



# Flexbox



{desafío}  
latam\_

# CSS Avanzado

Unidad	Clases (sincrónico)	Autoaprendizaje (asincrónico)	Tutoría (sincrónico)
Flexbox	2 horas	Desde 6 horas	2 horas
Grid	2 horas	Desde 6 horas	2 horas
Media Queries	2 horas	Desde 6 horas	2 horas
Animaciones de CSS	2 horas	Desde 6 horas	2 horas
<i>Receso</i>	<i>0 horas</i>	<i>0 horas</i>	<i>0 horas</i>



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

***Crear diseños responsivos usando la propiedad flexbox, conociendo previamente los conceptos básicos aplicados al uso de flexbox como método de posicionamiento y creación de diseños en CSS.***

- Unidad 1: Flexbox.
- Unidad 2: Grid.
- Unidad 3: Media Queries.
- Unidad 4: Animaciones con CSS.



Te encuentras aquí





# Inicio

{desafío}  
latam\_



`/* Alinear contenido con Flexbox. */`

`/* Crear un menú vertical */`

`/* Crear un layout con Flexbox */`

A vertical line runs down the center of the slide, separating the white background on the left from the orange background on the right. The line is composed of several overlapping, semi-transparent shapes: a large curly brace at the top, an '@' symbol in the middle, and a stylized figure at the bottom.

## Objetivos de la sesión

# 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.







¿Cuál es la propiedad CSS para establecer un contenedor como flexbox?

a) display: flex

b) flex-direction: row

c) flex-wrap: wrap



¿En cuál etiqueta deberíamos agregar la propiedad `display:flex`?

```
<div>  
  <p> Hola </p>  
  <p> Hola </p>  
  <p> Hola </p>  
</div>
```

a) En la etiqueta `<div>`

b) En cada etiqueta `<p>`

c) En la primera etiqueta `<p>`



¿Cuál es la propiedad CSS para establecer el flujo de los elementos flex en varias líneas?

a) flex-flow

b) flex-wrap

c) flex-direction



Activación de conceptos



Primer lugar:

\_\_\_\_\_



Segundo lugar:

\_\_\_\_\_



Tercer lugar:

\_\_\_\_\_



# Desarrollo

{desafío}  
latam\_



# Repaso Flexbox

*¿Qué es Flexbox?*

Flexbox o contenedores flex es una herramienta que nos permite alinear contenido de forma sencilla.



# Repaso Flexbox

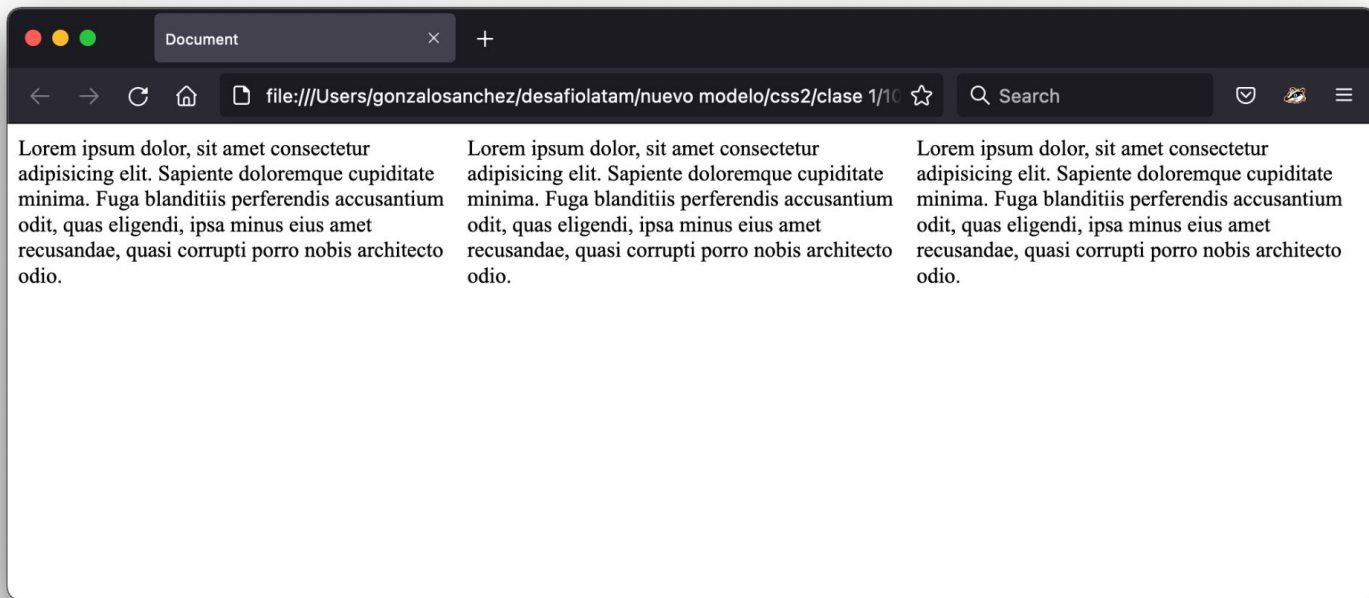
## Alineando contenido con flex

```
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 }
```

