**package** com.camera;

**import** java.util.InputMismatchException;

**import** java.util.Scanner;

**public** **class** UserInfo {

**private** String username;

**private** String password;

**private** **double** walletAmt;

**private** Scanner scanner = **new** Scanner(System.***in***);

//Setters & Getters

**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;

}

**public** **double** getWalletAmt() {

**return** walletAmt;

}

**public** **void** setWalletAmt(**double** walletAmt) {

**this**.walletAmt = walletAmt;

}

**public** **void** AddAmtToWallet() {

// Display current balance with formatting

System.***out***.println("Current Wallet Balance: INR " + walletAmt);

// Prompt for deposit choice with numerical options

**int** choiceInt = 0; // Initialize choiceInt to 0

**boolean** validChoice = **false**;

**while** (!validChoice) {

System.***out***.println("Do you want to add money? (1 for YES, 2 for NO)");

**try** {

choiceInt = scanner.nextInt();

scanner.nextLine(); // Consume newline

**if** (choiceInt == 1 || choiceInt == 2) {

validChoice = **true**;

} **else** {

System.***out***.println("Invalid choice. Please enter 1 for YES or 2 for NO.");

}

} **catch** (InputMismatchException e) {

System.***out***.println("Invalid input. Please enter a valid number (1 or 2).");

scanner.nextLine(); // Consume invalid input

}

}

**if** (choiceInt == 1) {

// Read amount from the keyboard

System.***out***.println("Enter amount to add to the wallet:");

**double** amountToAdd = scanner.nextDouble();

scanner.nextLine(); // Consume newline

// Add amount to walletAmt variable

walletAmt += amountToAdd;

// Update wallet amount within the UserInfo instance itself

**this**.setWalletAmt(walletAmt);

// Show updated wallet balance with formatting

System.***out***.println("Your Wallet Balance Updated Successfully Current wallet balance: INR " + walletAmt);

} **else** {

// User chose not to add money, no further action needed

}

}

}

**package** com.camera;

**public** **class** Camera {

**private** **int** camid;

**private** String brand;

**private** String model;

**private** **float** rentperday;

**private** String status;

**public** Camera(**int** camid, String brand, String model, **float** rentperday, String status) {

**super**();

**this**.camid = camid;

**this**.brand = brand;

**this**.model = model;

**this**.rentperday = rentperday;

**this**.status = status;

}

**public** **int** getCamid() {

**return** camid;

}

**public** **void** setCamid(**int** camid) {

**this**.camid = camid;

}

**public** String getBrand() {

**return** brand;

}

**public** **void** setBrand(String brand) {

**this**.brand = brand;

}

**public** String getModel() {

**return** model;

}

**public** **void** setModel(String model) {

**this**.model = model;

}

**public** **float** getRentperday() {

**return** rentperday;

}

**public** **void** setRentperday(**float** rentperday) {

**this**.rentperday = rentperday;

}

**public** String getStatus() {

**return** status;

}

**public** **void** setStatus(String status) {

**this**.status = status;

}

@Override

**public** String toString() {

**return** "Camera ID: " + **this**.camid +

", Brand: " + **this**.brand +

", Model: " + **this**.model +

", Rent per Day: " + **this**.rentperday +

", Status: " + **this**.status;

}

}

**package** com.camera;

**import** java.util.ArrayList;

**import** java.util.Comparator;

**import** java.util.List;

**import** java.util.Scanner;

**class** CameraOperations {

List<Camera> rentAcamera;

**public** CameraOperations() {

rentAcamera = **new** ArrayList<Camera>();

rentAcamera.add(**new** Camera(1, "Samsung", "DS123", 500.0f, "Available"));

rentAcamera.add(**new** Camera(2, "Sony", "HD214", 500.0f, "Available"));

rentAcamera.add(**new** Camera(3, "Panasonic", "DS123", 500.0f, "Available"));

rentAcamera.add(**new** Camera(4, "Canon", "XLR", 500.0f, "Available"));

rentAcamera.add(**new** Camera(5, "Fujitsu", "J5", 500.0f, "Rented"));

rentAcamera.add(**new** Camera(6, "Sony", "HD226", 500.0f, "Available"));

rentAcamera.add(**new** Camera(8, "LG", "L123", 500.0f, "Available"));

rentAcamera.add(**new** Camera(9, "Canon", "XPL", 500.0f, "Rented"));

rentAcamera.add(**new** Camera(10, "Chroma", "CT", 500.0f, "Available"));

rentAcamera.add(**new** Camera(13, "Canon", "Digital", 123.0f, "Available"));

rentAcamera.add(**new** Camera(14, "NIKON", "DSLR\_D7500", 500.0f, "Available"));

rentAcamera.add(**new** Camera(15, "Sony", "DSLR12", 200.0f, "Rented"));

rentAcamera.add(**new** Camera(19, "SONY", "SONY1234", 123.0f, "Available"));

rentAcamera.add(**new** Camera(20, "Canon", "5050", 25000.0f, "Available"));

rentAcamera.add(**new** Camera(21, "Nikon", "2030", 500.0f, "Available"));

