

HINTS

USO DE MEDIA QUERY

Al momento de dar responsividad a nuestros proyectos, no solamente tenemos la posibilidad de usar los Media Query como puntos de quiebres para definir las grillas. Si es de nuestro agrado, podemos también utilizar los Media Query dando distinto estilo en cada uno de los tamaños escribiendo las indicaciones en cada punto de quiebre. Esto lo usaremos principalmente cuando, en cierto tamaño de pantalla algún elemento no solo deba ocupar una distribución distinta si no, verse distinto en color, forma, etc.

Si tenemos el siguiente código HTML:

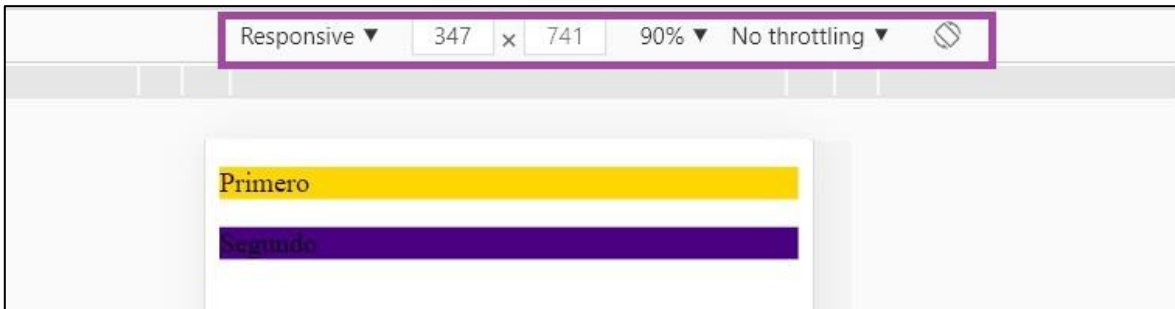
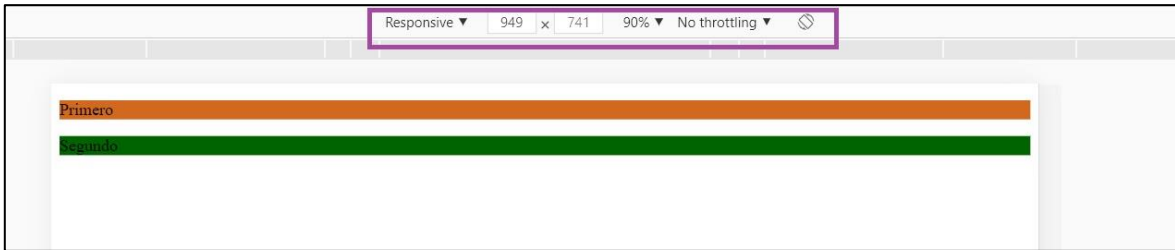
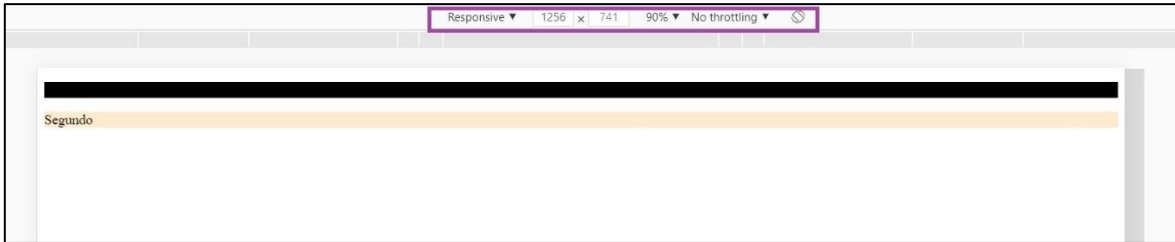
```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-
7 scale=1.0">
8   <link rel="stylesheet" href="estilo.css">
9   <title>Media Query</title>
10 </head>
11 <body>
12   <div class="primero"><p>Primero</p></div>
13   <div class="segundo"><p>Segundo</p></div>
14 </body>
15 </html>
```

En el CSS podemos dar un color distinto según el tamaño de la pantalla, sin tocar más el Documento HTML:

```
1 @media (min-width:1200px) {
2   .primero{
3     background-color: black;
4   }
5   .segundo{
6     background-color: blanchedalmond;
```

```
7      }
8  }
9  @media (min-width:992px) and (max-width:1199px){
10     .primero{
11         background-color: blue;
12     }
13     .segundo{
14         background-color: cadetblue;
15     }
16 }
17 @media (min-width:768px) and (max-width:991px){
18     .primero{
19         background-color: chocolate;
20     }
21     .segundo{
22         background-color: darkgreen;
23     }
24 }
25 @media (min-width:576px) and (max-width:767px){
26     .primero{
27         background-color: darkred;
28     }
29     .segundo{
30         background-color: deeppink;
31     }
32 }
33 @media (max-width:575px){
34     .primero{
35         background-color: gold;
36     }
37     .segundo{
38         background-color: indigo;
39     }
40 }
```

El resultado es:



OCULTAR UN ELEMENTO

En ocasiones podemos observar que las páginas web tienen algunos elementos que al verlas por un teléfono móvil no existen. Esto es parte también de los distintos diseños que se crean para cada tamaño de pantalla.

