Group Members : -

Roll no : 02 Name :Atharva shevate

Roll no : 20 Name :Sankalp Gawade

Roll no : 24 Name : Pranav karanjkar

Group Activity on JDBC Connectivity :- Gym Membership Management system

Introduction –

The Java project presented is a simple yet robust application for managing gym memberships. Leveraging the Java Database Connectivity (JDBC) API, the project facilitates seamless interaction with a MySQL database to store and manage member information. This project serves as a comprehensive solution for gym administrators to onboard new members, customize membership plans, and track membership status effectively.

Features –

1. **Database Integration:**
   The project integrates JDBC to connect with a MySQL database, allowing efficient storage and retrieval of member details.

2. **Membership Management:**
   Gym administrators can add new members dynamically, capturing essential information such as name, selected plan, membership start and end dates, and membership status.

3. **Dynamic Pricing and Plans:**
   The application offers flexibility in choosing membership plans, including basic, standard, premium, and personalized options. Pricing is

dynamically calculated based on selected plans and additional services, ensuring accurate billing for members.

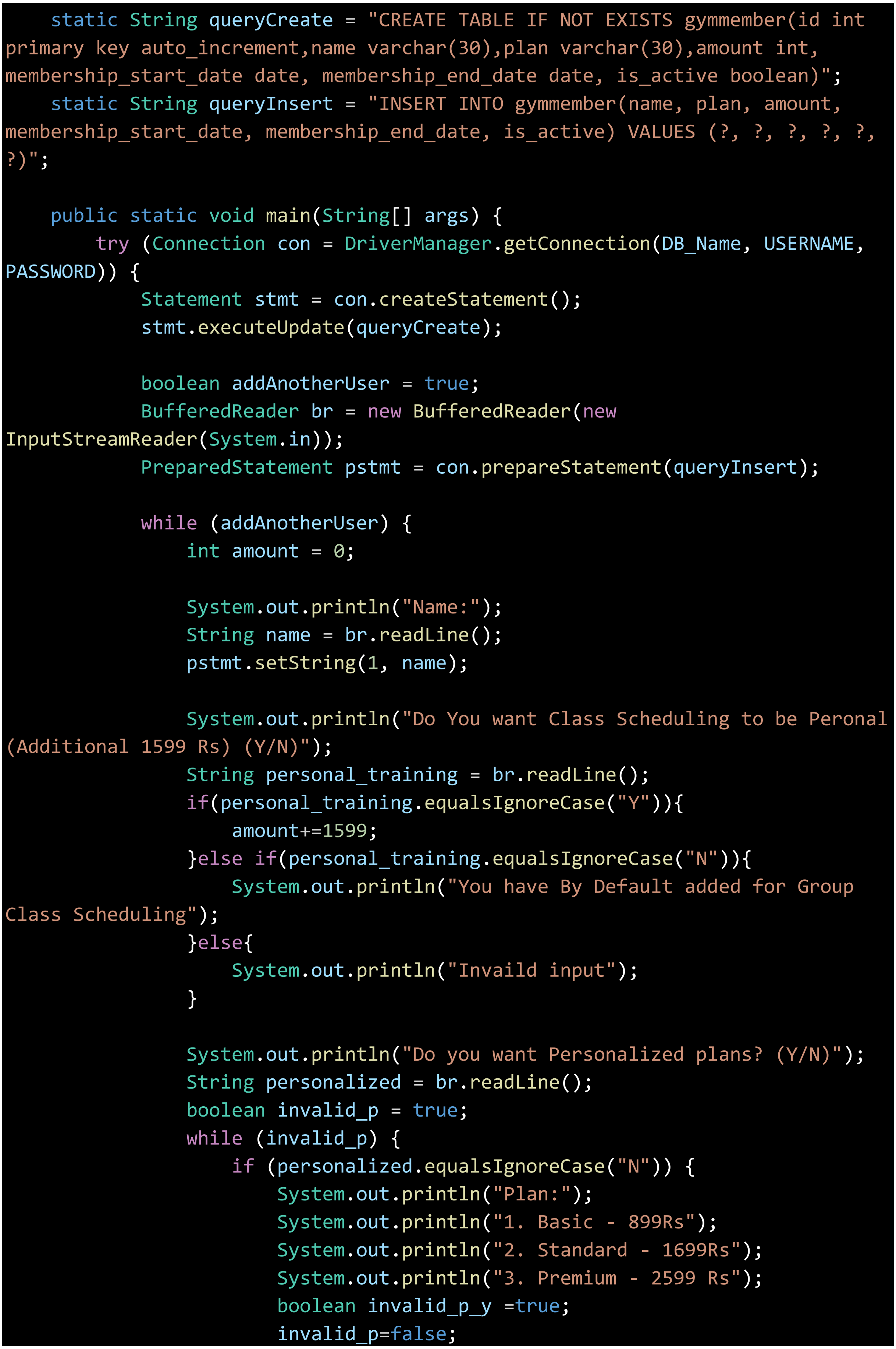
4. **Interactive User Interface:**
   The project features an interactive command-line interface, enabling seamless interaction with users for inputting member details and preferences.