}

**public** String addCamera(**int** camid, String brand, String model, **float** rentPerDay) {

**for** (Camera camera : rentAcamera) {

**if** (camera.getCamid() == camid) {

**return** "Camera ID already exists!";

}

}

Camera camera = **new** Camera(camid, brand, model, rentPerDay, "Available");

rentAcamera.add(camera);

**return** "YOUR CAMERA HAS BEEN ADDED SUCCESSFULLY TO THE LIST ";

}

**public** List<Camera> showAllCameras() {

rentAcamera.sort(Comparator.*comparingInt*(Camera::getCamid));

**if** (rentAcamera.isEmpty()) {

System.***out***.println("No cameras available!");

} **else** {

System.***out***.println("FOLLOWING IS THE LIST OF AVAILABLE CAMERA(S)");

System.***out***.println("CAMERA ID\tBRAND\t\tMODEL\t\tPRICE(PER DAY)\tSTATUS");

System.***out***.println("---------\t------\t\t-----\t\t--------------\t------");

**for** (Camera camera : rentAcamera) {

System.***out***.printf("%-10d\t%-10s\t%-14s\t%-16.1f\t%s\n", camera.getCamid(), camera.getBrand(),

camera.getModel(), camera.getRentperday(), camera.getStatus());

}

}

**return** rentAcamera;

}

**public** String deleteCamera(**int** camid) {

**for** (Camera camera : rentAcamera) {

**if** (camera.getCamid() == camid) {

**if** (!camera.getStatus().equals("Rented")) {

rentAcamera.remove(camera);

**return** "Camera deleted successfully!";

} **else** {

**return** "Cannot delete a rented camera!";

}

}

}

**return** "Camera not found!";

}

**public** String rentACamera(**int** camid, **double** walletAmt, UserInfo userInfo) {

**float** rentPerDay = 0.0f;

Camera foundCamera = **null**;

// Find the camera and retrieve its rent per day

**for** (Camera camera : rentAcamera) {

**if** (camera.getCamid() == camid) {

rentPerDay = camera.getRentperday();

foundCamera = camera;

**break**;

}

}

// If camera not found, return appropriate message

**if** (foundCamera == **null**) {

**return** "Camera not found!";

}

// Handle camera availability and wallet balance

**switch** (foundCamera.getStatus()) {

**case** "Available":

**if** (walletAmt >= rentPerDay) {

// Rent the camera

foundCamera.setStatus("Rented");

walletAmt -= rentPerDay;

// Update wallet amount in UserInfo instance

userInfo.setWalletAmt(walletAmt);

**return** "Camera rented successfully!";

} **else** {

System.***out***.println("YOUR CURRENT WALLET BALANCE IS\nINR." + walletAmt);

// Prompt for deposit choice, handling string input and invalid choices

**boolean** validChoice = **false**;

**int** choice = 0;

Scanner scanner = **null**;

**while** (!validChoice) {

System.***out***.println("ERROR: TRANSACTION FAILED DUE TO INSUFFICIENT WALLET BALANCE.PLEASE DEPOSIT THE AMOUNT TO YOUR WALLET.");

System.***out***.println("DO YOU WANT TO DEPOSIT MORE AMOUNT TO YOUR WALLET? (1 for YES, 2 for NO)");

scanner = **new** Scanner(System.***in***); // Assign a new Scanner inside the loop

String choiceString = scanner.nextLine();

**try** {

choice = Integer.*parseInt*(choiceString); // Attempt to parse as integer

**if** (choice == 1 || choice == 2) {

validChoice = **true**; // Only valid if 1 or 2

} **else** {

System.***out***.println("Invalid choice. Please enter 1 for YES or 2 for NO.");

}

} **catch** (NumberFormatException e) {

System.***out***.println("Invalid input. Please enter 1 for YES or 2 for NO.");

}

}

**if** (choice == 1) {

System.***out***.println("ENTER THE AMOUNT (INR) - ");

**double** amountToAdd = scanner.nextDouble();

scanner.nextLine(); // Consume newline

// Add amount to walletAmt variable

walletAmt += amountToAdd;

