

UD4	Práctica 1	Publicaciones
Integrantes	Sebastián Páez Delgado	
Repositorio	Unknown	

Desarrollo del ejercicio práctico:

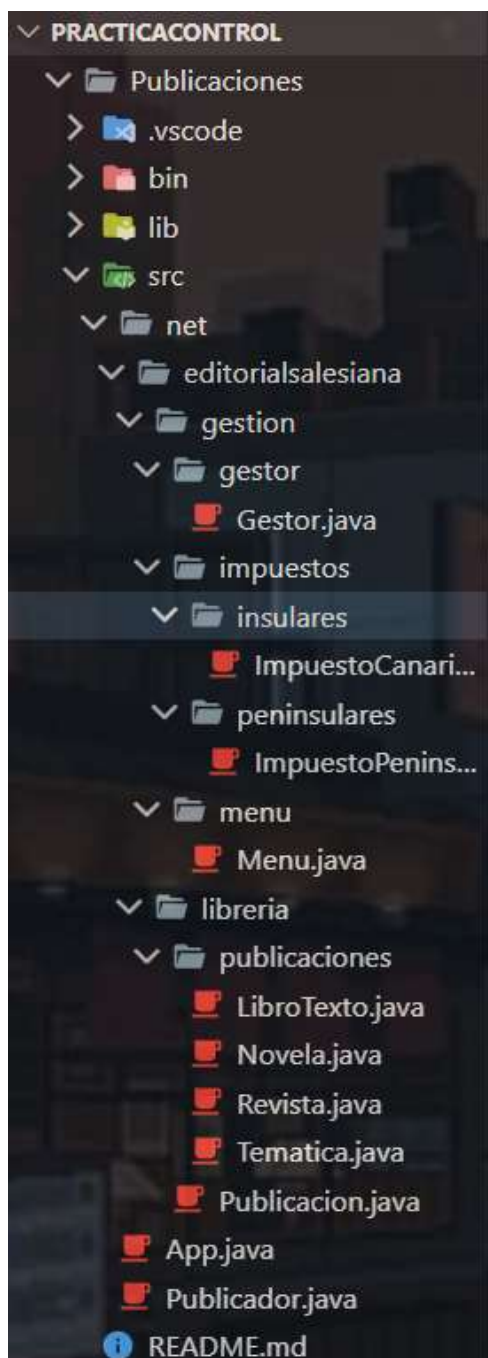


Presentado por el estudiante Juan Sebastián Páez Delgado – Salesianos la cuesta

Contenido

Contenido.....	2
Estructura de Paquetes	3
Clase Gestor	4
Añadir novela.....	4
añadir Novela.....	5
Añadir libro	6
Calcular impuestos	7
Impuesto Canario Como clase.....	7
Libro de texto Como clase	8
Novela Como Clase.....	9
Revista Como Clase.....	10
Tematica Como clase Enum	12
Publicacion Como Clase padre	12
Publicador Como clase main.....	14
Pruebas En Consola.....	14+
Caso A.....	16
Caso B.....	17
Caso C.....	18
Caso D.....	20
Caso E.....	21
Caso F	22
Despedida emotiva.....	22

Estructura de Paquetes



Estructuré el orden de los paquetes en base a su jerarquía, donde comenzamos desde nuestro dominio, pasando a nuestra editorial. Seguido nos dividimos en las diferentes áreas por la que nos moveremos.

Gestión, aquí irán todas aquellas clases encargadas del funcionamiento y ejecución de las acciones y métodos en nuestro código, siendo gestor el principal contribuyente. Donde caerán todos nuestros métodos útiles.

A continuación tenemos los impuestos en diferentes áreas del país España, peninsular e insular donde apuntamos únicamente a las islas canarias, donde estamos trabajando el proyecto actual.

Por último nuestro paquete menú, donde tendrá una clase quien devuelve un método estático para poder llamar al menú desde nuestra clase principal.

Librerías, como su nombre lo dice, almacenará todos los artículos de nuestra editorial, dándole atributos, quienes en su mayoría están en la clase Publicación, y de ella, heredará a sus clases hijas a quienes tendrán uno u otro atributo privado para ellas mismas.