5. **Membership Status Tracking**:
   Membership status is automatically determined based on the membership end date. Active memberships are flagged accordingly, providing administrators with real-time insights into active and expired memberships.

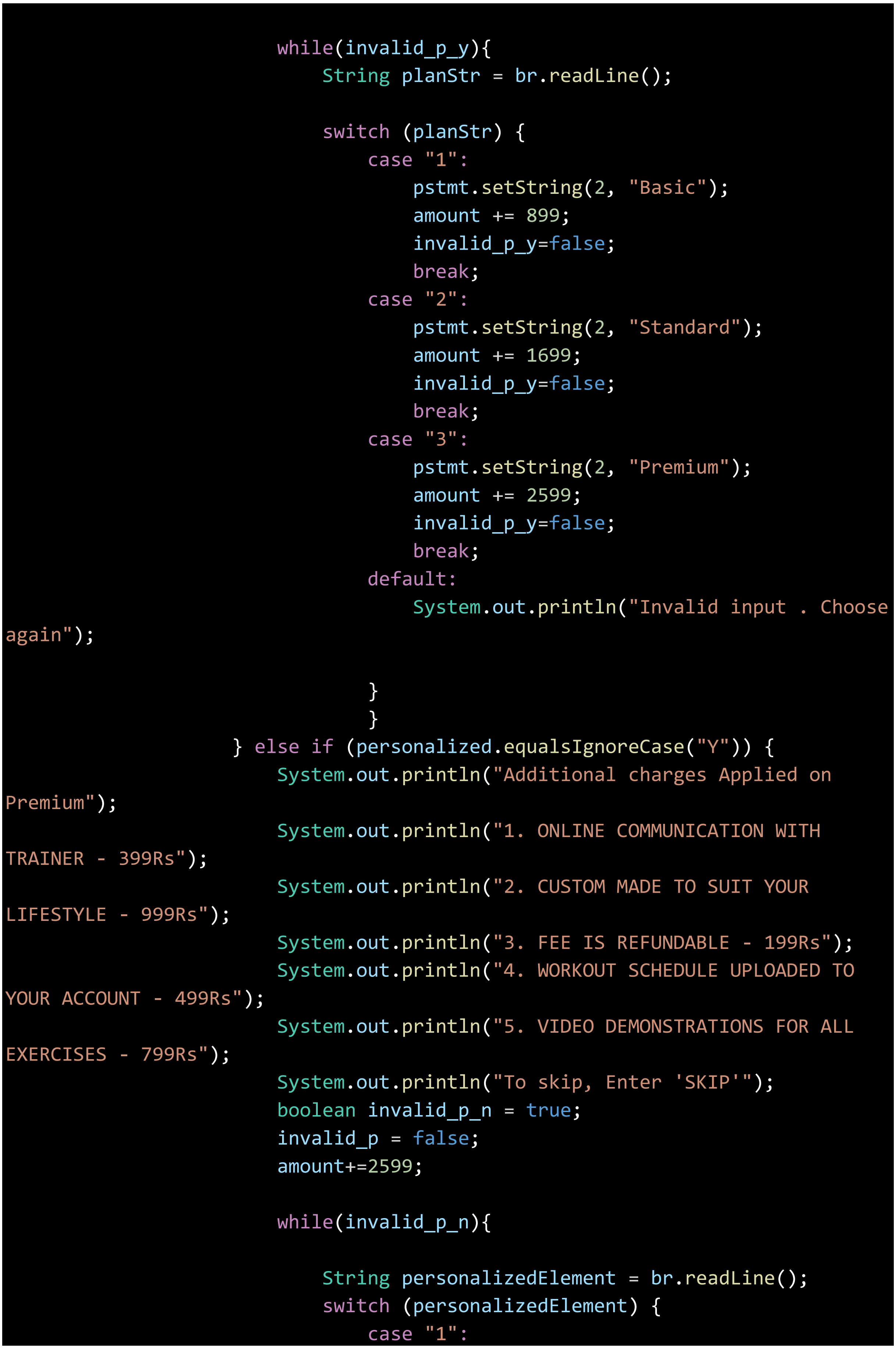
6. **Error Handling:**
   Robust error handling mechanisms are implemented to handle exceptions gracefully, ensuring uninterrupted operation and user-friendly error messages for debugging purposes.

