Task Calculation Application

PropertyClass.java

**package** Com.tax.calculation;

**public** **class** PropertyClass {

**double** value;

**int** built\_up\_area;

**int** age\_of\_land;

String located;

**double** tax;

**public** **double** getTax() {

**return** tax;

}

**public** **void** setTax(**double** tax) {

**this**.tax = tax;

}

**public** PropertyClass(**double** value, **int** built\_up\_area, **int** age\_of\_land, String located) {

**super**();

**this**.value = value;

**this**.built\_up\_area = built\_up\_area;

**this**.age\_of\_land = age\_of\_land;

**this**.located = located;

}

**public** **double** getValue() {

**return** value;

}

**public** **void** setValue(**double** value) {

**this**.value = value;

}

**public** **int** getBuilt\_up\_area() {

**return** built\_up\_area;

}

**public** **void** setBuilt\_up\_area(**int** built\_up\_area) {

**this**.built\_up\_area = built\_up\_area;

}

**public** **int** getAge\_of\_land() {

**return** age\_of\_land;

}

**public** **void** setAge\_of\_land(**int** age\_of\_land) {

**this**.age\_of\_land = age\_of\_land;

}

**public** String getLocated() {

**return** located;

}

**public** **void** setLocated(String located) {

**this**.located = located;

}

**public** PropertyClass() {

**super**();

}

}

VehicleClass.java

**package** Com.tax.calculation;

**public** **class** VehicleClass {

**int** registration\_number;

String brand;

**int** max\_velocity;

**int** no\_of\_seats;

String type\_of\_vehicle;

**double** purchase\_cost;

**double** vehicle\_tax;

**public** **int** getRegistration\_number() {

**return** registration\_number;

}

**public** **void** setRegistration\_number(**int** registration\_number) {

**this**.registration\_number = registration\_number;

}

**public** String getBrand() {

**return** brand;

}

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

**this**.brand = brand;

}

**public** **int** getMax\_velocity() {

**return** max\_velocity;

}

**public** **void** setMax\_velocity(**int** max\_velocity) {

**this**.max\_velocity = max\_velocity;

}

**public** **int** getNo\_of\_seats() {

**return** no\_of\_seats;

}

**public** **void** setNo\_of\_seats(**int** no\_of\_seats) {

**this**.no\_of\_seats = no\_of\_seats;

}

**public** String getType\_of\_vehicle() {

**return** type\_of\_vehicle;

}

**public** **void** setType\_of\_vehicle(String type\_of\_vehicle) {

**this**.type\_of\_vehicle = type\_of\_vehicle;

}

**public** **double** getPurchase\_cost() {

**return** purchase\_cost;

}

**public** **void** setPurchase\_cost(**double** purchase\_cost) {

**this**.purchase\_cost = purchase\_cost;

}

**public** **double** getVehicle\_tax() {

**return** vehicle\_tax;

}

**public** **void** setVehicle\_tax(**double** vehicle\_tax) {

**this**.vehicle\_tax = vehicle\_tax;

}

**public** VehicleClass(**int** registration\_number, String brand, **int** max\_velocity, **int** no\_of\_seats,

String type\_of\_vehicle, **double** purchase\_cost, **double** vehicle\_tax) {

**super**();

**this**.registration\_number = registration\_number;

**this**.brand = brand;

**this**.max\_velocity = max\_velocity;

**this**.no\_of\_seats = no\_of\_seats;

**this**.type\_of\_vehicle = type\_of\_vehicle;

**this**.purchase\_cost = purchase\_cost;

**this**.vehicle\_tax = vehicle\_tax;

}

**public** VehicleClass() {

**super**();

}

}

PropertyOperations.java

package Com.tax.calculation;

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class PropertyOperations {

PropertyClass pc = new PropertyClass();

List<PropertyClass> pl = new ArrayList<PropertyClass>();

