

User Manual

Gym Management System

2025

SPRINT – SD 13 – SUMMER 2025

JUSTIN GREENSLADE, ASHTON DENNIS, JOSEPH GALLANT

Group: 1 | Version 1.0

### Table of Contents

[Table of Contents 1](#_Toc206073730)

[1. Introduction 3](#_Toc206073731)

[2. System Requirements 3](#_Toc206073732)

[3. Installation Guide/Getting Started 3](#_Toc206073733)

[4. Database Diagram 4](#_Toc206073734)

[5. Project Structure 5](#_Toc206073735)

[5.1 Main project 5](#_Toc206073736)

[5.2 Libraries and scripts 5](#_Toc206073737)

[6. Class Diagrams 6](#_Toc206073738)

[6.1 User, Administrator, Member, Trainer, and Role 6](#_Toc206073739)

[6.2 Membership and MembershipType 7](#_Toc206073740)

[6.3 GymMerchandise and MerchandiseTypes 8](#_Toc206073741)

[6.4 WorkoutClass and WorkoutClassType 9](#_Toc206073742)

[7. Main Menu Overview 10](#_Toc206073743)

[7.1 Register 10](#_Toc206073744)

[7.2 Login 11](#_Toc206073745)

[7.3 Quit 11](#_Toc206073746)

[8. Admin Menu: 11](#_Toc206073747)

[8.1 View All Users: 12](#_Toc206073748)

[8.2 Delete a User: 12](#_Toc206073749)

[8.3 View Memberships & Revenue: 13](#_Toc206073750)

[8.4 Merchandise Management: 14](#_Toc206073751)

[Add new merchandise: 14](#_Toc206073752)

[Edit merchandise: 15](#_Toc206073753)

[Delete merchandise: 16](#_Toc206073754)

[Print all merchandise report and total stock value: 16](#_Toc206073755)

[Back to Admin Menu: 17](#_Toc206073756)

[9. Trainer Menu 17](#_Toc206073757)

[9.1 Manage my workout classes 17](#_Toc206073758)

[Create a Workout Class: 18](#_Toc206073759)

[Edit a Workout Class: 19](#_Toc206073760)

[Delete a Workout Class: 19](#_Toc206073761)

[View My Workout Classes: 19](#_Toc206073762)

[Back to Trainer Menu: 20](#_Toc206073763)

[9.2 Purchase a Membership: 20](#_Toc206073764)

[9.3 View Gym Merchandise: 20](#_Toc206073765)

[9.4 Logout: 21](#_Toc206073766)

[10. Member Menu 21](#_Toc206073767)

[10.1 Browse Workout Classes: 22](#_Toc206073768)

[10.2 View My Memberships: 22](#_Toc206073769)

[10.3 Purchase a Membership: 23](#_Toc206073770)

[10.4 View Gym Merchandise: 24](#_Toc206073771)

[11. Exiting the Program 24](#_Toc206073772)

[12. Troubleshooting Tips 25](#_Toc206073773)

[13. FAQs 25](#_Toc206073774)

[14. Contact Us 26](#_Toc206073775)

# Introduction

This Gym Management System is a Java console application designed to help a gym efficiently manage users, memberships, workout classes, and merchandise. The program supports different user roles such as Administrator, Trainer, and Member, each with their own set of capabilities.

# System Requirements

To run this program, you need:

* Java JDK 11 or higher installed
* A computer running Windows, Mac, or Linux
* Access to a terminal or command prompt

# Installation Guide/Getting Started

* Get the Project Files – On the project’s GitHub page, click Code → either Download ZIP (extract it after download) or Clone with Git (copy the link and run git clone <link> in your terminal).
* Open the Project in an Editor – Use Visual Studio Code (or IntelliJ/Eclipse), go to File → Open Folder, and select the extracted folder.
* Check Java Installation – Requires Java JDK 11+. In your terminal, run java -version. If not installed, download from [Oracle Java Downloads](https://www.oracle.com/java/technologies/javase-downloads.html).
* Open a Java File – In the Explorer panel, click any file ending in .java.
* Run the Program – Click the Run/▶️ button in your editor; the program will start in the terminal.
* Use the Program – The Main Menu will appear; type the number for your choice and press Enter.

## Logging system

The Gym Management System comes equipped with built-in logging capabilities. Errors and crashes, and general information such as data creation, updates and deletes are tracked in an appropriate file. Each entry includes a timestamp and description of the message. The files are stored in the root folder of the application.

* Error logs are tracked in the **gym\_management\_error\_log.txt** file.
* Information logs are tracked in the **gym\_management\_log.txt** file.

# Database Diagram

A screenshot of a computer screen

AI-generated content may be incorrect.

