package coches;

import java.util.Arrays;

import java.util.Scanner;

import coches.Coche.Accesorio;

public class Concesionario {

static Coche[] coches = new Coche[10];

public static void main(String[] args) {

// TODO Esbozo de método generado automáticamente

crearCoche("Toyota", "Dango", crearAccesorios(3));

crearCocheCombustible("Toyota", "Sami", crearAccesorios(1), "diesel");

crearCocheElectrico("Toyota", "Evis", crearAccesorios(1), 100);

crearCocheCombustible("Opel", "Corsa", crearAccesorios(7), "gasolina");

crearCocheElectrico("Mercedez", "Benz", crearAccesorios(5), 50);

mostrarCoches(coches);

coches[0].verPrecio();

}

public static Accesorio[] crearAccesorios(int menu) {

if (menu == 1) {

Accesorio[] accesorios3 = {Accesorio.TECHO, Accesorio.NAVEGADOR, Accesorio.ASIENTOS\_CALEFACTABLES};

return accesorios3;

}

else if (menu == 2) {

Accesorio[] accesoriosTechoNavegador = {Accesorio.TECHO, Accesorio.NAVEGADOR};

return accesoriosTechoNavegador;

}

else if (menu == 3) {

Accesorio[] accesoriosTechoAsiento\_Calefactable = {Accesorio.TECHO, Accesorio.ASIENTOS\_CALEFACTABLES};

return accesoriosTechoAsiento\_Calefactable;

}

else if (menu == 4) {

Accesorio[] accesoriosNavegadorAsiento\_Calefactable = {Accesorio.NAVEGADOR, Accesorio.ASIENTOS\_CALEFACTABLES};

return accesoriosNavegadorAsiento\_Calefactable;

}

else if (menu == 5) {

Accesorio[] accesoriosTecho = {Accesorio.TECHO};

return accesoriosTecho;

}

else if (menu == 6) {

Accesorio[] accesoriosNavegador = {Accesorio.NAVEGADOR};

return accesoriosNavegador;

}

else if (menu == 7) {

Accesorio[] accesoriosAsiento\_calefactables = {Accesorio.ASIENTOS\_CALEFACTABLES};

return accesoriosAsiento\_calefactables;

}

else {

Accesorio[] sin\_accesorios = new Accesorio[0];

return sin\_accesorios;

}

}

public static void crearCoche(String marca, String modelo, Accesorio[] accesorios) {

Coche normal = new Coche(marca, modelo, accesorios);

int contador = 0;

for (int posicion = 0; posicion < coches.length && contador != 1; posicion ++) {

if (coches[posicion] == null) {

coches[posicion] = normal;

normal.precioNormal();

contador++;

}

else if (coches[coches.length - 1] != null){

System.err.println("Error: El concesionario esta lleno");

contador++;

}

}

}

public static void crearCocheElectrico(String marca, String modelo, Accesorio[] accesorios, double potencia) {

Bateria b = new Bateria(potencia);

CocheElectrico electrico = new CocheElectrico(marca, modelo, accesorios, b);

electrico.setCoche\_electrico(electrico);

int contador = 0;

for (int posicion = 0; posicion < coches.length && contador != 1; posicion ++) {

if (coches[posicion] == null) {

coches[posicion] = electrico;

electrico.precioElectrico();

contador++;

}

else if (coches[coches.length - 1] != null){

System.err.println("Error: El concesionario esta lleno");

contador++;

}

}

}

public static void crearCocheCombustible(String marca, String modelo, Accesorio[] accesorios, String deposito) {

Deposito d = new Deposito(deposito);

CocheCombustible combustible = new CocheCombustible(marca, modelo, accesorios, d);

combustible.setCoche\_combustible(combustible);

int contador = 0;

for (int posicion = 0; posicion < coches.length && contador != 1; posicion ++) {

if (coches[posicion] == null) {

coches[posicion] = combustible;

combustible.precioCombustible();

contador++;

}

else if (coches[coches.length - 1] != null){

System.err.println("Error: El concesionario esta lleno");

contador++;

}

}

}

public static void mostrarCoches(Coche[]coches) {

ordenarPorPrecio(coches);

double precioFinal=0;

for(int i = 0; i < coches.length; i++) {

if(coches[i] != null) {

precioFinal = coches[i].getPrecioFinal() + precioFinal;

System.out.println(coches[i].toString());

System.out.println();

}

}

System.out.println("El precio total del concesionario es de: " + precioFinal);

System.out.println();

}

public static void informe(String marca) {

int numero\_coches = 0;

int numero\_coches\_electricos = 0;

int numero\_coches\_combustible = 0;

for(int i = 0; i < coches.length; i++) {

if(coches[i] != null && coches[i].getMarca().compareToIgnoreCase(marca) == 0) {

numero\_coches++;

if(coches[i].getCoche\_electrico() != null) {

numero\_coches\_electricos++;

}

if(coches[i].getCoche\_combustible() != null) {

numero\_coches\_combustible++;

}

}

}

int coches\_normales = numero\_coches - (numero\_coches\_electricos+numero\_coches\_combustible);

System.out.println("Hay " + numero\_coches + " coches de la marca " + marca);

System.out.println("Hay " + numero\_coches\_electricos + " coches electricos, " + numero\_coches\_combustible + " coches de combustible y " + coches\_normales + " coches normales");

}

public static void ordenarPorPrecio(Coche[]coches) {

for (int i = 0; i < coches.length - 1; i++) {

for (int j = 0; j < coches.length - i - 1; j++) {

if (coches[j] != null && coches[j + 1] != null && coches[j].getPrecioFinal() > coches[j + 1].getPrecioFinal()) {

Coche temp = coches[j];

coches[j] = coches[j + 1];

coches[j + 1] = temp;

}

}

}

}

}