



UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

**Centro de
e-Learning**

Diplomatura en programación web full stack con React JS

Centro de e-Learning SCEU UTN - BA.

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

www.sceu.frba.utn.edu.ar/e-learning



UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

**Centro de
e-Learning**

p. 2

Módulo 1: HTML y CSS

Unidad 4: CSS Intermedio

Centro de e-Learning SCEU UTN - BA.

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

www.sceu.frba.utn.edu.ar/e-learning



Presentación:

En esta unidad vamos a adentrarnos en uno de los tipos de posicionamiento más utilizados y sencillos, además vamos a lograr que nuestros diseños se adapten a las dimensiones del dispositivo en el cual se muestran. Con todo esto, vamos a dar por concluído nuestro paso por HTML y CSS para seguir avanzando luego hacia la fase de programación!



Objetivos:

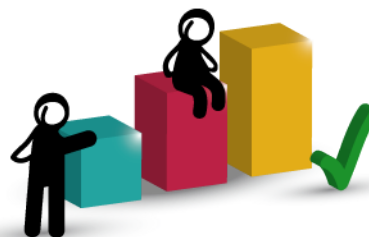
Que los participantes:

- Sepan realizar el posicionamiento por la tecnología de flexbox.
- Puedan realizar diseños adaptativos mediante el uso de media queries.



Bloques temáticos:

1. Uso avanzado de selectores
2. Introducción a Flexbox
3. Posicionamiento de elementos con flexbox
4. CSS media queries
5. Ejemplo de implementación
6. Trabajo Práctico



Consignas para el aprendizaje colaborativo

En esta Unidad los participantes se encontrarán con diferentes tipos de actividades que, en el marco de los fundamentos del MEC*, los referenciarán a tres comunidades de aprendizaje, que pondremos en funcionamiento en esta instancia de formación, a los efectos de aprovecharlas pedagógicamente:

- Los foros proactivos asociados a cada una de las unidades.
- La Web 2.0.
- Los contextos de desempeño de los participantes.

Es importante que todos los participantes realicen algunas de las actividades sugeridas y compartan en los foros los resultados obtenidos.

Además, también se propondrán reflexiones, notas especiales y vinculaciones a bibliografía y sitios web.

El carácter constructivista y colaborativo del MEC nos exige que todas las actividades realizadas por los participantes sean compartidas en los foros.

** El MEC es el modelo de E-learning colaborativo de nuestro Centro.*



Tomen nota:

Las actividades son opcionales y pueden realizarse en forma individual, pero siempre es deseable que se las realice en equipo, con la finalidad de estimular y favorecer el trabajo colaborativo y el aprendizaje entre pares. Tenga en cuenta que, si bien las actividades son opcionales, su realización es de vital importancia para el logro de los objetivos de aprendizaje de esta instancia de formación. Si su tiempo no le permite realizar todas las actividades, por lo menos realice alguna, es fundamental que lo haga. Si cada uno de los participantes realiza alguna, el foro, que es una instancia clave en este tipo de cursos, tendrá una actividad muy enriquecedora.

Asimismo, también tengan en cuenta cuando trabajen en la Web, que en ella hay de todo, cosas excelentes, muy buenas, buenas, regulares, malas y muy malas. Por eso, es necesario aplicar filtros críticos para que las investigaciones y búsquedas se encaminen a la excelencia. Si tienen dudas con alguno de los datos recolectados, no dejen de consultar al profesor-tutor. También aprovechen en el foro proactivo las opiniones de sus compañeros de curso y colegas.



1. Uso avanzado de selectores

Antes de seguir avanzando, vamos a mencionar algunas etiquetas que han surgido con HTML5.

Etiquetas semánticas

HTML5 incluye algunas etiquetas denominadas “etiquetas semánticas”, el nombre hace referencia a que las mismas son un intento por lograr que el código html sea más fácil de leer al describir claramente el propósito del bloque y/o elemento y/o sección. Su funcionamiento es el mismo que las etiquetas convencionales vistas en la Unidad 1.

La lista de los elementos semánticos incluidos en HTML5 es la siguiente:

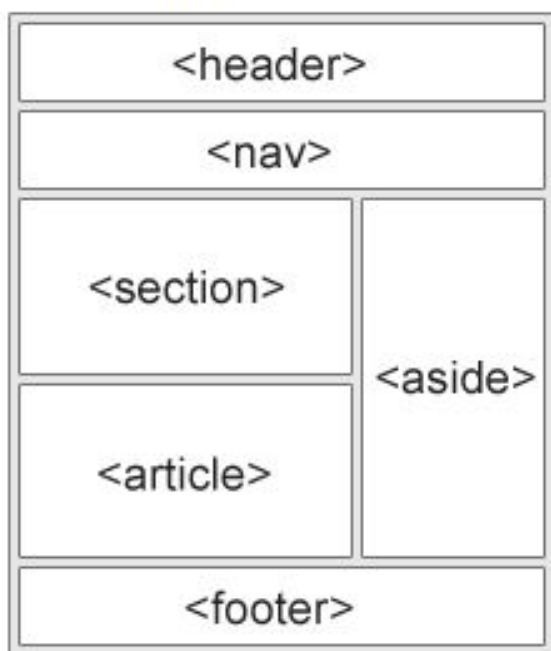
<article> <aside> <details> <figcaption> <figure> <footer> <header>	<main> <mark> <nav> <section> <summary> <time>
---	---

Los elementos marcados (en negrita) tienen un funcionamiento similar a la etiqueta <div> ya que agrupan otros elementos.



Ejemplo de uso

An example of semantic element layout by w3schools



Contraste entre el uso y no uso de etiquetas semánticas

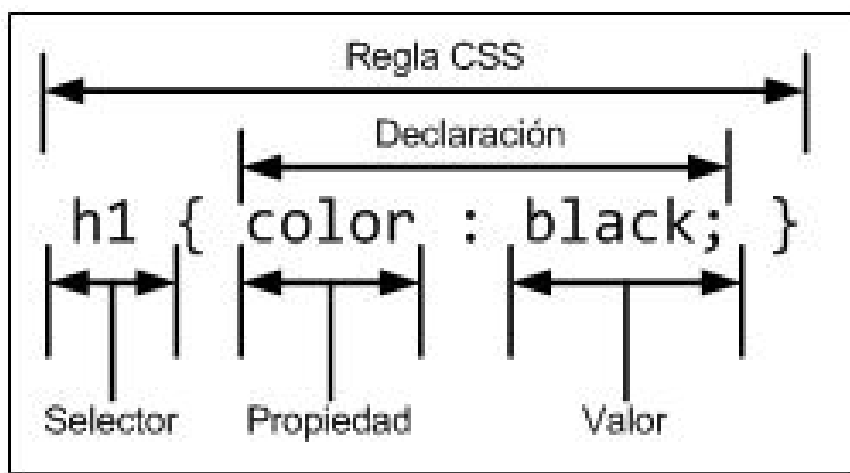


```
<header></header>
<section>
  <article>
    <figure>
      <img>
      <figcaption></figcaption>
    </figure>
  </article>
</section>
<footer></footer>
```

```
<div id="header"></div>
<div class="section">
  <div class="article">
    <div class="figure">
      <img>
      <div class="figcaption"></div>
    </div>
  </div>
</div>
<div id="footer"></div>
```

Uso avanzado de selectores

Cuando hablamos de selectores nos referimos a las etiquetas, las clases y los ids que usamos en CSS con el objetivo de aplicar diseño a los diferentes elementos del documento html que estamos trabajando.



