

Menu.java

```
1 /**
2  * @ (#) Menu.java
3  *
4  * Clase Menu.
5  * Esta clase recoge los diferentes menús donde se selecciona cada opción y a dicha opción
6  * va relacionada una llamada al método correspondiente.
7  *
8  * @author Yassine Marroun
9  * @version 1.00 2017/05/24
10 */
11 import java.util.Scanner;
12
13 public class Menu {
14
15     public static final Scanner sc = new Scanner(System.in);
16     private GestionTaller gestion = new GestionTaller();
17
18     // Menu Principal.
19
20     public int menuPrincipal(){
21
22         gestion.createDatosInicialesPruebas();
23
24         int menu1 = 0;
25         Integer numOpcs = 7;
26
27         do{
28             System.out.println("\n");
29             System.out.println("1. Crear clientes.");
30             System.out.println("2. Crear vehiculos.");
31             System.out.println("3. Fichas.");
32             System.out.println("4. Listar clientes.");
33             System.out.println("5. Listar vehiculos.");
34             System.out.println("6. Ofertas e Itv.");
35             System.out.println("7. Salir.");
36             System.out.print("Opcion> ");
37
38             menu1 = escanerInt();
39
40             if (menu1<1 || menu1>numOpcs) {
41                 System.out.println("Opcion incorrecta. Vuelve a probar");
42             }
43
44             switch (menu1){
45
46                 case 1:
47                     gestion.crearCliente();
48                     break;
49                 case 2:
50                     gestion.crearVehiculo();
51                     break;
52                 case 3:
53                     menuMecanico();
54                     break;
55                 case 4:
56                     gestion.listarClientes();
57                     break;
58                 case 5:
59                     gestion.listarVehiculos();
60                     break;
61                 case 6:
62
```

Menu.java

```

63         menuOfertaseItv();
64         case 7:
65             System.out.println("...");
66     }
67     } while(menu1 != numOpcs);
68
69     return menu1;
70 }
71
72
73 // Menu Mecanico.
74
75 public int menuMecanico(){
76
77     int menu1 = 0;
78     Integer numOpcs = 9;
79     String nombreMeca = "";
80     Integer numFicha = 0;
81     Ficha ficha = null;
82
83     do{
84         System.out.println("\n");
85         System.out.println("1. Alta de ficha");
86         System.out.println("2. Ver mis fichas");
87         System.out.println("3. Modificar ficha");
88         System.out.println("4. Cambiar estado fichas");
89         System.out.println("5. Asignar mecanico");
90         System.out.println("6. Ver fichas en un estado");
91         System.out.println("7. Ver fichas reparadas entre dos fechas");
92         System.out.println("8. Listar todas las fichas");
93         System.out.println("9. Salir");
94         System.out.print("Opcion> ");
95
96         menu1 = escanerInt();
97
98         if(menu1<1 || menu1>=numOpcs){
99             System.out.println("Opcion incorrecta. Vuelve a probar");
100         }
101
102         switch (menu1){
103
104             case 1:
105                 System.out.println("Alta de fichas");
106                 Integer nuevoNumero = gestion.getNumNuevaFicha();
107                 gestion.altaFichas(nuevoNumero);
108                 break;
109             case 2:
110                 System.out.println("Nombre mecanico: ");
111                 nombreMeca = sc.nextLine();
112                 gestion.visualizaFichasMecanico(nombreMeca.trim());
113                 break;
114             case 3:
115                 System.out.println("Modificar ficha numero:");
116                 numFicha = escanerInt();
117                 gestion.modificarFicha(numFicha);
118                 break;
119             case 4:
120                 System.out.println("Cambiar estado de ficha numero:");
121                 numFicha = escanerInt();
122                 ficha = gestion.getFicha(numFicha);
123                 if (ficha!=null){
124                     menuEstado(ficha);