Y finalmente tenemos nuestro publicador, quien se encargará de interactuar con el cliente o usuario final, y podrá elegir a través de las diferentes opciones que gestor presentará.

Clase Gestor

Añadir novela

```

1 package net.editorialsalesiana.gestion.gestor;
2
3 import java.util.Scanner;
4
5 import net.editorialsalesiana.gestion.impuestos.insulares.ImpuestoCanario;
6 import net.editorialsalesiana.gestion.impuestos.peninsulares.ImpuestoPeninsula;
7 import net.editorialsalesiana.libreria.Publicacion;
8 import net.editorialsalesiana.libreria.publicaciones.LibroTexto;
9 import net.editorialsalesiana.libreria.publicaciones.Novela;
10 import net.editorialsalesiana.libreria.publicaciones.Revista;
11 import net.editorialsalesiana.libreria.publicaciones.Tematica;
12

```

Dado que será la clase principal de métodos, importaremos cada clase para poder hacer uso de sus atributos y constructores para poder agregar la información a sus respectivos objetos.

```

public class Gestor {
    public static Revista añadirRevista() {
        String titulo = scanner.nextLine();

        System.out.println("¿Cuántas páginas tendrá?");
        int numeroPaginas = scanner.nextInt();
        scanner.nextLine();

        System.out.println("Ingrese un isbn");
        String isbn = scanner.nextLine();

        System.out.println("Ingrese un precio de distribución");
        float precioDistribucion = scanner.nextFloat();
        scanner.nextLine();

        System.out.println("Ingrese un precio público");
        float precioPublico = scanner.nextFloat();
        scanner.nextLine();

        System.out.println("Ingrese una fecha de edición");
        String fechaEdicion = scanner.nextLine();

        System.out.println("Ingrese el número de la revista");
        int numeroRevista = scanner.nextInt();
        scanner.nextLine();

        System.out.println("Ingrese el nombre de los artículos separados por comas (,)");
        String nombresArticulosSplit = scanner.nextLine();
        String[] nombresArticulos = nombresArticulosSplit.split(",");

        Revista nuevaRevista = new Revista(titulo, numeroPaginas, isbn, precioDistribucion, precioPublico,
            fechaEdicion,
            numeroRevista, nombresArticulos);
        for (int i = 0; i < publicaciones.length; i++) {
            if (publicaciones[i] == null) {
                publicaciones[i] = nuevaRevista;
                System.out.println("Revista agregada");
                break;
            }
        }

        if (publicaciones.length == 99) {
            scanner.close();
        }
        return nuevaRevista;
    }

    public static Novela añadirNovela() {

```


Nuestro método publico tipo revista añadirRevista se encargará a través de una serie de preguntas, de obtener los datos necesarios y asignarles los valores de estas preguntas a unas variables quienes consiguientemente se guardarán en un objeto Revista, despues de una pequeña validación donde se guardará en el próximo espacio vacío del array . y al llegar al límite del array, se cerrará nuestro escáner y terminará el programa.

```
public static Novela añadirNovela() {
    Scanner scannerr = new Scanner(System.in);

    System.out.println("Ingrese el titulo de la novela");
    String titulo = scannerr.nextLine();

    System.out.println("¿Cuántas paginas tendra?");
    int numeroPaginas = scannerr.nextInt();
    scannerr.nextLine();

    System.out.println("Ingrese un isbn");
    String isbn = scannerr.nextLine();

    System.out.println("Ingrese un precio de distribucion");
    float precioDistribucion = scannerr.nextFloat();
    scannerr.nextLine();

    System.out.println("Ingrese un precio publico");
    float precioPublico = scannerr.nextFloat();
    scannerr.nextLine();

    System.out.println("Ingrese una fecha de edicion");
    String fechaEdicion = scannerr.nextLine();

    System.out.println("Ingrese el nombre de el/la autor/a");
    String autor = scannerr.nextLine();

    Tematica tematica = null;

    while (tematica == null) {
        System.out.println(
            "Que tipo de tematica tendra esta novela?\n\t1) Negra\n\t2) Historica\n\t3) Juvenil\n" +
            "Por favor ingrese el numero de la tematica que gustaria elegir");
        int tematicaOpcionNumero = scannerr.nextInt();
        scannerr.nextLine();

        switch (tematicaOpcionNumero) {
```