En el siguiente ejemplo (donde utilizaremos el diseño de página web creado durante los *exercises* del CUE) se mostrará la creación de un `<div>` `class` "oculto" que utilizaremos para mostrar u ocultar el elemento dependiendo el tamaño de la pantalla. Este elemento lo colocaremos entre el `<main>` y el `<footer>` de nuestra página web.

```
1 <div class="oculto">
2     <p>TEXTO QUE OCULTAREMOS</p>
3 </div>
```

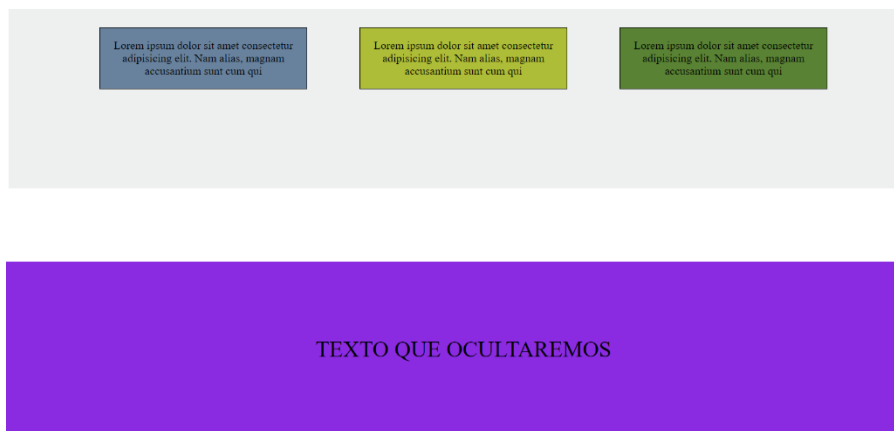
En nuestro CSS le daremos algunas indicaciones de estilo, como color de fondo o tamaño, solo para diferenciarlo bien.

```
1 .oculto{
2     background-color: blueviolet;
3     width: 100%;
4     text-align: center;
5     padding-top: 5%;
6     padding-bottom: 5%;
7     font-size: xx-large;
8 }
```

En nuestro navegador actualizaremos nuestra página web y la veremos en tamaño de *smartphone*. Podemos encontrar al final de la página un recuadro con el texto "texto que ocultaremos".



Al hacerlo en pantalla grande ocurre lo mismo.



En nuestro **CSS**, utilizando *Medias Query* de la misma forma que lo hicimos en el **HINTS** anterior, indicaremos que, aquellas pantallas menores a **990 px** no mostrarán este elemento. Para eso, usaremos la indicación **display none**.

Como el diseño comenzó en pantalla de tamaño pequeño, podríamos colocar en la indicación general el **display none** y, utilizando un punto de quiebre, indicar que las pantallas mayores a **990 px** si muestren elemento, usando **display block**.

A continuación, se muestran ambos ejemplos:

1. Primera forma:

```
1 .oculto{
2     background-color: blueviolet;
3     width: 100%;
4     text-align: center;
5     padding-top: 5%;
6     padding-bottom: 5%;
7     font-size: xx-large;
8 }
9 /*
10
11 Media Query. Pantallas menores a 990 px
12 no mostrarán el elemento "oculto"
13 */
14 @media (max-width:992px) {
15     .oculto{
16         display: none;
17     }
18 }
```