Code –

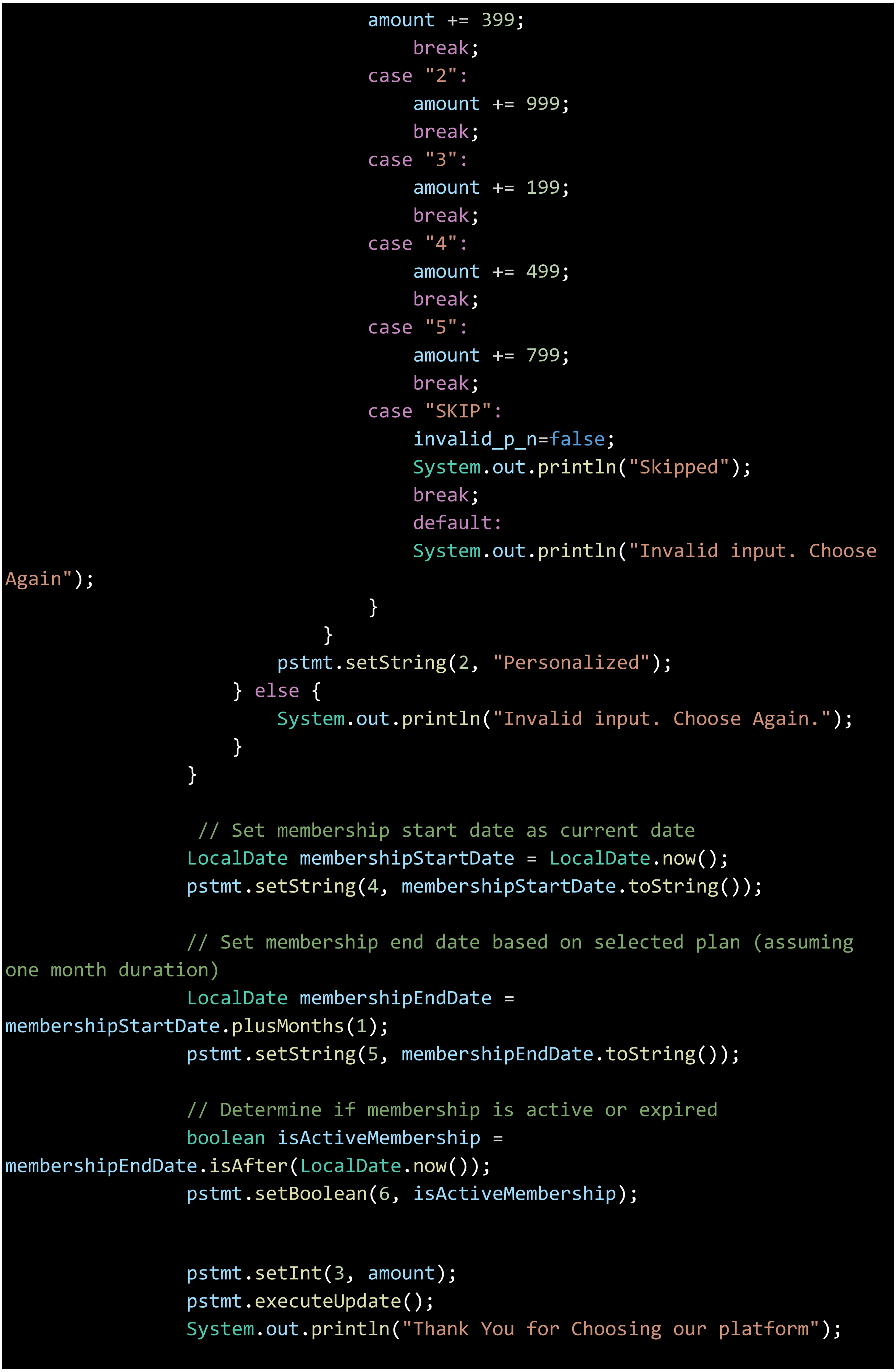
```
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.sql.*;
import java.time.LocalDate;
import java.time.format.DateTimeFormatter;

public class Main {
    static String DB_Name = "jdbc:mysql://localhost/gym";
    static String USERNAME = "root";
    static String PASSWORD = "root";
```