# Repaso Flexbox

## Alineando contenido con flex





# Repaso 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.

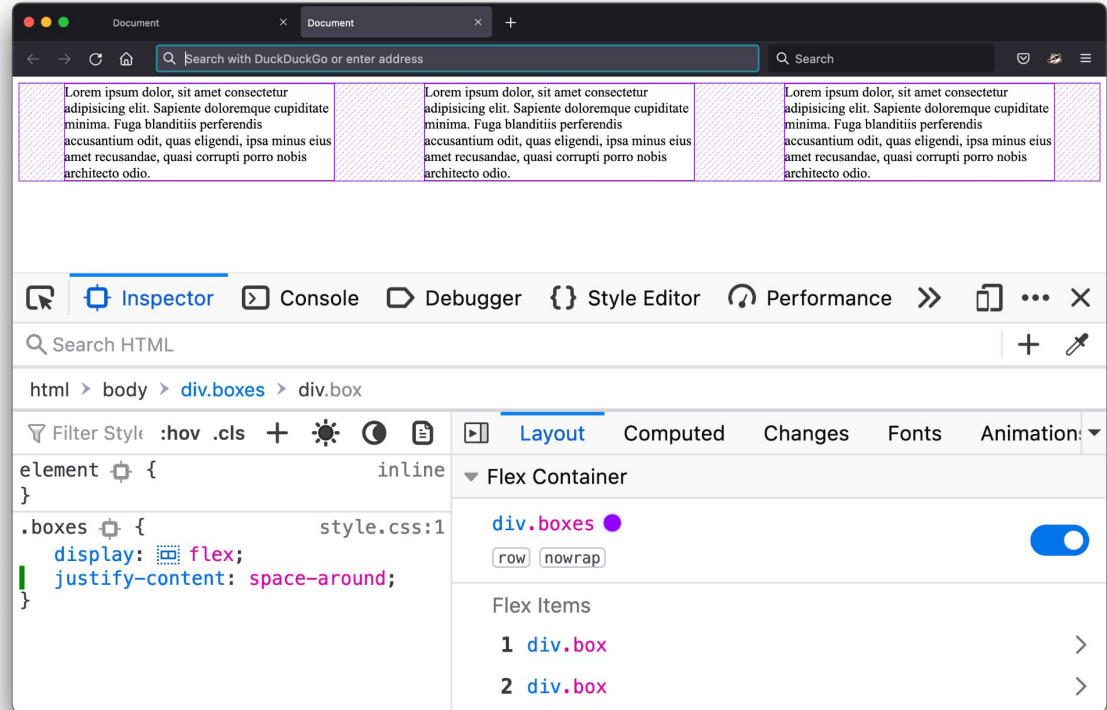


```
1  .boxes{
2    display: flex;
3    justify-content: flex-start;
4  }
5
6  .box{
7    width: 25%;
8  }
```

# Repaso Flexbox

## Alineando contenido

- Utilizaremos la propiedad **justify-content** con distintos valores para alinear el contenido.
- Podemos probar con los valores **flex-start**, **flex-end**, **center** y **space-around** entre otros.