2. Segunda forma:

```
1 .oculto{
2     background-color: blueviolet;
3     width: 100%;
4     text-align: center;
5     padding-top: 5%;
6     padding-bottom: 5%;
```

```

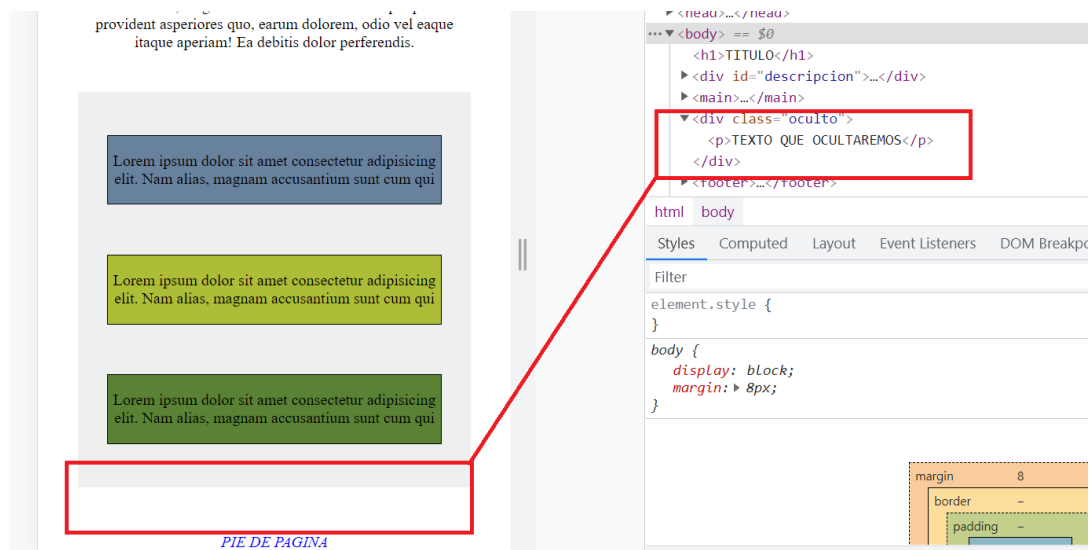
7   font-size: xx-large;
8   display: none;
9 }
10 /*
11
12 Media Query. Pantallas mayores a 990 px
13 mostrarán el elemento "oculto"
14 */
15 @media (min-width:992px) {
16   .oculto{
17     display: block;
18   }
19 }

```

Con ambas indicaciones obtenemos el mismo resultado, consiguiendo que en pantallas pequeñas el elemento quede oculto y en pantallas grandes el elemento se muestre.

Esto es muy importante para poder realizar nuestros diseños “minimalistas” en pantallas de *smartphone* y al mismo tiempo, poder mostrar más información en pantallas más grandes que cuentan con más espacio.

Es importante observar que a pesar de encontrarse oculto el elemento a la vista del usuario, si revisamos el inspector de elementos, podemos verlo dentro del código.



provident asperiores quo, earum dolore, odio vel eaque itaque aperiam! Ea debitis dolor perferendis.

Lorem ipsum dolor sit amet consectetur adipisicing elit. Nam alias, magnam accusantium sunt cum qui

Lorem ipsum dolor sit amet consectetur adipisicing elit. Nam alias, magnam accusantium sunt cum qui

Lorem ipsum dolor sit amet consectetur adipisicing elit. Nam alias, magnam accusantium sunt cum qui

PIE DE PAGINA

```

<neaa>...</neaa>
...<body> == $0
  <h1>TITULO</h1>
  <div id="descripcion">...</div>
  <main>...</main>
  <div class="oculto">
    <p>TEXT QUE OCULTAREMOS</p>
  </div>
  <footer>...</footer>

```

html body

Styles Computed Layout Event Listeners DOM Breakpo

Filter

```

element.style {
}

body {
  display: block;
  margin: 8px;
}

```

margin 8

border -

padding -

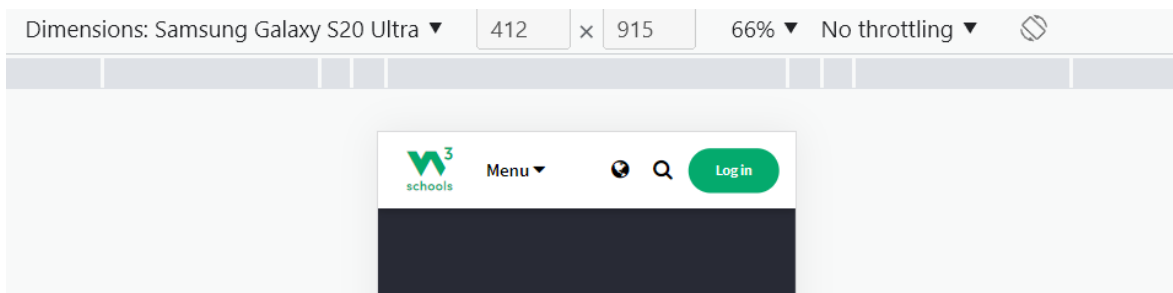
PROBAR DISPOSITIVOS

Cuando ingresamos al inspector de elementos, podemos utilizar la herramienta de “dimensiones”. Esto nos permite observar como se visualiza nuestra página web en distintos tamaños de pantalla.