Scanner s = new Scanner(System.in);

public void addProperty(ArrayList<PropertyClass> pl) throws ExceptionClass

{

System.out.println("ENTER THE PROPERTY DETAILS -");

System.out.print("ENTER THE BASE VALUE OF LAND - ");

double basevalue=s.nextDouble();

if(basevalue<=0)

{

throw new ExceptionClass("Base value should be non zero and positive only");

}else

{

pc.setValue(basevalue);

}

System.out.print("ENTER THE BUILT-UP AREA OF LAND - ");

pc.setBuilt\_up\_area(s.nextInt());

System.out.print("ENTER THE AGE OF LAND - ");

int age=s.nextInt();

if(age<=0)

{

throw new ExceptionClass("Age of building should be non-zero positive");

}

else

{

pc.setAge\_of\_land(age);

}

System.out.print("IS THE LAND LOCATED IN CITY?(Y:YES,N:NO) - ");

String located=s.next();

if(located.equals("y") || located.equals("n") || located.equals("Y")||located.equals("N"))

{

pc.setLocated(located);

}

else

{

throw new ExceptionClass("Enter only y for YES and n for NO");

}

PropertyClass pc1 = new PropertyClass(pc.getValue(), pc.getBuilt\_up\_area(), pc.getAge\_of\_land(),

pc.getLocated());

pl.add(pc1);

}

public void setTax(ArrayList<PropertyClass> pl) throws ExceptionClass {

System.out.println("ENTER THE PROPERTY ID TO CALCULATE THE TAX - ");

int id =s.nextInt();

if(pl.size()==0)

{

throw new ExceptionClass("List is empty");

}

if(id<0 && id>pl.size())

{

throw new ExceptionClass("Id value must be starting from 1");

}

else

{

if(pl.get(id-1).located.equalsIgnoreCase("y"))

{

pl.get(id-1).tax=(pl.get(id-1).built\_up\_area\*pl.get(id-1).age\_of\_land\*pl.get(id-1).value)+(0.5\*pl.get(id-1).built\_up\_area);

}

else

{

pl.get(id-1).tax=(pl.get(id-1).built\_up\_area\*pl.get(id-1).age\_of\_land\*pl.get(id-1).value);

}

System.out.println("PROPERTY TAX FOR PROPERTY ID - "+id+" IS "+pl.get(id-1).tax);

}

}

public void displayDetails(ArrayList<PropertyClass> pl)

{

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

System.out.println("ID\t\tBUILT-UP AREA\tBASE PRICE\tAGE(YEARS)\tIN CITY\t\tPROPERTY TAX");

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

int i=1;

for (PropertyClass pc : pl) {

System.out.print(i+"\t\t");

System.out.print(pc.built\_up\_area+ "\t\t");

System.out.print(pc.value + "\t\t");

System.out.print(pc.age\_of\_land + "\t\t\t");

System.out.print(pc.located+"\t\t");

System.out.print(pc.tax);

System.out.println();

i++;

}

}

}

VehicleOperations.java

package Com.tax.calculation;

import java.io.BufferedReader;

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class VehicleOperations {

VehicleClass vc = new VehicleClass();

Scanner s = new Scanner(System.in);

List<VehicleClass> vl = new ArrayList<VehicleClass>();

public void addVehicelDetails(ArrayList<VehicleClass> vl) throws ExceptionClass

{

System.out.print("ENTER THE VEHICLE REGISTRATION NUMBER");

int reg\_number=s.nextInt();

int dummy=reg\_number;

int count=0;

while(dummy!=0)

{

dummy/=10;

count++;

}

if(count!=4||reg\_number==0000)

{

throw new ExceptionClass("Please enter the valid registration number");

}else

{

vc.setRegistration\_number(reg\_number);

}

System.out.print("ENTER THE BRAND OF THE VEHICLE");

String brand=s.next();

vc.setBrand(brand);

System.out.print("ENTER THE MAXIMUM VELOCITY OF THE VEHICLE(KMPH) - ");

int velocity=s.nextInt();

if(velocity<120 || velocity>300 )

