COMP 3005 Final Project – Option Two – Armaan Wander

Entities:

Person – This is an entity that is used as a generalized entity that holds many attributes that are needed for members, trainers, and admin.

- PersonID
  - The primary key
- Username
  - This is unique to each person that is created and filled in by the user.
- Password
  - Filled in by the user.
- Name
  - This is displayed when the user logs into the system.
- AccountType
  - Holds if the account is a trainer, member, or admin.
- Height
- Weight
- Age
  - All three were filled in by user.
- Schedule
  - A 2D integer array that holds either a 0,1,2,3 or 4 depending on what the user has in that time slot.
  - Gets populated by the program when a new user is created

Member – This is derived from person; it holds extra information such as fitness goals and achievements. It is separated from the person entity because the trainers and admin have no use for these attributes so they would be wasted.

- FitnessGoal
  - It is text as goals is something that is more abstract so to allow for a wide range of goals text was the best option.
- FitnessAchievements
  - It is text for the same reason as FitnessGoal.

Trainer – This entity is derived from the person entity. It is separate for the same reasons as member.

- Specialization
  - Filled in by the user.

Room – This entity represents a physical room in the fitness center that can be booked for classes.

- Capacity
  - Initialized at runtime.
- Schedule
  - Works the same way as the one in the Person Entity

Class – This represents a class of members that has a trainer and will take place in a class.

- ClassName
    - Filled in by the admin.
- Weekday
    - Represents which day of the week the event will take place. It is an int so it can be used as an index for the Schedule arrays.
- Time
    - Same thing as weekday but for times in a day.
- NumberOfParticipants
    - This gets initialized to 0 at runtime and then increases when a new member joins the class and decrements when someone leaves the class.

PersonlSession – This represents a one-to-one session between a member and a trainer.

- Weekday
- Time
    - Both work the same way as the data members with the same name in the Class entity.

Routine – This entity represents a routine that a member can create.

- Routine
- Description
- Duration
    - All three are filled in by the user.
    - Duration is text as not all routines take up the whole 1-hour blocks that the schedule works with, so text allows the user to set any time from 1 hour to 5 minutes.
- TimesCompleted
    - Starts off at 0.
    - Increments when the user enters that they performed a routine.

ClassMemberBooking – This represents a one-to-one booking between a member and some class.

- Weekday
- Time
    - Both work the same way as the data members with the same name in the Class entity.