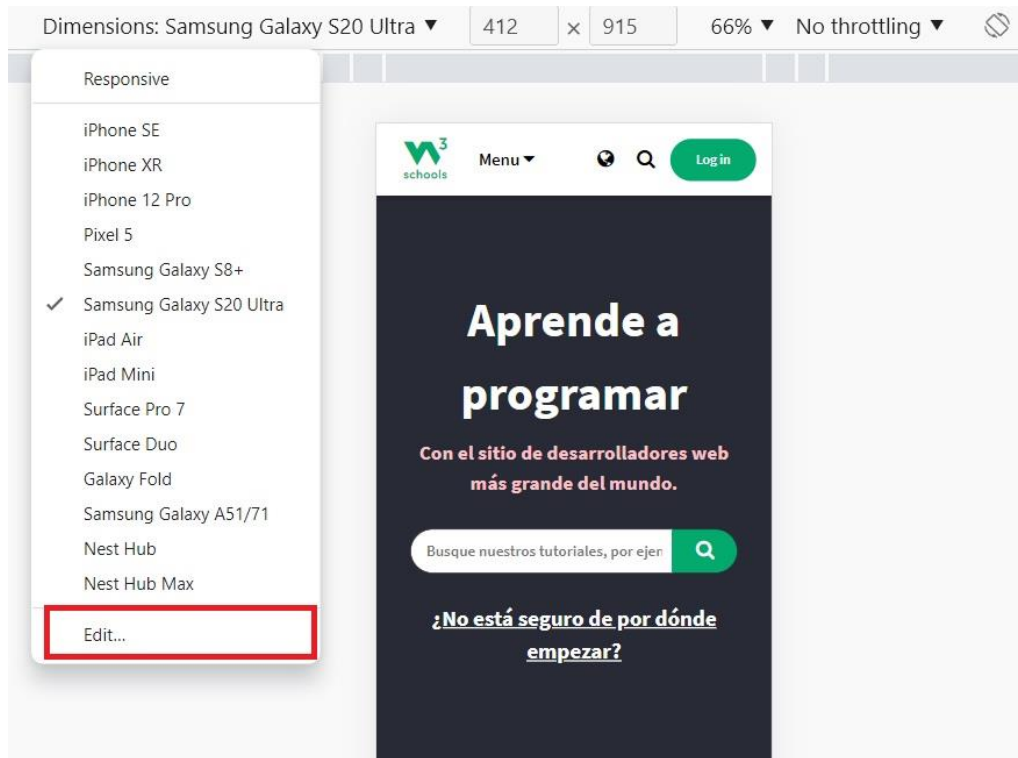
Existe la posibilidad de utilizar la herramienta en la opción “responsive” o responsiva, que nos da la posibilidad de mover el tamaño de pantalla manualmente para visualizarla.



Pero no solo tenemos esa posibilidad ya que, al mismo tiempo, esta herramienta nos permite elegir un tamaño específico de teléfono celular o Tablet y, de esta forma, ver como se visualiza la página web en estos dispositivos. Por ejemplo, podemos elegir evaluar nuestra página en un Samsung Galaxy S20 Ultra.

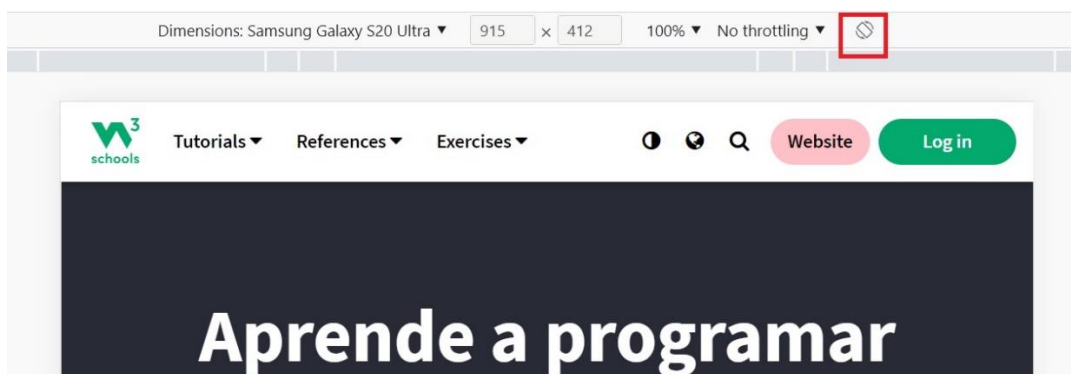


O cualquier dispositivo de la lista:



Además, si presionamos “editar” podemos cambiar las opciones que aparecen en la lista y seleccionar los dispositivos de nuestra preferencia.

Este no es el único elemento a nuestro favor para visualizar la responsividad de nuestra página, ya que presionamos el icono de móvil girando, podremos ver nuestro sitio en el dispositivo que seleccionemos, pero en versión horizontal.



Podemos corroborar que se encuentra en horizontal o en vertical viendo el tamaño en pixeles que nos aparece siempre en la pantalla, en este caso 915 x 412 px.