

Cliente.java

```
1 /**
2  * @ (#) Cliente.java
3  *
4  * Clase Cliente.
5  * Es una subclase que hereda de la clase madre Usuario.
6  * La clase GestionTienda contiene un mapa donde almacenaremos cada objeto creado de la
   clase Cliente,
7  * y además se desarrollan métodos para dar funcionalidad a dichos objetos.
8  * Una instancia de la clase Cliente, además de los ya heredados, va a tener los campos
   domicilio, telefono
9  * y tres ArrayList que almacenen objetos de tipo Electrodomestico, Ficha y Promo
   respectivamente.
10 * Constructor sin parámetros y constructor con los atributos heredados más los atributos
   particulares de esta clase.
11 * Métodos Getters y Setters de sus campos particulares.
12 * Por último, tenemos el método toString, con el que visualizamos los datos personales de
   un Cliente,
13 * un listado de los electrodomesticos que ha comprado, más la información que incluya o no
14 * en su correspondiente lista promós.
15 *
16 * @author Yassine Marroun
17 * @version 1.00 2018/05/04
18 */
19 package modelo.bd;
20 import utilidades.Utiles;
21
22
23 public class Cliente extends Usuario{
24
25     private String domicilio;
26     private String telefono;
27     private ArrayList<Electrodomestico> electrodomesticos;
28     private ArrayList<Ficha> fichas;
29     private ArrayList<Promo> promós;
30
31     public Cliente () {
32
33     }
34
35     public Cliente(String nombre, String apellidos, String dni, String domicilio, String
   telefono) {
36
37         super(nombre, apellidos, dni);
38         this.domicilio = domicilio;
39         this.telefono = telefono;
40         this.electrodomesticos = new ArrayList<Electrodomestico>();
41         this.fichas = new ArrayList<Ficha>();
42     }
43
44
45     public String getDomicilio() {
46         return domicilio;
47     }
48
49     public void setDomicilio(String domicilio) {
50         this.domicilio = domicilio;
51     }
52
53     public String getTelefono() {
54         return telefono;
55     }
56
57     public void setTelefono(String telefono) {
```

Cliente.java

```

58     this.telefono = telefono;
59 }
60
61 public ArrayList<Electrodomestico> getElectrodomesticos() {
62     return electrodomesticos;
63 }
64
65 public void setElectrodomesticos(ArrayList<Electrodomestico> electrodomesticos) {
66     this.electrodomesticos = electrodomesticos;
67 }
68
69 public ArrayList<Ficha> getFichas() {
70     return fichas;
71 }
72
73 public void setFichas(ArrayList<Ficha> fichas) {
74     this.fichas = fichas;
75 }
76
77 public ArrayList<Promo> getPromos() {
78     return promos;
79 }
80
81 public void setPromos(ArrayList<Promo> promos) {
82     this.promos = promos;
83 }
84
85
86 /*
87  El método datosCliente solicita que se introduzcan por teclado datos para cada uno de
88  los campos
89  con los que finalmente crea y devuelve un objeto de tipo Cliente, sin incluir las
90  listas,
91  ya que éstas se rellenan mediante otros métodos que iremos detallando.
92  */
93 public Cliente datosCliente() {
94     System.out.print("Introduzca el nombre del Cliente: ");
95     String nombre = Utiles.sc.nextLine();
96     System.out.print("Introduzca los apellidos del Cliente: ");
97     String apellidos = Utiles.sc.nextLine();
98     System.out.print("Introduzca el DNI: ");
99     String dni = Utiles.sc.nextLine();
100    System.out.print("Introduzca el DOMICILIO: ");
101    String domicilio = Utiles.sc.nextLine();
102    System.out.print("Introduzca el TELEFONO: ");
103    String telefono = Utiles.sc.nextLine();
104    Cliente cliente = new Cliente(nombre, apellidos, dni, domicilio, telefono);
105    return cliente;
106 }
107
108 @Override
109 public String toString(){
110     String datosCliente = super.toString() + " - " + domicilio + " - " + telefono +
111     "\n Electrodomesticos comprados: ";
112     for (Electrodomestico electro: electrodomesticos) {
113         electro.toStringFicha();
114         datosCliente = datosCliente + electro.toStringFicha();
115     }
116     datosCliente = datosCliente + "\n Tiene Promociones: ";
117     if (promos==null || promos.size()<=0){
118         datosCliente = datosCliente + "No.";
119     }
120 }

```

Cliente.java

```
118         } else {
119             datosCliente = datosCliente + "Si.";
120             datosCliente = datosCliente + getPromos().toString();
121         }
122
123         datosCliente = datosCliente + "\n ***** \n ";
124         return datosCliente;
125     }
126 }
```