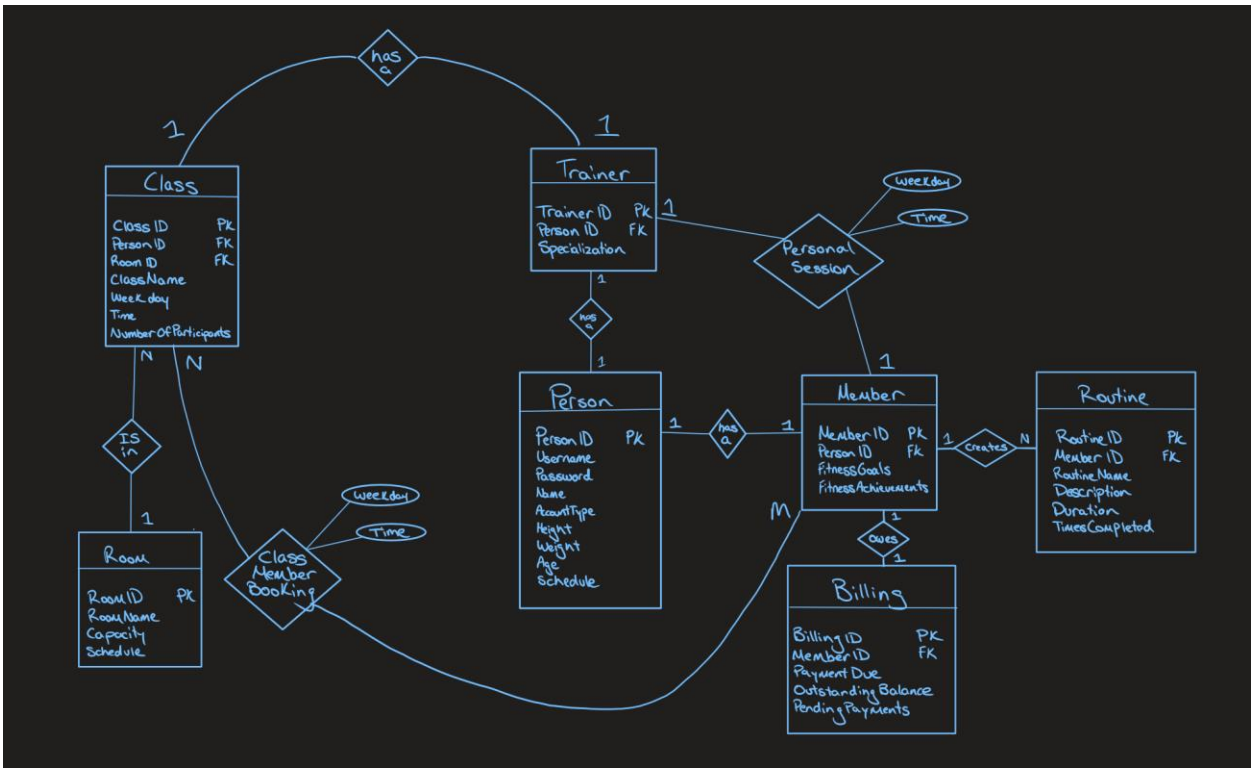
Billing – Manages billing for members.

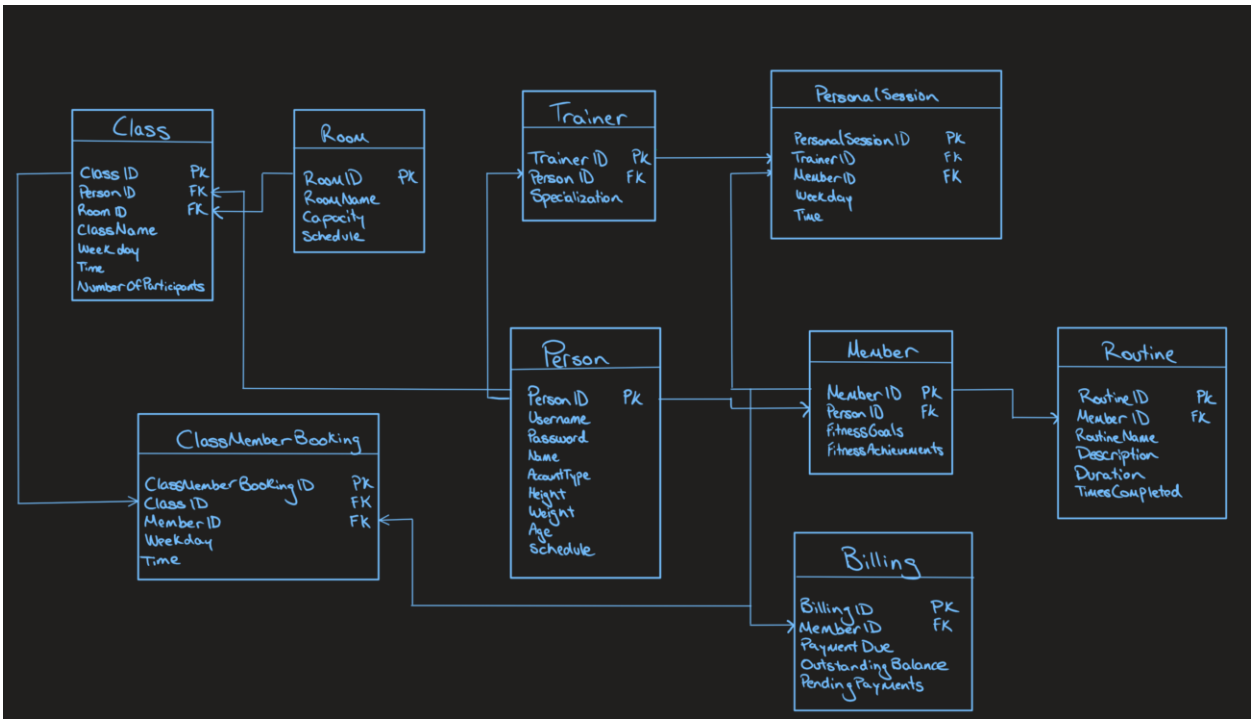
- PaymentDue
    - This holds the amount of money that the user owes since the last time payments were processed.
- OutstandingBalance
    - The amount of money that the user owes after payments are processed.
    - The program allows the user to over pay.

