



Alineamiento contenido con Flex

{desafío}
latam_



**Activen las cámaras los que puedan y
pasemos asistencia**

***Posicionar elementos en
maquetas web, usando las
reglas CSS necesarias para
realizar correctamente este
proceso.***

- Unidad 1: Introducción a HTML.
- Unidad 2: Introducción a CSS
- Unidad 3: Alineando contenido con flex
- Unidad 4: Bootstrap
- Unidad 5: Terminal, Git y Github
- Unidad 6: Trabajo colaborativo y Github Pages



Te encuentras aquí





Inicio

{desafío}
latam_



/* Utilizaremos flexbox para poner contenidos uno al lado de otro. */

- Crear una galería de imágenes.
- Crear un menú.
- Crear una sección de características.

Activación de conceptos

Contesta la pregunta correctamente y gana un punto

Instrucciones:

- Se realizará una pregunta, el primero en escribir “YO” por el chat, dará su respuesta al resto de la clase.
- El docente validará la respuesta.
- En caso de que no sea correcta, dará la oportunidad a la segunda persona que dijo “Yo”.
- Cada estudiante podrá participar un máximo de 2 veces.
- Al final, el/la docente indicará el 1º, 2º y 3º lugar.
- Esta actividad no es calificada, es solo una dinámica para recordar los conceptos clave para abordar esta sesión.





Activación de conceptos



¿Qué significa CSS?



Activación de conceptos



¿Qué atributo hay que utilizar para agregar CSS de forma Inline?



Activación de conceptos



¿Qué etiqueta hay que utilizar para agregar CSS a nivel de la página web?



Activación de conceptos



¿Qué etiqueta hay que utilizar para agregar CSS en un archivo externo?



Activación de conceptos



¿Cómo se debe escribir un selector para una etiqueta que tiene la clase xyz?



Activación de conceptos



Primer lugar:



Segundo lugar:



Tercer lugar:



Desarrollo

{desafío}
latam_



Flexbox

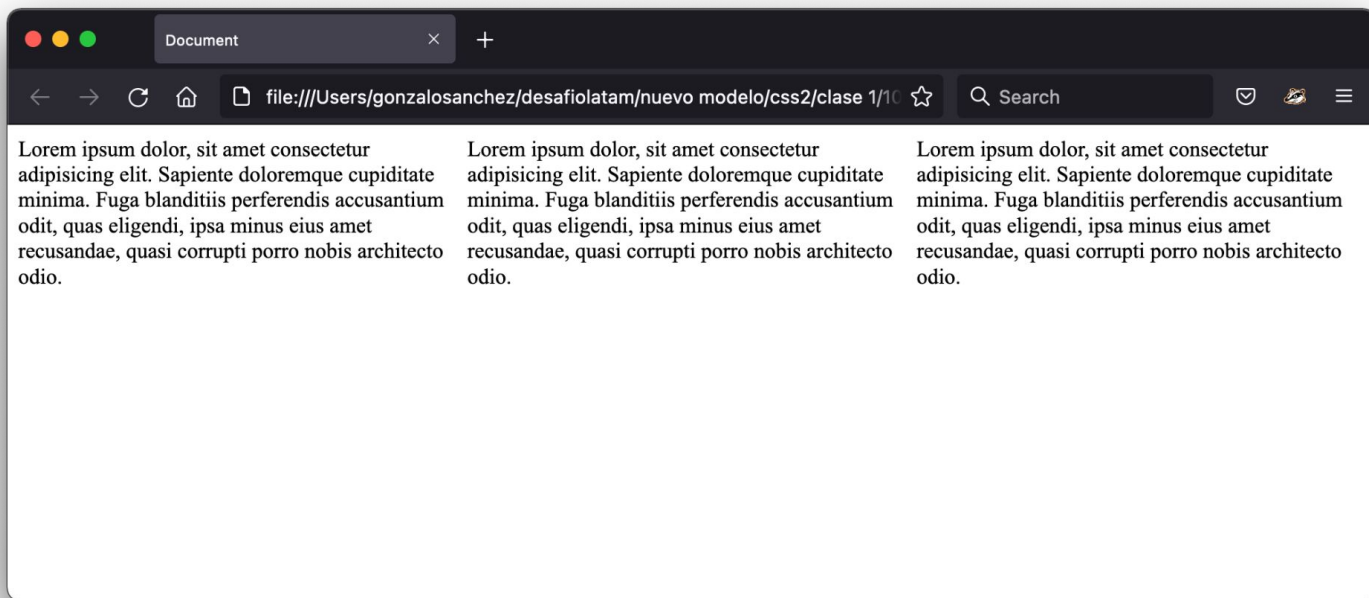
¿Qué es Flexbox?