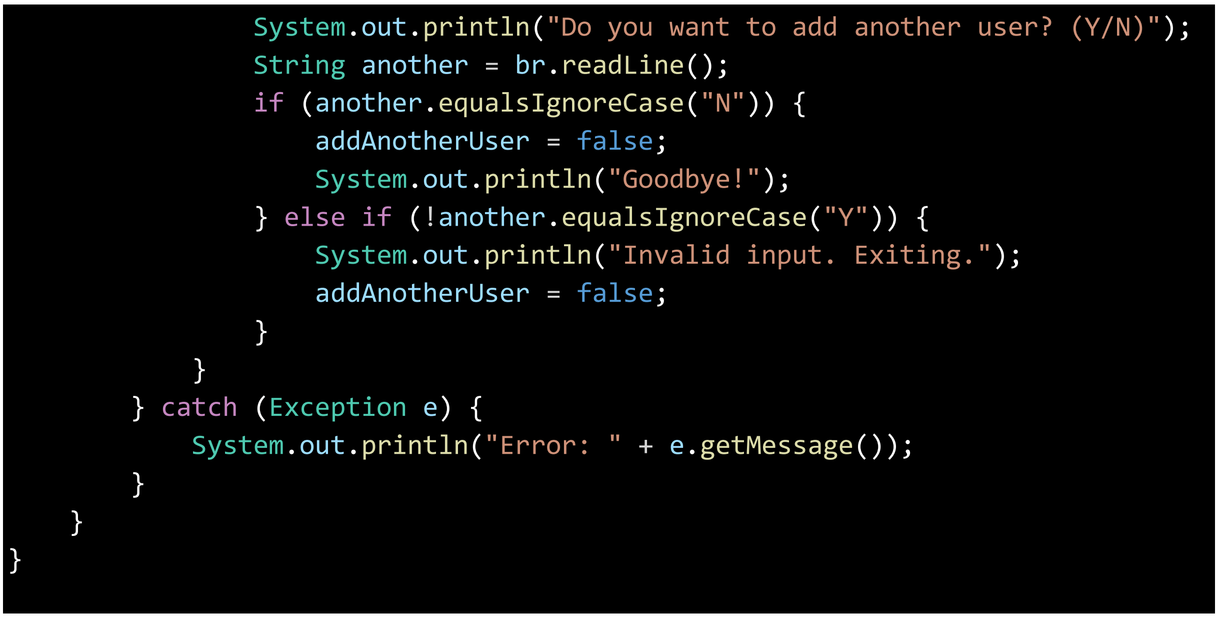
```
    static String queryCreate = "CREATE TABLE IF NOT EXISTS gymmember(id int
primary key auto_increment,name varchar(30),plan varchar(30),amount int,
membership_start_date date, membership_end_date date, is_active boolean)";
    static String queryInsert = "INSERT INTO gymmember(name, plan, amount,
membership_start_date, membership_end_date, is_active) VALUES (?, ?, ?, ?, ?,
?)";

    public static void main(String[] args) {
        try (Connection con = DriverManager.getConnection(DB_Name, USERNAME,
PASSWORD)) {
            Statement stmt = con.createStatement();
            stmt.executeUpdate(queryCreate);

            boolean addAnotherUser = true;
            BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));
            PreparedStatement pstmt = con.prepareStatement(queryInsert);

            while (addAnotherUser) {
                int amount = 0;

                System.out.println("Name:");
                String name = br.readLine();
                pstmt.setString(1, name);

                System.out.println("Do You want Class Scheduling to be Peronal
(Additional 1599 Rs) (Y/N)");
                String personal_training = br.readLine();
                if(personal_training.equalsIgnoreCase("Y")){
                    amount+=1599;
                }else if(personal_training.equalsIgnoreCase("N")){
                    System.out.println("You have By Default added for Group
Class Scheduling");
                }else{
                    System.out.println("Invaild input");
                }

                System.out.println("Do you want Personalized plans? (Y/N)");
                String personalized = br.readLine();
                boolean invalid_p = true;
                while (invalid_p) {
                    if (personalized.equalsIgnoreCase("N")) {
                        System.out.println("Plan:");
                        System.out.println("1. Basic - 899Rs");
                        System.out.println("2. Standard - 1699Rs");
                        System.out.println("3. Premium - 2599 Rs");
                        boolean invalid_p_y =true;
                        invalid_p=false;
```



```
                        while(invalid_p_y){
                            String planStr = br.readLine();

                            switch (planStr) {
                                case "1":
                                    pstmt.setString(2, "Basic");
                                    amount += 899;
                                    invalid_p_y=false;
                                    break;
                                case "2":
                                    pstmt.setString(2, "Standard");
                                    amount += 1699;
                                    invalid_p_y=false;
                                    break;
                                case "3":
                                    pstmt.setString(2, "Premium");
                                    amount += 2599;
                                    invalid_p_y=false;
                                    break;
                                default:
                                    System.out.println("Invalid input . Choose
again");

                                }
                                }
                    } else if (personalized.equalsIgnoreCase("Y")) {
                        System.out.println("Additional charges Applied on
Premium");
                        System.out.println("1. ONLINE COMMUNICATION WITH
TRAINER - 399Rs");
                        System.out.println("2. CUSTOM MADE TO SUIT YOUR
LIFESTYLE - 999Rs");
                        System.out.println("3. FEE IS REFUNDABLE - 199Rs");
                        System.out.println("4. WORKOUT SCHEDULE UPLOADED TO
YOUR ACCOUNT - 499Rs");
                        System.out.println("5. VIDEO DEMONSTRATIONS FOR ALL
EXERCISES - 799Rs");
                        System.out.println("To skip, Enter 'SKIP'");
                        boolean invalid_p_n = true;
                        invalid_p = false;
                        amount+=2599;

                        while(invalid_p_n){

                            String personalizedElement = br.readLine();
                            switch (personalizedElement) {
                                case "1":
```