- PendingPayments
    - is a payment that the user has made but the admin has not processed it yet.

Requirements	Assumptions Made	Representation in Assignment
The fitness system that must have three different types of users, members, admins, and trainers	N/A	Member and Trainer entities
A member must be able to create an account, manage their personal information, goals, health, view routines, achievements, health statistics, book/cancel personal session with trainers and join/leave classes.	For this to work the user must provide: Their username, password, name, height, weight, age, personal goals and achievements Additional requirements: There must be a trainer to book sessions with and to host classes, and there must be an admin who can create classes.	registerUser() updateUser() displayMemberDashboard bookPS() cancelPS() joinClass() leaveClass() viewRoutine()
A trainer must be able create their own account, manage their own schedule and view members based off their name.	The trainer must provide : Username, password, name, height, weight, age, and what they specialize in	registerTrainer() viewMembers() printSchedule() setAvailiability()
An admin must be able to create their own account, book rooms for classes, maintain equipment, bill members and process payments	The admin must provide : Username, password, name, height, weight, and age	createClass() deleteClass() bookMaintenance() deleteMaintenance() billMembers() processPayments()

Description of Methods can be found in the comments of the .java file on GitHub.


has a
1
1
Class
Class ID PK
Person ID FK
Room ID FK
ClassName
Week_day
Time
Number Of Participants
Trainer
Trainer ID PK
Person ID FK
Specialization
1
Personal Session
Week day
Time
has a
1
N
N
IS in
1
Person
Person ID PK
Username
Password
Name
AccountType
Height
Weight
Age
Schedule
1
has a
1
Member
Member ID PK
Person ID FK
FitnessGoals
FitnessAchievements
1
Creates
N
Routine
Routine ID PK
Member ID FK
RoutineName
Description
Duration
TimesCompleted
Room
Room ID PK
RoomName
Capacity
Schedule
Class Member Booking
Week day
Time
M
1
Owes
1
Billing
Billing ID PK
Member ID FK
Payment Due
Outstanding Balance
Pending Payments


Class
Class ID PK
Person ID FK
Room ID FK
ClassName
Week_day
Time
Number Of Participants
Room
Room ID PK
RoomName
Capacity
Schedule
Trainer
Trainer ID PK
Person ID FK
Specialization
PersonalSession
PersonalSession ID PK
Trainer ID FK
Member ID FK
Week day
Time
Person
Person ID PK
Username
Password
Name
AccountType
Height
Weight
Age
Schedule
Member
Member ID PK
Person ID FK
FitnessGoals
FitnessAchievements
Routine
Routine ID PK
Member ID FK
RoutineName
Description
Duration
TimesCompleted
ClassMemberBooking
ClassMemberBooking ID PK
Class ID FK
Member ID FK
Week day
Time
Billing
Billing ID PK
Member ID FK
Payment Due
Outstanding Balance
Pending Payments