

```

1
2 //-----PRINCIPAL.JAVA-----
3 public class Principal {
4     public static void main(String[] args) {
5
6         ParqueMovil German = new ParqueMovil("Parquel");
7
8         German.preguntaIniciaParque();
9         German.menuprincipal();
10    }
11 }
12
13 //-----OPCIONESMENU.JAVA-----
14 public enum OpcionesMenu {
15     ALTA,BAJA,MOSTRAR,LISTADO,SALIR
16 }
17
18 //-----MARCACOCHE.JAVA-----
19 public enum MarcaCoche {
20     FORD,MERCEDES,PEGASO
21 }
22
23 //-----VEHICULO.JAVA-----
24
25 public class Vehiculo {
26
27     private final String matricula;
28     private final MarcaCoche marca;
29     private final int cargaMaxima;
30     private int kilometraje;
31
32     public Vehiculo(String matricula, MarcaCoche marca, int cargaMaxima, int
33 kilometraje) {
34         this.matricula = matricula;
35         this.marca = marca;
36         this.cargaMaxima = cargaMaxima;
37         this.kilometraje = kilometraje;
38     }
39
40     public String getMatricula() {
41         return matricula;
42     }
43
44     public MarcaCoche getMarca() {
45         return marca;
46     }
47
48     public int getCargaMaxima() {
49         return cargaMaxima;
50     }
51
52     public int getKilometraje() {
53         return kilometraje;
54     }
55
56     public void setKilometraje(int kilometraje) {
57         this.kilometraje = kilometraje;
58     }
59
60     @Override
61     public String toString() {
62         return  marca +" "+
63             "|| "+matricula +" "+
64             "|| "+cargaMaxima +" "+ "...Carga MAX"+
65             "|| "+kilometraje+"...KMs";
66     }
67 }
68 //-----PARQUEMOVIL.JAVA-----
69
70 import javax.swing.*;
71
72 public class ParqueMovil {

```

```

73
74     String nombre;
75
76     private Vehiculo [] listaCoches = new Vehiculo[50];
77
78     public ParqueMovil(String nombre) {
79         this.nombre = nombre;
80     }
81
82     public Vehiculo[] getListaCoches() {
83         return listaCoches;
84     }
85
86     public void setListaCoches(Vehiculo[] listaCoches) {
87         this.listaCoches = listaCoches;
88     }
89
90     public void rellenarPrimeros(){
91         listaCoches[0]= new Vehiculo("4455CCC",MarcaCoche.MERCEDES,750,10000);
92         listaCoches[1]= new Vehiculo("5566DDD",MarcaCoche.FORD,1250,300);
93         listaCoches[2]= new Vehiculo("6677EEE",MarcaCoche.FORD,880,500);
94         listaCoches[3]= new Vehiculo("7788FFF",MarcaCoche.PEGASO,1230,15000);
95         listaCoches[4]= new Vehiculo("8899GGG",MarcaCoche.MERCEDES,445,23000);
96     }
97
98     public void muestraCoches() {
99         String salida="";
100
101         for (Vehiculo v : this.ordenaVehiculos()) {
102             if(v != null) {
103                 salida+=v+"\n";
104             }
105         }
106         JOptionPane.showMessageDialog(null,salida);
107     }
108
109     private Vehiculo[] ordenaVehiculos() {
110
111         Vehiculo aux;
112         Vehiculo[] vectorSinNulos=this.vectorSinNulos();
113
114         for(int i=0;i<vectorSinNulos.length-1;i++)
115             for(int j=0;j<vectorSinNulos.length-i-1;j++)
116
117                 if(vectorSinNulos[j+1].getMatricula().compareTo(vectorSinNulos[j].
118                 getMatricula())<0){//comparamos entre String y ordenamos de
119                 menor a mayor.
120                     aux=vectorSinNulos[j+1];
121                     vectorSinNulos[j+1]=vectorSinNulos[j];
122                     vectorSinNulos[j]=aux;
123                 }
124         return vectorSinNulos;
125     }
126
127     private Vehiculo[] vectorSinNulos() {
128
129         Vehiculo[] acotado;
130         int contador=0;
131         int contaplus=0;
132
133         for (Vehiculo v : this.listaCoches) { //Averiguo el tamaño ocupado del
134         vector.
135             if(v != null) {
136                 contador++;
137             }
138         }
139         acotado = new Vehiculo[contador]; //Utilizo vector auxiliar solo con las
140         posiciones ocupadas.
141         for (Vehiculo v : this.listaCoches) {
142             if(v != null) {
143                 acotado[contaplus]=v;
144                 contaplus++;
145             }
146         }

