# CAMERA RENTAL APPLICATION

```
package com.project;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
class Camera {
   private static int lastAssignedId = 0;
   private int cameraId;
   private String brand;
   private String model;
   private double rentalAmount;
   private String status;

   public Camera(String brand, String model, double rentalAmount) {
      this.cameraId = ++lastAssignedId;
      this.brand = brand;
      this.model = model;
      this.rentalAmount = rentalAmount;
      this.status = "Available";
          }
   public int getCameraId() {
      return cameraId;
   }
   public String getBrand() {
      return brand;
   }
   public String getModel() {
      return model;
   }
   public double getRentalAmount() {
      return rentalAmount;
   }
   public int getCameraID() {
      return cameraId;
          }
   public String getStatus() {
      return status;
   }
   public void setStatus(String status) {
      this.status = status;
   }
}
class User {
   private double walletAmount;
   public User(double walletAmount) {
      this.walletAmount = walletAmount;
   }
```

```
  public double getWalletAmount() {
      return walletAmount;
   }
  public void deposit(double amount) {
     walletAmount += amount;

  }
}
public class cameraRental {
   private List<Camera> cameraList;
   private User user;
   public cameraRental() {
      cameraList = new ArrayList<>();
      user = new User(1000);
   }
   public void addCamera(String brand, String model, double rentalAmount) {
      Camera camera = new Camera( brand, model, rentalAmount);
      cameraList.add(camera);
      System.out.println("YOUR CAMERA HAS BEEN SUCCESSFULLY ADDED TO THE LIST.");
   }

   public void displayCameraList() {
      if (cameraList.isEmpty()) {
         System.out.println("NO DATA PRESENT AT THIS MOMENT.");
      } else {

System.out.println("=================================================================
===");
         System.out.printf("%-10s %-15s %-10s %-15s %-10s%n", "Camera ID", "BRAND", "MODEL",
"PRICE(PER DAY)", "STATUS");

System.out.println("=================================================================
===");
         for (Camera camera : cameraList) {
            System.out.printf("%-10s %-15s %-10s %-15s %-10s%n",
                  camera.getCameraId(), camera.getBrand(), camera.getModel(),
camera.getRentalAmount(), camera.getStatus());
         }

System.out.println("=================================================================
===");
      }

   }
   public void removeCamera(int cameraId) {

      boolean found = false;
      for (Camera camera : cameraList) {
         if (camera.getCameraId() == cameraId) {
```

```
            if (camera.getStatus().equals("Rented")) {
                System.out.println("CANNOT REMOVE CAMERA AS THIS CAMERA IS CURRENTLY
RENTED");
            } else {
                cameraList.remove(camera);
                System.out.println("CAMERA SUCCESSFULLY REMOVED FROM THE LIST.");
            }
            found = true;
            break;
        }
    }
    if (!found) {
        System.out.println("CAMERA WITH THE GIVEN ID NOT FOUND IN THE LIST");
    }
  }

  public void rentCamera(int cameraId) {
          Camera rentedCamera = null;
    for (Camera camera : cameraList) {
        if (camera.getCameraId() == cameraId) {
            rentedCamera = camera;
            break;
        }
    }
    if (rentedCamera == null) {
        System.out.println("CAMERA WITH THE GIVEN ID NOT FOUND IN THE LIST");
        return;
    }
    if (rentedCamera.getStatus().equals("Rented")) {
        System.out.println("CAMERA IS ALREADY RENTED");
    } else if (user.getWalletAmount() >= rentedCamera.getRentalAmount()) {
        user.deposit(-rentedCamera.getRentalAmount());
        rentedCamera.setStatus("Rented");
        System.out.println("YOUR TRANSACTION FOR CAMERA - " + rentedCamera.getBrand() + " " +
rentedCamera.getModel() + " with rent INR " + rentedCamera.getRentalAmount() + " HAS
SUCCESSFULLY COMPLETED.");
    } else {
          System.out.println("ERROR : TRANSCATION FAILED DUE TO INSUFFICIENT WALLET
BALANCE. PLEASE DEPOSIT THE AMOUNT TO YOUR WALLET");
    }
  }


  public void viewWalletAmount() {
          System.out.println("YOUR CURRENT WALLET BALANCE IS: INR  "+  user.getWalletAmount());
          }
  public  void depositToWallet(double amount) {
    user.deposit(amount);
    double totalAmount = user.getWalletAmount();
```

```
      System.out.println("YOUR WALLET BALANCE UPDATED SUCCESSFULLY. CURRENT WALLET
BALANCE - " + totalAmount );

  }
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      cameraRental application = new cameraRental();

      boolean isLoggedIn = false;
