Camera Rental Application Project

```
package com.mphasis.Project1;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class CameraRentalApp {
    private List<Camera> cameraList;
    private double userWalletBalance;

    public CameraRentalApp() {
        cameraList = new ArrayList<>();
        userWalletBalance = 5000.0;
    }

    public void addCamera(String brand, String model, double rentPerDay) {
        Camera camera = new Camera(brand, model, rentPerDay);
        cameraList.add(camera);
    }
    public void addCamerasToList() {
        addCamera("Nikon"," Nik345", 500.00);
        addCamera("Canon","C123", 1000.00);
        addCamera("samsung","S123", 2000.00);
        addCamera("Sony","Sony123", 3000.00);
        }

    public void removeCamera() {
        if (cameraList.isEmpty()) {
            System.out.println("No cameras available.");
```

```
        } else {
            Scanner sc = new Scanner(System.in);
            System.out.println("Enter camera Id to remove:");
            int cameraIndex = sc.nextInt();
            sc.nextLine();

            if (cameraIndex >= 0 && cameraIndex < cameraList.size()) {
                Camera removedCamera = cameraList.remove(cameraIndex);
                System.out.println("Removed camera: " + removedCamera.getBrand() + " " +
removedCamera.getModel());
            } else {
                System.out.println("Invalid camera ID.");
            }
        }

    }

    public void displayCameraList() {
        if (cameraList.isEmpty()) {
            System.out.println("No cameras available.");
        } else {
             for (int i = 0; i < cameraList.size(); i++) {
                Camera camera = cameraList.get(i);
                String status = camera.isRented() ? "Rented" : "Available";
                System.out.println((i) + ". " + camera.getBrand() + " " + camera.getModel() + " - $" +
camera.getRentPerDay() + " per day (" + status + ")");


            }
        }
    }
```

```
public void rentCamera(int cameraIndex) {
    if (cameraIndex >= 0 && cameraIndex < cameraList.size()) {
        Camera camera = cameraList.get(cameraIndex);
        if (camera.isRented()) {
            System.out.println("Camera already rented.");
        } else {
            System.out.println("Renting camera: " + camera.getBrand() + " " + camera.getModel());
            camera.setRented(true);
            System.out.println("Camera rented successfully!");
        }
    } else {
        System.out.println("Invalid camera ID.");
    }
}
public void displaySubMenu() {
    Scanner scanner = new Scanner(System.in);

    while (true) {
        System.out.println("Wallet Menu:");
        System.out.println("1.Deposit Amount");
        System.out.println("2.View Wallet Balance");
        System.out.println("3.Go Back to Main Menu");
        int choice = scanner.nextInt();
        scanner.nextLine();

        switch (choice) {
            case 1:
                System.out.print("Enter amount to deposit: ");
                double amount = scanner.nextDouble();
                scanner.nextLine();
                depositAmount(amount);
```

```
            break;
        case 2:
            viewWalletBalance();
            break;
        case 3:
            return;
        default:
            System.out.println("Invalid choice...!!!!");
      }
      System.out.println();
   }
}

public void depositAmount(double amount) {
   if (amount <= 0) {
      System.out.println("Ivalid amount...");
   } else {
      userWalletBalance += amount;
      System.out.println("Deposit successful..");
   }
}

public void viewWalletBalance() {
   System.out.println("Current wallet balance: $" + userWalletBalance);
}

public void displayMenu() {
   Scanner scanner = new Scanner(System.in);

   while (true) {
      System.out.println("Main Menu:");
```

```
System.out.println("1.My Camera");
System.out.println("2.Rent a camera");
System.out.println("3.Display available cameras");
System.out.println("4.My Wallet");
System.out.println("5.Exit");
int choice = scanner.nextInt();
scanner.nextLine();

switch (choice) {
   case 1:
            PreviousMenu();
      int ch = scanner.nextInt();
            if(ch==1)
            {
                        System.out.print("Enter camera brand: ");
          String brand = scanner.nextLine();
          scanner.nextLine();
          System.out.print("Enter camera model: ");
          String model = scanner.nextLine();
          System.out.print("Enter per-day rent amount: ");
          double rentPerDay = scanner.nextDouble();
          scanner.nextLine();
          addCamera(brand, model, rentPerDay);
          break;
       }
            else if(ch==2)
            {
                    removeCamera();
            }
            else if(ch==3)
            {
```

```
            displayCameraList();


        }
        else if (ch==4)
        {
          return;
        }
        else
        {System.out.println("Oops... try again!");
        //System.out.println("Invalid Choice Selected Please select proper choice");
        }
    case 2:
    displayCameraList();
    System.out.print("Enter Camera ID :");
    int cameraIndex = scanner.nextInt();
    scanner.nextLine();
    rentCamera(cameraIndex);
    break;


case 3:
    displayCameraList();
    break;
case 4:
    displaySubMenu();
    break;
case 5:
        System.out.println("Closing your application... \nThank you!");
    System.exit(0);
default:
        System.out.println("Oops... try again!");
    break;
```

```
            }
            System.out.println();
        }
    }

    public void PreviousMenu()
    {
            System.out.println("1.Add");
            System.out.println("2.Remove");
            System.out.println("3.View My Cameras");
            System.out.println("4.Back to Previous Menu");
            //System.out.println("Enter a choice: ");


    }


    public static void main(String[] args) {
            CameraRentalApp obj=new CameraRentalApp();
            obj.addCamerasToList();
            User obj1=new User();
            obj1.login(obj);
    }
}
```

User:

```
package com.mphasis.Project1;
import java.util.Scanner;
```

```
class User {
    private String username;
    private String password;

public User() {
            username="Akhila";
            setPassword("akhila@1234");
}

    public void login(CameraRentalApp rentalSystem) {
        Scanner sc = new Scanner(System.in);
        System.out.println("      Hello.....!!!!!   ");
        System.out.println("      ******            ");
            System.out.println("+-----------------------------------------+");
                    System.out.println("    WELCOME TO CAMERA RENTAL APP    ");
                    System.out.println("+-----------------------------------------+");
                    System.out.println("ENTER LOGIN DETAILS -");

        System.out.print("USERNAME -");
        String uname = sc.nextLine();
        System.out.print("PASSWORD -");
        String pname = sc.nextLine();

        if (uname.equals(username) && pname.equals(getPassword())) {
            System.out.println("Login successful!!");
            rentalSystem.displayMenu();
        } else {
             System.out.println("Oops... try again!");
        }
    }
```

```
	public String getUsername() {
		return username;
	}

	public void setUsername(String username) {
		this.username = username;
	}

	public String getPassword() {
		return password;
	}

	public void setPassword(String password) {
		this.password = password;
	}

}
```

Camera:

```
package com.mphasis.Project1;

public class Camera {
	 private String brand;
	    private String model;
	    private double rentPerDay;
	    private boolean isRented;
	    public Camera(String brand, String model, double rentPerDay) {
```

```
        this.brand = brand;
        this.model = model;
        this.rentPerDay = rentPerDay;
    }

    public String getBrand() {
        return brand;
    }

    public String getModel() {
        return model;
    }

    public double getRentPerDay() {
            return rentPerDay;

    }

        public boolean isRented() {
            return isRented;
        }

        public void setRented(boolean isRented) {
            this.isRented = isRented;
        }

}
```