A cada selector es posible aplicarle una serie de declaraciones para lograr el efecto visual que deseamos.

Hasta ahora, el efecto visual se logra apenas se carga la página, es decir que todo el código css que incluimos, afecta a apariencia de cada elemento desde el momento mismo en que la página es cargada por el navegador.

CSS nos permite también definir estilos diferentes en función de algunas acciones que el usuario ejecute, por ejemplo, cuando pasa el mouse por encima de un elemento. Esta funcionalidad lleva el nombre de **pseudo-clases**

Una pseudo-clase define un estilo a aplicarse al elemento, sólo cuando el usuario realiza determinada acción.



Sintaxis:

selector:pseudo-class {propiedad:valor;}

Ejemplo:

div:hover {background-color: red;}

Este ejemplo muestra el estilo aplicado a todos los div del documento cuando el usuario pasa el mouse sobre el elemento. El estilo que aplicará será el de cambiar el color de fondo por rojo.

Ejemplo

Vamos a retomar el ejemplo de la unidad anterior. El html que vamos a utilizar es el siguiente

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <meta http-equiv="X-UA-Compatible" content="ie=edge">
7      <title>New Sorbetes</title>
8      <link rel="stylesheet" href="style.css">
9  </head>
10 <body>
11     <div class="caja_zapato">Sandalias</div>
12     <div class="caja_zapato">Zapatillas</div>
13     <div class="caja_zapato">Botas</div>
14 </body>
15 </html>
```



Y el CSS queda así

```
1  body {
2      margin: 100px;
3  }
4
5  .caja_zapato {
6
7      margin: 50px 80px 60px 70px;
8
9      padding: 25px;
10
11     border: 5px solid red;
12
13     display: inline;
14 }
15
16 .caja_zapato:hover {
17     background-color: red;
18 }
```

Como se ve al final del CSS, se agrega la pseudo-clase :hover para que cambie el background del elemento cuando el usuario pasa el mouse sobre el elemento. El efecto final es:



Estado normal



Estado :hover



Algunos ejemplos más sobre otros pseudo-clases

Pseudo-clases

:first-child

Selecciona el primer elemento hijo de un elemento.

Se seleccionará el primer elemento `` que sea hijo de un elemento y que se encuentre dentro del elemento `<p>`. Así se verá en rojo sólo el primer ``

Ejemplo:

```
p em:first-child {  
  color: red; }
```

```
<p>Lorem  
  <span><em>ipsum dolor</em></span>  
  sit amet, consectetur adipiscing elit.  
  Praesent odio sem, tempor quis,  
  <em>auctor eu</em>  
  , tempus at, enim. Praesent nulla ante,  
  <em>ultrices</em>  
  id, porttitor ut, pulvinar quis, dui.  
</p>
```

Centro de e-Learning SCEU UTN - BA.

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

www.sceu.frba.utn.edu.ar/e-learning



Pseudo-clases

:link y :visited

Aplicar diferentes estilos a los enlaces de una misma página.

:link enlaces que todavía no fueron visitados por el usuario

:visited enlaces que ya fueron visitados por el usuario

Son estados mutuamente excluyentes.

Estas propiedades dependen de que se haya borrado o no la lista de enlaces visitados.

Ejemplo

```
a: link {  
    color: red; }
```

```
a: visited {  
    color: green; }
```

Pseudo-clases

:hover :active y :focus

Para variar los estilos de un elemento en respuesta a las acciones del usuario.

Se pueden aplicar a cualquier elemento.

:hover Se activa cuando el usuario pasa el ratón o cualquier otro elemento apuntador por encima de un elemento.

Ejemplo

```
div:hover {  
    background-color: red;}
```



Pseudo-clases

:hover :active y :focus

:active Se activa cuando el usuario activa un elemento, por ejemplo cuando pulsa con el mouse sobre un elemento. El estilo se aplica durante un espacio de tiempo prácticamente imperceptible, ya que sólo dura desde que el usuario pulsa el botón del mouse hasta que lo suelta.

Ejemplo

```
a:active {  
    background-color: yellow;}
```

Pseudo-clases

:hover :active y :focus

:focus Se activa cuando el elemento está seleccionado.

Ejemplo

```
a:focus {  
    background-color: red;}
```




Pseudo-clases

:nth-of-type(patrón-para-los-elementos-coincidentes)

Selecciona uno o más elementos de un tipo dado, en función de su posición entre un grupo de hermanos.

Ejemplo