- Flexbox o contenedores flex es una herramienta que nos permite alinear contenido de forma sencilla.
- Fue ideado para la construcción de layouts que contengan elementos distribuidos en filas y columnas.



Flexbox

Alineando contenido con flex



Flexbox

Alineando contenido con flex

```
1 <div class="boxes">
2   <div class="box">
3     Lorem ipsum dolor, sit amet consectetur adipisicing elit. Sapiente
4     doloremque cupiditate minima. Fuga blanditiis perferendis accusantium
5     odit, quas eligendi, ipsa minus eius amet recusandae, quasi corrupti
6     porro nobis architecto odio.
7   </div>
8   <div class="box">
9     Lorem ipsum dolor, sit amet consectetur adipisicing elit. Sapiente
10    doloremque cupiditate minima. Fuga blanditiis perferendis accusantium
11    odit, quas eligendi, ipsa minus eius amet recusandae, quasi corrupti
12    porro nobis architecto odio.
13  </div>
14  <div class="box">
15    Lorem ipsum dolor, sit amet consectetur adipisicing elit. Sapiente
16    doloremque cupiditate minima. Fuga blanditiis perferendis accusantium
17    odit, quas eligendi, ipsa minus eius amet recusandae, quasi corrupti
18    porro nobis architecto odio.
19  </div>
20 </div>
```

```
1 .boxes{
2   display: flex;
3 }
```

Flexbox

Alineando contenido

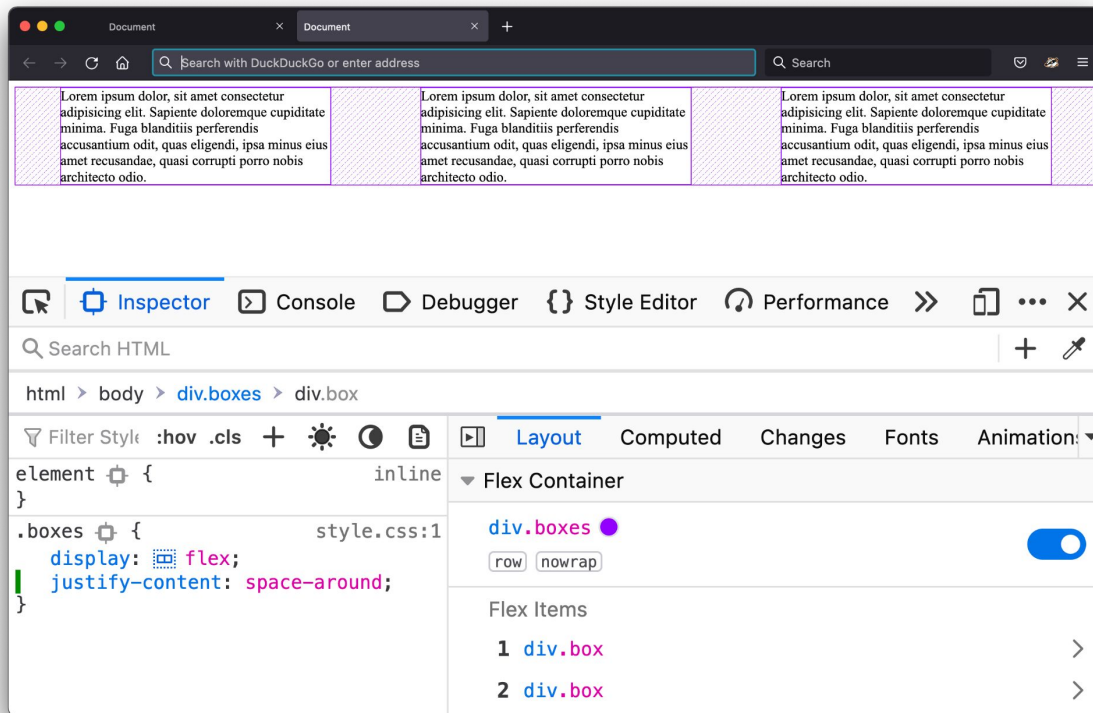
- Cuando las cajas suman menos del 100%, tendremos la oportunidad de escoger cómo alinear el contenido, a la izquierda, centrado, a la derecha, entre otras opciones.
- Utilizaremos la propiedad **justify-content** con distintos valores para alinear el contenido.



```
1  .boxes{
2    display: flex;
3    justify-content: flex-start;
4  }
5
6  .box{
7    width: 25%;
8  }
```