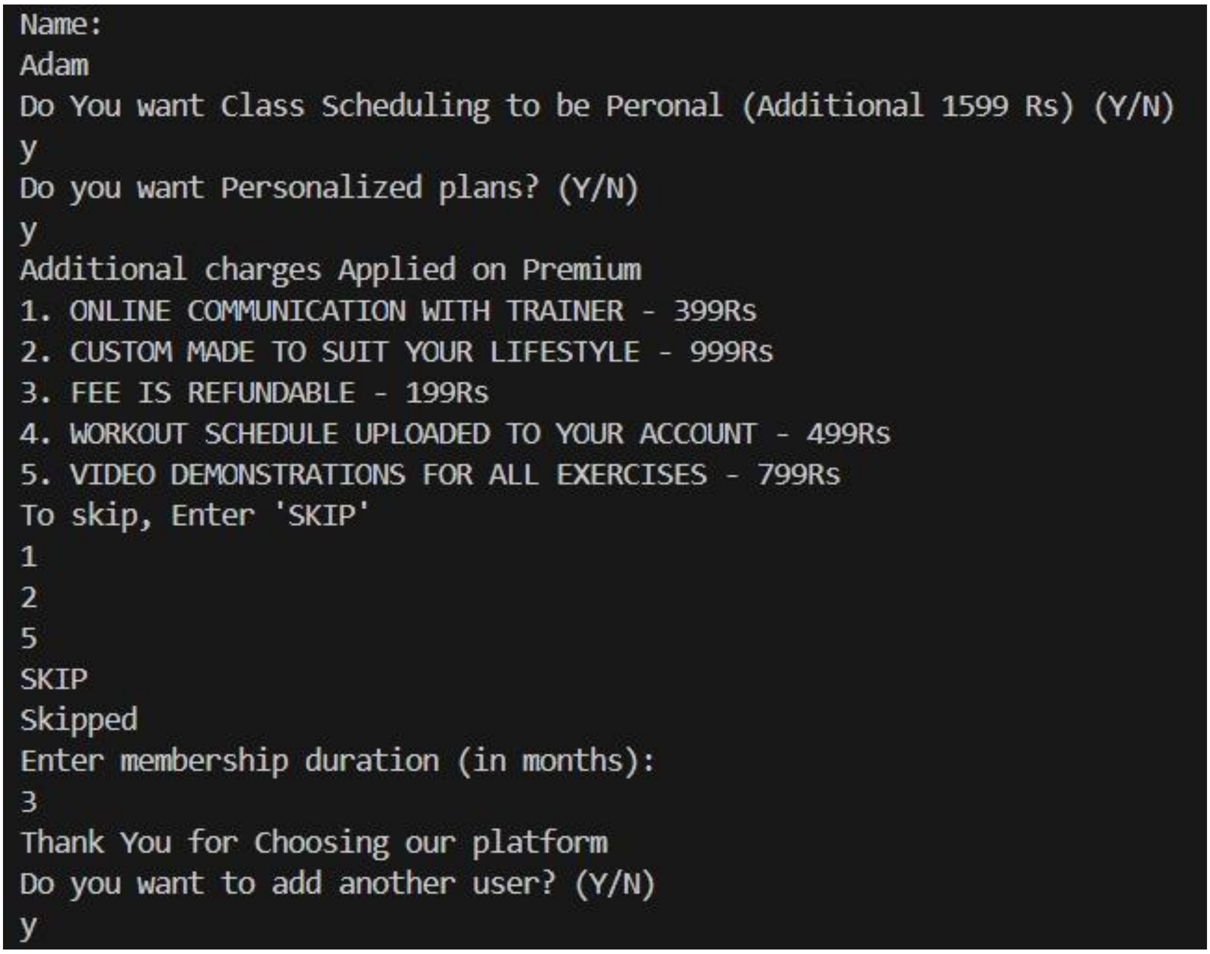


```
                                amount += 399;
                                    break;
                                case "2":
                                    amount += 999;
                                    break;
                                case "3":
                                    amount += 199;
                                    break;
                                case "4":
                                    amount += 499;
                                    break;
                                case "5":
                                    amount += 799;
                                    break;
                                case "SKIP":
                                    invalid_p_n=false;
                                    System.out.println("Skipped");
                                    break;
                                    default:
                                    System.out.println("Invalid input. Choose
Again");
                                }
                            }
                        pstmt.setString(2, "Personalized");
                    } else {
                        System.out.println("Invalid input. Choose Again.");
                    }
                }

                 // Set membership start date as current date
                LocalDate membershipStartDate = LocalDate.now();
                pstmt.setString(4, membershipStartDate.toString());

                // Set membership end date based on selected plan (assuming
one month duration)
                LocalDate membershipEndDate =
membershipStartDate.plusMonths(1);
                pstmt.setString(5, membershipEndDate.toString());

                // Determine if membership is active or expired
                boolean isActiveMembership =
membershipEndDate.isAfter(LocalDate.now());
                pstmt.setBoolean(6, isActiveMembership);


                pstmt.setInt(3, amount);
                pstmt.executeUpdate();
                System.out.println("Thank You for Choosing our platform");
```