```
/* Párrafos impares */
```

```
p:nth-of-type(2n+1) {  
  color: red;  
}
```

```
/* Párrafos pares */
```

```
p:nth-of-type(2n) {  
  color: blue;  
}
```

Pseudo-clases

:active

:any

:checked

:default

:dir()

:disabled

:empty

:enabled

:first

:first-child

:first-of-type

:fullscreen

:focus

:hover

:indeterminate

:in-range

:invalid

:lang()

:last-child

:last-of-type

:left

:link

:not()

:nth-child()

:nth-last-child()

:nth-last-of-type()

:nth-of-type()

:only-child

:only-of-type

:optional

:out-of-range

:read-only

:read-write

:required

:right

:root

:scope

:target

:valid

:visited

2. Introducción a Flexbox

Flexbox es un método para **distribuir el espacio entre ítems** de una interfaz y mejorar las capacidad de **alineación**. Es un método orientado a dispositivos móviles.

Resumidamente, flexbox nos va a ayudar a posicionar cada elemento dentro del documento html que estemos haciendo.

Algo importante a tener en cuenta es que flexbox maneja una sola dimensión de posicionamiento a la vez lo que quiere decir que vamos a tener que elegir si queremos posicionarlo respecto de las filas (row) o las columnas (column)



Contenedor flex

Un área del documento que contiene un flexbox se denomina contenedor flex.

display: flex / inline-flex; => todos los elementos internos se convierten en items flex

Centro de e-Learning SCEU UTN - BA.

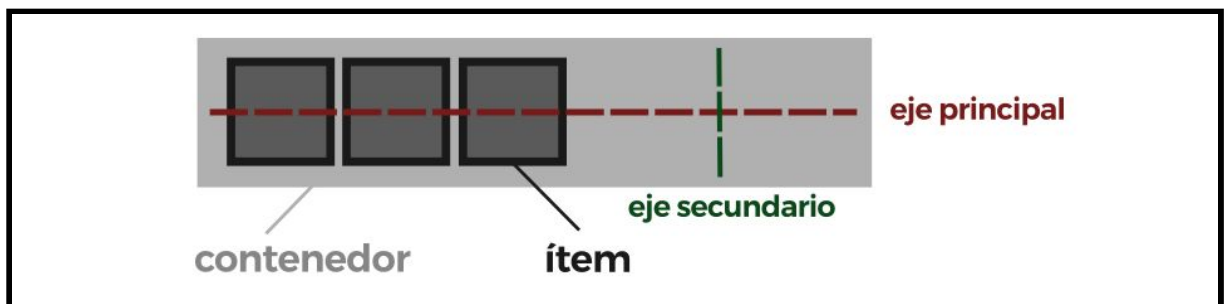
Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

www.sceu.frba.utn.edu.ar/e-learning



Valores predeterminados para los ítems flex:

- Los ítems se despliegan sobre una fila (la propiedad flex-direction por defecto es row).
- Los ítems empiezan desde el margen inicial sobre el eje principal.
- Los ítems no se ajustan en la dimensión principal, pero se pueden contraer.
- Los ítems se ajustarán para llenar el tamaño del eje cruzado.
- La propiedad flex-basis es definida como auto.
- La propiedad flex-wrap es definida como nowrap



flex-direction

Eje principal:

flex-direction: row/ row-reverse, column, column-reverse

Eje secundario es perpendicular al eje principal.

Si flex-direction es row / row-reverse => eje secundario corre por columnas

Si flex-direction es column / column-reverse => eje secundario corre por filas



3. Posicionamiento de elementos con flexbox

Vamos a trabajar con 4 declaraciones css para posicionar cada elemento y son:

- **display:** flex;
- **flex-direction:** row | column;
- **justify-content:** flex-start | flex-end | center | space-between | space-around;
- **align-items:** flex-start | flex-end | center | space-between | space-around | stretch;

display:flex;

display: flex es aplicado al contenedor para que el posicionamiento se aplique sobre los hijos.

El ancho de cada elemento se puede aplicar como
width o flex: <número>

Si tenemos 3 elementos con **flex:1**, los 3 tendrán el mismo ancho. Si uno tiene **flex:2** y los demás **flex:1**, el primero será más ancho que los demás

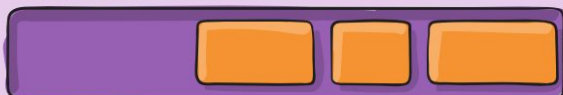


justify-content

flex-start



flex-end



center



space-between



space-around



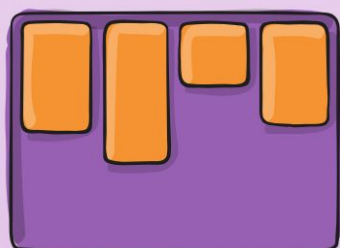
space-evenly



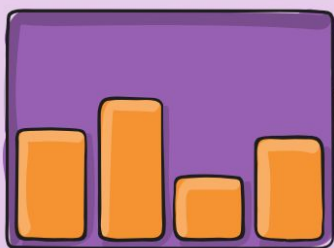


align-items

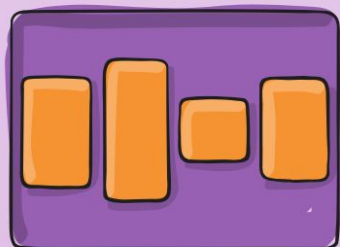
flex-start



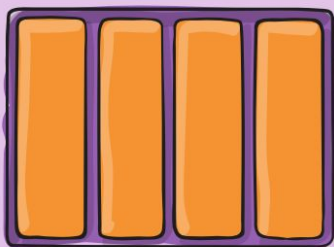
flex-end



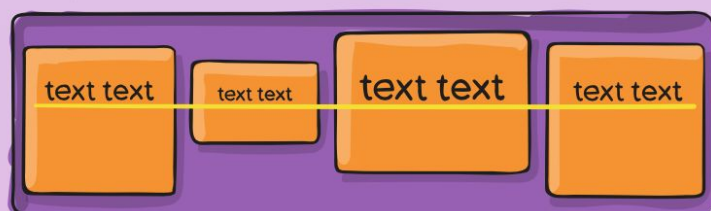
center



stretch



baseline



Ejemplo

Y seguimos con nuestro ejemplo de la unidad anterior para ver cómo trabaja flexbox para posicionar elementos.

Centro de e-Learning SCEU UTN - BA.

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

www.sceu.frba.utn.edu.ar/e-learning



```
otros > index.html > html > body
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <meta http-equiv="X-UA-Compatible" content="ie=edge">
7      <title>New Sorbetes</title>
8      <link rel="stylesheet" href="style.css">
9  </head>
10 <body>
11     <main>
12         <div class="caja_zapato">Sandalias</div>
13         <div class="caja_zapato">Zapatillas</div>
14         <div class="caja_zapato">Botas</div>
15     </main>
16 </body>
17 </html>
```

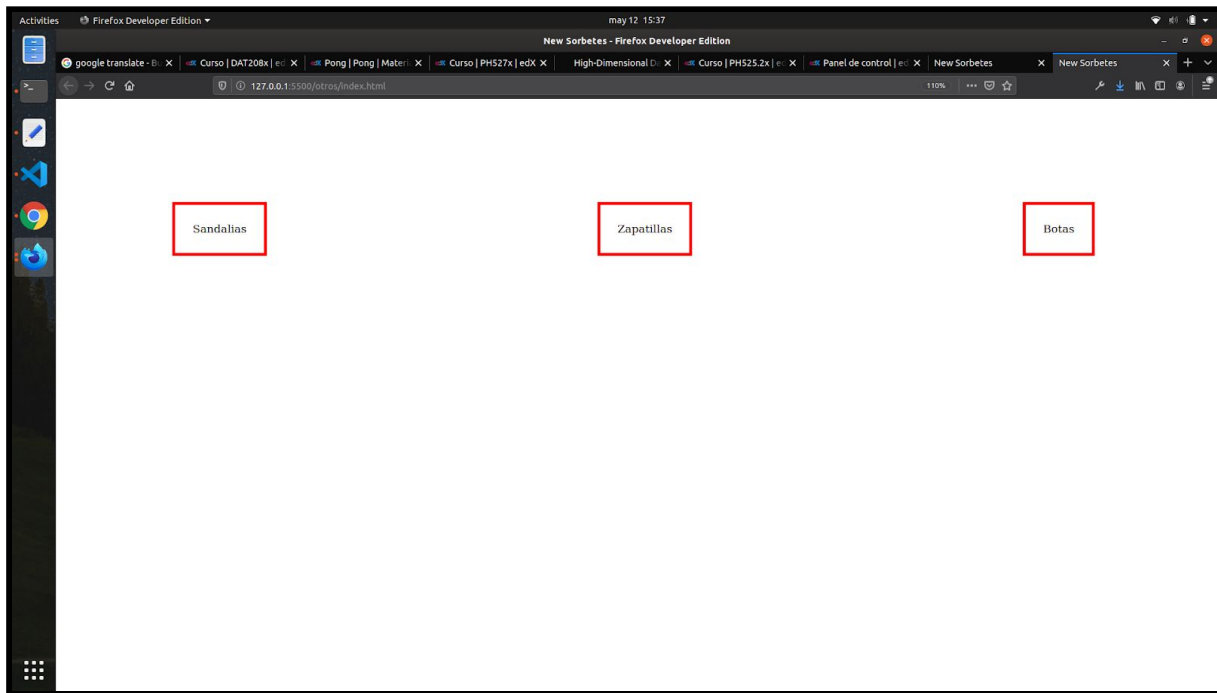
Agregamos un contenedor para todas las cajas de zapatos. Aprovechando que al inicio de la unidad vimos las etiquetas semánticas, hacemos uso de `<main>` como contenedor de las cajas.



