

Fitness Tracker

**Protiti Tarafder
6236037**

Data Structures and Object Oriented Programming

Table of Content

1. Project Description
2. Program Features and Screenshots
3. Challenges
4. Learning Outcomes

1. Project Description

The Fitness Tracker System is an application where users could monitor their workouts, nutrition and fitness goals. They can log anything from exercises, calorie intake (meals), set up fitness goals, and view their progress. It is used to manage their fitness routine and health. There is also an advanced user, the premium user, where they have access to more advanced analysis (monthly reports and goal comparisons).

Based on the type of user you are (premium user or free user), the amount of services will change.

1.Free User

- i.Track workout (sets, reps and duration)
- ii.Track meals (calories)
- iii.Set personal fitness goals
- iv.Limited view of accomplished workout (when logged)

2.Premium User (Has all the FreeUser services and:)

- i.Monthly reports (summary of burned calories and tracked amount of times they worked out)
- ii.Goal comparisons (tracks if the user is meeting their goal using percentage e.g. "You are 30% toward your goal of burning 1000 kcal this month")
- iii.Breakdown of what type of workout has been accomplished (cardio, strength, etc.)
- iv.Export their reports(CSV or text)

2. Program Features and Screenshots

The input is: (the set goals output has been excluded as it is mentioned here in input)

```
FreeUser user1 = new FreeUser( username: "pro");
PremiumUser user2 = new PremiumUser( username: "bob");

user1.setGoal("Burn 200 kcal");
user1.logWorkout(new CardioWorkout( exerciseName: "Jogging", new Date(), durationMinutes: 20));
user1.logWorkout( name: "Biking", duration: 25);

user2.setGoal("Burn 1500 kcal by next month");
user2.addWorkout(new StrengthWorkout( exerciseName: "Bench Press", new Date(), sets: 3, repsPerSet: 12));
user2.logWorkout( name: "StairMaster", duration: 35);

System.out.println("\n FREE USER");
user1.viewProgressReport();
user1.isOnTrackForGoal();
System.out.println("Logged workouts:");
user1.workoutHistory.forEach( Workout w -> System.out.println(w));

System.out.println("\n PREMIUM USER");
user2.viewProgressReport();
System.out.println(user2.generateMonthlyReport( month: 05, year: 2026));
user2.isOnTrackForGoal();
user2.generateReport();
user2.exportReport( filename: "bob_report.csv");
System.out.println("Premium user report exported to csv");
```

Free User: here can be seen the results of the free user

```
FREE USER
pro's progress report
Total Calories: 292.5
Amount of Logged Workouts: 2
Advanced progress report is a Premium User feature.
Goal tracking is a Premium User feature.
Logged workouts:
Jogging, Cardio, Mon May 12 00:10:39 EDT 2025
Biking, Cardio, Mon May 12 00:10:39 EDT 2025
```

Premium User: here can be seen the results and report of the premium user

```
PREMIUM USER
Progress report for bob
- Bench Press, Mon May 12 00:10:39 EDT 2025, 18.0 kcal
- StairMaster, Mon May 12 00:10:39 EDT 2025, 227.5 kcal
bob burn 0.00 kcal in 2026/5
Tracking goal progress for bob
Generating report for bob
Total calories burned: 245.5
Total workouts2
- Bench Press, Mon May 12 00:10:39 EDT 2025, 18.0
- StairMaster, Mon May 12 00:10:39 EDT 2025, 227.5
Exported to bob_report.csv
Premium user report exported to csv
```

The project meets almost all functionalities mentioned in the README file except for the goal tracking by percentage and the user upgrading. This is due to time restraints, but as well as complexity for the goal tracking by percentage as it would require the user to constantly be updating and tracking. For the user upgrading, its more of a mock functionality as the app is meant to have two clear distinctions between users.

3. Challenges

- File I/O and CSV were the hardest, especially when it came to the exporting report part. The logic was hard to understand and some errors would show up while coding especially when it came to using the try-catch and error handling.
- The goal tracking by percentage and user upgrading (explained above).
- Making sure the testing was accurate and worked fine. As well as making sure I was testing correctly (what does and doesn't need testing). There's multiple functions to this project and so making sure the essentials were being tested and not a small method that doesn't need actual testing, was hard to properly figure out.

4. Learning Outcomes

- Learning how to use stream and lambda, through this project I fully understood how to implement this into my code.
- I was able to practice error handling and testing outside of an already assigned assignment. I had to figure it out on my own and make sure I was using the right error handling on top of figuring out why my try-catch would pop up error.
- I learned that there could be a few ways to simplify the code (ex: advanced loops) instead of having to write a huge time consuming block.