{

throw new ExceptionClass("Velocity must be in a range between 120kmph-300kmph");

}else

{

vc.setMax\_velocity(velocity);

}

System.out.print("ENTER CAPACITY(NUMBER OF SEATS) OF THE VEICLE - ");

int seats=s.nextInt();

if(seats<2 || seats >50)

{

throw new ExceptionClass("Seats range should be 2 to 50");

}

else

{

vc.setNo\_of\_seats(seats);

}

System.out.print("CHOOSE THE TYPE OF VEHICLE - \n1.PETROL DRIVEN\n2.DIESEL DRIVEN\n3.CNG/LPG DRIVEN");

int vchoice=s.nextInt();

if(vchoice<0 || vchoice>3)

{

throw new ExceptionClass("Select with the range only");

}

else

{

switch(vchoice)

{

case 1:vc.setType\_of\_vehicle("PETROL");

break;

case 2:vc.setType\_of\_vehicle("DIESEL");

break;

case 3:vc.setType\_of\_vehicle("CNG/LPG");

break;

}

}

System.out.print("ENTER THE PURCHASE COST OF THE VEHICLE-");

double cost=s.nextDouble();

if(cost<50000 || cost >100000)

{

throw new ExceptionClass("cost must be within range of 50000 - 100000");

}

else

{

vc.setPurchase\_cost(cost);

}

vl.add(new VehicleClass(vc.getRegistration\_number(),vc.getBrand(),vc.getMax\_velocity(),vc.getNo\_of\_seats(),vc.getType\_of\_vehicle(),vc.getPurchase\_cost(),vc.getVehicle\_tax()));

}

public void setTax(ArrayList<VehicleClass> vl) throws ExceptionClass

{ if(vl.size()==0)

{

throw new ExceptionClass("Cannot perform on empty list");

}

else

{

System.out.print("ENTER THE REGISTRATION NO OF VEHICLE TO CALCULATE TAX - ");

int reg\_no=s.nextInt();

for(VehicleClass vc1:vl)

{

if(vc1.registration\_number == reg\_no)

{

if(vc1.type\_of\_vehicle.equals("PETROL"))

{

vc1.vehicle\_tax=Math.round(vc1.max\_velocity+0.1\*vc1.purchase\_cost);

}

else if(vc1.type\_of\_vehicle.equals("DIESEL"))

{

vc1.vehicle\_tax=Math.round(vc1.max\_velocity+0.11\*vc1.purchase\_cost);

}

else if(vc1.type\_of\_vehicle.equals("CNG/LPG"))

{

vc1.vehicle\_tax=Math.round(vc1.max\_velocity+0.12\*vc1.purchase\_cost);

}

System.out.println("VEHICLE TAX FOR REGISTRATION NO - "+reg\_no + " IS "+vc1.vehicle\_tax);

}

else

{

System.out.println("Reg number not Found");

}

}

}

}

public void displayVehicel(ArrayList<VehicleClass> vl)

{

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

System.out.println("| REGISTRATION\_NO\tBRAND\tMAX.VELOCITY\tNO.OF.SEATS\tVEHICLE TYPE\tPURCHASE COST\tVEHICLE TAX |");

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

for (VehicleClass vc : vl) {

System.out.print(vc.registration\_number+ "\t\t");

System.out.print(vc.brand + "\t\t");

System.out.print(vc.max\_velocity + "\t\t\t");

System.out.print(vc.no\_of\_seats+"\t\t");

System.out.print(vc.type\_of\_vehicle+"\t\t");

System.out.print(vc.purchase\_cost+"\t\t");

System.out.print(vc.vehicle\_tax);

System.out.println();

}

}

public void totalTax(ArrayList<PropertyClass> pl,ArrayList<VehicleClass> vl)

{

double propertytax=0;

double vehicletax=0;

for(PropertyClass pc:pl)

{

propertytax+=pc.tax;