      String text = "WELCOME TO CAMERA RENTAL APP";
      int length = text.length() + 4;

      System.out.println("+" + "-".repeat(length) + "+");

      System.out.println("|  " + text + "  |");

      System.out.println("+" + "-".repeat(length) + "+");
      System.out.println("PLEASE LOGIN TO CONTINUE -");


      while (!isLoggedIn) {

         System.out.print("USERNAME - ");
        String username = scanner.next();
         System.out.print("PASSWORD - ");
         String password = scanner.next();


         if (validateLogin(username, password)) {
            isLoggedIn = true;

         } else {
            System.out.println("INVALID USERNAME OR PASSWORD. PLEASE TRY AGAIN.");
         }
      }
      while (true) {


         System.out.println("1. MY CAMERA\n2. RENT A CAMERA\n3. VIEW ALL CAMERA\n4. MY
WALLET\n5. EXIT ");
         int ch =scanner.nextInt();
         switch(ch) {

         case 1:
           System.out.println("1. ADD\n2. REMOVE\n3. VIEW MY CAMERAS\n4. GO TO PREVIOUS
MENU");
            int choice = scanner.nextInt();
            switch(choice) {
               case 1:
```

```
                System.out.print("ENTER THE CAMERA BRAND : ");
                String brand = scanner.next();

                System.out.print("ENTER THE MODEL: ");
                String model = scanner.next();

                System.out.print("ENTER THE PER DAY PRICE (INR): ");
                double rentalAmount = scanner.nextDouble();
                application.addCamera(brand, model, rentalAmount);

                break;
            case 2:
                    application.displayCameraList();
                    System.out.print("ENTER THE CAMERA ID TO BE REMOVED : ");
                int cameraId = scanner.nextInt();
                application.removeCamera(cameraId);
                break;
             case 3:
                application.displayCameraList();
                break;
             case 4:
                    System.out.println("GO TO PREVIOUS MENU");
                    System.out.println("1. ADD\n2. REMOVE\n3. VIEW MY CAMERAS\n4. GO TO
PREVIOUS MENU");
                 int choiceMenu = scanner.nextInt();
                 switch(choiceMenu) {
                 case 1:

                     System.out.print("ENTER THE CAMERA BRAND : ");
                    String brand1 = scanner.next();

                    System.out.print("ENTER THE MODEL: ");
                    String model1 = scanner.next();

                    System.out.print("ENTER THE PER DAY PRICE (INR): ");
                    double rentalAmount1 = scanner.nextDouble();
                    application.addCamera(brand1, model1, rentalAmount1);

                    break;

                case 2:
                     application.displayCameraList();
                     System.out.print("ENTER THE CAMERA ID TO BE REMOVED : ");
                    int cameraId1 = scanner.nextInt();
                    application.removeCamera(cameraId1);
                    break;
                case 3:
                    application.displayCameraList();
```

```
                    break;

                case 4:
                    System.out.println("back to menu");
                    break;

                }
            }

        break;
        case 2:
          System.out.println("FOLLOWING IS THE LIST OF AVALIABLE CAMERA(S) - ");
          application.displayCameraList();
          System.out.print("ENTER THE CAMERA ID TO RENT:");
          int CameraID = scanner.nextInt();
           application.rentCamera(CameraID);
           break;

        case 3:
           application.displayCameraList();
           break;

        case 4:
          application.viewWalletAmount();

            System.out.println("DO YOU WANT TO DEPOSIT MORE AMOUNT TO YOUR WALLET? (1.
YES 2. NO)");
            int c = scanner.nextInt();
            switch (c) {
                case 1:
                    System.out.print("ENTER THE AMOUNT (INR) - ");
                    double additionalAmount = scanner.nextDouble();
                     application.depositToWallet( additionalAmount);
                    break;
                case 2:
                    System.out.println("Exiting the deposit process.");

                    }
            break;
        case 5:
            System.out.println("THANKS FOR USING THIS APPLICATION");
            System.exit(0);
        default:
            System.out.println("Invalid choice. Please try again.");
        }


    }
    }
```

```
    private static boolean validateLogin(String username, String password) {

                return username.equals("admin") && password.equals("admin123");
        }
}
```