// Update wallet amount in UserInfo instance

userInfo.setWalletAmt(walletAmt);

System.***out***.println(

"YOUR WALLET BALANCE UPDATED SUCCESSFULLY. CURRENT WALLET BALANCE\nINR." + walletAmt);

}

**return** " ";

}

**case** "Rented":

**return** "Camera is already rented!";

**default**:

// Handle unexpected cases

**return** "An unexpected error occurred.";

}

}

}

**package** com.camera;

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Scanner;

**public** **class** CameraMain {

**public** **static** **void** main(String[] args) {

System.***out***.println("+------------------------------------------+");

System.***out***.println("| WELCOME TO CAMERA RENTAL APPLICATION |");

System.***out***.println("+------------------------------------------+");

UserInfo user = **new** UserInfo();

user.setUsername("admin");

user.setPassword("admin@123");

user.setWalletAmt(10000.00);

System.***out***.println("PLEASE LOGIN TO CONTINUE");

Scanner scanner = **new** Scanner(System.***in***);

System.***out***.println("Enter username:");

String username = scanner.nextLine();

System.***out***.println("Enter Password:");

String password = scanner.nextLine();

**if** (username.equalsIgnoreCase(user.getUsername()) && password.equalsIgnoreCase(user.getPassword())) {

**boolean** exitApp = **false**;

System.***out***.println("Login Successful");

CameraOperations cameraOps = **new** CameraOperations();

**while** (!exitApp) {

System.***out***.println("Choose an option:");

System.***out***.println("1. My Camera");

System.***out***.println("2. Rent A Camera");

System.***out***.println("3. View All Camera");

System.***out***.println("4. My Wallet");

System.***out***.println("5. Exit");

**int** choice = scanner.nextInt();

scanner.nextLine();

**switch** (choice) {

**case** 1:

System.***out***.println("My Camera Menu:");

System.***out***.println("1. Add Camera");

System.***out***.println("2. Remove Camera");

System.***out***.println("3. View Cameras");

System.***out***.println("4. Back to Main Menu");

**int** subChoice = scanner.nextInt();

scanner.nextLine();

**switch** (subChoice) {

**case** 1:

System.***out***.println("Enter camera ID:");

**int** camId = scanner.nextInt();

scanner.nextLine();

System.***out***.println("Enter camera brand:");

String brand = scanner.nextLine();

System.***out***.println("Enter camera model:");

String model = scanner.nextLine();

System.***out***.println("Enter rent per day:");

**float** rentPerDay = scanner.nextFloat();

scanner.nextLine();

String addStatus = cameraOps.addCamera(camId, brand, model, rentPerDay);

System.***out***.println(addStatus);

**break**;

**case** 2:

cameraOps.showAllCameras();

System.***out***.println("Enter camera ID to delete:");

**int** camIdToDelete = scanner.nextInt();

scanner.nextLine();

String deleteStatus = cameraOps.deleteCamera(camIdToDelete);

System.***out***.println(deleteStatus);

**break**;

**case** 3:

List<Camera> cameras = cameraOps.showAllCameras();

**if** (cameras.isEmpty()) {

System.***out***.println("No cameras available!");

} **else** {

System.***out***.println("FOLLOWING IS THE LIST OF AVAILABLE CAMERA(S)");

System.***out***.println("CAMERA ID\tBRAND\t\tMODEL\t\tPRICE(PER DAY)\tSTATUS");

System.***out***.println("---------\t------\t\t-----\t\t--------------\t------");

**for** (Camera camera : cameras) {

System.***out***.println(camera.toString());

}

}

**break**;

**case** 4:

// Back to Main Menu

System.***out***.println(" Back to Main Menu");

**continue**;

**default**:

System.***out***.println("Invalid choice.");

**break**;

}

**break**;

**case** 2:

cameraOps.showAllCameras();

System.***out***.println("Enter camera ID to rent:");

**int** camIdToRent = scanner.nextInt();

scanner.nextLine();

String rentStatus = cameraOps.rentACamera(camIdToRent, user.getWalletAmt(), user);

System.***out***.println(rentStatus);

**break**;

**case** 3:

cameraOps.showAllCameras();

**break**;

**case** 4:

user.AddAmtToWallet();

**break**;

**case** 5:

exitApp = **true**;

**break**;

**default**:

System.***out***.println("Invalid choice.");

**break**;

}

}

System.***out***.println("Exiting application. Have a nice day!");

} **else** {

System.***out***.println("Incorrect username/password");

}

}

}