Nuestro método añadir Novela es más de lo mismo, con el pequeño cambio a la hora de agregar su temática, quien mediante una clase Enum, creada a su mismo nivel en el paquete de publicaciones, Creará un objeto temática que será null para posteriormente asignarle un valor usando un bucle de una pregunta y una asignación de valor siempre que este esté vacío, donde según lo escogido por el usuario en el switch, temática nula tomará un valor.

```
while (tematica == null) {
    System.out.println(
        "Que tipo de tematica tendra esta novela?\n\t1) Negra\n\t2) Historica\n\t3) Juvenil\n" +
        "Por favor ingrese el numero de la tematica que gustaria elegir");
    int tematicaOpcionNumero = scannerr.nextInt();
    scannerr.nextLine();

    switch (tematicaOpcionNumero) {
        case 1:
            tematica = Tematica.NEGRA;
            break;
        case 2:
            tematica = Tematica.HISTORICA;
            break;
        case 3:
            tematica = Tematica.JUVENIL;
            break;
        default:
            System.out.println(
                "Por favor, ingrese alguno de los numeros dados antes segun la tematica que quiera");
            break;
    }
}
```

```
Novela novelita = new Novela(titulo, numeroPaginas, isbn, precioDistribucion, precioPublico,
    fechaEdicion, autor, tematica);

for (int i = 0; i < publicaciones.length; i++) {
    if (publicaciones[i] == null) {
        publicaciones[i] = novelita;
        System.out.println(x:"Novela agregada");
        break;
    }
}

if (publicaciones.length == 99) {
    System.out.println(x:"Son demasiadas novelas, por favor revise su colección");
    System.exit(status:0);
    scannerr.close();
}

return novelita;
}
```

Ya luego de tener todos los valores necesarios de nuestro objeto novela, podemos asignarlos e introducirlos dentro del primer hueco vacío de nuestro array. Y una vez el array llegue hasta 99, el Escáner se cerrará.

Obviando las preguntas que necesitaremos para crear nuestro objeto libro

```
System.out.println(x:"Que temas tendra, ingrese sus temas separados por comas (,)?");
String temas = scannerr.nextline();
String[] temasArray = temas.split(regex:",");
LibroTexto librito = new LibroTexto(titulo, numeroPaginas, isbn, precioDistribucion, precioPublico,
    fechaEdicion, nombreAsignatura, temasArray);

for (int i = 0; i < publicaciones.length; i++) {
    if (publicaciones[i] == null) {
        publicaciones[i] = librito;
        System.out.println(x:"librito agregada");
        break;
    }
}

if (temasArray.length== 99) {
    System.out.println(x:"Son demasiados temas pa un libro");
    System.exit(status:0);
    scannerr.close();
}

return librito;
}
```

Pedimos nuestros temas separados por coma, para poder crear un array en base a esa cadena de texto, donde el separador será la coma, una vez hecho esto la información se agregará al objeto como un array. Y el escanear se cerrará tras llegar a 99.

```
public static void calcularImpuestos() {  
  
    Scanner scanner = new Scanner(System.in);  
    System.out.println(x:"Ingrese el isbn de la obra de la cual quiere saber los impuestos?");  
    String opcionUsuario = scanner.nextLine();  
  
    String mensajePublicaciones = "";  
  
    for (int i = 0; i < publicaciones.length; i++) {  
        if (publicaciones[i] != null && opcionUsuario.equalsIgnoreCase(publicaciones[i].getIsbn())) {  
            float impuesto1 = ImpuestoPeninsula.sacarIva(publicaciones[i].getPrecioPublico());  
            float impuesto2 = ImpuestoCanario.sacarIgic(publicaciones[i].getPrecioPublico());  
            mensajePublicaciones = "El IVA es " + impuesto2 + " y el IGIC es " + impuesto1;  
            System.out.println(mensajePublicaciones);  
            break;  
        }  
    }  
    if (publicaciones.length > 100) {  
        scanner.close();  
    }  
}  
}
```

Por último de nuestra clase gestor con un método void calcular impuestos pediremos el ISBN al usuario ya que este atributo es un identificador único de cada objeto.

