** (no transient dependencies exist)

4. **Trainer (ID, fname, lname, DateOfBirth, Gender, Email, PhoneNumber)**

- The following FDs are known:
    - ID → fname
    - ID → lname
    - ID → DateOfBirth
    - ID → Gender
    - ID → Email
    - ID → PhoneNumber

- **Note:** Email or phone is not unique to a trainer (an email or phone number can be used by more than one member). Decision made to allow parents to register kids as members using their emails and phone numbers. This also permits shared emails etc. Identification of a member is done solely based on ID

- **PK**: {ID}

- **2NF:**
    - We can conclude this relation is **in 2NF** since all FDs are full functional dependencies (all non-prime attributes depend on the full primary key)

- **3NF:**
    - This relation does not include any dependencies between non-prime attributes, therefore is by default **in 3NF** (no transient dependencies exist)

5. **IndividualAvailability (TrainerID, StartHour, EndHour, Date)**

- The following FDs are known:
    - None, Every attribute is part of the PK.

- **PK**: {TrainerID, StartHour, EndHour, Date}

- **2NF:**
    - This relation is **in 2NF** by default since it has no Functional Dependencies.
- **3NF:**
    - This relation is **in 3NF** by default since it has no Functional Dependencies.

6. **Registers (MemberID, FitnessClassID)**

- The following FDs are known:
    - None, Every attribute is part of the PK.
- **PK**: {MemberID, FitnessClassID}

- **2NF:**
    - We can conclude this relation is **in 2NF** since all FDs are full functional dependencies (all non-prime attributes depend on the full primary key).
- **3NF:**
    - This relation is **in 3NF** by default since it has no Functional Dependencies.

7. **Room (ID, Location)**

- The following FDs are known:
    - ID → Location
- **PK**: {ID}

- **2NF:**
    - This relation is **in 2NF** by default since it has no Functional Dependencies.
- **3NF:**
    - This relation does not include any dependencies between non-prime attributes, therefore is by default **in 3NF** (no transient dependencies exist).

8. **AdministrationStaff (ID, fname, lname, DateOfBirth, Gender, Email, PhoneNumber)**

- The following FDs are known:
    - ID → fname
    - ID → lname
    - ID → DateOfBirth
    - ID → Gender
    - ID → Email
    - ID → PhoneNumber

- **Note:** Email or phone is not unique to an Admin Staff member (an email or phone number can be used by more than one member). Decision made to allow parents to register kids as members using their emails and phone numbers. This also permits shared emails etc. Identification of a member is done solely based on ID

- **PK**: {ID}

- **2NF:**
    - We can conclude this relation is **in 2NF** since all FDs are full functional dependencies (all non-prime attributes depend on the full primary key)

- **3NF:**
    - This relation does not include any dependencies between non-prime attributes, therefore is by default **in 3NF** (no transient dependencies exist)

9. **FitnessClass (ID, Capacity, Date, StartHour, EndHour, RoomID, TrainerID, AdminID)**

- The following FDs are known:
    - ID → Capacity
    - ID → Date
    - ID → StartHour
    - ID → EndHour
    - ID → RoomID
    - ID → TrainerID
    - ID → AdminID

- **PK**: {ID}

- **2NF:**

    - We can conclude this relation is **in 2NF** since all FDs are full functional dependencies (all non-prime attributes depend on the full primary key)

- **3NF:**

    - This relation does not include any dependencies between non-prime attributes, therefore is by default **in 3NF** (no transient dependencies exist)