# Project Structure

## Main project

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | |  | | --- | | *DatabaseConnection* in **DBManager** is responsible for handling connections to the database and creating the database and tables when it is executed. | | *GymMerchandise* and *GymMerchandiseType* from **GymMerchandise** are entities used to represent the gym merchandise that is available for purchase. | | *GymMerchDAO* in **GymMerchManagement** handles creating, reading, updating, and deleting GymMerchandise from the database. | | *LoggingManagement* in **Logging** handles the writing of errors and infos to log files. | | *Main* in **MainProgram** is the main class used to launch the Gym Management System. | | *Menu* in **MainProgram** contains all the menu functions for each role. It is called by the *Main*. | | *MembershipsDAO* and *MembershipTypesDAO* in **MembershipManagement** handle creating, reading, updating, and deleting Memberships and MembershipTypes from the database. | | *Membership* and *MembershipType* in **Memberships** are entities used to represent the memberships assigned to a user. | | *UserDAO* and *UserService* in **UserManagement** handle the Login and Registration functionalities, as well as creating, reading, updating, and deleting Users from the database. | | *Administrator*, *Trainer*, and *Member* are entities used to represent users, inherited from the **User** class. They are assigned a Role. | | *WorkoutClass* and *WorkoutClassType* in **WorkoutClasses** are entities used to represent the Workout Classes taught by a Trainer. | | *WorkoutClassesDAO* and *WorkoutClassTypesDAO* in **WorkoutClassManagement** handle creating, reading, updating, and deleting WorkoutClasses and WorkoutClassTypes from the database. | |
|  |  |

## Libraries and scripts

|  |  |
| --- | --- |
| A screenshot of a computer  AI-generated content may be incorrect. | -The **lib** folder contains all the required libraries for BCrypt, used for password encryption, and PostgresSQL, used to access a Postgres database.  -The **scripts** folder contains the necessary script to create the database and the insertion of starting data, such as roles and users. |

# Class Diagrams

## User, Administrator, Member, Trainer, and Role

|  |
| --- |
| **User** |
| |  |  | | --- | --- | | -userId:int  -username:String  -password:String  -firstName:String  -lastName:String  -streetAddress:String | -city:String  -province:String  -postalCode:String  -email:String  -phone:String  -role:Role | |
| |  |  |  | | --- | --- | --- | | +User(userId:int, username:String, password:String, firstName:String, lastName:String, streetAddress:String, city:String, province:String, postalCode:String, email:String, phone:String, role:Role)  +User(user:User) | +getUserId():int  +getUsername():String  +getPassword():String  +getFirstName():String  +getLastName():String  +getStreetAddress():String  +getCity():String  +getProvince():String  +getPostalCode():String  +getEmail():String  +getPhone():String  +getRole():Role  +getFullName():String | +setUserId(userId: int):void  +setUsername(username:String):void  +setPassword(password:String): void  +setFirstName(firstName:String):void  +setLastName(lastName:String):void  +setStreetAddress(streetAddress:String):void  +setCity(city: String):void  +setProvince(province: String):void  +setPostalCode(postalCode: String):void  +setEmail(email: String): void  +setPhone(phone: String):void  +setRole(role: Role):void | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | **Administrator** | | +Administrator(userId:int, username:String, password:String, firstName:String, lastName:String, streetAddress:String, city:String, province:String, postalCode:String, email:String, phone:String, role:Role)  +Administrator(user:User) | | |  | | --- | | **Member** | | +Member(userId:int, username:String, password:String, firstName:String, lastName:String, streetAddress:String, city:String, province:String, postalCode:String, email:String, phone:String, role:Role)  +Member(user:User) | | |  | | --- | | **Trainer** | | +Trainer(userId:int, username:String, password:String, firstName:String, lastName:String, streetAddress:String, city:String, province:String, postalCode:String, email:String, phone:String, role:Role)  +Trainer(user:User) | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  | | --- | | **Role** | | -id:int  -name:String | | +Role(id:int, name:string)  +getId():int  +setId(id:int)  +getName():String  +setName(name:String) | | Three types of users can be created from the User class: Administrator, Member, and Trainer. They are assigned a Role, which defines their permissions. |

## Membership and MembershipType

|  |
| --- |
| **Membership** |
| -membershipId:int  -membershipType:MembershipType  -member:User  -startDate:Date  -endDate:Date |
| +Membership(membershipId:int, membershipType:MembershipType, member:User, startDate:Date, endDate:Date)  +getMembershipId():int  +setMembershipId(membershipId:int):void  +getMembershipType():MembershipType  +setMembershipType(membershipType:MembershipType):void  +getMember():User  +setMember(member:User):void  +getStartDate():Date  +setStartDate(startDate:Date):void  +getEndDate():Date  +setEndDate(endDate:Date):void |

|  |
| --- |
| **MembershipType** |
| -id:int  -name:String  -description:String  -cost:double |
| +MembershipType(id:int, name:String, description:String, cost:double)  +getId(): int  +getName():String  +getDescription():String  +getCost():double  +setId(id:int):void  +setName(name:String):void  +setDescription(description:String):void  +setCost(cost:double):void |

Members and trainers can purchase a gym Membership. A Membership is assigned a User, and a a MembershipType which includes the cost, name and description.

