

```

1 package vista;
2
3 import static com.coti.tools.Esdia.*;
4 import static com.coti.tools.OpMat.printToScreen3;
5 import controlador.Controlador;
6 import java.io.IOException;
7 import java.util.ArrayList;
8 import modelo.Profesteca;
9
10 public class Vista {
11
12     Profesteca profesteca = new Profesteca();
13     Controlador controlador = new Controlador();
14
15     public void menuPrincipal() throws IOException, Exception{
16         profesteca.importarDatos();
17         controlador.recibirPunteroAProfesteca(profesteca);
18
19         String [] possbopc = {"A","B","M","C","L","S"};
20         boolean exit = false;
21         while(!exit){
22             String mainMenu="%n== BASE DE DATOS DE PROFESORES =="
23                 + "%nA) Dar de alta un profe."
24                 + "%nB) Dar de baja un profe."
25                 + "%nM) Modificar algun atributo de un profesor."
26                 + "%nC) Consultar asignaturas impartidas por un profesor."
27                 + "%nL) Listado de todos los profes ordenados por nombre y
edad."
28                 + "%nS) Salir"
29                 + ">"
30                 + "%n";
31             String option = readString(mainMenu,possbopc).toLowerCase();
32             switch (option) {
33                 case "a" → {
34                     darAltaProfesor();
35                 }
36                 case "b" → {
37                     darBajaProfesor();
38                 }
39                 case "m" → {
40                     modificarProfesor();
41                 }
42                 case "c" → {
43                     consultarAsignaturasProfesor();
44                 }
45                 case "l" → {
46                     listadoProfesores();
47                 }
48                 case "s" → {
49                     exit = yesOrNo("¿Desea realmente salir?");
50                     exportarBinario();
51                 }
52                 default → System.out.printf("Opcion incorrecta%n%n");
53             }
54         }
55     }
56
57     /**
58     * Importar profesteca.

```

```

59     */
60     public void pasarPunteroAProfesteca(){
61         controlador.recibirPunteroAProfesteca(profesteca);
62     }
63
64     /**
65         * Leer y pasar al controlador los datos del profesor para darlo de
alta.
66     */
67     public void darAltaProfesor(){
68         String nombre;
69         int edad;
70         int cuantas;
71         ArrayList<String> asignaturas = new ArrayList<>();
72
73         nombre = readString("Ingrese el nobre del profesor: ");
74         edad = readInt("Ingrese edad del profesor: ");
75         cuantas = readInt("Ingrese cuantas asignaturas imparte el profesor:
");
76         for(int i=0;i<cuantas;i++){
77             asignaturas.add(readString("Ingrese nombre de la asignatura: "));
78         }
79         controlador.darAltaProfesor(nombre,edad,asignaturas);
80     }
81
82     /**
83         * Leer y pasar al controlador los datos del profesor para darlo de baja.
84     */
85     public void darBajaProfesor(){
86         String nombre = readString("Ingrese el noombre del profesor que desea
borrar: ");
87         if(controlador.darBajaProfesor(nombre)){
88             System.out.println("Profesor borrado exitosamente de la base de
datos.");
89         }else
90             System.out.println("El profesor que desea borrar, no se encuentra
en la base de datos.");
91     }
92
93     /**
94         * Modificar los datos del profesor seleccionado.
95     */
96     public void modificarProfesor(){
97         String nuevoAtributo;
98         String nombre = readString("Ingrese el nombre del profesor que desea
modificar: ");
99
100         if(controlador.existeProfesor(nombre))
101         {
102             boolean exit = false;
103             while(!exit){
104                 String mainMenu="n== QUE ATRIBUTO DESEA MODIFICAR =="
105                     + "%nN) Nombre."
106                     + "%nE) Edad."
107                     + "%nS) Salir"
108                     + ">"
109                     + "%n";
110                 String option = readString(mainMenu).toLowerCase();
111                 switch (option) {
112                     case "n" →{

```

```

113         nuevoAtributo = readString("Ingrese el nuevo nombre:
114     ");
115     controlador.modificarAtributoProfesor(nombre,option,nuevoAtributo);
116     }
117     case "e" →{
118         nuevoAtributo = readString("Ingrese la nueva edad:
119     ");
120     controlador.modificarAtributoProfesor(nombre,
121     option,nuevoAtributo);
122     }
123     case "s" → {
124         exit = yesOrNo("¿Desea realmente salir?");
125     }
126     }
127     }else
128     System.out.println("El profesor que desea modificar, no se
129     encuentra en la base de datos.");
130 }
131 /**
132  * Consultar asignaturas de un profesor
133  * Comprueba si existe el profesor y consulta e imprime las asignaturas.
134  */
135 public void consultarAsignaturasProfesor(){
136     String nombre = readString("Ingrese el nombre del profesor a
137     consultar: ");
138     if(controlador.existeProfesor(nombre)){
139         String asignaturas =
140         controlador.consultarAsignaturasProfesor(nombre);
141         System.out.println(asignaturas);
142     }
143 }
144 /**
145  * Crea matriz de profesores y lo imprime por pantalla a través de
146     printToScreen3
147  * @throws Exception
148  */
149 public void listadoProfesores() throws Exception{
150     String[][] matrizProfesores = controlador.listadoProfesores();
151     printToScreen3(matrizProfesores);
152 }
153 /**
154  * Exporta El ArrayList de los profesores a un archivo en .bin.
155  */
156 public void exportarBinario(){
157     controlador.exportarBinario();
158 }

```