

# DISEÑO DE INTERFACES WEB

## U.D. 5 (parte 1/2): Librerías de CSS (Bootstrap)



# Librerías de CSS

- Son bibliotecas de estilos visuales genéricos ya preparados para usar que se emplean para desarrollar páginas Web. Además, suelen añadir utilidades como componentes para hacer cuadros de diálogo, tablas, carrusel de imágenes, etc.
- La más usada es Bootstrap de Twitter (ya liberada). Originalmente llamada Blueprint, fue creada por Mark Otto y Jacob Thornton en 2011 como un marco de trabajo (framework).
- Otras: Foundation, Bulma. <https://geekflare.com/es/best-css-frameworks/>

# Librerías de CSS

- Ventajas:
  - Acelera el desarrollo de las páginas Web.
  - Definiciones de estilos de tipografía para los elementos HTML, así no hay que definirlos para cada proyecto.
  - Facilita el trabajo en equipo ya que puede mantener la misma línea de desarrollo de las hojas de estilo.
  - Creación de una rejilla para colocar los elementos que forman parte de la página Web. Ésta divide los espacios en columnas, ayuda a posicionar con CSS los componentes de una manera precisa y versátil.
  - Garantiza que lo que se desarrolle sea compatible con todos los navegadores.

# Librerías de CSS

- Desventajas:

- Se utilizan muchos estilos y no se emplean todos. Esta utilización de gran cantidad elementos y código hace que la página Web tarde más en cargarse.
- Salvo que se personalice, lo que requiere más conocimientos, la página Web se va a parecer otras miles que se han desarrollado con la misma librería o framework.
- Tiene curva de aprendizaje, lo primero es aprender a usarla.

# Bootstrap

- Es una **biblioteca multiplataforma** o conjunto de herramientas de código abierto **para desarrollos Web responsive** con HTML, CSS y JavaScript.
- Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como extensiones de JavaScript adicionales.
- Sitio oficial: <https://getbootstrap.com>



# Bootstrap

- Primero en promover la filosofía “mobile-first” (diseñar primero para tamaños de pantalla más pequeños).
- Consiguió el diseño responsivo (óptima visualización en distintos tipos de dispositivos finales) introduciendo la idea de rejilla (grid).
- Utiliza SASS (antiguamente LESS), un preprocesador de CSS (herramienta que permite escribir pseudocódigo CSS que luego será compilado y convertirse en CSS).

# Descarga

- Hay varias formas, si no se desea alojarlo, se puede incluirlo desde una CDN (Content Delivery Network) vía jsDelivr.
- Como resultado, se cargará desde la caché cuando se visite tu sitio Web, lo que conduce a un tiempo de carga más rápido. Además, la mayoría de los CDN se asegurarán de que una vez que un usuario le solicite un archivo, se entregará desde el servidor más cercano a ellos, lo que también conduce a un tiempo de carga más rápido.

# Plantilla básica

- Bootstrap necesita un Doctype de HTML5.
- Dentro de la etiqueta `<head>` solamente se anexa el archivo CSS.
- Al final de `<body>` se anexa el archivo JS:  
Colocarlo dentro de él ya que el contenido del sitio Web se cargará más rápido, y solo después de eso se procesará el archivo JS.



# Contenedores

- Bootstrap requiere de un elemento contenedor para envolver el contenido del sitio Web.
- Hay dos clases de contenedores para elegir:
  - `.container`: Proporciona un contenedor de ancho fijo sensible.
  - `.container-fluid`: Proporciona un contenedor de ancho completo que abarca todo el ancho de la ventana gráfica.
- Más en: <https://esdocu.net/bootstrap/5.3/layout/containers>

# Contenedores

- Breakpoints:

