

Vehiculo.java

```
1 /**
2  * @ (#) Vehiculo.java
3  *
4  * Clase Vehiculo.
5  * Es la superclase de las clases Coche, Moto y VehiculoGrande.
6  * Representa a cada uno de los vehículos que entran en el taller.
7  *
8  * @author Yassine Marroun
9  * @version 1.00 2017/05/24
10 */
11
12
13 public class Vehiculo {
14
15     protected String marca;
16     protected String modelo;
17     protected String matricula;
18     protected Enumerados.Combustible combustible;
19     protected String dniCliente;
20     private Itv itv;
21
22
23     public Vehiculo () {
24
25     }
26
27     public Vehiculo (String marca, String modelo, String matricula, Enumerados.Combustible
28 combustible, String dniCliente) {
29
30         this.marca = marca;
31         this.modelo = modelo;
32         this.matricula = matricula;
33         this.combustible = combustible;
34         this.dniCliente = dniCliente;
35         this.itv = new Itv();
36     }
37
38     public String getMarca() {
39         return marca;
40     }
41
42     public void setMarca(String marca) {
43         this.marca = marca;
44     }
45
46     public String getModelo() {
47         return modelo;
48     }
49
50     public void setModelo(String modelo) {
51         this.modelo = modelo;
52     }
53
54     public String getMatricula() {
55         return matricula;
56     }
57
58     public void setMatricula(String matricula) {
59         this.matricula = matricula;
60     }
61 }
```

Vehiculo.java

```

62     public Enumerados.Combustible getCombustible() {
63         return combustible;
64     }
65
66     public void setCombustible(Enumerados.Combustible combustible) {
67         this.combustible = combustible;
68     }
69
70     public String getDniCliente() {
71         return dniCliente;
72     }
73
74     public void setDniCliente(String dniCliente) {
75         this.dniCliente = dniCliente;
76     }
77
78     public Itv getItv() {
79         return itv;
80     }
81
82     public void setItv(Itv itv) {
83         this.itv = itv;
84     }
85
86
87     // Con el método nuevoVehiculo obtenemos los datos de un vehículo cuando acude al
88     // taller.
89     // Se crea una instancia de clase Coche, Moto o VehiculoGrande en función del tipo
90     // seleccionado.
91
92     public Vehiculo nuevoVehiculo() {
93         System.out.print(Enumerados.menuTiposVehiculos());
94         System.out.print("Introduzca el tipo de vehículo: ");
95         Integer tipoVehi = Menu.escanerInt();
96         Enumerados.TipoVehiculo enumTv = Enumerados.TipoVehiculo.values()[tipoVehi];
97         System.out.print("Introduzca la Marca del vehículo: ");
98         String marca = Menu.sc.nextLine();
99         System.out.print("Introduzca el Modelo: ");
100        String modelo = Menu.sc.nextLine();
101        System.out.print("Introduzca la Matricula: ");
102        String matricula = Menu.sc.nextLine();
103        System.out.print(Enumerados.menuTiposCombustible());
104        System.out.print("Introduzca el tipo de Combustible: ");
105        Integer combustible = Menu.escanerInt();
106        Enumerados.Combustible tipoComb = Enumerados.Combustible.values()[combustible];
107        System.out.print("Introduzca el dni del cliente: ");
108        String dniCliente = Menu.sc.nextLine();
109        Vehiculo vehiculo = null;
110
111        if (Enumerados.TipoVehiculo.COCHES.equals(enumTv)){
112            System.out.print(Enumerados.menuTiposCoche());
113            System.out.print("Introduzca el tipo de coche: ");
114            Integer itc = Menu.sc.nextInt();
115            Menu.sc.nextLine();
116            Enumerados.TipoCoche tipo = Enumerados.TipoCoche.values()[itc];
117            vehiculo = new Coche(tipo, marca.trim(), modelo.trim(), matricula.trim(),
118            tipoComb, dniCliente.trim());
119        }
120
121        if (Enumerados.TipoVehiculo.MOTOS.equals(enumTv)){
122            System.out.print(Enumerados.menuTiposMoto());
123            System.out.print("Introduzca el tipo de moto: ");

```

Vehiculo.java

```

121         Integer itc = Menu.sc.nextInt();
122         Menu.sc.nextLine();
123         Enumerados.TipoMoto tipo = Enumerados.TipoMoto.values()[itc];
124         vehiculo = new Moto(tipo, marca.trim(), modelo.trim(), matricula.trim(),
    tipoComb, dniCliente.trim());
125     }
126
127     if (Enumerados.TipoVehiculo.GRANDE.equals(enumTv)){
128         System.out.print(Enumerados.menuTiposMoto());
129         System.out.print("Introduzca el tipo de vehiculo grande: ");
130         Integer itc = Menu.sc.nextInt();
131         Menu.sc.nextLine();
132         Enumerados.TipoGrandes tipo = Enumerados.TipoGrandes.values()[itc];
133         vehiculo = new VehiculoGrande(tipo, marca.trim(), modelo.trim(),
    matricula.trim(), tipoComb, dniCliente.trim());
134     }
135     return vehiculo;
136 }
137
138
139 // Para visualizar los datos de un objeto, en este caso tenemos dos métodos:
140 // toStringAmpliado, con el que mostramos todos los atributos de un vehiculo.
141 // toString, muestra tres atributos, del que se hace uso cuando queremos listados.
142
143 public String toStringAmpliado() {
144     String datos = "\n Marca: " + marca +
145         "\n Modelo: " + modelo +
146         "\n Matricula: " + matricula +
147         "\n Combustible: " + combustible.toString() +
148         "\n a pasar Itv: " + siNo(itv.isaPasarItv()) +
149         "\n reparado Itv: " + siNo(itv.isReparado()) +
150         "\n Dni del Cliente: " + dniCliente +
151         "\n ***** \n ";
152     return datos;
153 }
154
155 @Override
156 public String toString() {
157     String datos = " " + marca + " " + modelo + " " + matricula;
158     return datos;
159 }
160
161
162 // El método siNo tiene la función de devolver un Si o un No según el valor del
    parámetro boolean recibido.
163 // Con ello visualizamos en el toStringAmpliado el valor que devuelven isaPasarItv e
    isReparado.
164
165 private String siNo(Boolean tipoBool){
166     if (tipoBool){
167         return "Si";
168     } else {
169         return "No";
170     }
171 }
172 }

```