# Repaso Flexbox

## Alineando contenido

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



Perro



Perro



Perro



Perro



Perro



Perro



Perro



Perro

# Centrando contenido Flexbox

*El eje principal*

**Texto encima de la imagen de fondo**

Descripción de la sección

# Alineando contenido con Flexbox

*El eje principal*

```
.hero{  
  display: flex;  
  justify-content: center;  
  height: 500px;  
}
```

CSS

```
<div class="hero">  
  <div class="content">  
    <h1>Texto encima de la imagen  
de fondo</h1>  
    <p>Descripción de la  
sección</p>  
  </div>  
</div>
```

HTML

# Alineando contenido con Flexbox


*El eje secundario*

**Texto encima de la imagen de fondo**

Descripción de la sección

# Alineando contenido con Flexbox

## *El eje secundario*



```
.hero{  
  display: flex;  
  justify-content: center;  
  height: 500px;  
  align-items: center;  
}
```

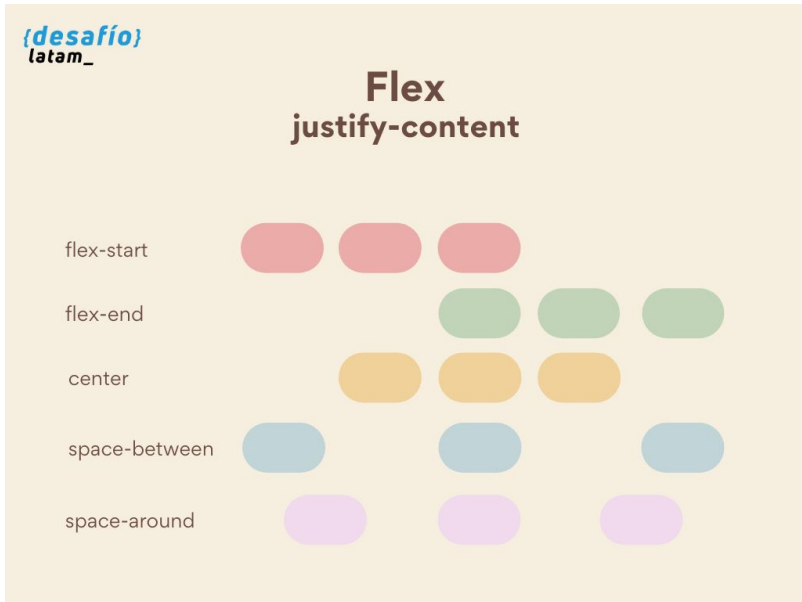
CSS

```
<div class="hero">  
  <div class="content">  
    <h1>Texto encima de la imagen  
de fondo</h1>  
    <p>Descripción de la  
sección</p>  
  </div>  
</div>
```

