

Ejercicio para entregar

Diseñe jerarquías de clases para representar los siguientes conjuntos de objetos:

a. Los diferentes productos que se pueden encontrar en una tienda de electrónica tienen un conjunto de características comunes (precio, código de barras...) y una serie de características específicas de cada producto. Entre estos productos podemos encontrar distintos tipos de aplicaciones, entre las cuales hay aplicaciones para practicar deporte, aplicaciones para reproducir música y aplicaciones para expertos informáticos (trabajar con bbdd o programación)

Implemente en Java las jerarquías de clases que haya diseñado (incluyendo sus variables de instancia, sus constructores y sus métodos get/set). A continuación, crear un carrito de la compra en el que se pueden incluir productos (los productos se incluyen directamente en el programa de prueba) y emitir un ticket en el que figuren los datos de cada producto del carrito, incluyendo su precio y el importe total de la compra.

En un documento copia el código y haz capturas de pantalla donde se vea el correcto funcionamiento del programa. Incluye también una carpeta con los .java.
Sube el archivo comprimido resultante al apartado correspondiente en el fpribera.es

.....

❖ Yo particularmente he interpretado el ejercicio de la siguiente manera:

1. He creado una clase padre abstracta llamada producto
 2. Después he creado 3 clases hijas → deporte, música e informática
 3. Y por último una clase única llamada carrito donde invoco a las clases hija, introduzco los datos de los artículos e imprimo el ticket de venta para el cliente.
- En Deporte apps de musculación, dietas, mantenimiento, etc.
 - En Música apps de reproducción de ficheros de audio en varios formatos.
 - En Informática apps con lenguajes de programación en diferentes entornos de desarrollo.

Nota importante: aunque he creado variables para conseguir que el programa estuviera lo más completo posible, en esta ocasión en el ticket no he utilizado todos los parámetros, pero he decidido dejarlo incluido, para futuras versiones en las que imprima facturas con mas detalle.

Esta es la clase padre (Abstracta):

```
package Tienda;
```

```
//Esta es la clase padre(Abstracta)
```

```
public abstract class producto {
```

```
    //En adelante app = aplicación o programa
```

```
    public String nombre; //Nombre de la app de cara al cliente
```

```
    public int codproduc; //Codigo de la app de cara al cliente
```

```
    public double precio; //Precio de la app de cara al cliente
```

```
    public String proveedor; //Empresa que desarrolla la app
```

```
    public int codprovee; //Código de artículo de proveedor de la app
```

```
    public int nunids; //Cantidad adquirida
```

```
    public double importe; //Importe total dependiendo de la cantidad  
    adquirida
```

```
    //Constructor sin parámetros
```

```
    public producto() {
```

```
        nombre="";
```

```
        codproduc=0;
```

```
        nunids=0;
```

```
        precio=0.0;
```

```
        proveedor="";
```

```
        codprovee=0;
```

```
        importe=0.0;
```

```
    }
```

```
    //Constructor con parámetros
```

```
    public producto(int codproduc, String nombre, int nunids, double  
precio) {
```

```
        this.codproduc=codproduc;
```

```
        this.nombre=nombre;
```

```
        this.nunids=nunids;
```

```
        this.precio=precio;
```

```
        importe=nunids*precio;
```

```
    }
```

```
    //Getters & Setters
```

```
    public String getNombre() {
```

```
        return nombre;
```

```
    }
```

```
    public void setNombre(String nombre) {
```

```
        this.nombre = nombre;
```

```
    }
```

```
    public int getCodproduc() {
```

```
        return codproduc;
```

```
}

public void setCodproduc(int codproduc) {
    this.codproduc = codproduc;
}

public double getPrecio() {
    return precio;
}

public void setPrecio(double precio) {
    this.precio = precio;
}

public String getProveedor() {
    return proveedor;
}

public void setProveedor(String proveedor) {
    this.proveedor = proveedor;
}

public int getCodprovee() {
    return codprovee;
}

public void setCodprovee(int codprovee) {
    this.codprovee = codprovee;
}

public int getNunids() {
    return nunids;
}

public void setNunids(int nunids) {
    this.nunids = nunids;
}

public double getImporte() {
    return importe;
}

public void setImporte(double importe) {
    this.importe = importe;
}
}
```

Esta es la clase deporte (hija) de la clase padre (producto):

```
package Tienda;

//Esta es una clase hija de la clase producto(padre)
public class deporte extends producto {

    //ejercicio = ejercicio físico a realizar
    public String ejercicio;//musculaci¿%n, dieta, mantenimiento,
artes marciales, etc.

    //Constructor sin parámetros
    public deporte() {
        super();
        ejercicio="";
    }
    //Constructor con parámetros
    public deporte(String ejercicio, String nombre, int codproduc,
double precio,int nunids) {
        super(codproduc, nombre, nunids, precio);
        setEjercicio(ejercicio);
    }

    public String getEjercicio() {
        return ejercicio;
    }

    public void setEjercicio(String ejercicio) {
        this.ejercicio = ejercicio;
    }
}
```

