

Esta clase va a ser

- grabada

Hoy jugaremos con...

Kahoot!

**Puedes descargar la app en tu
teléfono celular o jugar online...**

Clase 12. DESARROLLO WEB

@ Media + Pseudo clases

Temario

11

Frameworks CSS + Bootstrap

- ✓ Librerías y Frameworks
- ✓ Bootstrap
- ✓ Estructura de Bootstrap

12

@Media + Pseudoclasses

- ✓ [Responsive](#)
- ✓ [Pseudo Clases](#)
- ✓ [Algunos ejemplos](#)

13

SASS

- ✓ Procesadores CSS
- ✓ Nesting, import y vars

Objetivos de la clase



Repasar diseño responsive con Bootstrap.



Conocer el uso de pseudo clases

CLASE N°11

Glosario

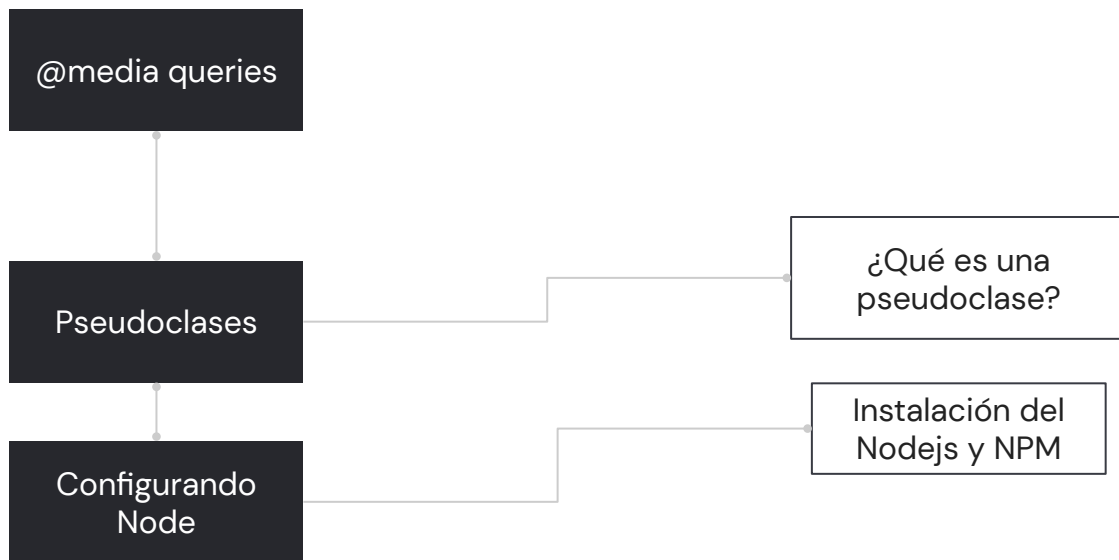
Librería: colección de elementos que suelen tener un propósito concreto.

Framework: conjunto de herramientas y código, para trabajar de acuerdo con una metodología, utilizando determinados patrones.

Bootstrap: Es un framework que permite crear interfaces web con CSS y JavaScript. Es responsive

Responsive design: Que adapta la interfaz del sitio web al tamaño del dispositivo en que se visualice.

MAPA DE CONCEPTOS



Estado del Proyecto Final

Estado del Proyecto Final

La clase anterior conocimos la segunda pre-entrega del Proyecto Final. Ahora, contaremos con un tiempo de la clase para compartir dudas e inquietudes sobre sus trabajos.

¿Quién quiere comenzar? 🙌

En esta instancia no nos focalizaremos en casos muy específicos, sino en las consultas generales que puedan ayudar a toda la comisión.

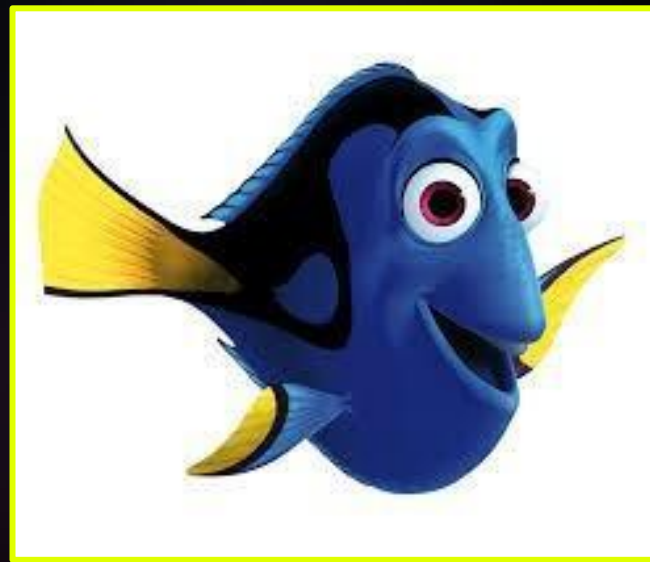
Responsive



PARA RECORDAR

Vamos a repasar responsive

Volvemos a tocar el tema, pues para aprovechar Bootstrap, es fundamental conocer y entender muy bien el responsive. Cualquier cambio que se realice altera el diseño en todos los dispositivos.

