

# Práctica Tema 3

## Tienda Virtual

### Introducción

En el desarrollo de una aplicación para la gestión de una tienda virtual, se plantea la creación de un sistema que permita administrar la información detallada de los productos y servicios ofrecidos.

### Requisitos:

Cada uno de estos elementos que se pueden vender, tanto productos como servicios, deben tener al menos un nombre, un precio base, y características específicas que se detallan más adelante.

Existe un método para obtener el precio que en principio se calcula sobre el precio base más impuestos.

Por tanto, debe haber una variable de clase que almacene el % del impuesto aplicable. En principio se establecerá en un 7% de incremento del precio. Debe existir un método que permita modificar dicho impuesto.

Los productos deben contener información adicional como el nombre del fabricante, además del nombre y el precio base. También tienen asociado un peso y un volumen, esenciales para calcular el coste de envío.

Por ejemplo: Una aspiradora con nombre "Cyclone 23" de la marca "Braun" con precio de 89,99€ pesa 4.250 gramos y un volumen de 3.300 cm<sup>3</sup>.

El coste de envío se calculará a partir de un coste fijo de 2€ más el peso multiplicado por 0,0002€ por gramo y el volumen se incrementa en 1€ cada vez que se pasen los 1.000 cm<sup>3</sup>. Es decir, de 0 a 999 cm<sup>3</sup> no incrementa por volumen. De 1.000cm<sup>3</sup> a 1.999 cm<sup>3</sup> se incrementa en 1€, de 2.000 cm<sup>3</sup> a 2.999 cm<sup>3</sup> se incrementa en 2€. Y así sucesivamente. Por tanto, los gastos de envío para el "Cyclone 23" es de  $2 + (4.250 * 0,0002) + 4€ = 2 + 0,85 + 4 = 6,85€$ .

Algunos productos son perecederos y poseen una fecha límite de caducidad. Estos productos deben implementar un método para indicar si ya está caducado y otro para indicar cuántos días quedan para expirar el producto.

El precio de los productos perecederos disminuirá en un 10% un mes antes de caducar, y en un 25% diez días antes de la fecha límite.

Por ejemplo: Una jamón con nombre "Reserva Iberico" de la marca "Joselito" con precio de 215€ pesa 6.790 gramos y un volumen de 4.000 cm<sup>3</sup> que caduca el 15/03/2024.

Los servicios pueden ser de cuatro tipos:

- Eventos: son servicios que tienen una fecha de ejecución del evento. Por tanto, tienen que almacenar dicha fecha y además implementar los métodos de si el evento ya ha sido y cuántos días quedan para el evento.  
El precio de los eventos se incrementará un 20% una semana antes del evento, y un 50% si se contrata el mismo día del evento.
  - Estos servicios están vinculados a eventos con fechas específicas, como entradas para conciertos u obras teatrales.
  - Deben contar con métodos para:
    - Indicar si el evento ya ha ocurrido o no.
    - Mostrar la cantidad de días restantes para la realización del evento.
- Servicios con Sesiones:
  - Este tipo de servicio implica una contratación de un número fijo de sesiones, por ejemplo, "10 Clases de Piano".
  - Se debe incluir un mecanismo para registrar y disminuir el número de sesiones contratadas a medida que se consumen.
- Servicios Mixtos:
  - Combina características de los Servicios de Eventos y los Servicios con Sesiones.
  - Por ejemplo: "12 clases de Yoga en el mes de noviembre".
  - Tiene un número fijo de sesiones contratadas que deben consumirse antes de una fecha límite o caducidad.
  - Debe incluir los mecanismos de los eventos y los servicios por sesiones.
- Servicios Normales (de uso ilimitado):
  - Estos servicios no encajan en las categorías anteriores.
  - Por ejemplo, la compra de un libro electrónico, un disco o una película..
  - No tienen fecha de caducidad ni una cantidad limitada de sesiones contratadas.

Es crucial desarrollar una "estructura" que sirva como tienda para gestionar los productos y servicios.

La tienda debe almacenar todos los elementos que se venden en la tienda virtual.

La tienda debe ser capaz de hacer lo siguiente:

- Insertar elementos.
- Mostrar los elementos almacenados con sus características. Incluyendo el precio de venta.
- Mostrar sólo los productos.
- Mostrar sólo los servicios.
- Mostrar sólo los elementos que tienen una fecha de expiración.
- Mostrar los elementos que se pueden vender. Es decir, los que no han caducado.

Los precios se deben mostrar redondeados a dos decimales.

## Tareas a Realizar:

- Implementar las clases que consideres necesarias con sus propiedades y métodos de forma que cumplan con los requisitos indicados.
- Implementa en un fichero PHP la creación de una tienda a la que se le van a añadir los siguientes elementos (se añaden por código sin interactuar con el usuario):
  - 5 productos
    - 2 productos no perecederos
    - 1 producto perecedero que haya caducado
    - 1 producto perecedero que caduque en 2 o 3 días
    - 1 producto perecedero que caduque en 20 o 25 días.
  - servicios
    - 3 eventos: uno caducado, otro que caduque hoy y otro que caduque en unos meses.
    - 2 servicios por sesiones: uno al que le quedan pocas sesiones y otro al que no le queda ninguna.
    - 2 servicios mixtos: uno caducado y otro no.
    - 2 servicios normales.
- Crea otra página PHP que llame a la anterior y muestre el listado de los elementos vendibles con sus características y su precio final, filtrando según se le indique en la URL. Estos filtros pueden ser:
  - Mostrar todos los elementos almacenados.
  - Mostrar sólo los productos.
  - Mostrar sólo los servicios.
  - Mostrar sólo los elementos que tienen una fecha de expiración.
  - Mostrar los elementos que se pueden vender. Es decir, los que no han caducado.
- Crea otra página de prueba que solo tenga un listado de enlaces a la página anterior realizando cada uno de los filtros.

## Observaciones:

- Realiza un diagrama de clases a papel con todas las clases, sus herencias, relaciones e interfaces de forma que te sirva de ayuda para el desarrollo. También puedes entregarlo y se valorará positivamente.
- La estructura del código debe ser eficiente y reutilizable. Usa herencia, clases y métodos abstractos y/o interfaces para ello.
- No uses propiedades o métodos públicos si no son necesarios.
- Se valorará el uso de autoloading y namespaces.

## Entrega:

Entrega la carpeta con el proyecto o un enlace al repositorio. Recuerda que la carpeta **vendor** no debe incluirse.

También puedes entregar en la carpeta el diagrama de clases creado.

## Ampliación:

Voluntariamente y con puntuación extra.

Crea los test correspondientes a las Clases implementadas en PHPUnit.