## GymMerchandise and MerchandiseTypes

|  |
| --- |
| **GymMerchandise** |
| -id:int  -merchandiseName:String  -merchandisePrice:double  -quantityInStock:int  -merchandiseType:MerchandiseType |
| +GymMerchandise(id:int, merchandiseName:String, merchandisePrice:double, quantityInStock:int, merchandiseType:MerchandiseType)  +getId():int  +setId(id: int):void  +getMerchandiseName():String  +setMerchandiseName(name: String):void  +getMerchandisePrice():double  +setMerchandisePrice(merchandisePrice: double):void  +getQuantityInStock():int  +setQuantityInStock(quantityInStock: int):void  +getMerchandiseType():MerchandiseType  +setMerchandiseType(merchandiseType: MerchandiseType):void |

|  |
| --- |
| **MerchandiseType** |
| -id:int  -merchandiseTypeName:String |
| +MerchandiseType(id:int, typeName:String)  +getId():int  +setId(id:int):void  +getMerchandiseTypeName():String  +setMerchandiseTypeName(typeName:String):void |

The GymMerchandise class keeps track of the products available for purchase by the members and trainers. Each GymMerchandise requires a MerchandiseType to be assigned.

## WorkoutClass and WorkoutClassType

|  |
| --- |
| **WorkoutClass** |
| -id:int  -workoutClassType:WorkoutClassType  -description:String  -trainer:Trainer  -dateTime:DateTime |
| +WorkoutClass(id:int, workoutClassType:WorkoutClassType, description:String, trainer:Trainer, dateTime:DateTime)  +getId():int  +setId(id: int):void  +getWorkoutClassType():WorkoutClassType  +setWorkoutClassType(workoutClassType:WorkoutClassType):void  +getDescription():String  +setDescription(description:String):void  +getTrainer():Trainer  +setTrainer(trainer:Trainer):void  +getDateTime():DateTime  +setDateTime(dateTime:DateTime):void |

|  |
| --- |
| **WorkoutClassType** |
| -id:int  -name:String  -description:String |
| +WorkoutClassType(id:int, name:String, description:String)  +getId():int  +setId(id: int):void  +getName():String  +setName(name: String):void  +getDescription():String  +setDescription(description: String):void |

The WorkoutClass class is responsible for the Workout Classes offered by a trainer. A trainer is a type of user. A WorkoutClass must be assigned a WorkoutClassType.

# Main Menu Overview

When you start the Gym Management System, you are greeted with the **Main Menu**. This menu lets you either register as a new user, log in if you already have an account, or quit the program.

The menu options are:

A screenshot of a computer

AI-generated content may be incorrect.

## Register

Selecting **Register** (option 1) allows you to create a new account in the system. You will be prompted to enter your personal details:

A screenshot of a computer

AI-generated content may be incorrect.

* **Username:** Choose a unique username for your account.
* **Password:** Enter a secure password.
* **First Name and Last Name:** Your full name as it will appear in the system.
* **Address, City, Province, Postal Code:** Your mailing information.
* **Email:** A valid email address for notifications.
* **Phone Number:** Your contact number.
* **Role:** Choose your role in the gym — Admin, Trainer, or Member.

After completing these steps, your account will be registered, and you can then log in with your credentials.

## Login

Selecting **Login** (option 2) lets you access your existing account. You will need to enter:

A screenshot of a computer screen

AI-generated content may be incorrect.

* **Username:** Your registered username.
* **Password:** Your account password.

If your credentials are correct, you will be logged in and redirected to the menu corresponding to your role:

* **Administrator Menu** if you are an admin.
* **Trainer Menu** if you are a trainer.
* **Member Menu** if you are a gym member.

If the login fails, you will be prompted to enter your credentials again.

## Quit

Selecting **Quit** (option 3) will exit the Gym Management System. A thank-you message will be displayed before the program closes.

# Admin Menu:

Once you log in as an **Administrator**, you will be presented with the **Admin Menu**, which allows you to manage users, view memberships and revenue, and handle gym merchandise.

The menu options are:

A screenshot of a computer program

AI-generated content may be incorrect.

## View All Users:

Select option **1** to see a complete list of all users registered in the system, along with their contact information and roles.

A screenshot of a computer

AI-generated content may be incorrect.

* The list includes User ID, Username, Full Name, Email, Phone, Address, and Role.
* This helps you quickly access details of all users, including admins, trainers, and members.

After viewing the list, you can press Enter to return to the Admin Menu.

## Delete a User:

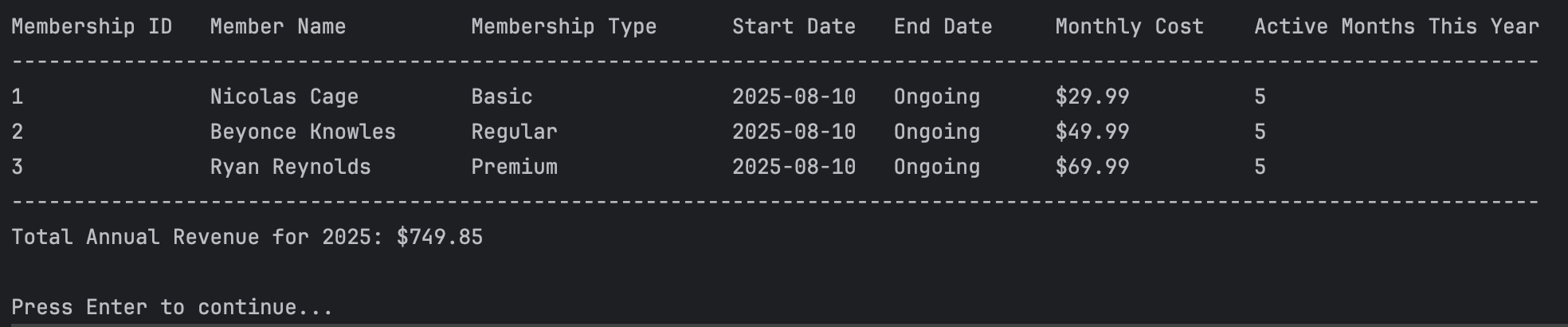
Select option **2** to delete a user from the system.