	<b>xs</b> <576px	<b>sm</b> ≥576px	<b>md</b> ≥768px	<b>lg</b> ≥992px	<b>xl</b> ≥1200px	<b>xxl</b> ≥1400px
<b>Container</b> <small>max-width</small>	None (auto)	540px	720px	960px	1140px	1320px
<b>Class prefix</b>	<small>.col-</small>	<small>.col-sm-</small>	<small>.col-md-</small>	<small>.col-lg-</small>	<small>.col-xl-</small>	<small>.col-xxl-</small>
<b># of columns</b>	12					
<b>Gutter width</b>	1.5rem (.75rem on left and right)					
<b>Custom gutters</b>	<a href="#">Yes</a>					
<b>Nestable</b>	<a href="#">Yes</a>					
<b>Column ordering</b>	<a href="#">Yes</a>					

Extra small, Small, Medium, Large, Extra large y Extra extra large

- Grid system (rejilla): <https://esdocu.net/bootstrap/5.3/layout/grid/>

# Ejercicios

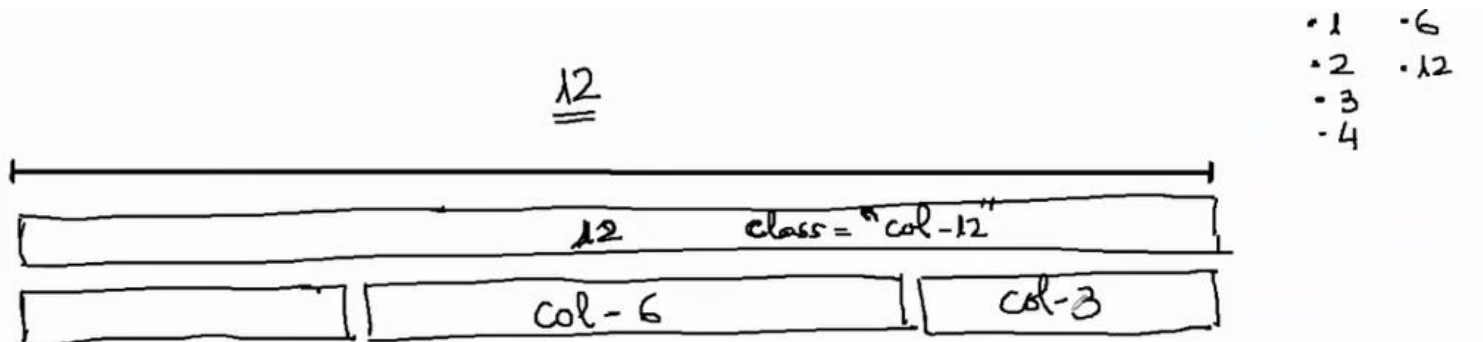
- Realiza todos los ejercicios del bloque 'Ejercicios 5.1'.

# Sistema de rejilla responsive

- Bootstrap incluye un sistema de rejilla ***responsive*** de 12 columnas (si no se desea utilizar las 12 individualmente, se puede agruparlas para crear columnas más anchas).
- Este sistema de rejilla o cuadrículas funciona así:
  - Dentro de un contenedor: `<div class="container"></div>`  
`<div class="container-fluid"></div>`
  - Se crean filas (row): `<div class="row"></div>`
  - Y dentro de cada fila se crean columnas (col): `<div class="celda col-12">CELDA 1.1</div>`
  - Solamente las columnas pueden ser “hijos” inmediatos de las filas.
  - Las columnas se crean con la clase “col”, pero admiten ciertas modificaciones.

# Sistema de rejilla responsive

- Al definir la estructura, se le indica el ancho de la columna de la siguiente forma (ejemplos):
  - Si se crea 1 columna: Ocupará el 100% del ancho de la fila `class="col-12"`.
  - Si se crean 3 columnas y una es el doble de las otras 2: Una ocupará el 50% `class="col-6"` y las otras 2 el 25% cada una `class="col-3"`.

