**SOURCE CODE**

**1.Main Class**

package com.cemerarental.main;

import com.cemerarental.utitlity.CameraRentalUtility;

public class App {

public static void main(String[] args) {

// **TODO** Auto-generated method stub

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

System.***out***.println("Welcome to Camera Rental App");

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

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

CameraRentalUtility.*login*();

System.***out***.println("Thank you Visit Again!");

}

}

**2. Utility Class**

package com.cemerarental.utitlity;

import java.util.Scanner;

import com.camerarental.service.CameraService;

public class CameraRentalUtility {

static Scanner *sc* = new Scanner(System.***in***);

//login method

public static void login() {

//reading useername and password through keyboard

System.***out***.print("USERNAME - ");

String emailid = *sc*.next();

System.***out***.print("PASSWORD - ");

String password = *sc*.next();

//checking username and password correchat or not

if(emailid.equalsIgnoreCase("harish@gmail.com") && password.equals("harish@123")) {

System.***out***.println("SUCCESSFULLY LOGIN ");

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

//if username and password correct calling mainOption method

*mainOption*();

}else {

//if username or password wrong displaying message

System.***out***.println("USERNAME OR PASSWORD INCORRECT ");

}

}

//mainOption method

private static void mainOption() {

CameraService cs = new CameraService();

int choice;

String con="";

do {

//printing options

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

System.***out***.println("1: MY CAMERA ");

System.***out***.println("2: RENT CAMERA ");

System.***out***.println("3: VIEW ALL CAMERAS ");

System.***out***.println("4: MY WALLET ");

System.***out***.println("5: EXIT");

System.***out***.print("PLEASE ENTER YOUR CHOICE: ");

//reading the choice from above list

choice = *sc*.nextInt();

switch(choice) {

//if choice is 1 it will call subMenuOption method

case 1: *subMenuOption*(cs);

break;

//if choice is 2 it will call rentCamera method

case 2: cs.rentCamera();

break;

//if choice is 3 it will call viewAllCamera method

case 3: cs.viewAllCamera();

break;

//if choice is 4 it will call myWallet method

case 4: cs.myWallet();

break;

//if choice is 5 it will call closeApp method

case 5: *closeApp*();

break;

//if choice is not in above list it will print wrong choice

default:System.***out***.println("WRONG CHOICE ");

break;

}

//it will ask yes or no for to continue or exit

System.***out***.print("DO YOU WANT TO CONTINUE(YES/NO) ");

con = *sc*.next();

if(con.equalsIgnoreCase("yes")) {

con = "1";

}else {

con = "0";

}

}while(con.equalsIgnoreCase("1"));

}

//subMenuOption method

private static void subMenuOption(CameraService cs) {

int choice;

String con="";

//CameraService cs = new CameraService();

do {

//printing the sub menu list

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

System.***out***.println("1: ADD");

System.***out***.println("2: REMOVE");

System.***out***.println("3: VIEW ALL CAMERAS ");

System.***out***.println("4. GO TO PREVIOUS MENU ");

System.***out***.print("PLEASE ENTER YOUR CHOICE: ");

//reading choice through keyboard

choice = *sc*.nextInt();

switch(choice) {

//if choice is 1 it will call addCamera method

case 1: cs.addCamera();

break;

//if choice is 2 it will call removeCamera method

case 2: cs.removeCamera();

break;

//if choice is 3 it will call viewAllCamera method

case 3: cs.viewAllCamera();

break;

//if choice is 4 it will call mainOption method

case 4: *mainOption*();

break;

//if choice is not in above list then print wrong choice

default:System.***out***.println("WRONG CHOICE ");

break;

}

System.***out***.print("DO YOU WANT TO CONTINUE(YES/NO) ");

//it will ask yes or no for to continue or not

con = *sc*.next();

if(con.equalsIgnoreCase("yes")) {

con = "1";

}else {

con = "0";

}

}while(con.equalsIgnoreCase("1"));

}

//closeApp method

private static void closeApp() {

//printing thank you message

System.***out***.println("\nClosing your application... \nThank you!");

}

}

**3. Camera class**

package com.camerarental.bean;

public class Camera {

//camera properties

private int id;

private String brand;

private String model;

private float perDayPrice;

private String status;

//camera properties getter and setter methods

public String getStatus() {

return status;

}

public void setStatus(String status) {

this.status = status;

}

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 getPerDayPrice() {

return perDayPrice;

}

public void setPerDayPrice(float perDayPrice) {

this.perDayPrice = perDayPrice;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

//parameter less constructor

public Camera() {

super();

}

//parameterized constructor

public Camera(String brand, String model, float perDayPrice, int id, String status) {

super();

this.brand = brand;

this.model = model;

this.perDayPrice = perDayPrice;

this.id = id;

this.status = status;