* You will be prompted to enter the **User ID** of the user you want to remove.
* When a user is deleted, all related memberships are also removed automatically to maintain data integrity.
* If you enter an invalid User ID or a non-numeric value, an error message will prompt you to try again.

After deletion, press Enter to return to the Admin Menu.

## View Memberships & Revenue:

Select option **3** to view a detailed report of all gym memberships, including member names, membership types, start and end dates, monthly costs, and the number of active months within the current year. 

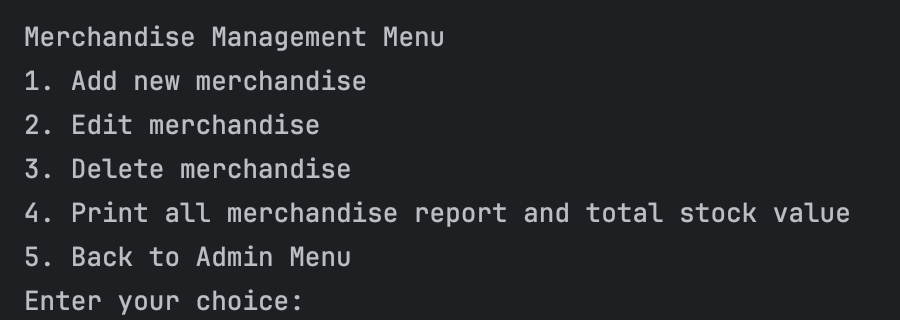
* The system calculates the **total annual revenue** generated from all active memberships for the current year.
* Ongoing memberships without an end date are counted through December 31st of the current year.
* This report helps administrators monitor membership distribution and financial performance.

After reviewing the report, press Enter to return to the Admin Menu.

## Merchandise Management:

Select option **4** to open the **Merchandise Management Menu**, where you can manage gym merchandise inventory.

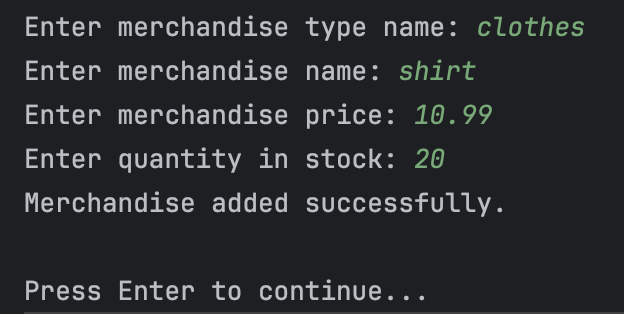
The merchandise menu provides the following options:



### Add new merchandise:

Select **option 1** to add a new item to the gym’s merchandise inventory.

You will be prompted to enter:



* **Merchandise Type Name:**

If the type already exists (e.g., "T-Shirt", "Water Bottle"), it will be reused.

If it does not exist, the system will automatically create it.

* **Merchandise Name:**

This must be unique within the same merchandise type.

If the same name already exists under that type, the system will stop you and return to the menu.

* **Merchandise Price:**

Enter a valid decimal number (e.g., 19.99).

* **Quantity in Stock:**

Enter a whole number (e.g., 50).

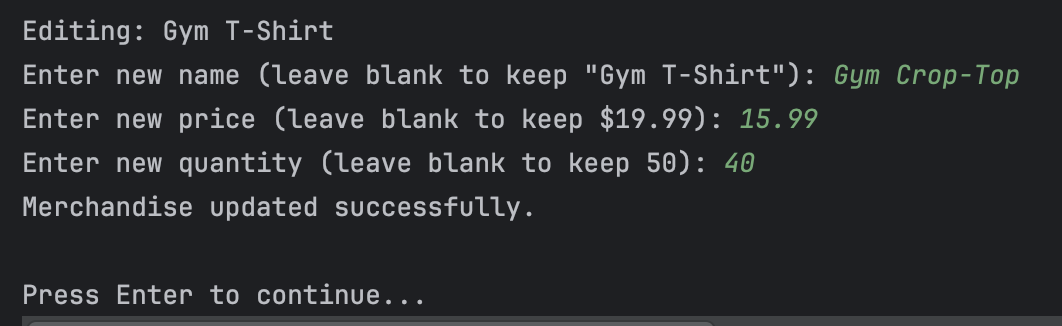
Once all fields are entered, the system saves the new merchandise and confirms:  
 Merchandise added successfully.

Press **Enter** to return to the Merchandise Management Menu.

### Edit merchandise:

Select **option 2** to modify an existing merchandise item.

You will be prompted to:



* **Enter the ID** of the merchandise you want to edit.