```
1  body {
2    margin: 100px;
3  }
4
5  .caja_zapato {
6
7    margin: 50px 80px 60px 70px;
8
9    padding: 25px;
10
11   border: 5px solid red;
12
13 }
14
15 .caja_zapato:hover {
16   background-color: red;
17 }
18
19 main {
20   display: flex;
21   flex-direction: row;
22   justify-content: space-between;
23   align-items: center;
24 }
```

En este caso estamos indicando que los elementos dentro de la caja main deben estar equidistantes unos de otros, utilizando el espacio sobrante para separarlo de forma pareja (`justify-content: space-between;`)

Y el resultado final es:





4. CSS media queries

Antes de hablar de media queries debemos hablar sobre el diseño responsivo o sensible. Se trata de técnicas que permiten hacer webs adaptables a las condiciones del dispositivo que las accede. Son técnicas adaptativas enfocadas en las dimensiones de la pantalla.

Responsive: característica de un sistema que tiene respuesta hacia el medio que le rodea.

Consejos de diseño

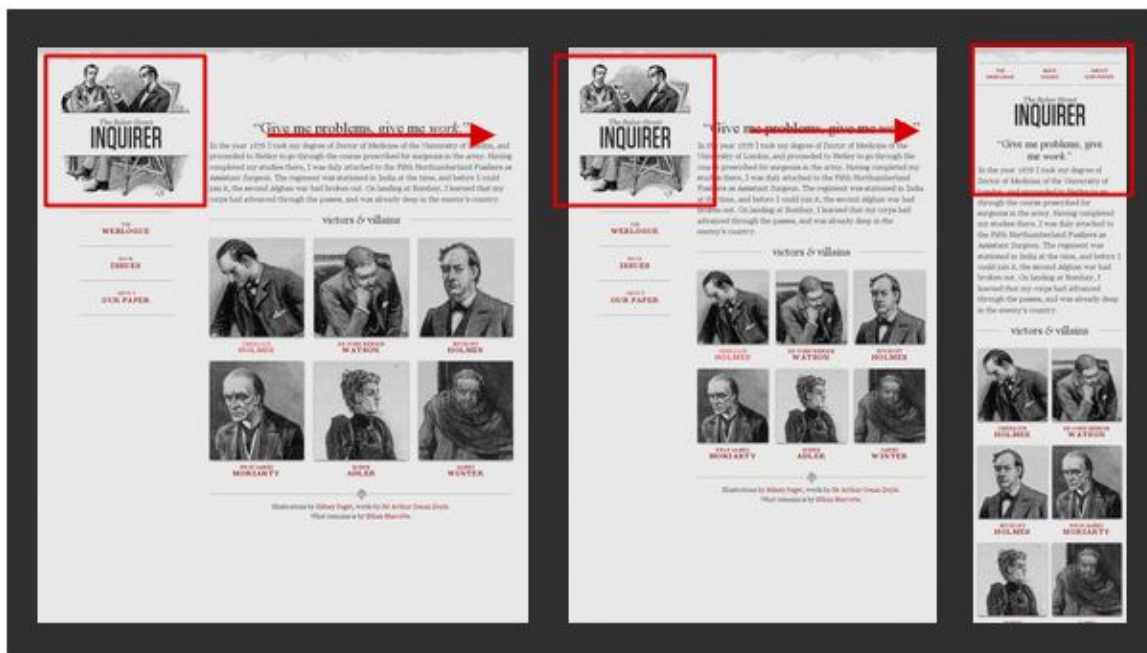
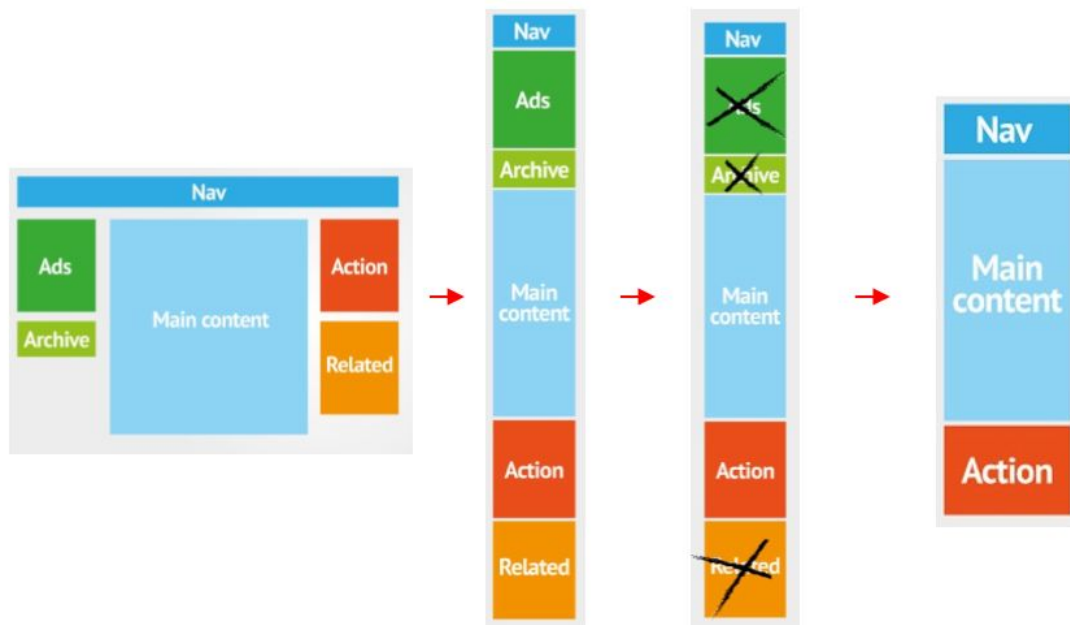
- No usar estilos in-line
- La web no debe verse igual en todos los dispositivos
- Usar unidades relativas

También hay que adaptarse a las características de la navegación en distintos dispositivos.

Las técnicas responsive se focalizan en diseñar para:

- ➔ Todos los navegadores y sistemas
- ➔ Todas las resoluciones de pantalla
- ➔ Todas las velocidades de conexión

La web no debe necesariamente verse igual en todos los tipos de dispositivos.





Flujo de desarrollo de un sitio responsive

1. Crear un HTML con el contenido que deseamos mostrar. Lo más semántico posible
2. Aplicar formato con CSS
 - a. Recursos de diseño estético: estilos que se agregan para mejorar la estética
 - b. Diseño de layout: jerarquizar la información y presentarla con una estructura que permita el mejor entendimiento.
 - a. Imágenes Flexibles
 - b. Sistema de grilla o cajas fluidas
 - c. Media queries

Siempre tener en cuenta que algunos estilos pueden no ser soportados por todos los navegadores o ser interpretados de forma diferente.

Media Queries

Media queries CSS3 son las herramientas para especificar estilos respecto a las condiciones del cliente web que está visitando el sitio.

Son las primeras construcciones CSS que nos permiten definir estilos **condicionales** (aplicables en determinadas situaciones)

Media queries = consultas sobre las características del medio donde se está visualizando la web.

Qué es un condicional?

"Si ocurre esto, hacé tal cosa"

Ejemplos:

- Si la pantalla del usuario tiene estas características, entonces aplica estos estilos

Centro de e-Learning SCEU UTN - BA.

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

www.sceu.frba.utn.edu.ar/e-learning



- Si se imprime el documento en la impresora, aplica estos estilos.
- Si la pantalla del dispositivo tiene estas dimensiones y además está situado en posición horizontal (landscape), entonces aplica este CSS.

Sintaxis

```
@media not|only mediatype and (caracteristica-media and|or|not caracteristica-media) {  
    CSS-Codigo;  
}
```

Alternativa 1: atributo LINK

```
<link rel="stylesheet" href="mis-estilos.css">
```

Se pueden especificar condiciones para que se apliquen los estilos enlazados.

Para especificar que sólo se deben aplicar los estilos si la página se muestra para impresión:

```
<link rel="stylesheet" href="mis-estilos.css" media="print">
```

Lo nuevo:

Para aplicar estilos sólo cuando la pantalla del usuario (o la ventana del navegador en caso de computadoras) tenga un ancho de 1200 px.