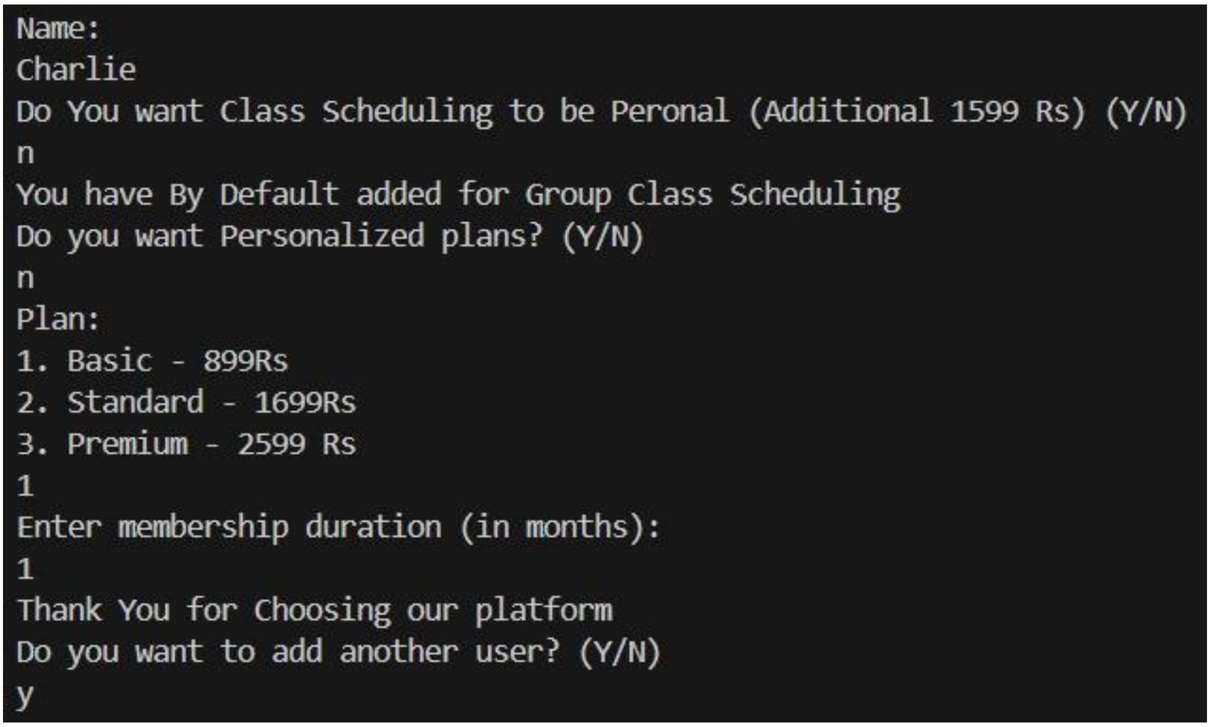


```
                System.out.println("Do you want to add another user? (Y/N)");
                String another = br.readLine();
                if (another.equalsIgnoreCase("N")) {
                    addAnotherUser = false;
                    System.out.println("Goodbye!");
                } else if (!another.equalsIgnoreCase("Y")) {
                    System.out.println("Invalid input. Exiting.");
                    addAnotherUser = false;
                }
            }
        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
        }
    }
}
```