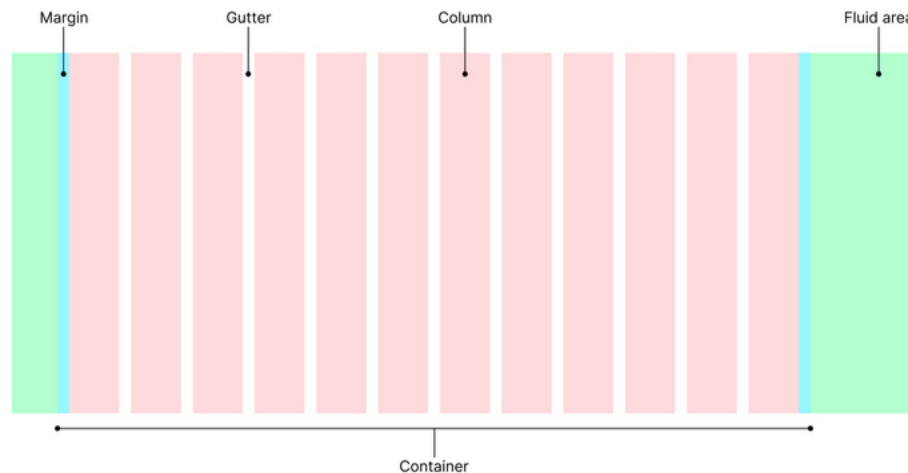


# Ejercicios

- Realiza todos los ejercicios del bloque 'Ejercicios 5.2'.
- Realiza todos los ejercicios del bloque 'Ejercicios 5.3'.
- Realiza todos los ejercicios del bloque 'Ejercicios 5.4'.

# Gutters

- Son el “padding” (espaciado) entre las columnas.
- Se utilizan para espaciar y alinear de manera *responsive* el contenido en el sistema de cuadrícula de Bootstrap. Por defecto es 24px y 0.



- Más en: <https://esdocu.net/bootstrap/5.3/layout/gutters>

# Botones

- Bootstrap incluye varios estilos de botones predefinidos.
- Más en: <https://esdocu.net/bootstrap/5.3/components/buttons>



# Ejercicios

- Realiza el 'Ejercicio 5.5'.

# Grupos de Listas

- Son un componente flexible y poderoso para mostrar una serie de contenido en formato listado.
- Suele utilizarse para crear menús de navegación, sistemas de pestañas, lista de elementos, etc.
- Más en: <https://esdocu.net/bootstrap/5.3/components/list-group>

# Ejercicios

- Realiza el 'Ejercicio 5.6'.

# Alertas

- Bootstrap incluye varios estilos de alertas predefinidos.
- También se puede insertar código **jQuery** y **JavaScript** para, por ejemplo, cerrar ventanas de avisos.
- Bootstrap 5 deja de usar jQuery:  
<https://platzi.com/tutoriales/1331-bootstrap/9730-por-que-bootstrap-5-dejo-de-usar-jquery-y-que-es-lo-que-viene/>  
<https://www.tortugacode.com/como-cambiar-de-jquery-a-vanilla-js-con-bootstrap-5/>
- Más en: <https://esdocu.net/bootstrap/5.3/components/alerts>

# Ejercicios

- Realiza el 'Ejercicio 5.7'.
- Realiza todos los ejercicios del bloque 'Ejercicios 5.8'.

# Navbar

- Es un barra o encabezado de navegación adaptable.
- Más en: <https://esdocu.net/bootstrap/5.3/components/navbar>  
<https://getbootstrap.com/docs/5.3/components/navbar/>  
<https://blog.hubspot.es/website/bootstrap-navbar#crear-navbar-bootstrap>

# Display

- Oculta de manera rápida los componentes.
- Más en: <https://esdocu.net/bootstrap/5.1/utilities/display/>  
<https://getbootstrap.com/docs/5.3/utilities/display/>

# Ejercicios

- Realiza el 'Ejercicio 5.9'.