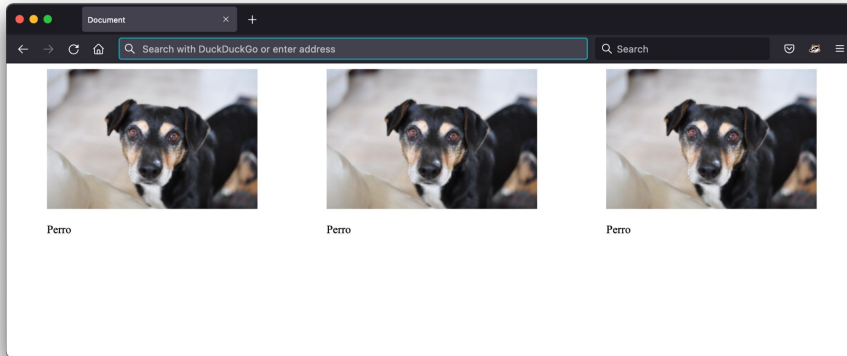
Flexbox

Utilicemos el inspector de elementos para alinear el contenido

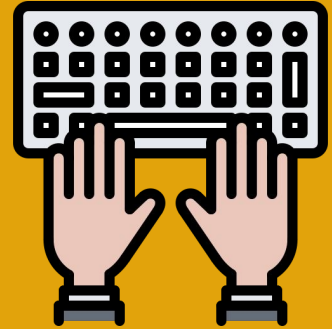


Utilizando flexbox

1. Crear un archivo index.html
2. Dar una estructura base
3. Crear un div contenedor padre con display:flex
4. Crear 3 div contenedores hijos
5. Agregar una imagen pequeña de 300px a cada contenedor, o ajustarla por CSS
6. Agregar un texto debajo de cada imagen
7. Utilizar el inspector de elementos para separar los contenedores utilizando la propiedad justify-content



Ejercicio ¡Manos al teclado!

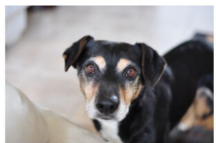


Por defecto, Flexbox utiliza **flex-wrap: nowrap**; esto hace que los elementos aparezcan en una única línea, por lo que si queremos hacer algo como una galería de imágenes con flex, tenemos que cambiarla por **flex-wrap: wrap**.

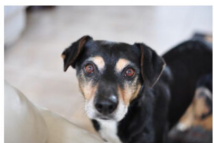
```
.container{  
  display: flex;  
  flex-wrap: wrap;  
}
```

Utilizando flexbox

1. En el archivo index.html creado anteriormente, agrega varias imágenes más (incluyendo div y texto)
2. Al contenedor padre, agrega la propiedad flex-wrap:wrap
3. Agrega la clase .box img{ max-width: 100% } para ajustar el tamaño de las imágenes y que no quede una sobre otra.
4. Puedes modificar el ancho de las cajas con un número menor a 25% para dejar espacio entre las imágenes.



Perro



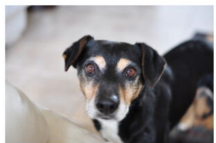
Perro



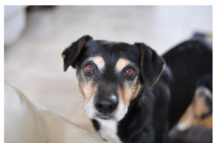
Perro



Perro



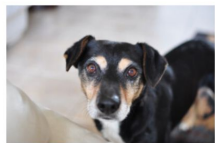
Perro



Perro



Perro



Perro

Ejercicio ¡Manos al teclado!



/* Utilizaremos flexbox para poner contenidos uno al lado de otro. */

- Crear una galería de imágenes. ✓
- Crear un menú.
- Crear una sección de características.