Esta es la clase música (hija) de la clase padre (producto):

```
package Tienda;

//Esta es una clase hija de la clase producto(padre)
public class musica extends producto {

    //formato = tipo de música que va a reproducir
    public String formato; //Soporta mp3, mp4, ogg, etc

    //Constructor sin parámetros
    public musica() {
        super();
        formato="";
    }

    //Constructor con parámetros
    public musica(String formato, String nombre, int codproduc, double
precio, int nunids) {
        super(codproduc, nombre, nunids, precio);
        setFormato(formato);
    }

    public String getFormato() {
        return formato;
    }

    public void setFormato(String formato) {
        this.formato = formato;
    }
}
```

Esta es la clase informática (hija) de la clase padre (producto):

```
package Tienda;

//Esta es una clase hija de la clase producto(padre)
public class informatica extends producto{

    //lsop = lenguajes soportados por la app para programar
    public String lsop;

    //Constructor sin parámetros
    public informatica() {
        super();
        lsop="";
    }
    //Constructor con parámetros
    public informatica(String lsop, String nombre, int codproduc,
double precio, int nunids) {
        super(codproduc, nombre, nunids, precio);
        setLsop(lsop);
    }

    public String getLsop() {
        return lsop;
    }

    public void setLsop(String lsop) {
        this.lsop = lsop;
    }
}
```

Esta es la clase carrito en la que invoco las clases hija, introduzco datos de los artículos a vender por la tienda e imprimo el ticket de venta:

```
package Tienda;

//Esta es una clase unica en la que invocamos clases hijas
public class carrito {

    //Variables e invocaciones de clase
    String tcarrito;//ticket de carrito
    private deporte dep;
    private musica mus;
    private informatica inf;

    //Constructor donde introducimos los datos de los artículos
    carrito() {
        tcarrito="";
        dep = new deporte("Pesas", "Musculos",11111, 10.0, 2);
        mus = new musica("MP3","Wmedia",12345,5.0,3);
        inf = new informatica("Varios", "Eclipse",77777,250.0,2);
    }
    //Getters & Setters
    public deporte getDep() {
        return dep;
    }

    public void setDep(deporte dep) {
        this.dep = dep;
    }

    public musica getMus() {
        return mus;
    }

    public void setMus(musica mus) {
        this.mus = mus;
    }

    public informatica getInf() {
        return inf;
    }

    public void setInf(informatica inf) {
```

```

        this.inf = inf;
    }
    //Método que genera un numero aleatorio para el numero del ticket

    public void SetTCarrito() {
        int numero=(int)(Math.random()*9999)+1;//Número aleatorio
entre 1 y 9999
        System.out.println("");
        System.out.println("Nº TICKET: "+numero);
    }

    //Método para imprimir el ticket
    public void ImprimirTicket() {
        dep.importe = dep.precio*dep.nunids;
        inf.importe = inf.precio*inf.nunids;
        mus.importe = mus.precio*mus.nunids;

        double totalsiniva=dep.importe+inf.importe+mus.importe;

        double iva=(totalsiniva*21)/100;

        double total=(totalsiniva+iva);

        /*El TIPO es ejercicio a realizar en Deportes
        lenguajes a programar en Informatica
        y formato de musica a reproducir en Música*/

        //Aquí construyo el formato del ticket, que puedo
manipular sin problema
        SetTCarrito();//Mostramos el número de ticket
        System.out.println("");

        System.out.println("\tCODIGO"+ "\tTIPO"+ "\tNOMBRE"+ "\tTUNIDADES"+ "
PRECIO"+ "  IMPORTE");

        System.out.println("\t"+dep.codproduc+"\t"+dep.ejercicio+"\t"+dep.n
ombre+"  "+dep.nunids+
            "\t  "+dep.precio+"\t  "+dep.importe);

        System.out.println("\t"+inf.codproduc+"\t"+inf.lsop+"\t"+inf.nombre
+"\t  "+inf.nunids+
            "\t  "+inf.precio+"\t  "+inf.importe);

        System.out.println("\t"+mus.codproduc+"\t"+mus.formato+"\t"+mus.nom
bre+"\t  "+mus.nunids+
            "\t  "+mus.precio+"\t  "+mus.importe);
        System.out.println("");
    }

```



```

        System.out.println("\t\t\t\t\t TOTAL SIN IVA:
"+totalsiniva+"€");
        System.out.println("\t\t\t\t\t IVA 21%: "+iva+"€");
        System.out.println("\t\t\t\t\t *****");
        System.out.println("\t\t\t\t\t TOTAL");
        System.out.println("\t\t\t\t\t "+total+"€");
        System.out.println("\t\t\t\t\t *****");
    }
    //Método main
    public static void main(String[] args) {

        carrito c=new carrito();

        c.ImprimirTicket();
    }
}

```

Aquí va una captura del resultado del ticket de venta en consola:

<terminated> carrito [Java Application] C:\Program Files\Java\jre1.8.0_191\bin\javaw.exe (10 feb. 2019 23:00:45)

Nº TICKET: 6732

CODIGO	TIPO	NOMBRE	UNIDADES	PRECIO	IMPORTE
11111	Pesas	Musculos	2	10.0	20.0
77777	Varios	Eclipse	2	250.0	500.0
12345	MP3	Wmedia	3	5.0	15.0
				TOTAL SIN IVA: 535.0€	
				IVA 21%: 112.35€	

				TOTAL	
				647.35€	