If the ID does not match any merchandise, you will see:  
 Merchandise not found.  
 and be returned to the menu.

* If found, the system will display the current merchandise details and allow you to:

**Enter a new name** (leave blank to keep the current one)

**Enter a new price** (leave blank to keep the current one)

**Enter a new quantity** (leave blank to keep the current one)

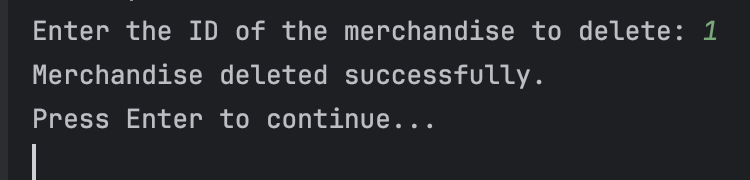
When all changes are entered, the system updates the record and confirms:  
 Merchandise updated successfully.

Press **Enter** to return to the Merchandise Management Menu.

### Delete merchandise:

Select **option 3** to remove merchandise from inventory.

You will be prompted to:



* **Enter the ID** of the merchandise you wish to delete.

If found, the merchandise will be permanently deleted, and the system will confirm:  
 Merchandise deleted successfully.

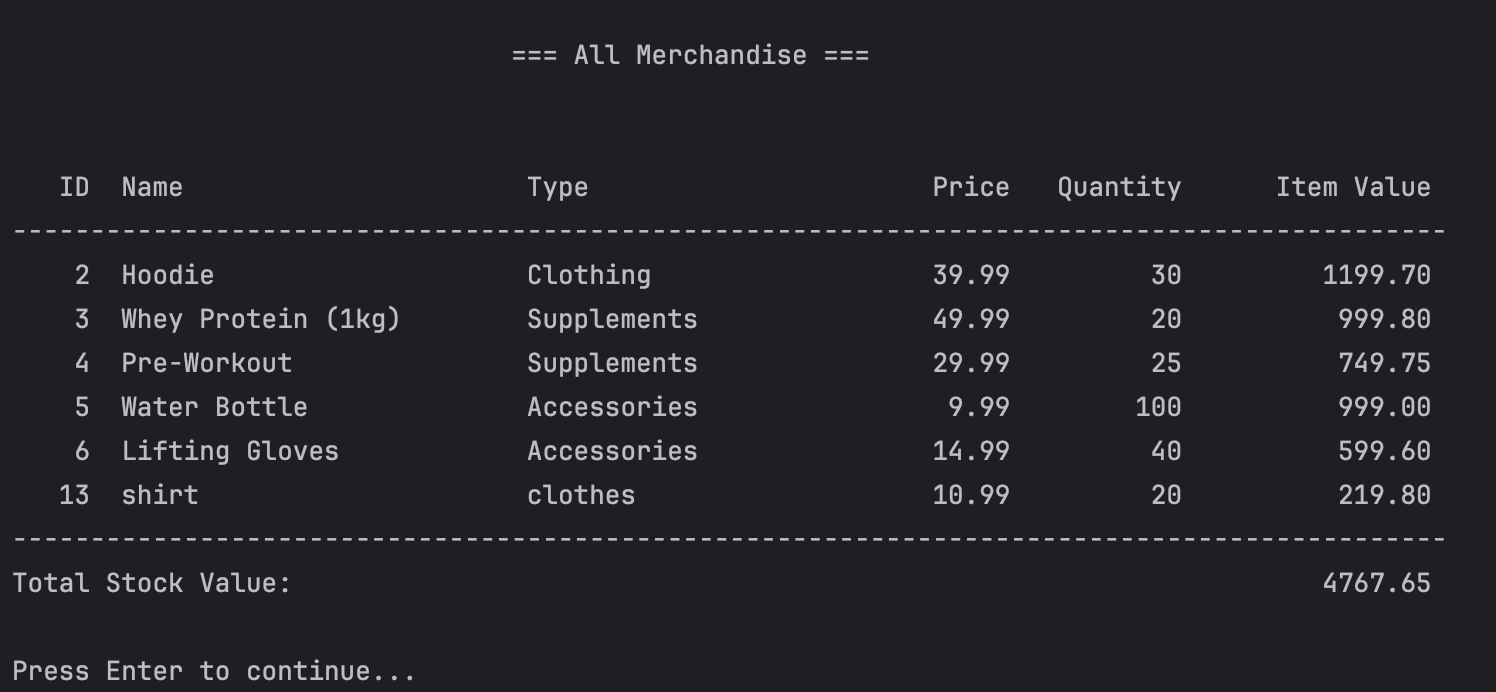
If no matching ID exists, you will see:  
 No merchandise found with that ID.

Press **Enter** to return to the Merchandise Management Menu.

### Print all merchandise report and total stock value:

Select **option 4** to view a formatted report of all merchandise in the system, along with the total stock value.

The report displays:



At the bottom, the **Total Stock Value** is calculated and displayed.

Press **Enter** to return to the Merchandise Management Menu.

### Back to Admin Menu:

- Select **option 5** to exit the Merchandise Management Menu and return to the main Admin Menu.

# Trainer Menu

Once you log in as a **Trainer**, you will be presented with the **Trainer Menu**, which allows you to manage a trainer's workout classes (create, update, delete or view), purchase a membership, show the gym merchandise or log out.