Construyendo un menú con flex

Parte 1: menu y flex



```
1 <ul>
2   <li> Item 1 </li>
3   <li> Item 2 </li>
4   <li> Item 3 </li>
5 </ul>
```



```
1 ul {
2   display: flex;
3   gap: 20px;
4   list-style-type: none;
5 }
```

La idea es similar, si aplicamos flex sobre ``, los hijos pasan a convertirse en elementos flex; los que podemos separar utilizando la propiedad `gap` y remover el disco utilizando `list-style-type:none`

Construyendo un menú con flex

Parte 1: menu y flex

Después, con CSS podemos darle estilo.

Una idea sencilla podría ser la siguiente.

```
1 body, ul {  
2   margin: 0px;  
3 }  
4 ul {  
5   display: flex;  
6   gap: 20px;  
7   list-style-type: none;  
8   background-color: black;  
9   color: white;  
10 }  
11 li {  
12   padding: 1.2rem;  
13 }  
14 li:hover {  
15   background-color: white;  
16   color: black;  
17 }
```

Item 1

Item 2

Item 3

Prueba pasar el mouse por sobre el ítem 2.

Pseudoselectores

¿Qué son?

- Los **pseudoselectores** en CSS se llaman así porque no seleccionan elementos reales en el árbol de documentos HTML, sino que seleccionan elementos basados en una condición o estado específico del elemento o de su contenido.

Pseudoselector	Ejemplo de Uso	Descripción
:hover	a:hover { color: red; }	Selecciona un elemento cuando el usuario pasa el cursor sobre él.
:active	a:active { color: blue; }	Selecciona un elemento cuando está siendo activamente seleccionado (por ejemplo, cuando un usuario hace clic en un enlace).

/* Utilizaremos flexbox para poner contenidos uno al lado de otro. */

- Crear una galería de imágenes. ✓
- Crear un menú. ✓
- Crear una sección de características.



Pregunta



¿Qué significa body, ul {...}?



Pregunta



¿Qué sucede en el ejercicio anterior si cualquiera de los 2 (body o ul) tiene un valor distinto a cero?



Pregunta



¿Qué es un pseudoselector?

Íconos

Agrandando un ícono

- Ingresar a fontawesome.com
- Crear un kit.
- Seguir las instrucciones para integrarlo.
- Luego, buscar un ícono de tu gusto y copiarlo a la página.

1

Add Your Kit's Code to a Project

```
<script src="https://kit.fontawesome.com/9e2d2bddba.js" crossorigin="anonymous"></script>
```

Copy Kit Code!

Copy your kit's code into the `<head>` of each template or page in your project where you want to use Font Awesome via this kit. If you've got other versions or flavors of Font Awesome already in use, make sure to remove them, mmkay?

Here you go! Check out the example HTML below.

```
<!doctype html>
<html>
  <head>
    <!-- Place your kit's code here -->
    <script src="https://kit.fontawesome.com/9e2d2bddba.js" crossorigin="anonymous"></script>
  </head>

  <body>
    <!-- Ready to use Font Awesome. Activate interlock. Dynotherms - connected. Infracells - up. Icons ar
    <i class="fas fa-thumbs-up fa-5x"></i>
  </body>
</html>
```



Download Example File

Íconos

Ejemplo

Donde:

- fa-solid es el tipo de ícono (también puedes probar fa-regular).
- fa-thumbs-up es el ícono de dedo hacia arriba.
- fa-4x es una clase que amplifica el tamaño original del ícono.

```
<i class="fa-solid fa-thumbs-up fa-4x"></i>
```


Construyendo una sección de características

Características



Lorem ipsum dolor sit amet consectetur
adipiscing elit. Quae officiis repellat
ipsam, suscipit, eius magni totam
voluptate aliquid minus dolorum earum
at qui perspiciatis blanditiis ad nisi vitae
necessitatibus maiores!



Lorem ipsum dolor sit amet consectetur
adipiscing elit. Quae officiis repellat
ipsam, suscipit, eius magni totam
voluptate aliquid minus dolorum earum
at qui perspiciatis blanditiis ad nisi vitae
necessitatibus maiores!



Lorem ipsum dolor sit amet consectetur
adipiscing elit. Quae officiis repellat
ipsam, suscipit, eius magni totam
voluptate aliquid minus dolorum earum
at qui perspiciatis blanditiis ad nisi vitae
necessitatibus maiores!