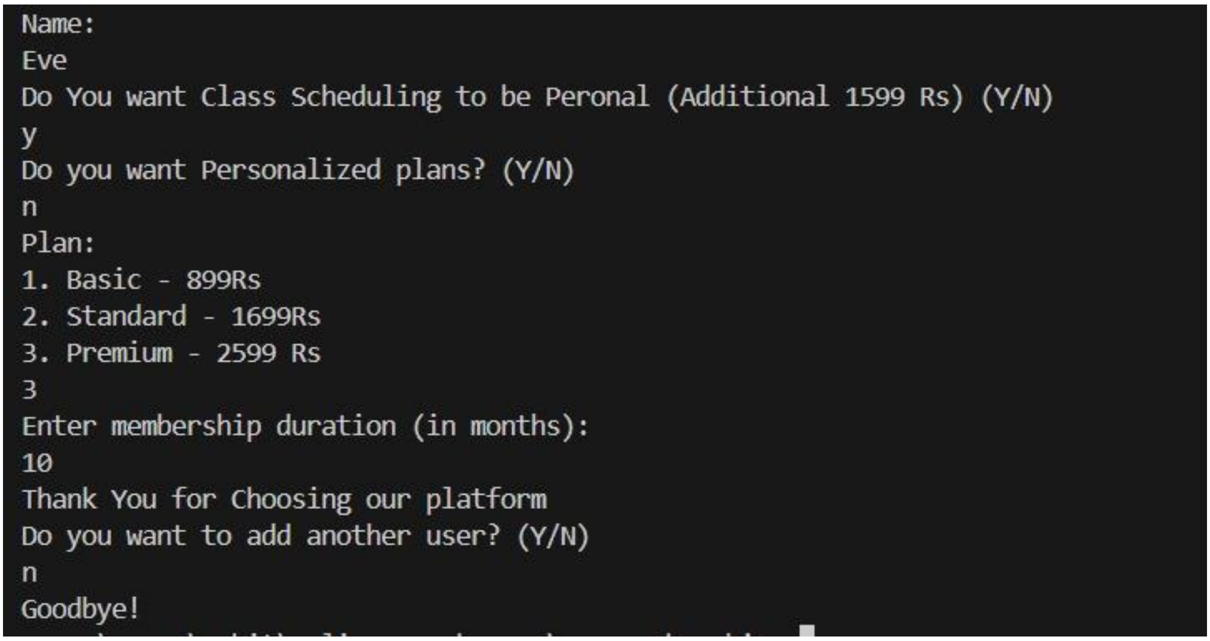
Output –



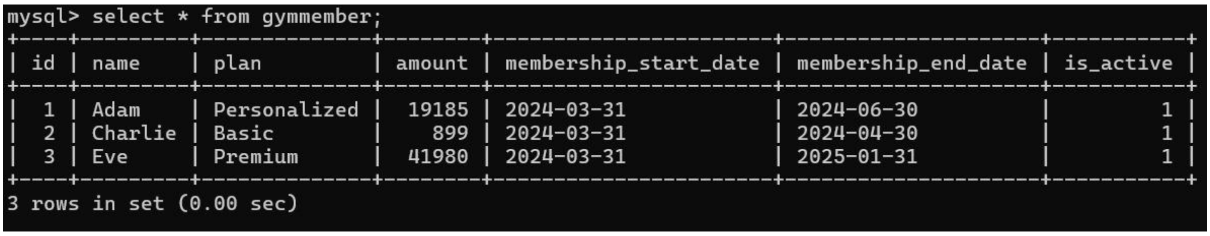
```
Name:
Adam
Do You want Class Scheduling to be Peronal (Additional 1599 Rs) (Y/N)
y
Do you want Personalized plans? (Y/N)
y
Additional charges Applied on Premium
1. ONLINE COMMUNICATION WITH TRAINER - 399Rs
2. CUSTOM MADE TO SUIT YOUR LIFESTYLE - 999Rs
3. FEE IS REFUNDABLE - 199Rs
4. WORKOUT SCHEDULE UPLOADED TO YOUR ACCOUNT - 499Rs
5. VIDEO DEMONSTRATIONS FOR ALL EXERCISES - 799Rs
To skip, Enter 'SKIP'
1
2
5
SKIP
Skipped
Enter membership duration (in months):
3
Thank You for Choosing our platform
Do you want to add another user? (Y/N)
y
```



```
Name:
Charlie
Do You want Class Scheduling to be Peronal (Additional 1599 Rs) (Y/N)
n
You have By Default added for Group Class Scheduling
Do you want Personalized plans? (Y/N)
n
Plan:
1. Basic - 899Rs
2. Standard - 1699Rs
3. Premium - 2599 Rs
1
Enter membership duration (in months):
1
Thank You for Choosing our platform
Do you want to add another user? (Y/N)
y
```



```
Name:
Eve
Do You want Class Scheduling to be Peronal (Additional 1599 Rs) (Y/N)
y
Do you want Personalized plans? (Y/N)
n
Plan:
1. Basic - 899Rs
2. Standard - 1699Rs
3. Premium - 2599 Rs
3
Enter membership duration (in months):
10
Thank You for Choosing our platform
Do you want to add another user? (Y/N)
n
Goodbye!
```



```
mysql> select * from gymmember;
+----+---------+--------------+--------+-----------------------+---------------------+-----------+
| id | name    | plan         | amount | membership_start_date | membership_end_date | is_active |
+----+---------+--------------+--------+-----------------------+---------------------+-----------+
|  1 | Adam    | Personalized |  19185 | 2024-03-31            | 2024-06-30          |         1 |
|  2 | Charlie | Basic        |    899 | 2024-03-31            | 2024-04-30          |         1 |
|  3 | Eve     | Premium      |  41980 | 2024-03-31            | 2025-01-31          |         1 |
+----+---------+--------------+--------+-----------------------+---------------------+-----------+
3 rows in set (0.00 sec)
```

Implementation Details

- **Database Schema Creation:** The project initiates by establishing a connection to a MySQL database and executing a SQL query to create a table named gymmember. This table consists of columns such as id, name, plan, amount, membership_start_date, membership_end_date, and is_active to store member details.

- **User Input Handling:** The application utilizes a BufferedReader to interactively capture user inputs from the command line interface. Users provide information such as their name, preference for class scheduling, and choice of membership plan.