The menu options are:

A screenshot of a computer

AI-generated content may be incorrect.

## Manage my workout classes

**Option 1** will open a submenu to manage your workout classes:

A screenshot of a black screen

AI-generated content may be incorrect.

### Create a Workout Class:

Select **option 1** to create a new workout class.

You will be prompted to:

A screenshot of a computer program

AI-generated content may be incorrect.

* Enter the Workout Class Type ID, or type **l** to list all available class types.
* Enter a description for the new workout class.
* Enter the date of the class in **YYYY-MM-DD** format.
* Enter the time of the class in **HH:MM** format (24-hour clock).
* If all information is valid, the system will confirm:  
   *Workout class with ID: [ID] created successfully.*
* Press Enter to return to the Workout Class Management Menu.

Enter the **Workout Class Type ID** if you would like to use an existing one:

A screenshot of a computer program

AI-generated content may be incorrect.

### Edit a Workout Class:

* Select **option 2** to edit a class you have previously created.
* You will be prompted to select the class ID, then update its details.

A screenshot of a computer program

AI-generated content may be incorrect.

* Press Enter to return to the Workout Class Management Menu when finished.

### Delete a Workout Class:

* Select **option 3** to remove one of your classes from the schedule.
* You will be prompted to enter the ID of the class you wish to delete.

A black background with white text

AI-generated content may be incorrect.

* Press Enter to return to the Workout Class Management Menu when finished.

### View My Workout Classes:

* Select **option 4** to view a list of all the classes you have scheduled.
* The display includes class IDs, names, descriptions, and scheduled times.

A black screen with white text

AI-generated content may be incorrect.

* Press Enter to return to the Workout Class Management Menu when finished.

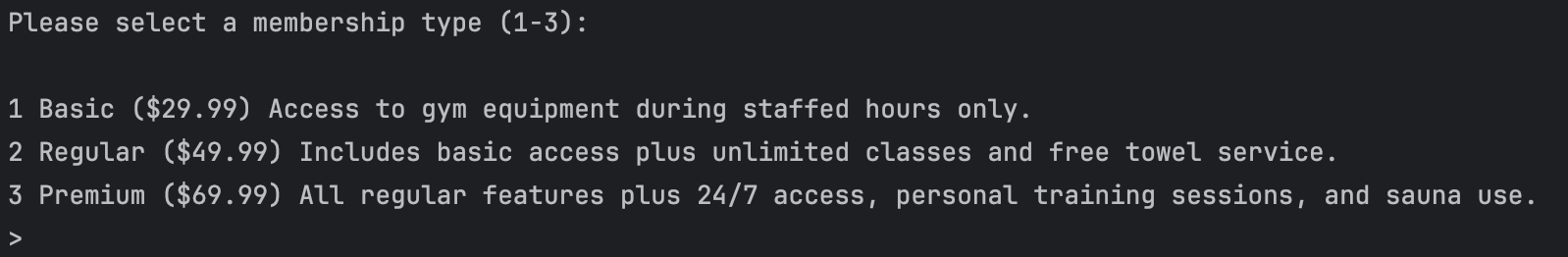
### Back to Trainer Menu:

- Select **option 5** to exit the Workout Class Management Menu and return to the main Trainer Menu

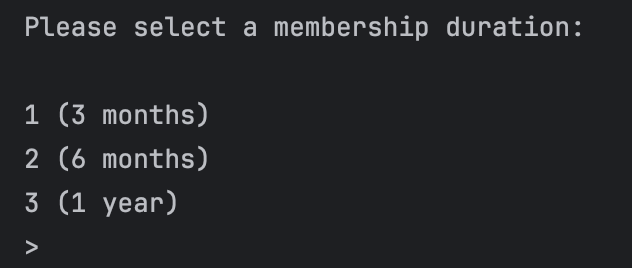
## Purchase a Membership:

Select **option** **2** to purchase a new gym membership.

You will be prompted to:



* Enter the Membership Type ID you would like to purchase.



* Enter the duration you want for your membership (3 months, 6 months, or 1 year).

If the membership is created successfully, the system will confirm

Press Enter to return to the previous menu when finished.

## View Gym Merchandise:

* Select **option** **3** to show all the gym merchandise.
* This will display a table showing all merchandise currently in the system, including the merchandise ID, name, type, and price.



Press Enter to return to the previous menu when finished.

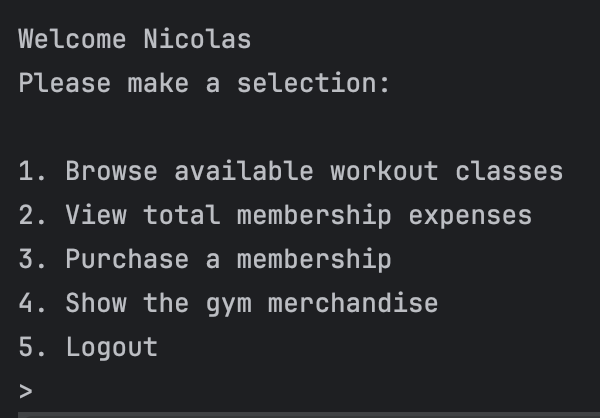
## Logout:

Select **option 4** to log out of the trainer account and return to the main login screen.