El String mensaje lo cree por fuera para poder asignarle un valor mas adelante y poder imprimirlo tras una validación para sacar los impuestos a los valores que no sean nulos y que sean iguales a nuestra ISBN dada de manera previa. Para luego tomar la información del array donde esta almacenado la publicación con el isbn igual al nuestro, tomar su precio al público y usando el método estático de calcular IVA e IGIC retornamos los valores en un sout. Terminando así nuestra clase gestor

Impuesto Canario

En nuestro método del impuesto canario usamos un constructor de valor único creado en nuestra clase padre publicación previamente, para tomar el precio al público, luego con este valor lo multiplicamos por 1*07 que seria el calculo para el impuesto IGIC.

La misma forma se usa para calcular el IVA, solo que en vez de un 1.07 usaremos un 1.21, poniendo F's despues del numero para indicar que son valores floats

```
package net.editorialsalesiana.gestion.impuestos.insulares;

import net.editorialsalesiana.libreria.Publicacion;

public class ImpuestoCanario extends Publicacion {

    public ImpuestoCanario(float precioPublico) {
        super(precioPublico);
    }

    public static float sacarIgc(float precioPublico) {
        return precioPublico * 1.07f;
    }
}
```

```
package net.editorialsalesiana.gestion.impuestos.peninsulares;

import net.editorialsalesiana.libreria.Publicacion;

public class ImpuestoPeninsula extends Publicacion {

    public ImpuestoPeninsula(float precioPublico) {
        super(precioPublico);
    }

    public static float sacarIva(float precioPublico) {
        return precioPublico * 1.21f;
    }
}
```

Libro de texto

```
package net.editorialsalesiana.libreria.publicaciones;

import net.editorialsalesiana.libreria.Publicacion;

public class LibroTexto extends Publicacion {

    private String asignatura;
    private String[] temas = new String[110];

    public LibroTexto(String titulo, int numeroPaginas, String isbn, float precioDis-
tribucion, float precioPublico,
        String fechaEdicion, String asignatura, String[] temas) {
        super(titulo, numeroPaginas, isbn, precioDistribucion, precioPublico, fechaEdi-
cion);
        this.asignatura = asignatura;
        this.temas = temas;
    }

    public String getAsignatura() {
        return asignatura;
    }

    public String[] getTemas() {
        return temas;
    }

    public void setTemas(String[] temas) {
        this.temas = temas;
    }

    @Override
    public String toString() {
        String mensaje = "_____<====<" + getTitulo() + ">====_\n" +
            "*Asignatura: <" + getAsignatura() + ">\n";
    }
}
```



```
    mensaje += "*Lista de temas:\n";
    for (String tema : temas) {
        mensaje += " \t - " + tema + "\n";
    }

    mensaje += "*Precio de distribución: <" + getPrecioDistribucion() + ">Euros\n"
+
        "*Precio de venta al publico: <" + getPrecioPublico() + ">Euros\n" +
        "=====";
    return mensaje;
}
}
```

Nuestra clase libro de texto se encargará de marcar sus atributos individuales como privados ya que solo las usará ella. Consiguientemente usamos el constructor y llamamos a la clase padre para poder heredar el resto de atributos. Nuestro array temas fue usado en la clase gestor para poder añadir los temas mediante un método toSplit, aunque este array en la clase libro no se utiliza para guardarlos, si no para darle un valor array al constructor. Luego irán nuestros métodos de acceso, el override con una concatenación en la variable mensaje, y un for each para la impresión de nuestro array