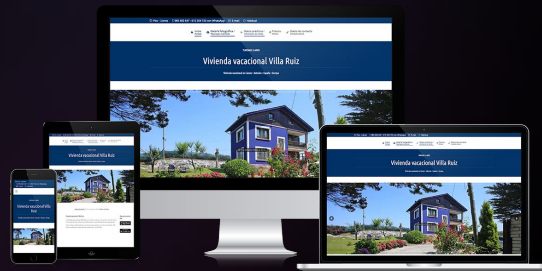




PARA RECORDAR

Página responsive

El sistema detecta automáticamente el ancho de la pantalla y, a partir del mismo, adapta todos los elementos de la página, desde el tamaño de letra hasta las imágenes y los menús, ofreciendo al usuario la mejor experiencia posible.





PARA RECORDAR

Recordemos



Responsive Web Design

Mobile First Web Design



Media Queries

Recordemos los rangos que define Bootstrap:

- ✓ Pantallas extra pequeñas (móviles) $< 576\text{px}$.
- ✓ Pantallas pequeñas (_sm, tablets _en vertical) $\geq 576\text{px}$.
- ✓ Pantallas medianas (md, para tablets en horizontal) $\geq 768\text{px}$.
- ✓ Pantallas grandes (lg, tamaño escritorio) $\geq 992\text{px}$.
- ✓ Pantallas extra grandes (xl, escritorio grande) $\geq 1200\text{px}$.

[Breakpoints en Bootstrap](#)

Ejemplo de Media Queries



Si quisiera que en las pantallas extra pequeñas (xs) el color de fondo que aplica la clase **.miestilo** sea rojo, y para el resto de tamaños sea verde, podrías hacer:

```
.miestilo {  
    background-color: green;  
}  
  
@media (max-width: 768px) {  
    .miestilo {  
        background-color: red;  
    }  
}
```

Ejemplo de Media Queries



Si quisiera variar la alineación del texto que se aplica en una clase, a partir de las pantallas tipo escritorio:

```
.miestilo {  
    text-align: center;  
}  
  
@media (max-width: 992px) {  
    .miestilo {  
        text-align: left;  
    }  
}
```


Ejemplo de Media Queries



Puedes también modificar el cuerpo de texto, si lo utilizas en px para diferentes pantallas:

```
@media (min-width: 600px) {  
  div.example {  
    font-size: 80px;  
  }  
@media (max-width: 600px) {  
  div.example {  
    font-size: 30px;  
  }  
}
```

Recuerda que esto no es necesario si utilizas el font-size en "em".z

Operador “AND”



Puedes sumar diferentes indicaciones con las Media Queries, las cuales se utilizan con el operador “and”. En este caso, el estilo que definido se reproducirá en pantallas que van de 400px a 700px:

```
@media (max-width: 700px) and (min-width: 400px) {  
  .miestilo {  
    text-align: left;  
  }  
}
```

Orientación



En este caso, sólo se reproducirá el estilo si la ventana tiene un ancho de de 700px o más, y la pantalla está en formato horizontal.

```
@media (min-width: 700px) and (orientation: landscape)
{
    .miestilo {
        text-align: left;
    }
}
```

Ejemplo de Media Queries



Modificación de menú de posición horizontal en pantallas grandes, a vertical en pantallas chicas.

HTML

```
<div class="topnav">
  <a href="#">Home</a>
  <a href="#">Nosotros</a>
  <a href="#">Contacto</a>
</div>
```

CSS

```
/* Top navigation bar */
.topnav {
  overflow: hidden;
  background-color: #333;}

/* Topnav links */
.topnav a {
  float: left;
  display: block;
  color: #f2f2f2;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none; }

@media screen and (max-width: 600px) {
  .topnav a {
    float: none;
    width: 100%;
  }
}
```

Ejemplo de Media Queries



Puedes modificar la distribución de las columnas para que en pantallas pequeñas se vean una abajo de la otra:

Desktop

Responsive Four Column Layout

Resize the browser window to see the responsive effect. On screens that are 992px wide or less, the columns will resize from four columns to two columns. On screens that are 600px wide or less, the columns will stack on top of each other instead of next to eachother.