# Member Menu

Once you log in as a Member, you will be presented with the Member Menu, which allows you to browse available workout classes, view your total membership expenses, purchase a membership, show the gym merchandise, or log out.

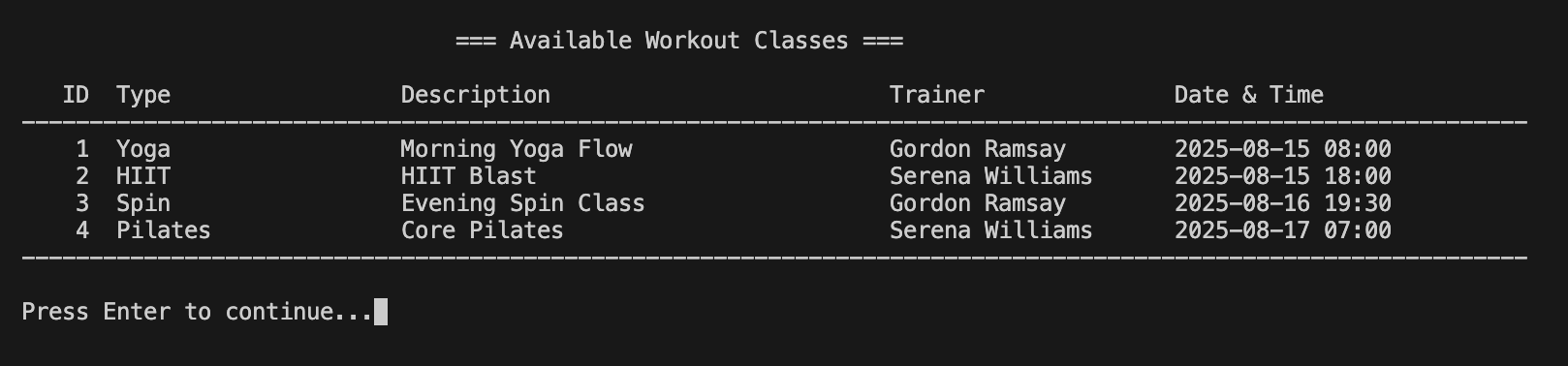
The menu options are:



## Browse Workout Classes:

Select option 1 to view a list of all workout classes currently available at the gym.

The list displays:



* Workout Class ID
* Workout Class Type
* Description
* Trainer
* Date and time of the class

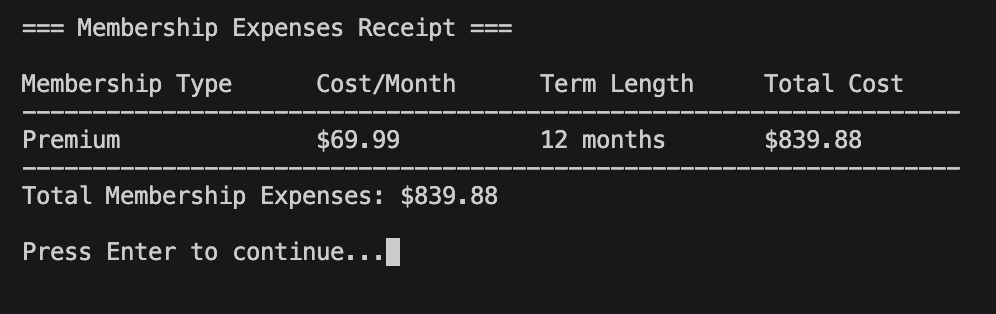
If no workout classes are available, a message will be displayed.

Press Enter to return to the menu.

## View My Memberships:

Select **option** 2to view total membership expenses.

When you choose this option, the system will display a table containing:



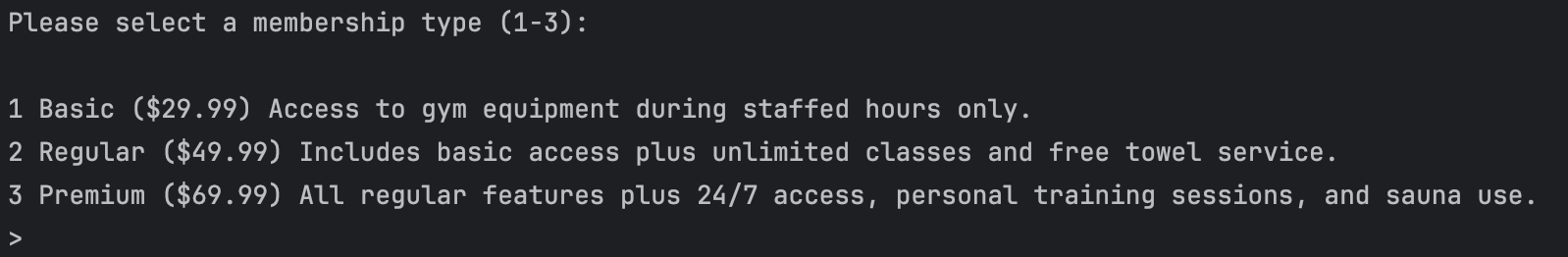
* **Membership Type** – The name of the membership plan you purchased.
* **Cost/Month** – The price you pay each month for that membership.
* **Term Length** – The total number of months in the membership term.
* **Total Cost** – The cost per month multiplied by the number of months.