# Tablas

- Documentación y ejemplos para el diseño de tablas (dado su uso frecuente en complementos de JavaScript).
- Más en: <https://esdocu.net/bootstrap/5.3/content/tables>  
<https://www.eniun.com/tablas-bootstrap-5-clases-estilo/>

# Card

- Es un contenedor de información flexible. Incluye opciones para encabezados y pie de página, una amplia variedad de contenido, colores de fondo contextuales y potentes opciones de visualización.
- Más en: <https://esdocu.net/bootstrap/5.3/components/card>  
<https://www.tutorialesprogramacionya.com/bootstrap4ya/detalleconcepto.php?punto=43&codigo=43&inicio=40>

# Carousel

- Es un componente de presentación de dispositivos para recorrer elementos (imágenes, textos, vídeos, etc.) como un carrusel.
- Más en: <https://esdocu.net/bootstrap/5.3/components/carousel>  
<https://blog.hubspot.es/website/bootstrap-carousel>

# Collapse

- Es un componente que se utiliza para mostrar y ocultar contenido. Los botones o anclajes se utilizan como activadores que se asignan a elementos específicos que alterna.
- Más en: <https://esdocu.net/bootstrap/5.3/components/collapse>

# Modal

- Es un componente que se utiliza para agregar cuadros de diálogo para lightboxes, notificaciones de usuario o contenido completamente personalizado.
- Más en: <https://esdocu.net/bootstrap/5.3/components/modal/>

# Formularios

- Permiten al usuario introducir datos los cuales son enviados a un servidor Web para ser procesados.
- Más en: <https://esdocu.net/bootstrap/5.3/forms/overview>  
<https://blog.hubspot.es/website/formulario-bootstrap>

# Ejercicios

- Realiza el 'Ejercicio 5.10'.

# DISEÑO DE INTERFACES WEB

**U.D. 5 (parte 2/2):  
Preprocesadores de CSS (LESS)**





# Preprocesadores de CSS

- Son herramientas que permiten escribir pseudocódigo CSS que luego será compilado y convertirse en CSS nativo.
- Lenguajes de hojas de estilos (extensiones de CSS) que permiten añadir nuevas funcionalidades para hacer el trabajo de diseño y maquetación del sitio Web más sencillo.
- Ejemplo: LESS ***LEaner Style Sheets*** (lenguaje de hojas de estilo). Creado por Alexis Sellier en 2009.
- Otros: SASS, Stylus. <https://es.bitdegree.org/tutoriales/preprocesador-css/>

# LESS



- Es un programa que preprocesa un archivo con un formato propio (.less) para convertirlo en un archivo CSS convencional, que se podrá utilizar en cualquier navegador como si se hubiese escrito directamente en CSS nativo.
- Al ser una extensión de CSS, el lenguaje LESS no es solo retrocompatible con CSS, sino que proporciona características extra usando sintaxis CSS. Hace que aprender LESS sea sencillo, y si se tienen dudas, siempre se podrá recurrir a CSS.
- Sitio oficial: <https://lesscss.org>

# Compilación

- Se puede compilar LESS en el lado del cliente, en el lado del servidor, usar un compilador online, una aplicación de escritorio, un plug-in para un IDE o un automatizador de tareas.

- Compilador en el lado cliente:

```
<link rel="stylesheet/less" type="text/css"
href="styles.less" />
<script
src="https://cdn.jsdelivr.net/npm/less"></script>
```

- Compilador online: <https://lesscss.org>

# Variables

- En CSS no hay variables.
- Permiten especificar valores que van a ser utilizados en un único lugar, y además poder reusarlas en otras partes de la hoja de estilo, pudiendo realizar cambios globales modificando una única línea de código.
- Solo se pueden modificar en los archivos “.less” (antes de compilar): Como solo existen en Less en el momento en el que se compilan, y cuando se convierten en CSS desaparecen al sustituirse por su valor, vistas desde el CSS se podrían considerar como **constantes**.