Construyendo una sección de características

- Todo el contenido va a quedar sobre una sección (o div) que tiene fondo gris y texto centrado.
- Dentro de la sección tendremos un `<h2>` con el texto “características”.



Construyendo una sección de características



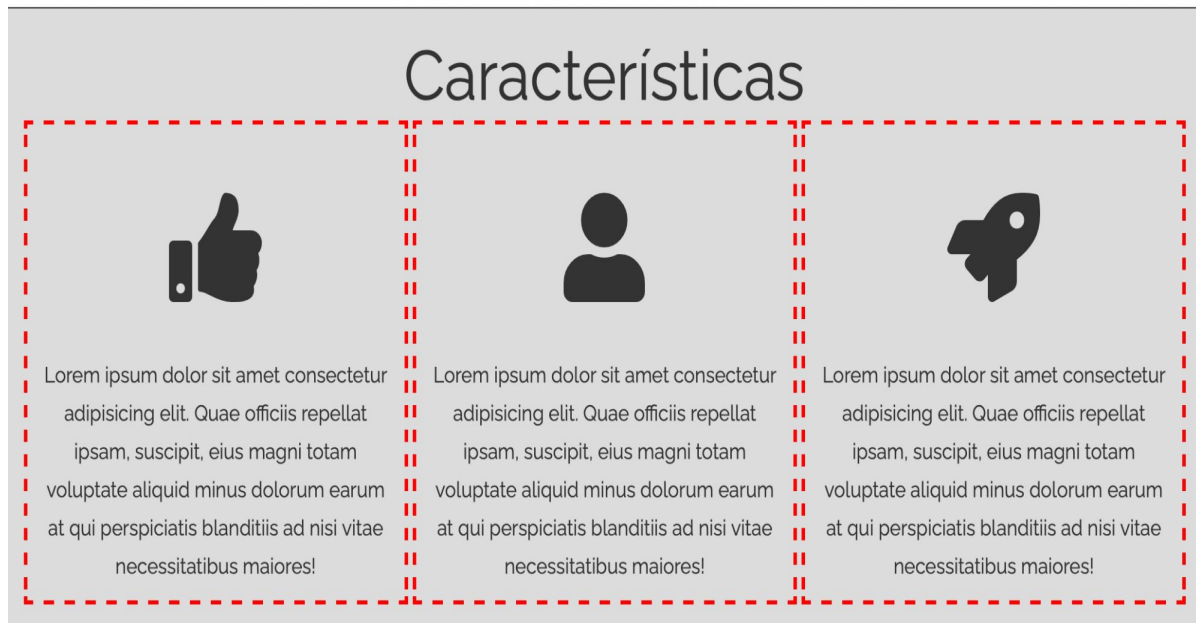
```
<section class="caracteristicas">
  <h2>Características</h2>
</section>
```

```
body {
  margin: 0px;
}

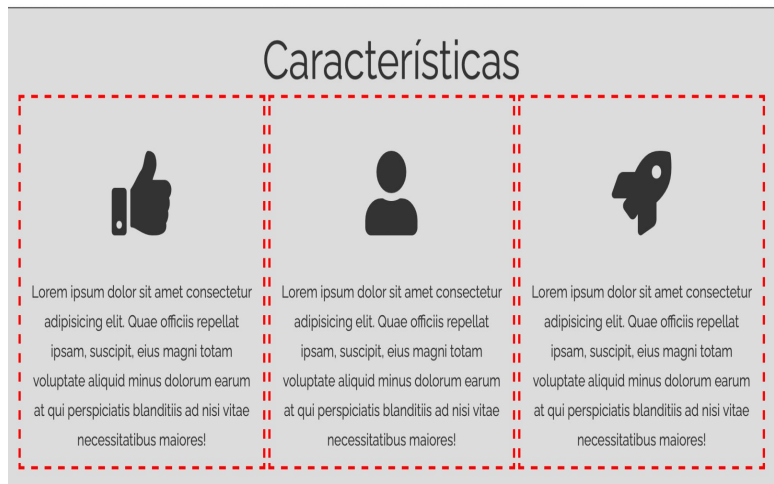
.caracteristicas {
  background-color: #f1f1f1;
  text-align: center;
}
```

Construyendo una sección de características

- Dentro de la sección, debajo del título, crearemos un div con el estilo display: flex.
- Dentro de este div, tendremos 3 divs adicionales, y dentro de cada uno, un párrafo con el texto lorem ipsum.
- Finalmente agregamos los íconos.