Column 1 Some text..	Column 2 Some text..
Column 3 Some text..	Column 4 Some text..

Mobile

Responsive Four Column Layout

Resize the browser window to see the responsive effect. On screens that are 992px wide or less, the columns will resize from four columns to two columns. On screens that are 600px wide or less, the columns will stack on top of each other instead of next to eachother.

Column 1 Some text..
Column 2 Some text..
Column 3 Some text..

[Ver el código](#)

Ejemplo de Media Queries

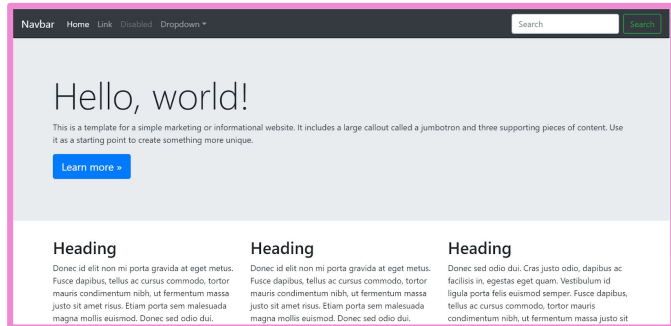
Puedes llegar a hacer cosas muy avanzadas y personalizar completamente el aspecto de una web según el tamaño del dispositivo.

- ✓ Cambiar el tamaño y la posición de una imagen.
- ✓ Modificar la posición de cualquier elemento.
- ✓ Cambiar el tamaño de letra, la fuente o su color.
- ✓ Aplicar combinaciones de estilos avanzados.

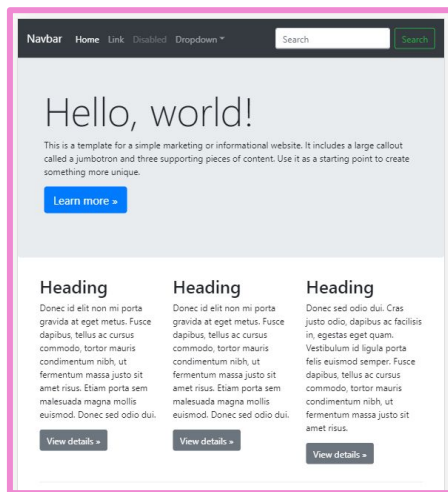
¡Cualquier cosa que se te ocurra! 🚀

Ejemplo

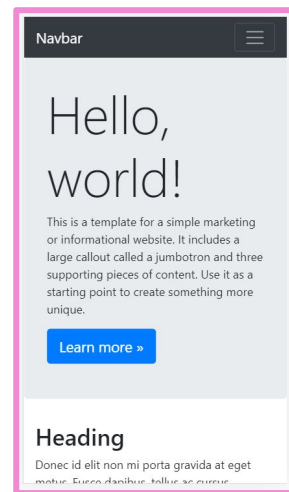
Desktop



Tablet



Mobile





Break

¡10 minutos y volvemos!

¿Preguntas?

#CoderTip: Ingresa al [siguiente link](#) y revisa el material interactivo que preparamos sobre **Preguntas Frecuentes**, estamos seguros de que allí encontrarás algunas respuestas.

Pseudo clases

¿Qué es una pseudo clase? 🤔

Una pseudoclase CSS es una palabra clave que se añade a los selectores y que especifica un estado especial del elemento seleccionado.

Por ejemplo, `:hover` aplicará un estilo cuando el usuario haga hover sobre el elemento especificado por el selector.

`selector:pseudoclase { propiedad: valor; }`

¿Qué es una pseudo clase? 🤔

Las pseudoclasas, junto con los pseudoelementos, permiten aplicar un **estilo** a un elemento no solo en relación con el contenido del árbol de documento, sino **también en relación a factores externos** como:

- ✓ El historial del navegador (:visited, por ejemplo),
- ✓ El estado de su contenido (como :checked en algunos elementos de formulario),
- ✓ La posición del ratón (como :hover que permite saber si el ratón está encima de un elemento o no).

Algunos ejemplos

```
a:hover{ font-weight:bold; background-color:blue;  
color:white; text-decoration:underline; }  
a:link{ text-decoration:none; color:red; }  
a:visited{ color:#999999; }  
a:active{ background-color:yellow; color:black; }
```

Ejemplos más complejos

Otra pseudo clase muy útil pero poco utilizada, aplica a todos los elementos A, excepto a los incluidos como elementos B. Es decir, podríamos aplicar unos estilos a todos los elementos "div" y evitar que éstos se apliquen a otras capas con un id o class determinado. En el siguiente ejemplo hacemos justo esto, evitando que se apliquen los estilos a las capas "padre".