HTML

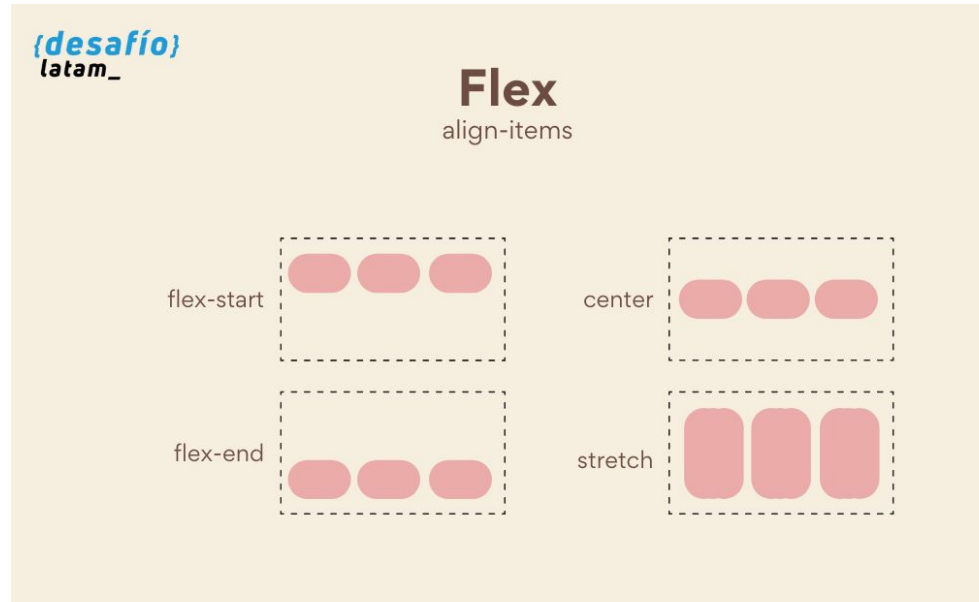
# Alineando contenido con Flexbox

## *Posibles valores*



**{desafío}**  
**latam\_**

**Eje  
principal**



**Eje  
secundario**



# Centrando contenido Flexbox

*Combinando lo aprendido con imágenes de fondo*



Vive el futuro

Sitio corporativo de una empresa de arquitectura

# Centrando contenido Flexbox

## *Combinando lo aprendido con imágenes de fondo*



- Para resolverlo, tenemos que preguntarnos: ¿el texto está a la izquierda, al centro o a la derecha? En función de eso, escogemos el valor de justify-content
- Luego tenemos que preguntarnos: ¿el texto está arriba, al medio o abajo? En función de eso, escogemos el valor de align-items
- Finalmente, jugamos con las propiedades de padding, altos de las secciones, colores de fondo para lograr el efecto deseado.

# Centrando contenido Flexbox

## Implementando Flex con una imagen de fondo

```
body{
  margin: 0;
  font-family: 'Roboto',
sans-serif;

}

.hero {
  display: flex;
  align-items: flex-end;
  justify-content: center;
  height: 800px;
  background-image:
url("corporate_building.png");
  background-size: cover;
}
```

```
.content {
  font-size: 1.4em;
  font-stretch: expanded;
  height: 8em;
  width: 100%;
  padding: 1em 8em;
  background-color:
rgba(15,15,15, 0.6);
  text-align: center;
  color: white;
  text-shadow: 1px 1px #000;
}
```


```
<section class="hero">
  <div class="content">
    <h1> Vive el futuro </h1>
    <p> Sitio corporativo de una
empresa de Arquitectura </p>
  </div>
</section>
```

## Utilizando Flexbox

1. Descarga una imagen, puedes utilizar un banco de imágenes como <https://unsplash.com/>
2. Crea un archivo html nuevo, dentro del body con la sección hero, genera un div con la clase content, un h1 y un p con contenido. (Igual que en el ejercicio anterior).
3. Utilizando CSS, agrega la imagen de fondo, agrega flex sobre la sección y utilizando las propiedades align-items y justify-content
4. Utiliza las propiedades de height, background-size y background-position para ajustar la imagen y obtener un resultado interesante.