Construyendo una sección de características



```
<section class="caracteristicas">
  <h2>Características</h2>
  <div class="flex">
    <div>
      <i class="fa-regular fa-thumbs-up fa-4x"></i>
      <p>
        Lorem ipsum dolor sit amet consectetur adipiscing elit. Quae officiis repellat ipsam, suscipit, eius magni totam voluptate aliquid minus dolorum earum at qui perspicatis blanditiis ad nisi vitae necessitatibus maiores!
      </p>
    </div>
    <div>
      <i class="fa-solid fa-user fa-4x"></i>
      <p>
        Lorem ipsum dolor sit amet consectetur adipiscing elit. Quae officiis repellat ipsam, suscipit, eius magni totam voluptate aliquid minus dolorum earum at qui perspicatis blanditiis ad nisi vitae necessitatibus maiores!
      </p>
    </div>
    <div>
      <i class="fa-solid fa-rocket fa-4x"></i>
      <p>
        Lorem ipsum dolor sit amet consectetur adipiscing elit. Quae officiis repellat ipsam, suscipit, eius magni totam voluptate aliquid minus dolorum earum at qui perspicatis blanditiis ad nisi vitae necessitatibus maiores!
      </p>
    </div>
  </div>
</section>
```

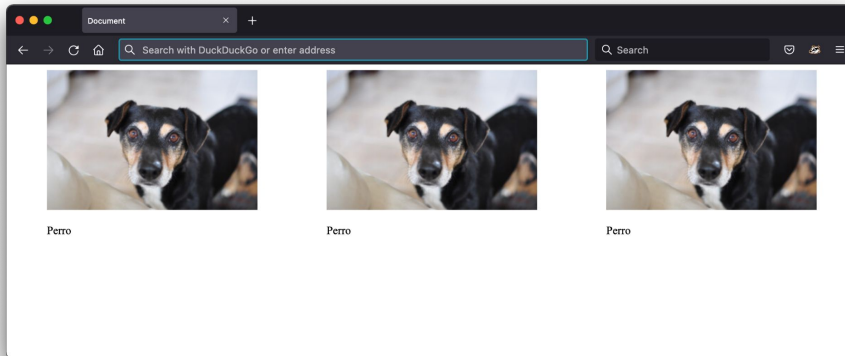
```
body {
  margin: 0px;
}

.flex {
  display: flex;
  gap: 20px;
}

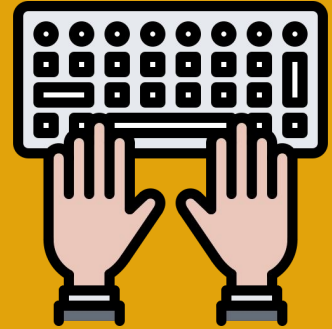
.caracteristicas {
  background-color: #f1f1f1;
  padding: 20px;
  text-align: center;
}
```

Repite los pasos desde cero

1. Crear un archivo index.html
2. Dar una estructura base
3. Crear una sección con el texto “características”
4. Crea el contenedor padre (div) para aplicar display:flex
5. Crear 3 div contenedores hijos, con un ícono y un párrafo de texto
6. Con CSS, agrega la propiedad display, color de fondo, cambia márgenes, gap, alineamiento de texto y lo que estimes conveniente. Muestra tus resultados.



Ejercicio ¡Manos al teclado!



/* Utilizaremos flexbox para poner contenidos uno al lado de otro. */

- Crear una galería de imágenes. ✓
- Crear un menú. ✓
- Crear una sección de características. ✓

Objetivo



Cierre

{desafío}
latam_



¿Existe algún concepto que no
hayas comprendido?

Reflexionemos

- Revisar la guía que trabajarán de forma autónoma.
- Revisar en conjunto el desafío.

¿Qué sigue?



*Academia de
talentos digitales*

www.desafiolatam.com



/DesafioLatam



/DesafioLatam



/DesafioLatam



/DesafioLatam