Novela

```
package net.editorialsalesiana.libreria.publicaciones;

import net.editorialsalesiana.libreria.Publicacion;

public class Novela extends Publicacion {

    private Tematica tematica;
    private String autor;

    public Novela(String titulo, int numeroPaginas, String isbn, float precioDistribucion, float precioPublico, String fechaEdicion, String autor, Tematica tematica) {
        super(titulo, numeroPaginas, isbn, precioDistribucion, precioPublico, fechaEdicion);
        this.autor = autor;
        this.tematica = tematica;
    }

    public Tematica getTematica() {
        return tematica;
    }
}
```

+

e

Revista

p

```
        this.numeroRevista = numeroRevista;
        this.nombresArticulos = nombresArticulos;
    }

    public int getNumeroRevista() {
        return this.numeroRevista;
    }

    public void setNumeroRevista(int numeroRevista) {
        this.numeroRevista = numeroRevista;
    }

    public String[] getNombreArticulo() {
        return this.nombresArticulos;
    }

    public void setNombresArticulos(String[] nombresArticulos) {
        this.nombresArticulos = nombresArticulos;
    }

    @Override
    public String toString() {

        String mensaje = "_____===<" + getTitulo() + ">===_____\\n" +
        "*Numero de revista: <"
            + getNumeroRevista() + ">\\n" +
            "*Lista de articulos: \\n";

        for (String titulosArticulos : nombresArticulos) {
            mensaje += " \\t - " + titulosArticulos + "\\n";
        }

        mensaje += "*Precio de distribucion: <" + getPrecioDistribucion() + ">Eu-
ros\\n"
            + "*Precio de venta al publico: <"
            + getPrecioPublico() +
            ">Euros\\n
=====";
        return mensaje;
    }
}
```

Para hacer amena la documentación, esta clase es exactamente lo mismo que las dos anteriores, usando un toSplit en nuestra clase gestor para ingresar el valor de los diferentes artículos y luego un for each en nuestro override, el resto son los métodos de acceso y constructores de la respectiva clase.

Tematica

```
package net.editorialsalesiana.libreria.publicaciones;

public enum Tematica {
    NEGRA,
    HISTORICA,
    JUVENIL
}
```

Tematica , aquí pusimos los valores de las temáticas que podrían tener nuestras novelas, luego accedimos a ellas mediante el nombre de la clase



Publicacion

```
package net.editorialsalesiana.libreria;

public class Publicacion {

    protected String titulo;
    protected int numeroPaginas;
    protected String isbn;
    protected float precioDistribucion;
    protected float precioPublico;
    protected String fechaEdicion;

    public Publicacion(String titulo, int numeroPaginas, String isbn, float precioDistribucion, float precioPublico, String fechaEdicion) {
        this.titulo = titulo;
        this.numeroPaginas = numeroPaginas;
        this.isbn = isbn;
        this.precioDistribucion = precioDistribucion;
        this.precioPublico = precioPublico;
        this.fechaEdicion = fechaEdicion;
    }
}
```



```
}

public Publicacion(float precioPublico) {
    this.precioPublico = precioPublico;
}

public Publicacion() {

}

public String getTitulo() {
    return titulo;
}

public void setTitulo(String titulo) {
    this.titulo = titulo;
}

public int getNumeroPaginas() {
    return this.numeroPaginas;
}

public void setNumeroPaginas(int numeroPaginas) {
    this.numeroPaginas = numeroPaginas;
}

public String getIsbn() {
    return this.isbn;
}

public void setIsbn(String isbn) {
    this.isbn = isbn;
}

public float getPrecioDistribucion() {
    return this.precioDistribucion;
}

public void setPrecioDistribucion(float precioDistribucion) {
    this.precioDistribucion = precioDistribucion;
}

public String getFechaEdicion() {
    return this.fechaEdicion;
}

public void setFechaEdicion(String fechaEdicion) {
    this.fechaEdicion = fechaEdicion;
}
```

```
public float getPrecioPublico() {  
    return this.precioPublico;  
}  
  
public void setPrecioPublico(float precioPublico) {  
    this.precioPublico = precioPublico;  
}  
  
}
```

Nuestra clase Padre, de donde la gran mayoría de atributos salieron, usando las variables protegidas para que sus hijos y solo sus hijos puedan heredarlas, también nuestros métodos de acceso públicos para que los métodos de otras clases puedan acceder al valor de las variables y usarlas en sus respectivas acciones.