```
<link rel="stylesheet" href="mis-estilos.css" media="(min-width: 1200px)">
```



Alternativa 2: @media

Se incorporan los estilos en una construcción @media condicional.

```
@media (min-width: 500px) {  
  h1{  
    margin: 1%;  
  }  
  .estiloresponsive{  
    float: right;  
    padding-left: 15px;  
  }  
}
```

Operadores lógicos

Los disponibles son:

- and
- not
- only
- or

Ejemplos

```
@media (max-width: 600px) and (orientation: landscape) {  
  h1{  
    color: red;  
  }  
}
```



Los smartphones suelen simular tamaños de pantalla más grandes por lo que será necesario incluir el “viewport” en el HTML

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

```
@media tv and (min-width: 1200px){  
  h1{  
    margin: 10%;  
  }  
}
```

Aplica la regla en dispositivos de tipo televisión y cuya resolución mínima de anchura sea de 1200 píxeles.

Sirve para pantallas de 600px mínimo, también para todos los dispositivos handheld (de mano) que estén en posición vertical.

```
@media (min-width: 600px), handheld and (orientation: portrait) {  
  h1{  
    color: green;  
  }  
}
```

Orden de colocación de los media queries

Es importante por:

Centro de e-Learning SCEU UTN - BA.

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

www.sceu.frba.utn.edu.ar/e-learning



1. La cascada
2. Mobile First

Orden habitual para MQ (media queries):

1. Los estilos globales se colocan fuera de cualquier MQ. Por regla Mobile First, estas serán las que se apliquen a los móviles.
2. Colocar los MQ necesarios para los siguientes dispositivos de menor anchura.
3. Ir colocando los MQ en orden de anchura (menor a mayor)

Rangos de media queries

Usar min-width y max-width a la vez

Ejemplo

```
@media (min-width: 700px) and (max-width: 800px){  
  .lateral{  
    width: 33%;  
    float: right;  
    background-color: #6ee;  
  }  
}
```

Breakpoints - Ejemplo



```
/* Extra small devices (phones, 600px and down) */
@media only screen and (max-width: 600px) {...}

/* Small devices (portrait tablets and large phones, 600px and up) */
@media only screen and (min-width: 600px) {...}

/* Medium devices (landscape tablets, 768px and up) */
@media only screen and (min-width: 768px) {...}

/* Large devices (laptops/desktops, 992px and up) */
@media only screen and (min-width: 992px) {...}

/* Extra large devices (large laptops and desktops, 1200px and up) */
@media only screen and (min-width: 1200px) {...}
```

```
@media only screen and (orientation: landscape) {
  body {
    background-color: lightblue;
  }
}
```

Breakpoints - OCULTAR ELEMENTO

```
/* If the screen size is 600px wide or less, hide the element */
@media only screen and (max-width: 600px) {
  div.example {
    display: none;
  }
}
```

Breakpoints - CAMBIAR EL TAMAÑO DE LA FUENTE



```
/* If the screen size is 601px or more, set the font-size of <div> to 80px */
@media only screen and (min-width: 601px) {
  div.example {
    font-size: 80px;
  }
}

/* If the screen size is 600px or less, set the font-size of <div> to 30px */
@media only screen and (max-width: 600px) {
  div.example {
    font-size: 30px;
  }
}
```

Tipos

all	Todos los dispositivos de tipo media
print	Impresora
screen	Pantalla
speech	Se utiliza para los lectores de pantalla que "leen" en voz alta

Características

any-hover	¿Hay algún mecanismo de entrada disponible para que el usuario se pueda desplazar sobre los elementos?(Agregado en Media Queries Nivel 4)
any-pointer	¿Hay algún mecanismo de entrada disponible que sea un dispositivo señalador?, ¿qué tan preciso es? (Agregado en Media Queries Nivel 4)
aspect-ratio	La relación entre el ancho y la altura de la ventana gráfica.

Centro de e-Learning SCEU UTN - BA.

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

www.sceu.frba.utn.edu.ar/e-learning



color	El número de bits por componente de color para el dispositivo de salida
color-gamut	El rango aproximado de colores que son compatibles con el agente de usuario y el dispositivo de salida (agregado en el Nivel 4 de Consultas de Medios)
color-index	Número de colores que puede mostrar el dispositivo
grid	Si el dispositivo es una cuadrícula o mapa de bits
height	Altura del viewport

any-hover	¿Hay algún mecanismo de entrada disponible para que el usuario se pueda desplazar sobre los elementos?(Agregado en Media Queries Nivel 4)
any-pointer	¿Hay algún mecanismo de entrada disponible que sea un dispositivo señalador?, ¿qué tan preciso es? (Agregado en Media Queries Nivel 4)
aspect-ratio	La relación entre el ancho y la altura de la ventana gráfica.
color	El número de bits por componente de color para el dispositivo de salida
color-gamut	El rango aproximado de colores que son compatibles con el agente de usuario y el dispositivo de salida (agregado en el Nivel 4 de Consultas de Medios)
color-index	Número de colores que puede mostrar el dispositivo
grid	Si el dispositivo es una cuadrícula o mapa de bits
height	Altura del viewport



any-hover	¿Hay algún mecanismo de entrada disponible para que el usuario se pueda desplazar sobre los elementos?(Agregado en Media Queries Nivel 4)
any-pointer	¿Hay algún mecanismo de entrada disponible que sea un dispositivo señalador?, ¿qué tan preciso es? (Agregado en Media Queries Nivel 4)
aspect-ratio	La relación entre el ancho y la altura de la ventana gráfica.
color	El número de bits por componente de color para el dispositivo de salida
color-gamut	El rango aproximado de colores que son compatibles con el agente de usuario y el dispositivo de salida (agregado en el Nivel 4 de Consultas de Medios)
color-index	Número de colores que puede mostrar el dispositivo
grid	Si el dispositivo es una cuadrícula o mapa de bits
height	Altura del viewport

any-hover	¿Hay algún mecanismo de entrada disponible para que el usuario se pueda desplazar sobre los elementos?(Agregado en Media Queries Nivel 4)
any-pointer	¿Hay algún mecanismo de entrada disponible que sea un dispositivo señalador?, ¿qué tan preciso es? (Agregado en Media Queries Nivel 4)
aspect-ratio	La relación entre el ancho y la altura de la ventana gráfica.
color	El número de bits por componente de color para el dispositivo de salida
color-gamut	El rango aproximado de colores que son compatibles con el agente de usuario y el dispositivo de salida (agregado en el Nivel 4 de Consultas de Medios)
color-index	Número de colores que puede mostrar el dispositivo
grid	Si el dispositivo es una cuadrícula o mapa de bits
height	Altura del viewport



any-hover	¿Hay algún mecanismo de entrada disponible para que el usuario se pueda desplazar sobre los elementos?(Agregado en Media Queries Nivel 4)
any-pointer	¿Hay algún mecanismo de entrada disponible que sea un dispositivo señalador?, ¿qué tan preciso es? (Agregado en Media Queries Nivel 4)
aspect-ratio	La relación entre el ancho y la altura de la ventana gráfica.
color	El número de bits por componente de color para el dispositivo de salida
color-gamut	El rango aproximado de colores que son compatibles con el agente de usuario y el dispositivo de salida (agregado en el Nivel 4 de Consultas de Medios)
color-index	Número de colores que puede mostrar el dispositivo
grid	Si el dispositivo es una cuadrícula o mapa de bits
height	Altura del viewport



5. Ejemplo de implementación

Siguiendo con nuestro ejemplo, vamos a implementar un cambio de color y posición de las cajas de zapatos cuando se vea en dispositivos con un ancho menor a 480px.

El html sigue igual

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <meta http-equiv="X-UA-Compatible" content="ie=edge">
7      <title>New Sorbetes</title>
8      <link rel="stylesheet" href="style.css">
9  </head>
10 <body>
11     <main>
12         <div class="caja_zapato">Sandalias</div>
13         <div class="caja_zapato">Zapatillas</div>
14         <div class="caja_zapato">Botas</div>
15     </main>
16 </body>
17 </html>
```

