package java\_program;

import java.util.Date;

import java.util.Scanner;

import java.util.regex.Pattern;

public class CustomException extends Exception{

public CustomException (String str)

{

// calling the system of parent Exception

super(str);

}

}

class DriverInfo {

private int id;

private String phoneNumber;

public DriverInfo() {

super();

}

public DriverInfo(int id, String phoneNumber) {

super();

this.id = id;

this.phoneNumber = phoneNumber;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getPhoneNumber() {

return phoneNumber;

}

public void setPhoneNumber(String phoneNumber) {

this.phoneNumber = phoneNumber;

}

}

class GenerateParkingLotInfo {

TicketInfo ticketInfo = new TicketInfo();

public static boolean plateNumberValidator(String plateNumber) {

try {

boolean validated = true;

if(plateNumber == null || !(Pattern.matches("^[A-Z]{1,3}-[0-9]{1,6}$", plateNumber))) {

validated = false;

System.err.println("Insert a valid plate number");

throw new CustomException("Insert a valid plate number");

}

return validated;

}catch (CustomException e) {

// e.printStackTrace();

return false;

}

}

public boolean phoneNumberValidator(String phoneNumber) {

try {

boolean validated = true;

if(phoneNumber == null || !(Pattern.matches("^[1-9][0-9]{3,14}$", phoneNumber))) {

validated = false;

System.err.println("Insert a valid phone number");

throw new CustomException("Insert a valid phone number");

}

return validated;

}catch (CustomException e) {

// e.printStackTrace();

return false;

}

}

public boolean minuteValidator(String minute) {

try {

boolean validated = true;

if(minute == null || !(Pattern.matches("^[1-9][0-9]{0,6}$", minute.toString()))) {

validated = false;

System.err.println("Insert a valid minute");

throw new CustomException("Insert a valid minute");

}

return validated;

}catch (CustomException e) {

// e.printStackTrace();

return false;

}

}

public void generateTicketInfo() {

String generateInput;

String plateNumber = null;

String phoneNumber = null;

String minute = null;

Date ticketGeneratedDate = new Date();

TicketInfo ticketInfo = new TicketInfo();

Scanner input = new Scanner(System.in);

System.out.println("Enter your vehicle model: ");

ticketInfo.setModel(input.next());

System.out.println("Enter your vehicle color: ");

ticketInfo.setColor(input.next());

// do..while loop for accepting license plate info

do{

System.out.println("Enter your license plate (ex, ABC-1234): ");

plateNumber = input.next();

}while(!plateNumberValidator(plateNumber));

ticketInfo.setLicPlate(plateNumber);

// do..while loop for accepting staying time info

do{

System.out.println("How long are you staying? Put in minutes: ");

minute = input.next();

}while(!minuteValidator(minute));

ticketInfo.setTime(Integer.parseInt(minute));

// do..while loop for accepting phone number info

do{

System.out.println("Enter your phone number: ");

phoneNumber = input.next();

}

while(!phoneNumberValidator(phoneNumber));

ticketInfo.setPhoneNumber(phoneNumber);

ticketInfo.setTicketGeneratedDate(ticketGeneratedDate);

System.out.println("\n");

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

System.out.println("\t\*\*\*\*\* PARKING TICKET \*\*\*\*\*");

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

System.out.println(""

+ "| Date: " + ticketInfo.getTicketGeneratedDate() + "\n"

+ "| Model: " + ticketInfo.getModel() + "\n"

+ "| Color: " + ticketInfo.getColor() + "\n"

+ "| License Plate: " + ticketInfo.getLicPlate());

System.out.println("| Total = $" + ticketInfo.getTotalPrice());

System.out.println("\_\_\_\_\_\_\_\_\_\_\_");

System.out.println("Thank you!\n");

System.out.println("Do you want to continue? Insert Yes(Y) or press any key to exit ");

generateInput = input.next();

// recursive

if(generateInput.equalsIgnoreCase("Yes") ||

generateInput.equalsIgnoreCase("Y"))

{

this.generateTicketInfo();

}else {

System.exit(0);

}

}

public static void main(String[] args) {

GenerateParkingLotInfo generateParkingLotInfo = new GenerateParkingLotInfo();

generateParkingLotInfo.generateTicketInfo();

}

}

class TicketInfo extends VehicleInfo{

private int id;

private Integer time;

private Date ticketGeneratedDate;

//price per hour is 15 and equal to 0.25 USD per minute

final double PRICEPERMINUTE = 0.25;

public TicketInfo() {

super();

// TODO Auto-generated constructor stub

}

public TicketInfo(int id, String model, String licPlate, String color) {

super(id, model, licPlate, color);

// TODO Auto-generated constructor stub

}

public TicketInfo(int id, Integer time) {

super();

this.id = id;

this.time = time;

}

public double getTotalPrice(){

return getTime() \* PRICEPERMINUTE;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public Integer getTime() {

return time;

}

public void setTime(Integer time) {

this.time = time;

}

public Date getTicketGeneratedDate() {

return ticketGeneratedDate;

}

public void setTicketGeneratedDate(Date ticketGeneratedDate) {

this.ticketGeneratedDate = ticketGeneratedDate;

}

}

class VehicleInfo extends DriverInfo {

private int id;

private String model;

private String licPlate;

private String color;

public VehicleInfo() {

super();

// TODO Auto-generated constructor stub

}

public VehicleInfo(int id, String model, String licPlate, String color) {

super();

this.id = id;

this.model = model;

this.licPlate = licPlate;

this.color = color;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getModel() {

return model;

}

public void setModel(String model) {

this.model = model;

}

public String getLicPlate() {

return licPlate;

}

public void setLicPlate(String licPlate) {

this.licPlate = licPlate;

}

public String getColor() {

return color;

}

public void setColor(String color) {

this.color = color;

}

}