# Variables

- Ejemplo:

```
// LESS
```

```
@color: #4D926F;  
@background: #3d3d3d;  
  
#header {  
  color: @color;  
  background: @background;  
}  
h2 {  
  color: @color;  
}
```

```
/* CSS compilado */
```

```
#header {  
  color: #4D926F;  
  background: #3d3d3d;  
}  
h2 {  
  color: #4D926F;  
}
```

# Mixins

- Permiten incluir todas las propiedades de una clase en otra diferente añadiendo simplemente el nombre de la clase como una propiedad más. El comportamiento es similar al de las constantes o variables pero con clases enteras.
- Pueden comportarse también como funciones y tomar argumentos.
- Permiten repeticiones de código más eficientes y limpias, así como modificaciones de código más fáciles y rápidas.

# Mixins

- Ejemplo:

// LESS

```
.rounded-corners (@radius: 5px) {  
  -webkit-border-radius: @radius;  
  -moz-border-radius: @radius;  
  border-radius: @radius;  
}  
  
#header {  
  .rounded-corners;  
}  
#footer {  
  .rounded-corners(10px);  
}
```

/\* CSS compilado \*/

```
#header {  
  -webkit-border-radius: 5px;  
  -moz-border-radius: 5px;  
  border-radius: 5px;  
}  
#footer {  
  -webkit-border-radius: 10px;  
  -moz-border-radius: 10px;  
  border-radius: 10px;  
}
```

# Anidamientos

- CSS admite anidamiento lógico, pero los bloques de código no están anidados.
- LESS permite anidar los selectores dentro de otros. Esto hace la herencia clara y las hojas de estilo más cortas.



# Anidamientos

- Ejemplo:

// LESS

```
#header {  
  h1 {  
    font-size: 26px;  
    font-weight: bold;  
  }  
  p {  
    font-size: 16px;  
    a {  
      text-decoration: none;  
      color: red;  
      &:hover {  
        border-width: 1px;  
        color: #fff;  
      }  
    }  
  }  
}
```

/\* CSS compilado \*/

```
#header h1 {  
  font-size: 26px;  
  font-weight: bold;  
}  
#header p {  
  font-size: 16px;  
}  
#header p a {  
  text-decoration: none;  
  color: red;  
}  
#header p a:hover {  
  border-width: 1px;  
  color: #fff;  
}
```

# Operaciones y Funciones

- Las operaciones permiten la suma, resta, división y multiplicación de valores de propiedades y colores, que se pueden utilizar para crear relaciones complejas entre propiedades.
- Las funciones son transformaciones uno a uno con código JavaScript, permitiendo la manipulación de valores.

# Operaciones y Funciones

- Ejemplo:

// LESS

```
@the-border: 1px;
@base-color: #111;
@red:        #842210;

#header {
  color: @base-color * 3;
  border-left: @the-border;
  border-right: @the-border * 3;
}
#footer {
  color: @base-color + #003300;
  border-color: desaturate(@red, 10%);
}
```

/\* CSS compilado \*/

```
#header {
  color: #333333;
  border-left: 1px;
  border-right: 3px;
}
#footer {
  color: #114411;
  border-color: #7d2717;
}
```

# Ejercicios resueltos

- Vídeos:

<https://www.youtube.com/watch?v=BszfHCbswPY>

<https://www.youtube.com/watch?v=J8JzVo4BNLs>

<https://www.youtube.com/watch?v=7dpagECJ-Rk>

- Ejemplos:

<https://programandoointentandolo.com/2017/07/tutorial-less-4-variables-less.html>

<https://programandoointentandolo.com/2017/08/tutorial-less-7-mixins-less.html>

<https://programandoointentandolo.com/2017/08/tutorial-less-8-mixins-parametricos.html>

<https://programandoointentandolo.com/2017/08/tutorial-less-9-mixins-condicionales.html>

<https://programandoointentandolo.com/2017/09/tutorial-less-10-mixins-rekursivos-bucles.html>

<https://programandoointentandolo.com/2017/10/tutorial-less-13-funciones-colores-less.html>

<http://amatellanes.github.io/lesscss.org/>