

# PROGRAMACION WEB - MEDIA QUERIES Y BOOTSTRAP

## Abstract

El diseño responsive es posible gracias a las "Media Queries" de CSS. Las Media Queries son una forma de aplicar condicionales a las reglas de CSS. Éstas le dicen al navegador qué reglas debe ignorar o aplicar dependiendo del dispositivo del usuario.

## 1. Media Queries

Las Media queries le permiten aplicar estilos CSS según el tipo general de un dispositivo (como impresión o pantalla) u otras características como la resolución de la pantalla o el ancho del viewport del navegador. Las media queries se utilizan para lo siguiente:

- Para aplicar estilos condicionalmente utilizando las reglas de arroba CSS @media e @import.
- Para segmentar medios específicos para style, link, source y otros HTML con el atributo media. .

Una media query se compone de un tipo de medio opcional y cualquier cantidad de expresiones de características de medios, que pueden combinarse opcionalmente de varias maneras usando operadores lógicos. Las consultas de medios no distinguen entre mayúsculas y minúsculas. Para ampliar esta información, usted puede acceder [aquí](#) y [aquí también](#) que es de donde he tomado cierta información.

Todo esto es importante para el diseño web responsivo, ésto hace que la página web se vea bien en todos los dispositivos. Para ello se utiliza solamente HTML y CSS. No es un Javascript ni un programa. Para revisar un ejemplo completo al respecto, por favor entre [Aquí](#)

## 2. Breakpoints: Categorías

Por convención los tamaños son los siguientes:

Tamaño	Dispositivo
320px	Para dispositivos con pantallas pequeñas, como los teléfonos en modo vertical
480px	Para dispositivos con pantallas pequeñas, como los teléfonos, en modo horizontal
600px	Tabletas pequeñas, como el Amazon Kindle (600×800) y Barnes & Noble Nook (600×1024), en modo vertical
768px y 1023px	Tabletas de diez pulgadas como el iPad (768×1024), en modo vertical
1024px	Tabletas como el iPad (1024×768), en modo horizontal, así como algunas pantallas de ordenador portátil, netbook, y de escritorio
1200px	Para pantallas panorámicas, principalmente portátiles y de escritorio

**NOTA IMPORTANTE:** Existe una forma muy sencilla de hacer esto de manera alternativa, utilizando media grids, a través de Grid minmax.

Para ello revisar los siguientes links

- [Grid minmax](#)
- [Grid autocolumns](#)

## 3. Pseudo-clases y Pseudo-elementos

Se utiliza una pseudoclase para definir un estado especial de un elemento.

Por ejemplo, se puede utilizar para:

Aplicar estilo a un elemento cuando un usuario pasa el mouse sobre él o para diseñar los enlaces visitados y no visitados de manera diferente. Otro uso, es por ejemplo para cambiar el estilo a un elemento cuando recibe el foco.

### Sintaxis:

```
selector:pseudo-class {  
  property: value;  
}
```

Es de vital importancia notar que, no todos los elementos HTML pueden soportar todas las pseudoclases, para acceder a los ejemplos y a la lista de elementos permitidos, por favor haga clic [aquí](#)

Los pseudoelementos persiguen un objetivo muy similar; sin embargo la diferencia es que las pseudoclases indican un estado y el pseudo-elemento una característica o acción sobre un elemento

## 4. BEM

BEM significa Modificador de Bloques de Elementos (Block Element Modifier) por sus siglas en inglés. Sugiere una manera estructurada de nombrar a sus clases, basada en las propiedades del elemento en cuestión. BEM tiene como horizonte modularizar lo máximo posible cada uno de los bloques de código dispuesto. Se centra en tres parámetros o variables posibles: bloques (div, section, article, ul, ol, etc.), elementos (a, button, li, span, etc.) y modificadores. Estos últimos

se definen de acuerdo a la posterior utilización que haga el desarrollador a cargo.

Puede revisar lo contenido en el siguiente [link](#)

## 5. Frameworks y librerías

Dentro de CSS, nosotros podemos crearnos nuestras propias librerías o podemos hacer uso de librerías existentes.

En el caso de CSS, existen ciertas librerías, que nos permiten trabajar con distintos objetivos, una librería muy interesante es [Normalize.css](#). Ésta hace que los navegadores muestren todos los elementos de manera más consistente y acorde con los estándares modernos. Se dirige precisamente sólo a los estilos que necesitan normalizarse. Como práctica, requiero que ingrese a la página de Normalize y la utilice con la finalidad de revisar cómo funciona y las formas existentes de usar esta librería.

Utilizar también [Balloon.css](#) y revisar los ejemplos y [Colofilter.css](#).

Buscar más librerías y ver cómo usarlas para los diferentes elementos y páginas que guste. Pejm. para transparencias, estilos atractivos de letras, etc.

## 6. CMS

CMS son las siglas de Content Management System, en español Sistema de Gestión de Contenidos. En síntesis, un CMS le permite crear, organizar, publicar y eliminar contenidos de SU sitio web.

Como un ejemplo de un CMS y como práctica en clases, desplegaremos un CMS y lo desplegaremos en nuestro servidor local. Como ejemplo puede ser [Joomla](#).

## 7. Bootstrap

Bootstrap es un framework front-end utilizado para desarrollar aplicaciones web y sitios mobile first, o sea, con un layout que se adapta a la pantalla del dispositivo utilizado por el usuario. Ver los siguientes ejemplos [aquí](#).

Otro producto un tanto parecido podría ser [Tailwind](#) aunque es mucho más largo de usar y difícil.

Revisar también [pico.css](#), un software muy bonito y minimalista.

Para implementar un framework es necesario documentarse mucho, ésto para poder utilizarlos ya que tenemos dos estilos, los que son opinionated y los unopinionated.

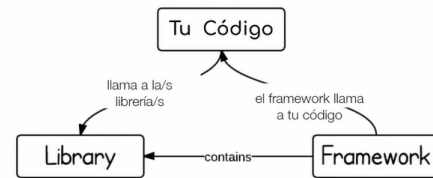
Bootstrap es opinionated y nos dice cómo podemos hacer algo, luego si queremos, pues o personalizamos.

### ACTIVIDAD IMPORTANTE

Al respecto, requiero que por favor verifiquen qué es un framework opinionated y uno non-oninionated. Hay diferencia con aquellos obstinados? o es lo mismo.

Cómo funciona bootstrap, se presenta en la imagen a continuación:

## ¿Cómo funciona?



Bootstrap usa varias librerías, nos promete un desarrollo rápido y responsive, son sus dos ofertas más importantes, por lo que no nos interesa que librerías usa, sino entender cómo usar. Cada framework tiene una versión y evoluciona, por lo que no importa el uso de las librerías particulares, sino el framework en sí. Tenemos que entender cómo éste funciona.

Es necesario verificar la compatibilidad con los navegadores [aquí](#)

Vamos a ver cómo se ve una página con Bootstrap [aquí](#). A continuación, en dicha página revisaremos cómo funciona el uso de componentes en bootstrap.