```

Menu.java

```

125         } else{
126             System.out.println("Numero de ficha inexistente");
127         }
128         break;
129     case 5:
130         System.out.println("Asignar mecanico a ficha numero:");
131         numFicha = escanerInt();
132         System.out.println("Nombre mecanico: ");
133         nombreMeca = sc.nextLine();
134         gestion.asignarFichaAMecanico(numFicha, nombreMeca.trim());
135         break;
136     case 6:
137         System.out.println(Enumerados.menuEstadoReparacion());
138         System.out.println("Estado buscado:");
139         Integer iEstado = escanerInt();
140         Enumerados.EstadoReparacion estado =
Enumerados.EstadoReparacion.values()[iEstado];
141         if (estado!=null){
142             gestion.visualizaFichasEstado(estado);
143         } else{
144             System.out.println("Estado inexistente");
145         }
146         break;
147     case 7:
148         gestion.fichasEntreFechas();
149         break;
150     case 8:
151         gestion.visualizaFichasTodas();
152         break;
153     default:
154         break;
155     }
156     } while(menu1 <= numOpcs);
157     return menu1;
158 }
159
160
161 // Menu Estado.
162
163 private void menuEstado(Ficha ficha){
164     int submenu1 = 0;
165     Integer numOpcs = 0;
166
167     do{
168         for (Enumerados.EstadoReparacion estado: Enumerados.EstadoReparacion.values()){
169             System.out.println(estado.ordinal() + ". " + estado.name());
170         }
171         numOpcs = Enumerados.EstadoReparacion.values().length;
172         System.out.print(numOpcs + ". Volver");
173
174         System.out.print("\n Submenu de estado reparacion > ");
175         submenu1 = escanerInt();
176
177         if(submenu1<0 || submenu1>=numOpcs){
178             System.out.println("Opcion incorrecta. Vuelve a probar");
179         } else{
180             if (Enumerados.EstadoReparacion.values()
[submenu1].equals(Enumerados.EstadoReparacion.PARADO)){
181                 System.out.println("Motivo del Estado Parado");
182                 menuParado(ficha);
183             }
184         }

```

Menu.java

```

185
186         if (submenu1<numOpcs){
187             ficha.setEstado(Enumerados.EstadoReparacion.values()[submenu1]);
188         }
189
190     } while (submenu1 > numOpcs);
191     // Este menu solo se repite si la opcion es incorrecta. Si es correcta se modifica
    el estado.
192 }
193
194
195 // Menu Parado.
196
197 private int menuParado(Ficha ficha){
198     System.out.println("\n Submenu de Parado");
199     int submenu2 = 0;
200     Integer numOpcs = 2;
201     do{
202         System.out.println("0. Faltan Piezas");
203         System.out.println("1. Confirmar Cliente");
204         System.out.println("2. Volver");
205         System.out.print("Opcion Submenu 2 > ");
206
207         submenu2 = escanerInt();
208
209         if (submenu2<numOpcs){
210             ficha.setParado(Enumerados.Parado.values()[submenu2]);
211         }
212     }while (submenu2 > numOpcs);
213     // Este menu solo se repite si la opcion es incorrecta. Si es correcta se modifica
    el motivo.
214
215     return submenu2;
216 }
217
218
219 // Menu Ofertas e ITV.
220
221 public int menuOfertaseItv(){
222
223     int menuOfe = 0;
224     Integer numOpcs = 6;
225
226
227     do{
228         System.out.println("\n");
229         System.out.println("1. Alta de oferta");
230         System.out.println("2. Ofertar a cliente");
231         System.out.println("3. Enviar ofertas");
232         System.out.println("4. Alta Pasar Itv");
233         System.out.println("5. Modificar estado Itv");
234         System.out.println("6. Salir");
235         System.out.print("Opcion> ");
236
237         menuOfe = escanerInt();
238
239         if(menuOfe<1 || menuOfe>numOpcs){
240             System.out.println("Opcion incorrecta. Vuelve a probar");
241         }
242
243         switch (menuOfe){
244

```

Menu.java

```
245         case 1:
246             gestion.crearOferta();
247             break;
248         case 2:
249             gestion.ofertaACliente();
250             break;
251         case 3:
252             gestion.enviarOfertas();
253             break;
254         case 4:
255             gestion.pasarItv();
256             break;
257         case 5:
258             gestion.modificarEstadoItv();
259             break;
260         default:
261             break;
262     }
263     } while(menuOfe < numOpcs);
264     return menuOfe;
265 }
266
267
268 // Ya que en esta clase necesitamos obtener por teclado muchos datos de tipo Integer,
269 // este método escanerInt evita posibles fallos de programa recogiendo los errores con
un catch.
270
271 public static Integer escanerInt(){
272     try{
273         Integer numero = sc.nextInt();
274         sc.nextLine();
275         return numero;
276     }
277     catch(Exception e){
278         sc.nextLine();
279         return -1;
280     }
281 }
282 }
```