La parte de css para computadoras (ancho mayor a 480px) sigue igual también



```
1  body {
2    margin: 100px;
3  }
4
5  .caja_zapato {
6
7    margin: 50px 80px 60px 70px;
8
9    padding: 25px;
10
11    border: 5px solid red;
12
13  }
14
15  .caja_zapato:hover {
16    background-color: red;
17  }
18
19  main {
20    display: flex;
21    flex-direction: row;
22    justify-content: space-between;
23    align-items: center;
24  }
25
```

Pero le agregamos la parte de media queries



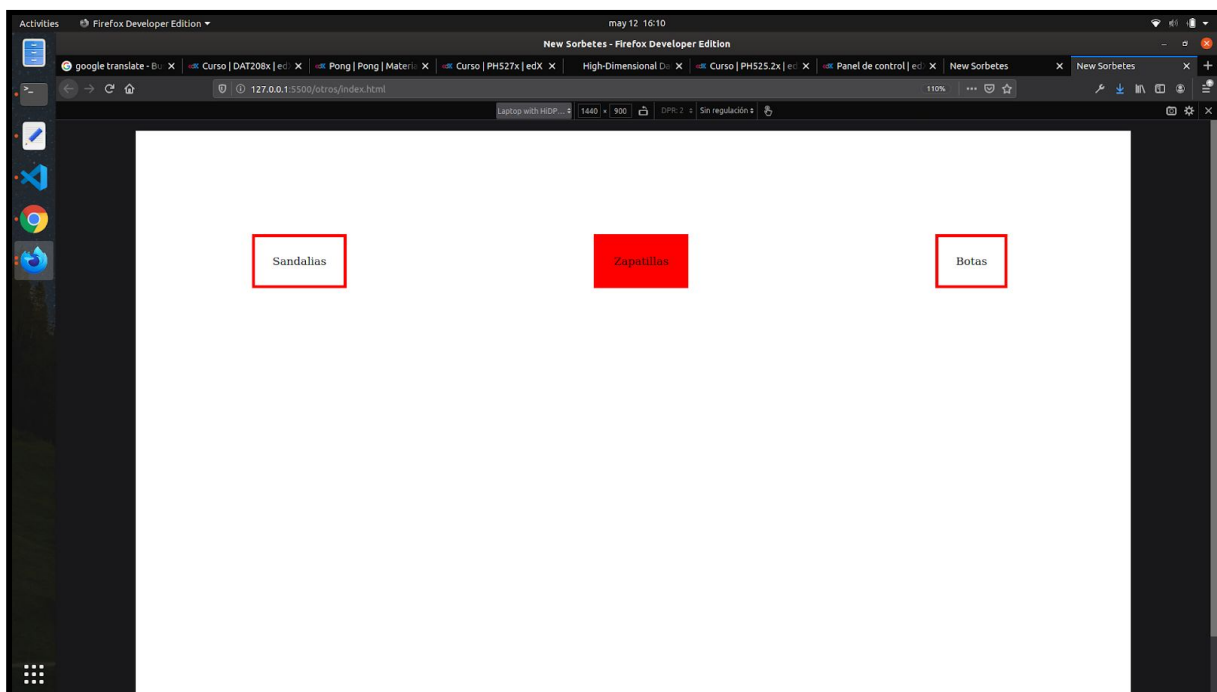
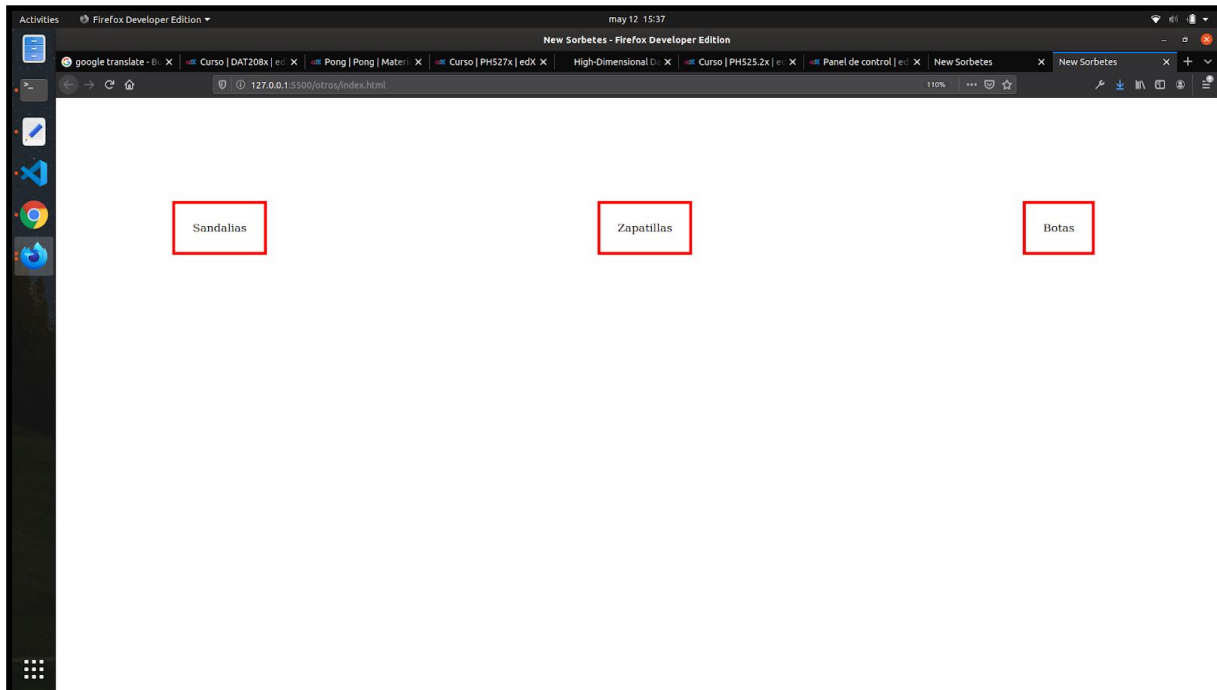
```
26
27 @media (max-width: 480px) {
28
29     main {
30         display: flex;
31         flex-direction: column;
32         justify-content: space-between;
33         align-items: center;
34     }
35
36
37     .caja_zapato {
38
39         margin: 50px 80px 60px 70px;
40
41         padding: 25px;
42
43         border: 5px solid blue;
44
45     }
46
47
48     .caja_zapato:hover {
49         background-color: blue;
50     }
51
52 }
```

