

# DEW1

**Martínez Sánchez, Jaime –**  
**[martinezsanchezjaime@gmail.com](mailto:martinezsanchezjaime@gmail.com)**  
**[jaimarsa@inf.upv.es](mailto:jaimarsa@inf.upv.es)**

**Grupo: 3TI21**

**Identificación: dew-jaimarsa-1718.dsic.cloud**

## AUTOEVALUACIÓN

**Actividad 3: 3**

**Actividad 4: 3**

**Actividad 4.1: 2**

**Actividad 5: 1**

# ACTIVIDADES

**Actividad 3:** En esta actividad, lo que hemos hecho ha sido arreglar los elementos de HTML 4 que han quedado obsoletos con respecto a la nueva versión, HTML 5. Para ello, lo que he hecho ha sido eliminar las etiquetas `<tt>` que están obsoletas, y sustituirlo por elementos CSS.

Para llevar a cabo esto, lo que he hecho ha sido definir un nuevo estilo CSS en la cabecera, al que he llamado Estilo4, que es una letra con fuente monoespacio, que es lo que se encargaba de hacer la etiqueta `<tt>`, y entonces le hemos asignado este estilo con etiqueta `<span>` a aquellas palabras que lo necesitaban.

Por otra parte, para solucionar el error que había en la línea 46 y 47, lo que he hecho ha sido meter todo el texto en el mismo `<li>` para evitar el problema que nos encontrábamos al no empezar con ninguna etiqueta.

Si seguimos bajando, nos encontramos con el problema de que la etiqueta `<ul>` no permite que haya inmediatamente dentro de ella, otra etiqueta `<ul>`. Para solucionarlo, lo que hemos hecho ha sido meter la segunda etiqueta `<ul>` en un `<li>`, ya que `<ul>` sí que acepta esta etiqueta inmediatamente dentro de ella.

Por último, en la zona donde se encuentra la imagen, marcada por la etiqueta `<img>`, nos encontramos con que se necesita el atributo `alt`, que es el texto que acompaña a una imagen para poder diferenciarlas fácilmente. Por ello le hemos puesto a este atributo el nombre "lucene", que es el nombre que se le da a este archivo (lucene\_demo.png).

**Actividad 4:** Para la consecución de esta actividad, lo que hemos hecho ha sido crear un directorio en nuestro proyecto, al que hemos llamado `css`; en él hemos metido aquellos archivos con extensión `.css` necesarios, en este caso, el archivo que estaba en PoliformaT `estiloAzul.css`, y luego los archivos `estiloFondamento.css` y `estiloMontserrat.css`, que los vamos a necesitar para la actividad, descargado desde Google Fonts.

A continuación, lo que he hecho ha sido incluir en mi proyecto el archivo `ej1.html`, para darle los distintos estilos que hemos descargado. Para ello he creado tres copias de esta página: `ej1Estilo.html`, `ej1Fondamento.html` y `ej1Montserrat.html`, y les he referenciado a cada uno su correspondiente archivo `css`, gracias a una etiqueta `<link>`, en el que he indicado el directorio en el que se encontraban cada estilo, y de esta forma se han establecido los nuevos estilos de las distintas páginas.

**Actividad 4.1:** Para la realización de esta actividad, hemos necesitado de Bootstrap y las hojas de estilo Sketchy y Superhero, las cuales han sido descargadas desde Bootswatch, obteniendo sendos archivos bootstrap.css y bootstrap-min.css.

Lo que he hecho ha sido coger la página ej1.html que hemos descargado previamente y la hemos editado para que incluyera ambas hojas de estilo, en dos páginas diferentes, que se encuentran en dos directorios distintos, junto a sus hojas de estilo. El procedimiento es igual para ambos estilos, así que únicamente vamos a ver uno de ellos, en este caso el de Sketchy.

Para conseguir el estilo de Sketchy, lo que he hecho ha sido incluir los archivos bootstrap.css y bootstrap-min.css en el directorio correspondiente, y lo he referenciado desde la página. Para hacer esta actividad, haremos uso de las clases que se encuentran en bootstrap.

Lo primero que he hecho ha sido crear la barra de navegación. Para ello, en el principio del <body> de la página, hemos metido en una etiqueta <nav> la clase navbar, bg-dark (para el color de la barra), y fixed-top (para que la barra tenga la forma que se pide). Dentro de esta, hemos metido una etiqueta <div> con la clase navbar-dark, para darle el color al texto “DEW 1718/Sketchy”, creado con una etiqueta <a> y la clase navbar-brand. Tras esto, he creado los botones “Inicio”, “Los títulos...”, “Área de texto” y “Pie”, de la misma manera que el texto anterior, pero referenciándolos de manera que vayan a los sitios del texto que se les pide.

Lo siguiente que he hecho ha sido meter la primera parte de la página en una especie de recuadro, para ello he usado la clase jumbotron. A continuación, había que partir el texto sobrante en dos columnas. Para ello he usado primero la clase row, para que estuvieran en la misma fila, y dentro la clase col-6, para separarlo en columnas cada parte de la página. He usado col-6 porque era la que mejor aprovechaba el espacio a nuestra disposición.

Por último, hemos hecho una tabla sobre la parte final de la página, para lo que hemos usado la clase table-bordered dentro de la etiqueta <table>, para que la tabla tuviera los bordes que se piden.

**Actividad 5:** Para esta actividad, necesitamos la hoja de estilo `tsr.css`. Una vez referenciada, comienzo a hacer las diferentes condiciones.

En primer lugar, tenemos que comprobar que no hay ningún campo vacío; para ello se usa el atributo `required` en cada campo introducido con la etiqueta `<input>`, excepto en el apartado donde se elige el tipo de tarjeta, que lo que he hecho ha sido que siempre haya una opción marcada con el atributo `checked`. De esta manera, se asegura que no se enviará la orden hasta que estén completos todos los apartados.

Para comprobar que el precio por unidad es un entero positivo, hemos puesto el tipo como `"number"`, y el atributo `min` en 0. De la misma forma, para poder marcar más de un departamento, deberemos pulsar la tecla `Ctrl` de nuestro teclado mientras marcamos los distintos departamentos que queremos. Para que se elija un solo tipo de tarjeta, tenemos el `radiobutton`, que hace que solo puedas marcar una opción al mismo tiempo, ya que si eliges otra se elimina la que tenías marcado.

Los demás condiciones no son posibles de realizar con HTML y se necesitará JavaScript.