Ejemplos más complejos

```
div:not(.padre){  
    padding-left:40px;  
    padding-right:40px;  
}
```


Otros ejemplos...

La solución a todos los problemas de maquetación de antaño con las filas y los bloques.

👉 Con `:nth-child(N)` podremos aplicar sus estilos a todos los elementos hijos cuya posición sea un número "N" respecto a un padre. Este número N no tiene que ser necesariamente un número entero para especificar una posición fija (la posición 2, por ejemplo), ya que también permite insertar fórmulas y palabras específicas.

Otros ejemplos...

```
.padre div:nth-child(2){ font-weight:bold;  
color:orange; }
```

¡Atención!

Recuerda instalar **Node.js** para la próxima clase.



[Ver tutorial](#)

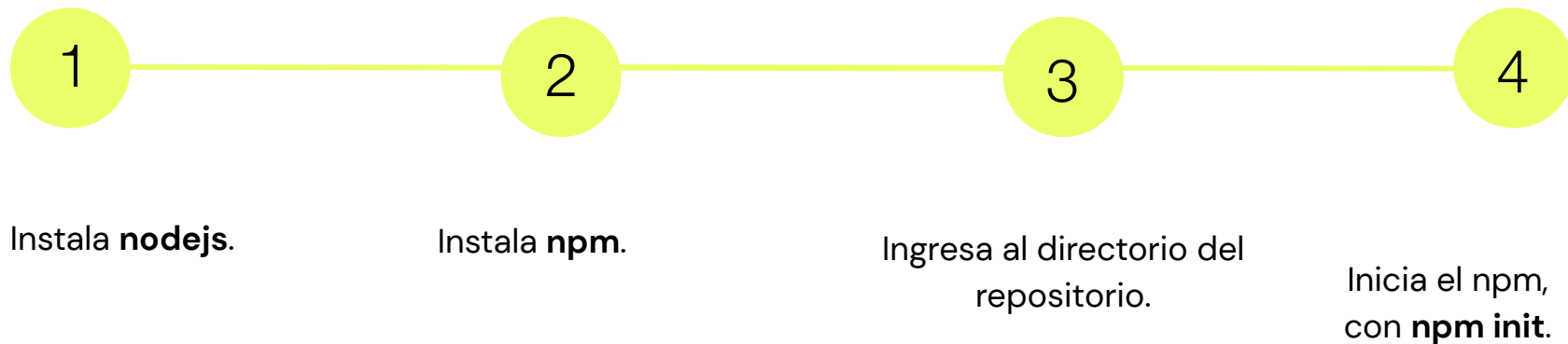
Configurando Node



Ejemplo en vivo

Instalación de Node.js y el NPM

Repasando la instalación del procesador



Repasando la instalación del procesador

5

Instala el nodemon:
**npm install -D
node-sass
nodemon.**

6

Crea la carpeta
SCSS y CSS y sus
archivos
respectivos.

7

**Edita el package.json e inserta los
lineas.**

- "build-css":
- "node-sass --include-path
scss scss/prueba.scss
css/pruebacss.css",
 - "watch-css": "nodemon -e
scss -x \"npm run build-css\""

8

Compila
con: **npm
run
watch-css.**



Repaso

¿Qué tal si hacemos una trivia para repasar lo aprendido en las últimas dos clases? 😊

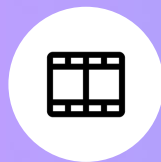


Encuesta

Por encuestas de Zoom

¡Terminamos el **módulo 6: Frameworks CSS!**

Cuéntanos qué temas te resultaron más complejos de entender. **Puedes elegir más de uno.** Vamos a retomar aquellos temas que resultaron de mayor dificultad en el próximo AfterClass.



¿Quieres saber más?
**Te dejamos material
ampliado de la clase**



MATERIAL AMPLIADO

Recursos multimedia

- ✓ [Themes y templates](#) | TE LO EXPLICO CON GATITOS
- ✓ [Íconos Material Design](#) | Material Design Icons

Disponible en nuestro repositorio.

¿Preguntas?

Resumen de la clase hoy

- ✓ Aplicar nuevas herramientas de Bootstrap.
- ✓ Aplicar un theme de Bootstrap.
- ✓ Repasar Responsive.

Opina y valora
esta clase

Muchas gracias.

#DemocratizandoLaEducación