El resultado en dispositivos grandes es el mismo que siempre

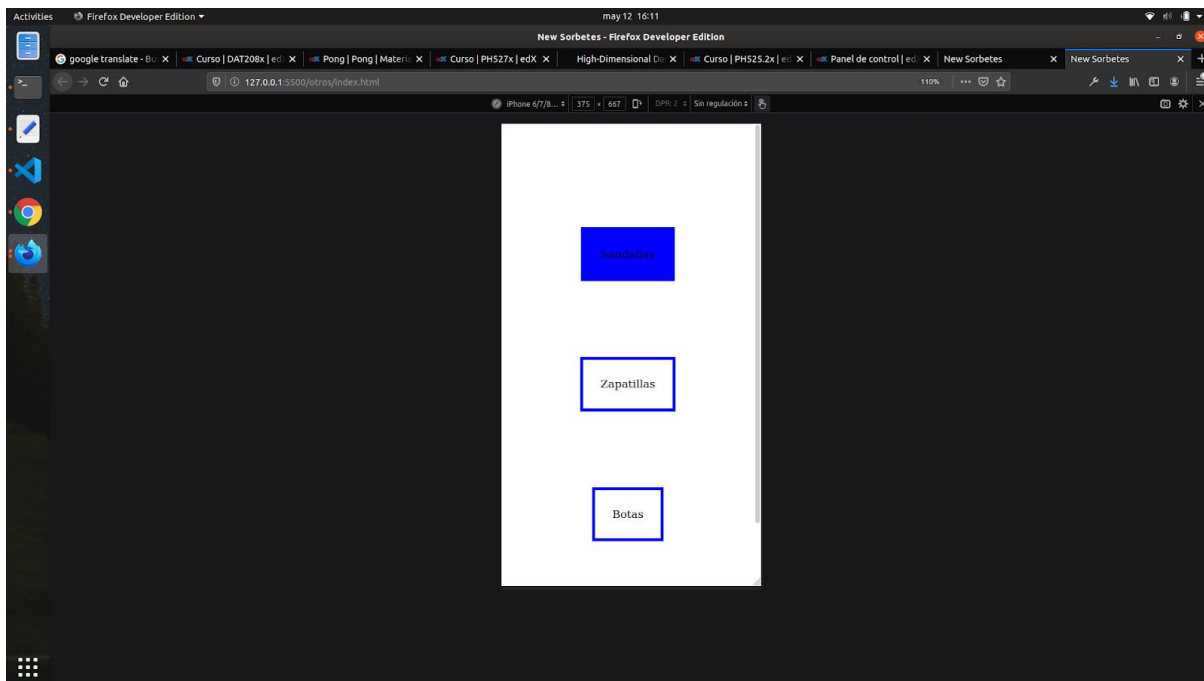
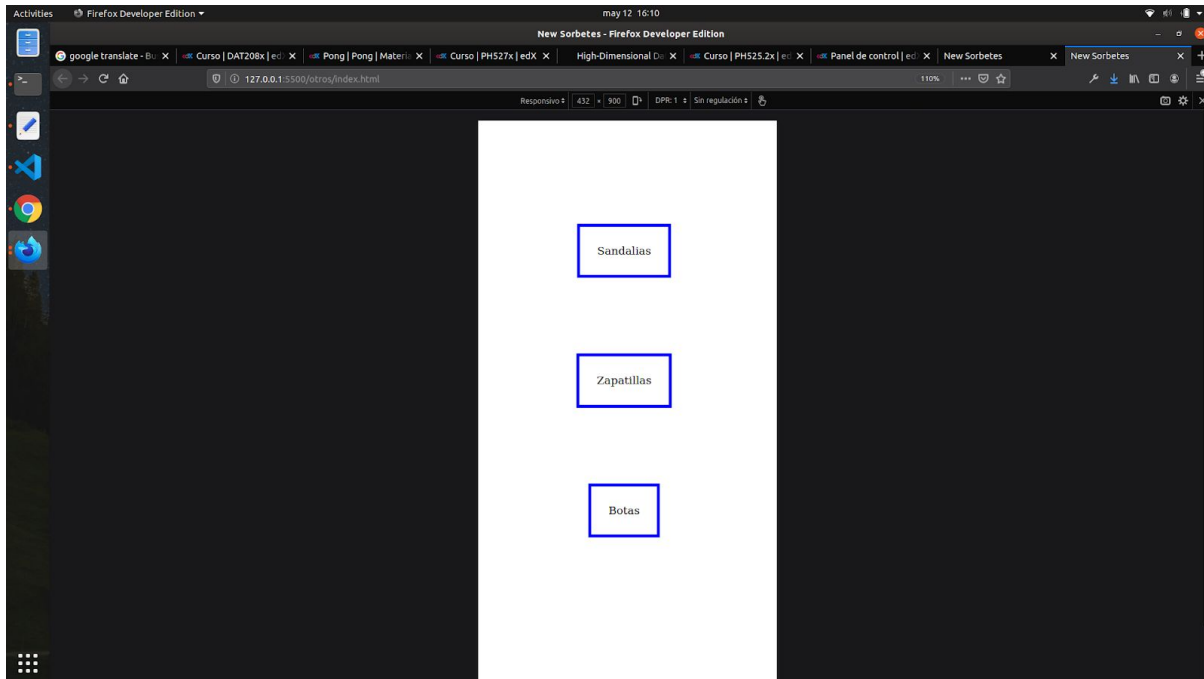
Centro de e-Learning SCEU UTN - BA.

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

www.sceu.frba.utn.edu.ar/e-learning



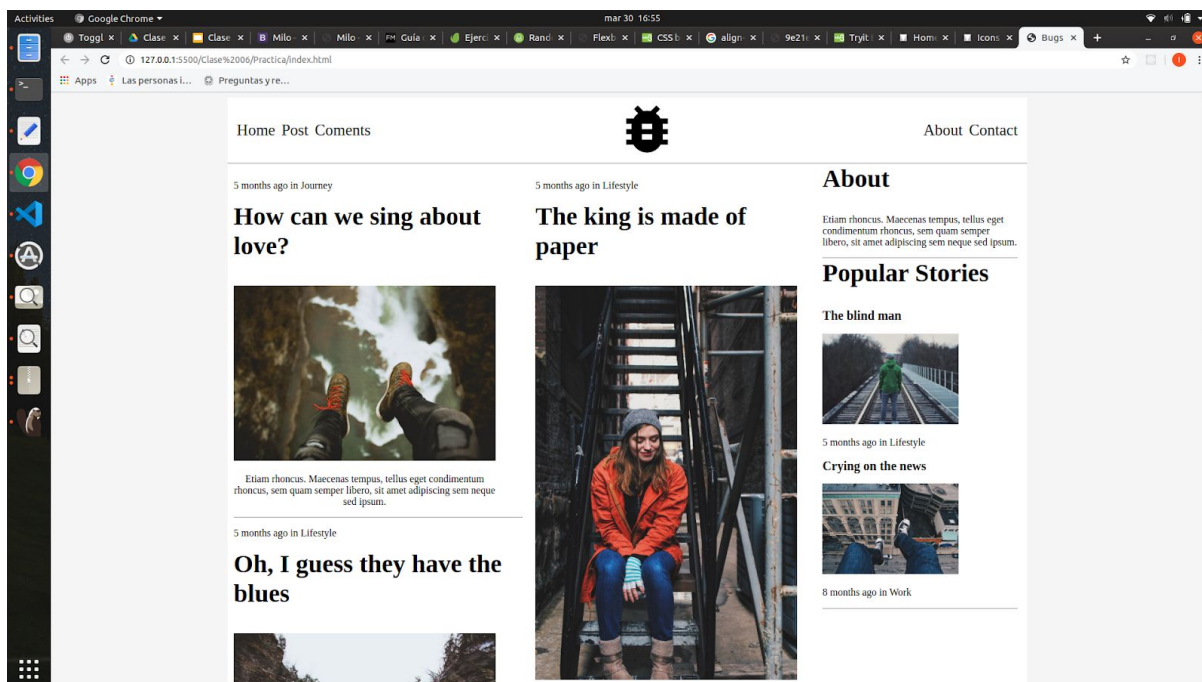
Pero cuando las dimensiones del dispositivo son inferiores a 480px