## Ejercicio ¡Manos al teclado!



*/\* Alinear contenido con Flexbox \*/* 

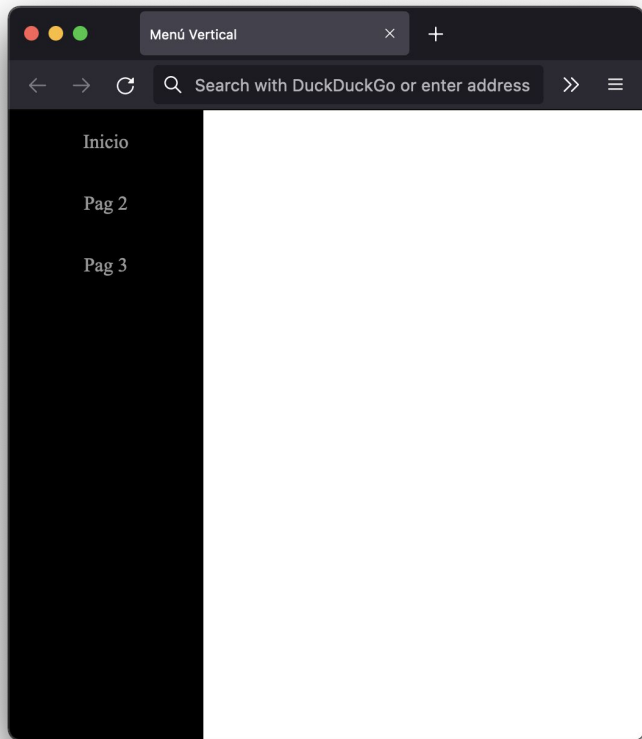
*/\* Crear un menú vertical \*/*

*/\* Crear un layout con Flexbox \*/*

## Objetivos de la sesión

## Flexbox tiene 2 ejes

*Principal y secundario*



- Usualmente, el eje principal es el horizontal y el eje secundario es el vertical, pero podemos cambiar esto y de esta forma podremos hacer cosas interesantes, como un menú vertical.

# Repaso Flexbox

## Creando un menú con flex



```
<menu>  
  <ul> ←  
    <li>  
      <a>
```

Para lograrlo, primero realizaremos un menú normal.

Aquí agregamos **display:flex** ya que es el contenedor padre de los elementos que alineamos uno al lado del otro.

```
body, menu{
  margin: 0;
}
menu {
  background-color:black;
}
menu ul{
  margin: 0;
  padding: 0;
  display: flex;
}
```

```
menu li{
  padding: 1em;
  list-style: none;
}
menu a{
  text-decoration: none;
  color: white;
  color: rgba(255, 255, 255, 0.6);
}
menu a:hover{
  color: white;
}
```

CSS

{desafío}  
latam\_

## Repaso Flexbox

### Creando un menú con flex

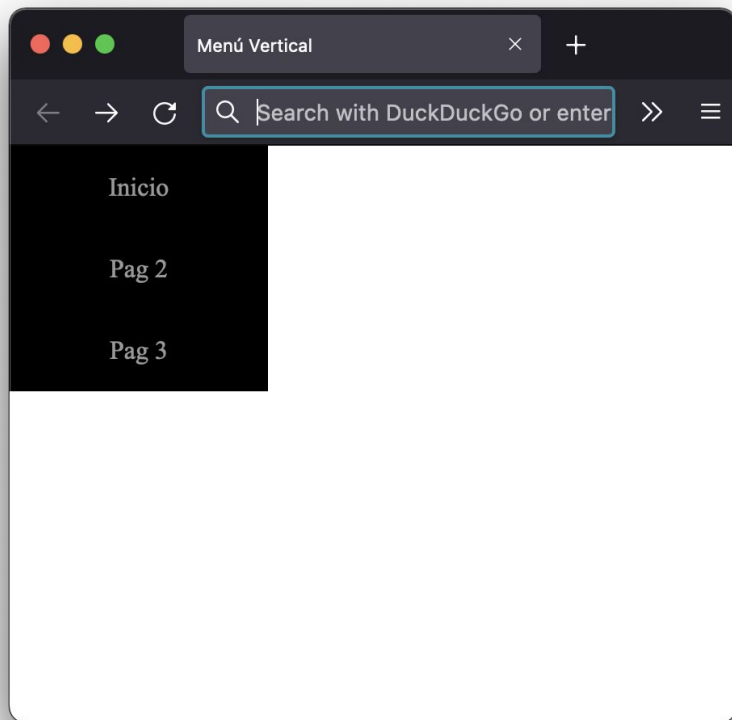
```
<menu>
  <ul>
    <li>
      <a href="#"> Inicio </a>
    </li>
    <li>
      <a href="#"> Pag 2 </a>
    </li>
    <li>
      <a href="#"> Pag 3 </a>
    </li>
  </ul>
</menu>
```