Publicador

```
import java.util.Scanner;  
  
import net.editorialsalesiana.gestion.gestor.Gestor;  
import net.editorialsalesiana.gestion.menu.Menu;  
import net.editorialsalesiana.libreria.publicaciones.LibroTexto;  
import net.editorialsalesiana.libreria.publicaciones.Novela;  
import net.editorialsalesiana.libreria.publicaciones.Revista;  
  
public class Publicador {  
    public static void main(String[] args) {  
  
        Scanner lector = new Scanner(System.in);  
  
        Menu menu = new Menu();  
  
        boolean seguir = true;  
  
        // crear gestor y poner los arrays ahí y crear el objeto aquí  
  
        while (seguir) {  
            menu.llamarMenu();  
            String opcion = lector.nextLine();  
  
            switch (opcion) {  
                case "a":  
                    Revista revista = Gestor.añadirRevista();  
                    System.out.println(revista.getTitulo() + " ha sido agregada");  
                    break;
```

```
        case "b":
            LibroTexto librito = Gestor.añadirLibro();
            System.out.println(librito.getTitulo() + " ha sido agregado");
            break;

        case "c":
            Novela novelita = Gestor.añadirNovela();
            System.out.println(novelita.getTitulo() + " Ha sido agregada");
            break;

        case "d":
            Gestor.mostrarPublicaciones();
            break;
        case "e":
            Gestor.calcularImpuestos();
            break;
        case "f":
            seguir = false;
            break;
        default:
            break;
    }
}

lector.close();
}
```

Nuestra clase publicador llamara al menú, y mediante un switch y una pregunta al usuario manejara sobre las opciones llamando los diferentes métodos desde otras clases, en la opción A creamos un objeto revista y de valor le pasamos el de nuestro método Revista añadirRevista quien devolvía un objeto tipo revista ya rellenito de informacion. Lo mismo sucede en la clase B y C.

Nuestra caso D está accediendo de manera estática al método mostrar publicaciones mediante un for y una validación para no mostrar los vacíos, pasará por el array de publicaciones e imprimirá todas

Nuestro caso E Calculara los impuestos invocando un método desde gestor y gestor invoca el método estático que son dos clases por sí mismas explicadas arriba.

Pruebas Por Consola

A continuación vamos a empezar con las pruebas de código para comprobar que cada acción esta funcionando de manera correcta y como fue pedida mediante el documento enviado por el profesor.

Caso A

```
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del programa
a
Ingrese el titulo de la revista
Pepe x tylor swtich (entiendes, por que switch de loop)
Cuantas paginas tendra?
12
Ingrese un isbn
12-42
Ingrese un precio de distribucion
3
Ingrese un precio publico
10
Ingrese una fecha de edicion
12/24/2024
Ingrese el numero de la revista
78
Ingrese el nombre de los articulos separados por comas (,)
están realmente saliendo pepesado y tylor switch??, el descenso a la locura de peporro
Revista agregada
Pepe x tylor swtich (entiendes, por que switch de loop) ha sido agregada
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del programa
d
Lista de publicaciones:
=====<Pepe x tylor swtich (entiendes, por que switch de loop)>=====
*Numero de revista: <78>
*Lista de articulos:
    - están realmente saliendo pepesado y tylor switch??
    - el descenso a la locura de peporro
*Precio de distribucion: <3.0>Euros
*Precio de venta al publico: <10.0>Euros
=====
```



```
=====
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del programa
a
Ingrese el titulo de la revista
Los calvos son buenos en java?
Cuantas paginas tendra?
12
Ingrese un isbn
4214
Ingrese un precio de distribucion
4
Ingrese un precio publico
14
Ingrese una fecha de edicion
12/6/89
Ingrese el numero de la revista
129
Ingrese el nombre de los articulos separados por comas (,)
Todos los calvos conocidos en los salesianos son buenos programadores en java
Revista agregada
Los calvos son buenos en java? ha sido agregada
=====QUE DESEA HACER=====
```

