

```
1 package PHI;
2
3 import java.text.ParseException;
4 import java.text.SimpleDateFormat;
5 import java.util.Date;
6 import java.util.Scanner;
7
8 public class Main {
9     public static void main(String[] args) {
10         /**
11          * The main method to demonstrate the functionality of the health data
12          * tracking system.
13          *
14          * Pre-condition:
15          * - Appropriate classes with constructors exist
16          * - Custom validation exception handling method exists
17          *
18          * Post-condition:
19          * - Prints the health data for a user, including blood pressure, cholesterol
20          * levels, blood glucose level, BMI, and custom health data.
21          * - Validates each health data entry before adding it to the user's health
22          * data list.
23          * - Handles and prints any validation errors that occur during the process.
24          */
25         try {
26             // Create a user with some health data
27             // Create a Scanner object to read input from the terminal
28             Scanner scanner = new Scanner(System.in);
29
30             // Prompt the user to enter their information
31             System.out.print("Enter your first name: ");
32             String firstName = scanner.nextLine();
33
34             System.out.print("Enter your last name: ");
35             String lastName = scanner.nextLine();
36
37             System.out.print("Enter your email: ");
38             String email = scanner.nextLine();
39
40             System.out.print("Enter your password: ");
41             String password = scanner.nextLine();
42
43             System.out.print("Enter your date of birth (yyyy-mm-dd): ");
44             String dobString = scanner.nextLine();
45             Date dob = new SimpleDateFormat("yyyy-MM-dd").parse(dobString);
46
47             System.out.print("Enter your gender: ");
48             String gender = scanner.nextLine();
49
50             System.out.print("Enter your phone number: ");
51             String phoneNumber = scanner.nextLine();
52
53             // Create a user with the provided information
54             User<HealthData<?>> user = new User<>(firstName, lastName, email,
55             password, dob, gender, phoneNumber);
56             User<HealthData<?>> user = new User<>("John", "Doe",
57             "johndoe@example.com", "Passw0rd", new Date(), "Male", "555-1234");
58         }
59     }
60 }
```

test

Thursday, June 22, 2023, 9:40 AM

```
55         // Add common health data
56         HealthData<?> commonHealth1 = new CommonHealthData(user.getFullName(), new
    Date(), "Blood Pressure", 120, 60);
57         ((CommonHealthData) commonHealth1).validate();
58         user.addHealthData(commonHealth1); // Add the health data before
    validation
59
60         HealthData<?> commonHealth2 = new CommonHealthData(user.getFullName(), new
    Date(), "Cholesterol", -120, 60, 140);
61         ((CommonHealthData) commonHealth2).validate();
62         user.addHealthData(commonHealth2);
63
64         HealthData<?> commonHealth3 = new CommonHealthData(user.getFullName(), new
    Date(), "Blood Glucose", 100);
65         ((CommonHealthData) commonHealth3).validate();
66         user.addHealthData(commonHealth3);
67
68         HealthData<?> commonHealth4 = new CommonHealthData(user.getFullName(), new
    Date(), "BMI", 145.0, 65.0);
69         ((CommonHealthData) commonHealth4).validate();
70         user.addHealthData(commonHealth4);
71
72         // Add custom health data
73         HealthData<?> customHealth = new CustomHealthData("Ankle pain", new Date
    (), "ankle pain on 5-12-2023");
74         user.addHealthData(customHealth);
75
76         // Print out John Doe's health data (original)
77         System.out.println("Original health data:");
78         printHealthData(user);
79
80         // Edit health data
81         int indexToEdit = 1; // Index of the health data entry to edit
82         HealthData<?> newHealthData = new CommonHealthData(user.getFullName(), new
    Date(), "Cholesterol", 130, 70, 300);
83         ((CommonHealthData) newHealthData).validate();
84
85         user.editHealthData(indexToEdit, newHealthData);
86
87         System.out.println("Updated health data:");
88
89         // Print out John Doe's health data (updated)
90         printHealthData(user);
91     } catch (HealthDataException e) {
92         System.out.println("Health data validation error: " + e.getMessage());
93     // } catch (ParseException e) {
94         // TODO Auto-generated catch block
95         //e.printStackTrace();
96     }
97 }
98
99 private static void printHealthData(User<HealthData<?>> user) {
100     System.out.println(user.getFullName() + "'s health data:");
101     for (HealthData<?> healthData : user.getHealthDataList()) {
102         System.out.println(healthData.getMetric() + ": Recorded at: " +
    healthData.getDate());
103         System.out.println("Metric: " + healthData.getMetric());
104
105         if (healthData instanceof CommonHealthData) {
```

test

Thursday, June 22, 2023, 9:40 AM

```
106         CommonHealthData commonHealthData = (CommonHealthData) healthData;
107
108         // Check if the health data is related to blood pressure
109         if (commonHealthData.getMetric().equals("Blood Pressure")) {
110             System.out.println("Systolic BP: " +
commonHealthData.getSystolicBP());
111             System.out.println("Diastolic BP: " +
commonHealthData.getDiastolicBP());
112             HealthDataChecker.checkBloodPressure(commonHealthData);
113         }
114
115         // Check if the health data is related to cholesterol levels
116         if (commonHealthData.getMetric().equals("Cholesterol")) {
117             System.out.println("ldl: " + commonHealthData.getLdlCholesterol
());
118             System.out.println("hdl: " + commonHealthData.getHdlCholesterol
());
119             System.out.println("tri: " +
commonHealthData.getTriglycerideCholesterol());
120
121             HealthDataChecker.checkCholesterol(commonHealthData);
122         }
123
124         // Check if the health data is related to blood glucose level
125         if (commonHealthData.getMetric().equals("Blood Glucose")) {
126             System.out.println("blood sugar: " +
commonHealthData.getGlucoseLevel());
127             HealthDataChecker.checkBloodGlucose(commonHealthData);
128         }
129
130         // Check if the health data is related to BMI
131         if (commonHealthData.getMetric().equals("BMI")) {
132             System.out.println("height: " + commonHealthData.getHeight());
133             System.out.println("weight: " + commonHealthData.getWeight());
134             System.out.println("bmi: " + commonHealthData.calculateBMI());
135             HealthDataChecker.checkBMI(commonHealthData);
136         }
137     } else if (healthData instanceof CustomHealthData) {
138         CustomHealthData customHealthData = (CustomHealthData) healthData;
139         System.out.println("Notes: " + customHealthData.getNotes());
140     }
141
142     System.out.println();
143 }
144 }
145 }
146
```