}

//Overriding toString method

*@Override*

public String toString() {

return " "+ id+" " + brand + " " + model + " " + perDayPrice + " "+ status;

}

}

**4.Service Class**

package com.camerarental.service;

import java.util.ArrayList;

import java.util.Iterator;

import java.util.List;

import java.util.Scanner;

import com.camerarental.bean.Camera;

public class CameraService {

static List<Camera> *listOfCamera* = new ArrayList<>();

static int *id* = 0;

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

//addCamera method

public void addCamera() {

//reading camera details to add camera

System.***out***.print("ENTER THE CAMERA BRAND -");

String brand = sc.next();

System.***out***.print("ENTER THE CAMERA MODEL - ");

String model = sc.next();

System.***out***.print("ENTER THE PER DAY PRICE (INR) - ");

float amount = sc.nextFloat();

//setting camera details

Camera cc = new Camera();

cc.setBrand(brand);

cc.setModel(model);

cc.setPerDayPrice(amount);

//listOfCamera.add(cc);

//if camera added then setting camera personal details

if(*listOfCamera*.add(cc)) {

*id*++;

cc.setId(*id*);

cc.setStatus("Available");

}

System.***out***.println("YOUR CAMERA HAS BEEN SUCCESSFULLY ADDED TO THE LIST.");

}

//removeCamera method

public void removeCamera() {

//if camera is not available we cant remove camera from list

if(*listOfCamera*.size() == 0) {

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

}

//if camera is available we can remove camera from list

else {

//calling viewAllCamera method to see available cameras to remove camera

viewAllCamera();

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

System.***out***.print("ENTER THE CAMERA ID TO REMOVE - ");

//reading id through keyboard to remove camera

int id1 = sc.nextInt();

int flag = 0;

Iterator<Camera>li = *listOfCamera*.iterator();

//checking with id camera available or not from the list of cameras

for(Camera cc : *listOfCamera*) {

if(cc.getId() == id1) {

//if id is available remove camera

*listOfCamera*.remove(id1-1);

*id*--;

flag++;

System.***out***.println("CAMERA SUCCESSFULLY REMOVED FROM THE LIST.");

break;

}

}

//if id is not available then print message

if(flag == 0) {

System.***out***.println("CAMERA NOT EXIST WITH THIS ID.");

}

}

}

//viewAllCamera method

public void viewAllCamera() {

//if no cameras print message

if(*listOfCamera*.size() == 0) {

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

}

else {

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

System.***out***.println("CAMERA ID BRAND MODEL PRICE(PER DAY) STATUS");

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

//if cameras available display all available cameras

Iterator<Camera> li = *listOfCamera*.iterator();

while(li.hasNext()) {

Camera cc = li.next();

System.***out***.println(cc); // toString method

}

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

}

}

static float *walletAmount* = 500;

//myWallet method

public void myWallet() {

System.***out***.println("YOUR CURRENT WALLET BALANCE IS INR."+*walletAmount*);

System.***out***.print("DO YOU WANT TO DEPOSIT MORE AMOUNT INTO YOUR WALLET?(YES/NO) - ");

String opt = sc.next();

if(opt.equalsIgnoreCase("yes")) {

System.***out***.print("ENTER YOUR AMOUNT (INR) - ");

int addWallet = sc.nextInt();

//adding amount to wallet amount

*walletAmount* += addWallet;

System.***out***.println("YOUR WALLET BALANCE UPDATED SUCCESSFULLY. CURRENT WALLET BALANCE (INR) - "+ *walletAmount*);

}

}

//rentCamera method

public void rentCamera() {

//calling viewAllCamera method to see available cameras to get a camera rent

viewAllCamera();

System.***out***.print("ENTER THE CAMERA ID YOU WANT TO RENT - ");

int id = sc.nextInt();

int flag = 0;

Iterator<Camera> li = *listOfCamera*.iterator();

while(li.hasNext()) {

Camera cam = li.next();

//checking camera available or not for rent and i have sufficient wallet amount or not to take camera rent

if(cam.getId() == id && cam.getStatus().equals("Available") && *walletAmount* >= cam.getPerDayPrice()) {

flag++;

cam.setStatus("Rented");

*walletAmount* -= cam.getPerDayPrice();

System.***out***.println("CAMERA RENTED SUCCESSFULLY ");

}

}

if(flag == 0) {

System.***out***.println("ERROR : TRANSACTION FAILED DUE TO INSUFFICIENT BALANCE PLEASE DEPOSIT AMOUNT TO YOUR WALLET\nCAMERA NOT EXIST WITH THIS ID.");

}

}

}