```

```

141         }
142         return acotado;
143     }
144
145     public void agregarCoche() {
146         Vehiculo local;
147
148         for (int i=0; i<listaCoches.length; i++){
149             if (listaCoches[i]==null){
150                 local=pedirDatosVehiculoNuevo();
151                 if(compruebaDuplicados(local)){
152                     System.out.println("Este vehiculo ya existe");
153                 }else{
154                     listaCoches[i]=local;
155                 }
156                 return;
157             }
158         }
159     }
160
161     private Vehiculo pedirDatosVehiculoNuevo() {
162         String matricula= JOptionPane.showInputDialog(null,"Introduce una
matricula:");
163         MarcaCoche marca
=(MarcaCoche)JOptionPane.showInputDialog(null,"Marca","Marca",JOptionPane.PLA
N_MESSAGE,null,MarcaCoche.values(),MarcaCoche.FORD);
164         int cargaMaxima= Integer.parseInt(JOptionPane.showInputDialog(null,"¿Peso
maximo soportado?"));
165         int kilometraje=
Integer.parseInt(JOptionPane.showInputDialog(null,"¿Kilometros del
vehiculo?"));
166
167         Vehiculo nuevoVehiculo = new Vehiculo(matricula, marca, cargaMaxima,
kilometraje);
168
169         return nuevoVehiculo;
170     }
171
172     public void preguntaIniciaParque() {
173
174         int respuesta=JOptionPane.showConfirmDialog(null,"¿Quieres inicializar el
parque movil?",null,JOptionPane.YES_NO_OPTION);
175
176         if (respuesta==0){
177             this.rellenarPrimeros();
178             this.muestraCoches();
179         }else {
180             }
181     }
182
183     public boolean compruebaDuplicados(Vehiculo nuevoVehiculo) {
184
185         for (Vehiculo v :
this.getListaCoches()) {
186             if (v != null) {
187                 if (v.getMatricula().equals(nuevoVehiculo.getMatricula())) {
188                     return true;
189                 }
190             }
191         }
192         return false;
193     }
194
195     private void bajaVehiculo() {
196
197         Vehiculo encontrado=this.buscaVehiculo();
198
199         if (encontrado==null){
200             return;
201         }
202         int posicionVehiculo=this.devuelvePosicionVehiculo(encontrado);
203         listaCoches[posicionVehiculo]=null;
204     }
205

```

```

206
207     private int devuelvePosicionVehiculo(Vehiculo encontrado) {
208
209         for (int i=0;i<listaCoches.length;i++){
210             if (encontrado==listaCoches[i]){
211                 return i;
212             }
213         }
214         return -1;
215     }
216
217     private Vehiculo buscaVehiculo() {
218         String matricula;
219
220         matricula=this.pedirMatricula();
221
222         for (Vehiculo v : this.listaCoches) {
223             if(v != null) {
224                 if (v.getMatricula().equalsIgnoreCase(matricula)){
225                     JOptionPane.showMessageDialog(null,v.toString());
226                     return v;
227                 }
228             }
229         }
230         JOptionPane.showMessageDialog(null,"El vehiculo introducido ;NO EXISTE!");
231         return null;
232     }
233
234     private String pedirMatricula() {
235
236         String matricula=JOptionPane.showInputDialog(null,"Introduce la matricula
237         del vehiculo");
238
239         return matricula;
240     }
241
242     public void menuprincipal() {
243
244         int opcion=JOptionPane.showOptionDialog(null,null,"Menu
245         Principal",JOptionPane.PLAIN_MESSAGE,JOptionPane.PLAIN_MESSAGE,null,OpcionesMe
246         nu.values(),null);
247
248         switch (opcion){
249
250             case 0:
251                 this.agregarCoche();
252                 this.menuprincipal();
253                 break;
254             case 1:
255                 this.bajaVehiculo();
256                 this.menuprincipal();
257                 break;
258             case 2:
259                 this.buscaVehiculo();
260                 this.menuprincipal();
261                 break;
262             case 3:
263                 this.muestraCoches();
264                 this.menuprincipal();
265                 break;
266             default:
267                 return;
268         }
269     }
270

```