- **Dynamic Pricing Calculation:** Based on user inputs, the application dynamically calculates the membership fee (amount) by considering the base plan cost and any additional services opted for by the member, such as personalized training or communication with trainers.

- **Membership Dates Management:** The project employs the LocalDate class from the java.time package to manage membership start and end dates. Membership start date is set to the current date, while the end date is calculated based on the selected plan, assuming a one-month duration. This ensures accurate tracking of membership periods.
- **Membership Status Determination:** The application evaluates the membership end date against the current date to determine whether the membership is active or expired. This information is stored in the is_active column of the database table, facilitating easy tracking of active and expired memberships.
- **PreparedStatement Execution:** To prevent SQL injection attacks and improve performance, the project utilizes PreparedStatement for executing parameterized SQL queries. This ensures secure and efficient insertion of member details into the database.
- **Interactive Loop:** The application operates within a loop to allow for the addition of multiple members in succession. After adding each member, users are prompted to decide whether to add another member or exit the application.
- **Error Handling:** Robust exception handling mechanisms are implemented using try-catch blocks to gracefully handle any errors that may arise during database operations, user input parsing, or SQL query execution. Detailed error messages are displayed to aid in debugging and troubleshooting.
-

Conclusion –

The Java project offers a robust and user-friendly solution for managing gym memberships efficiently. By leveraging JDBC for database connectivity and incorporating dynamic pricing, membership date management, and interactive user input handling, the application provides gym administrators with the tools needed to onboard new members, customize membership plans, and track membership statuses effectively. With its error-handling mechanisms and interactive command-line interface, the project ensures a seamless user experience while maintaining data integrity and security. Overall, this project serves as a reliable and comprehensive solution for gym management, enhancing the overall efficiency and satisfaction of both administrators and gym members.