}

for(VehicleClass vc:vl)

{

vehicletax+=vc.vehicle\_tax;

}

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

System.out.println("| SR. NO. PARTICULAR\t\tQUANTITY\tTAX |");

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

System.out.print("| 1\t");

System.out.print("PROPERTIES\t\t");

System.out.print(pl.size()+"\t\t");

System.out.print(propertytax+" |\t\t");

System.out.println();

System.out.print("| 2\t");

System.out.print("VEHICLES\t\t");

System.out.print(vl.size()+"\t\t");

System.out.println(vehicletax+" |\t\t");

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

System.out.println("| TOTAL-------------"+(pl.size()+vl.size())+"\t"+(propertytax+vehicletax)+" |");

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

}

}

ExceptionClass.java

**package** Com.tax.calculation;

**public** **class** ExceptionClass **extends** Exception{

String msg;

**public** ExceptionClass(String msg) {

**super**(msg);

}

}

Main.java

package Com.tax.calculation;

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class Main {

static final String username1="admin";

static String password1 = "admin";

public static void main(String[] args) throws ExceptionClass

{

List<PropertyClass> pl= new ArrayList<PropertyClass>();

List<VehicleClass> vl= new ArrayList<VehicleClass>();

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

System.out.println("| WELCOME TO TAXA(TAX CALCULATION APPLICATION) |");

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

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

Scanner s = new Scanner(System.in);

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

String username = s.next();

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

String password = s.next();

ArrayList<PropertyClass> properties = new ArrayList<>();

ArrayList<VehicleClass> vehicles = new ArrayList<>();

if(username.equals(username1)&&password.equals(password1))

{

boolean condition=true;

while(condition)

{

System.out.println("1.PROPERTY TAX\n2.VEHICLE TAX\n3.TOTAL\n4.EXIT");

int choice1=s.nextInt();

if(choice1<0)

{

throw new ExceptionClass("Enter only in range from 0");

}

boolean b=true;

while(b)

{

switch(choice1)

{

case 1:

PropertyOperations po = new PropertyOperations();

System.out.println("1.ADD PROPERTY DETAILS\n2.CALCULATE PROPERTY TAX\n3.DISPLAY ALL PROPERTIES\n4.BACK TO MAIN MENU");

switch (s.nextInt())

{

case 1:

try {

po.addProperty(properties);

} catch (ExceptionClass e) {

// TODO Auto-generated catch block

System.out.println(e.getMessage());

po.addProperty(properties);

}

break;

case 2:

try {

po.setTax(properties);

} catch (ExceptionClass e) {

// TODO Auto-generated catch block

System.out.println(e.getMessage());

po.addProperty(properties);

}

break;

case 3:

po.displayDetails(properties);

break;

case 4:

b = false;

break;

}

break;

case 2:

while(b)

{

VehicleOperations vo = new VehicleOperations();

System.out.println("1.ADD VEHICLE DETAILS\n2.CALCULATE VEHICLE TAX\n3.DISPLAY ALL VEHICLES\n4.BACK TO MAIN MENU");

switch (s.nextInt())

{

case 1:

try {

vo.addVehicelDetails(vehicles);

} catch (ExceptionClass e) {

// TODO Auto-generated catch block

System.out.println(e.getMessage());

}

break;

case 2:

try {

vo.setTax(vehicles);

} catch (ExceptionClass e) {

// TODO Auto-generated catch block

System.out.println(e.getMessage());

}

break;

case 3:

vo.displayVehicel(vehicles);

break;

case 4:

b = false;

break;

}

}

break;

case 3:

while(b)

{

VehicleOperations to = new VehicleOperations();

to.totalTax(properties, vehicles);

b=false;

}

break;

case 4:

System.out.println("THANK YOU VISIT AGAIN");

System.exit(0);

break;

default:

System.out.println("Invalid choise");

}

}

}

}

else

{

System.out.println("Invalid username/password");

}

}

}