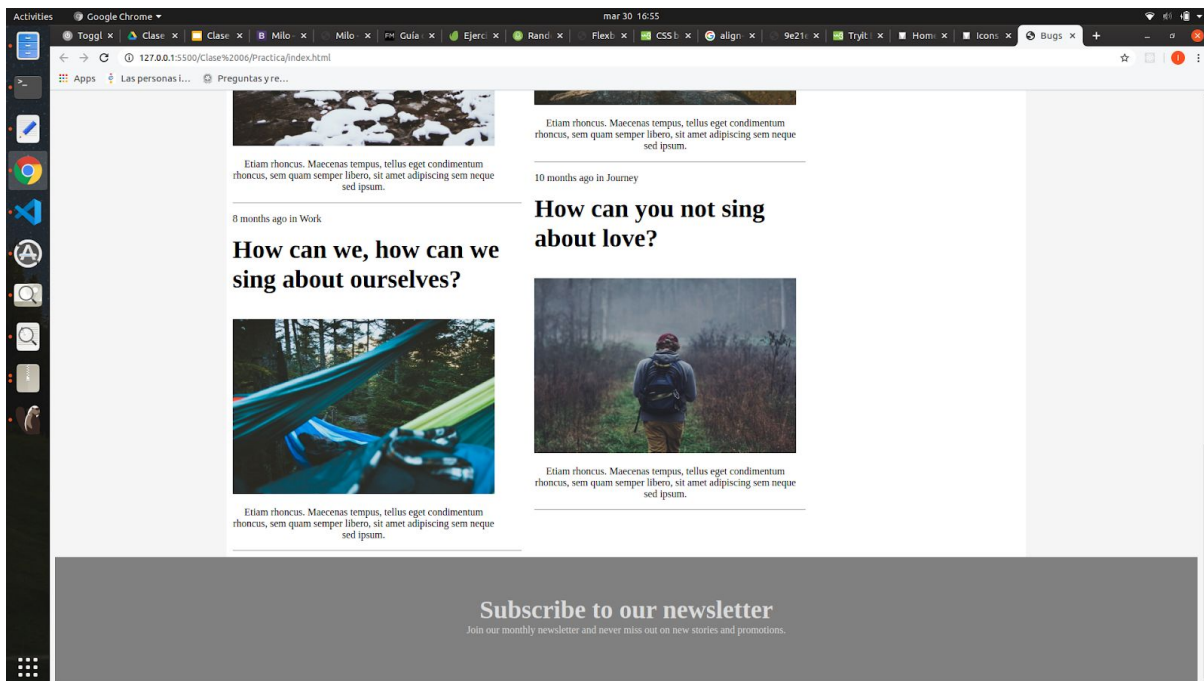
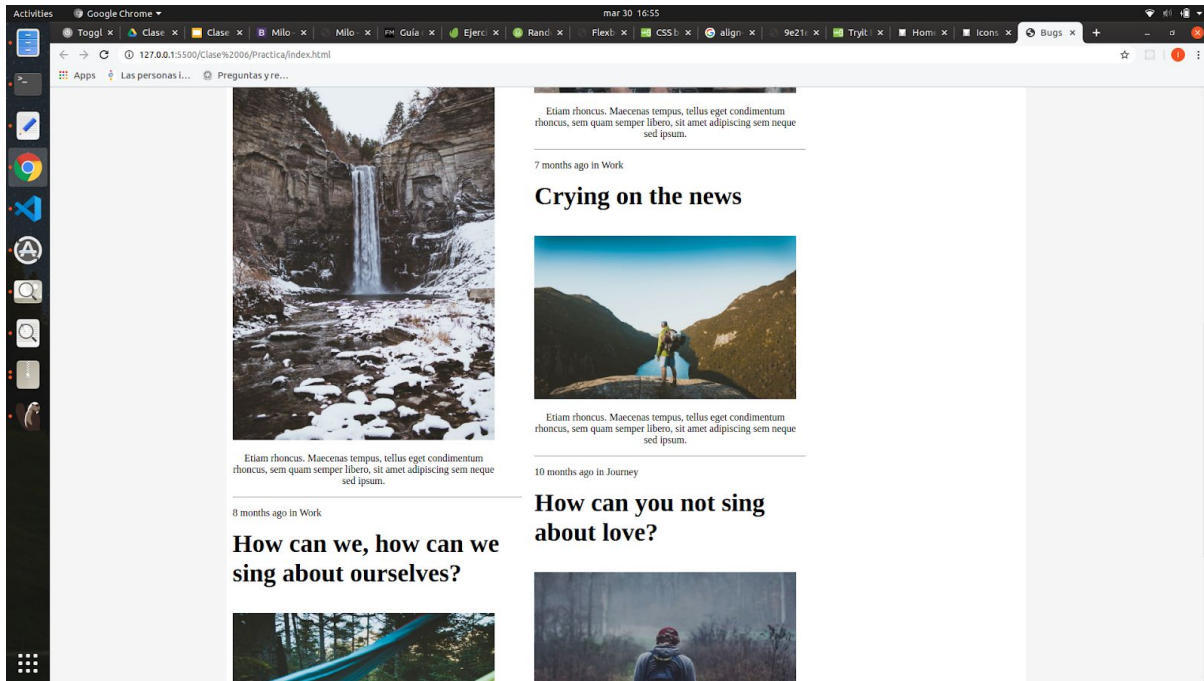




6. Trabajo Práctico

Desarrollar la siguiente página utilizando todo lo aprendido de etiquetas semánticas y posicionamiento flex





Las imágenes pueden conseguirlas en <https://unsplash.com/>

No es necesario que sean exactamente las mismas aunque han sido descargadas de ese sitio web.



Bibliografía utilizada y sugerida

MDN web docs CSS. (n.d.) Recuperado de:

<https://developer.mozilla.org/es/docs/Web/CSS>

MDN web docs HTML5. (n.d.) Recuperado de:

<https://developer.mozilla.org/es/docs/HTML/HTML5>

World Wide Web Consortium. (n.d.) Recuperado de <https://www.w3.org/>

w3schools.com. (n.d.) Recuperado de <https://www.w3schools.com/>



Lo que vimos:

- Posicionamiento flexbox
- Media queries



Lo que viene:

- Módulo 2: JavaScript