Caso B

```
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del programa
b
Ingrese el titulo del libro
larry poker
Cuantas paginas tendra?
15
Ingrese un isbn
2130
Ingrese un precio de distribucion
12
Ingrese un precio publico
19
Ingrese una fecha de edicion
12/665
Ingrese el nombre de la asignatura
matematica
Que temas tendra, ingrese sus temas separados por comas (,)?
larry, poker, freePalestina
librito agregada
larry poker ha sido agregado
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del programa
```

Caso C

```
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del programa
c
Ingrese el titulo de la novela
lord of the arrays
Cuantas paginas tendra?
124
Ingrese un isbn
12423
Ingrese un precio de distribucion
3213
Ingrese un precio publico
5467
Ingrese una fecha de edicion
12/65
Ingrese el nombre de el/la autor/a
santi
Que tipo de tematica tendra esta novela?
    1) Negra
    2) Historica
    3) Juvenil
Por favor ingrese el numero de la tematica que gustaria elegir
1
Novela agregada
lord of the arrays Ha sido agregada
=====QUE DESEA HACER=====
```

```
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del programa
c
Ingrese el titulo de la novela
popap
Cuantas paginas tendra?
12
Ingrese un isbn
32
Ingrese un precio de distribucion
32
Ingrese un precio publico
13121
Ingrese una fecha de edicion
12/65
Ingrese el nombre de el/la autor/a
pepe
Que tipo de tematica tendra esta novela?
1) Negra
2) Historica
3) Juvenil
Por favor ingrese el numero de la tematica que gustaria elegir
3
Novela agregada
popap Ha sido agregada
=====QUE DESEA HACER=====
```

Caso D

```
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del programa
d
Lista de publicaciones:
=====
===<Pepe x tylor swtich (entiendes, por que switch de loop)>===
*Numero de revista: <78>
*Lista de articulos:
    - estan realmente saliendo pepesado y tylor switch??
    - el descenso a la locura de peporro
*Precio de distribucion: <3.0>Euros
*Precio de venta al publico: <10.0>Euros
=====
===<Los calvos son buenos en java?>===
*Numero de revista: <129>
*Lista de articulos:
    - Todos los calvos conocidos en los salesianos son buenos programadores en java
*Precio de distribucion: <4.0>Euros
*Precio de venta al publico: <14.0>Euros
=====
===<larry poker>===
*Asignatura: <matematica>
*Lista de temas:
    - larry
    - poker
    - freePalestina
*Precio de distribución: <12.0>Euros
*Precio de venta al publico: <19.0>Euros
=====
===<lord of the arrays>===
*Tematica: <NEGRA>
*Autor: <santi>
*Numero de paginas: <124>
*Precio de distribucion: <3213.0>Euros
*Precio de venta al publico: <5467.0>Euros
=====
===<popap>===
*Tematica: <JUVENIL>
*Autor: <pepe>
*Numero de paginas: <12>
*Precio de distribucion: <32.0>Euros
*Precio de venta al publico: <13121.0>Euros
=====
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
```


Caso E

```
*Precio de venta al publico: <13121.0>Euros
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del progama
e
Ingresa el isbn de la obra de la cual quiere saber los impuestos?
12-42
El IVA es 10.700001 y el IGIC es 12.1
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del progama
e
Ingresa el isbn de la obra de la cual quiere saber los impuestos?
2130
El IVA es 20.330002 y el IGIC es 22.990002
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del progama
e
Ingresa el isbn de la obra de la cual quiere saber los impuestos?
32
El IVA es 14039.471 y el IGIC es 15876.41
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del progama
f
=====QUE DESEA HACER=====
a. Añadir una nueva revista
```

Caso F

```
=====QUE DESEA HACER=====
a. Añadir una nueva revista
b. Añadir un nuevo libro de texto
c. Añadir una nueva novela
d. Mostrar el listado de todas las publicaciones
e. Selección de una publicación y cálculo de su precio en Canarias y en península.
f. Salir del programa
f
PS D:\1DAM\FPS DAM\Programacion\ud4-herenciaStatic\Herencia\PracticaControl> █
```

Gracias por leer