After viewing your membership expenses, press **Enter** to return to the menu.

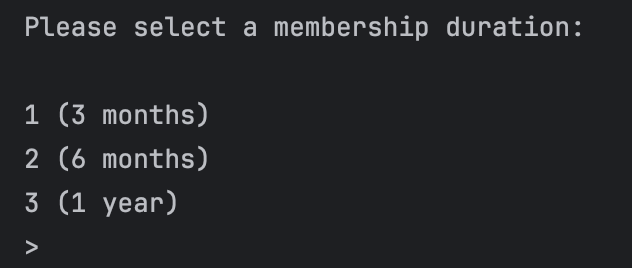
## Purchase a Membership:

Select **option** 3to purchase a new gym membership.

You will be prompted to:



* Enter the Membership Type ID you would like to purchase.



* Enter the duration you want for your membership (3 months, 6 months, or 1 year).

If the membership is created successfully, the system will confirm

Press Enter to return to the previous menu when finished.

## View Gym Merchandise:

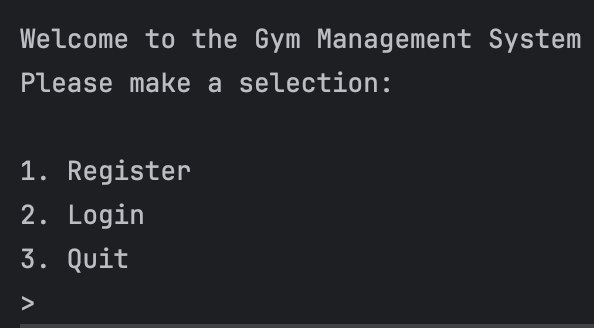
* Select **option** **4** to show all the gym merchandise.
* This will display a table showing all merchandise currently in the system, including the merchandise ID, name, type, and price



Press Enter to return to the previous menu when finished.

# Exiting the Program

* After you log out from your personal menu (Admin, Trainer, or Member), you will return to the **Main Menu**. Here you have three options:



* To exit the program, select option **3** from the Main Menu. The system will then display a goodbye message and close.

.

# Troubleshooting Tips

If you encounter any issues while using the Gym Management System, please try the following steps:

* **The program does not start or run:**
  + Verify that Java JDK 11 or higher is installed and properly configured on your system.
  + Confirm that you have opened the correct project folder in your IDE (such as Visual Studio Code or IntelliJ).
  + Use the Run or Play button in your editor to launch the application.
* **Unable to log in or register:**
  + Ensure that your username and password are entered correctly (case-sensitive).
  + Complete all required fields during registration, including username, password, email, and role.
* **Menu options show “invalid option” or do not respond:**
  + Enter the number corresponding to your menu choice and press Enter.
  + Avoid entering letters or special characters unless specifically requested.
* **Changes are not saved or data does not display:**
  + Confirm that the PostgreSQL database is running and properly connected.
  + Verify your database configuration settings in the project.
  + Make sure the necessary database tables exist and have been set up correctly.
* **Workout classes, memberships, or merchandise are missing:**
  + These items must be added by an Admin or Trainer before they appear.
  + Restart the program if new data was recently added.
  + Some features and data visibility depend on your user role.
* **Application crashes or displays errors:**
  + Review any error messages for guidance on the issue.
  + Restart the application and try again.
  + Consult the application’s log file for detailed error information if problems persist.
* **General recommendations:**
  + Always log out properly before closing the program to ensure data integrity.
  + Restart the application if you encounter unexpected behavior.

# FAQs

**Q: What user roles are available in the system?**  
 A: There are three roles—Admin, Trainer, and Member. Each role has specific access and functions within the system.

**Q: How do I register a new user?**  
 A: From the main menu, select “Register.” You’ll need to provide a username, password, email, phone number, address, and select a role.

**Q: How do I purchase a membership?**  
 A: Log in as a Member or Trainer and select the option to purchase a membership from your menu.

**Q: Can I view workout classes if I’m not a Trainer?**  
 A: Yes. Both Members and Trainers can browse available workout classes, but only Trainers can create, update, or delete classes.

**Q: How do I manage merchandise?**  
 A: Admins can add, update, or remove merchandise items and view stock reports. Trainers and Members can only view available merchandise.

**Q: What database is used, and do I need to set it up?**  
 A: The system uses PostgreSQL. You’ll need to install and configure the database, then run the provided SQL scripts to create tables and seed initial data.

**Q: How do I exit the program safely?**  
 A: From the main menu, choose the “Quit” option. Logging out from any user menu will return you to the main menu where you can then exit.

**Q: What if I enter an invalid menu option?**  
 A: The system will prompt you to enter a valid number. Just type the correct option number and press Enter.

# Contact Us

If you have any questions or need further assistance, feel free to reach out:

• **Email:**

Justin: Justin.Greenslade@keyin.com Joseph: Joseph.Gallant@keyin.com Ashton: Ashton.Dennis@keyin.com