

## 1) Carrito de la compra

- Crea una página PHP que permita elegir una serie de artículos, inventados por ti, de una tienda online. Cada checkbox o botón permite seleccionar un artículo, en el que se indica su precio.
- Al pulsar el botón “Enviar” del formulario, se indicará el detalle de la compra, así como el precio total de lo que hemos comprado.
- Si actualizamos los productos, el carrito deberá modificar los productos y precios de la nueva selección.

## 2) Gestión de festivos

Implementar una clase en Java llamada **Fecha**.

- Dicha clase tendrá cuatro atributos **dia, mes, anio y tipo** privados que serán cuatro números de tipo entero. Tipo nos indicará el formato de presentación de la fecha y puede tomar los siguientes valores:
  - ✓ Si tipo = 0 el formato de presentación será dd/mm/aaaa (**Ej. 26/01/2017**). Este será el tipo por defecto
  - ✓ Si tipo = 1 el formato de presentación será dd – nombre\_mes – aaaa (**Ej. 26 – Enero – 2017**)
  - ✓ Si tipo = 2 el formato de presentación será dd “de” nombre\_mes “de aaaa (**Ej. 26 de Enero de 2017**)
- También tendrá los siguientes métodos:
  - ✓ Constructor con cuatro parámetros que indicarán el día, el mes, el año y el tipo
  - ✓ Métodos Gets y Sets de todos los atributos
  - ✓ Método toString que devolverá un String de un objeto de la clase Fecha. El formato de la fecha que devolverá el String dependerá del formato de presentación indicado en el atributo tipo.

Se desea implementar una clase en Java llamada **Festivo** para gestionar las festividades de una localidad.

- Dispondrá de los siguientes atributos privados:
  - ✓ **nombre:** de tipo String que guardará el nombre de la festividad
  - ✓ **fecha:** objeto de la clase Fecha creada en el ejercicio 2 que guardará la fecha de la festividad
  - ✓ **numFestivos:** almacenará el número total de festivos. Su valor se incrementará cada vez que creamos un nuevo objeto de la clase Festivo y será común a todos los objetos de la clase.
- Tendrá los siguientes métodos:
  - ✓ Constructor con dos parámetros: nombre y fecha

- ✓ Métodos Gets y Sets
- ✓ Método diferencia que recibe un objeto de tipo Festivo como parámetro y devuelve un entero que indicará el número de días que hay entre el Festivo que invoca al método y el Festivo recibido como parámetro. La diferencia de días será siempre un número positivo, independientemente de cuál de los dos festivos es posterior. (No se tendrán en cuenta los años bisiestos)

Se pide implementar la clase Festivo y realizar un programa *index* que pida por pantalla los nombres, meses, días y años de 2 festivos y cree un objeto Festivo para cada uno de los festivos ingresados por el usuario. Posteriormente deberá llamar al método diferencia e indicará la diferencia de días entre ambos festivos.

Por pantalla se mostrará dicha diferencia con el formato del siguiente ejemplo:

**Fiesta 1:**

La Almudena

9 de Noviembre de 2017

**Fiesta 2:**

El Pilar

12 de Octubre de 2017

**Diferencia:** 28 días

### 3) Gestión de empleados

Crea una clase NIF que represente el DNI con su correspondiente letra.

Los atributos de la clase serán el **número de DNI y su letra**. La clase NIF dispondrá de los siguientes métodos:

- **Un constructor que reciba como parámetro el DNI y calcule y asigne la letra que le corresponde.**
- **Un método leer():** que pida por teclado el número de DNI y calcule a partir del DNI introducido la letra que le corresponde y se los asigne al objeto.
- **Método llamado toString()** que devuelva un String con el NIF de la siguiente forma: ocho dígitos (de tener menos de ocho dígitos se completará con ceros por la izquierda), un guión y la letra en mayúscula. Por ejemplo: 12345678-Z

**Método para obtener la letra del NIF:**

La letra del NIF se calculará usando un método privado. La forma de obtener la letra del NIF es la siguiente: Se obtiene el resto de la división entera del número de DNI entre 23 y se usa la siguiente tabla para obtener la letra que corresponde: 0 - T 1 - R 2 - W 3 - A 4 - G 5 - M 6 - Y 7 - F 8 - P 9 - D 10 - X 11 - B 12 - N 13 - J 14 - Z 15 - S 16 - Q 17 - V 18 - H 19 - L 20 - C 21 - K 22 - E

Crea una clase Empleado que tenga los siguientes atributos privados:

- **Nif.** (utilizar la clase NIF creada anteriormente)
- **Nombre.**
- **Sueldo base.**
- **Horas extra** realizadas en el mes.
- Tipo de **IRPF (%)**.
- **Casado o no.** (Será de tipo boolean)
- **Número de hijos.**
- **Importe de la hora extra.** Este será un atributo común para toda clase.

Los objetos Empleado se crean con un **constructor con un solo parámetro correspondiente al número de DNI (sin letra, la letra será calculada automáticamente por la clase nif)**.

Además de los **métodos getter/setter** la clase Empleado tendrá estos métodos:

- **Método para el cálculo del complemento** correspondiente a las horas extra realizadas.
- **Método para calcular el sueldo bruto** (sueldo base + complemento por horas extras)
- **Método para calcular las retenciones por IRPF.** El porcentaje de IRPF se aplica sobre el sueldo bruto, teniendo en cuenta que el porcentaje que hay que aplicar es el tipo menos 2 puntos si el empleado está casado y menos 1 punto adicional por cada hijo que tenga.
- **Método toString()** para mostrar los datos de los empleados de la siguiente forma:

12345678-A Lucas Guerrero Arjona  
Sueldo Base: 1150.0  
Horas Extras: 4  
tipo IRPF: 15.0  
Casado: S  
Número de Hijos: 2

Hacer un programa para probar la clase.