HTML



# Flexbox

## *Dirección de la columna*



- Es posible cambiar la dirección del eje de flex utilizando **flex-direction: column**; de esta forma, flex forma una columna en lugar de una fila.
- También tendremos que indicar un ancho para el menú.

# Repaso Flexbox

## Creando un menú con flex

```
body, menu{
  margin: 0;
}
menu {
  background-color:black;
  width: 10em;
}
menu ul{
  margin: 0;
  padding: 0;
  display: flex;
  flex-direction: column;
}
```

```
menu li{
  padding: 1em;
  list-style: none;
}
menu a{
  text-decoration: none;
  color: white;
  color: rgba(255, 255, 255, 0.6);
}
menu a:hover{
  color: white;
}
```


```
<menu>
  <ul>
    <li>
      <a href="#"> Inicio </a>
    </li>
    <li>
      <a href="#"> Pag 2 </a>
    </li>
    <li>
      <a href="#"> Pag 3 </a>
    </li>
  </ul>
</menu>
```

## Utilizando Flexbox

1. Repite los pasos del ejercicio anterior.
2. Luego, crea un menú.
3. Agrega una lista `<ul>` con 3 elementos y links.
4. Agrega un color de fondo al menú.
5. Utiliza `display:flex` en la lista.
6. Utiliza `flex-direction: column` en la lista.
7. Define un ancho para el menú.

## Ejercicio ¡Manos al teclado!



*/\* Alinear contenido con Flexbox \*/* 

*/\* Crear un menú vertical \*/* 

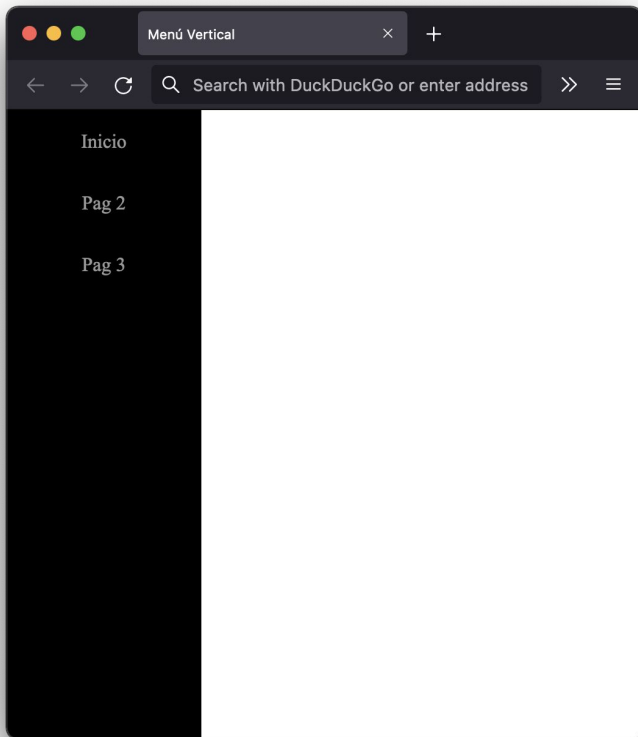
*/\* Crear un layout con Flexbox \*/*



## Objetivos de la sesión

# Flexbox

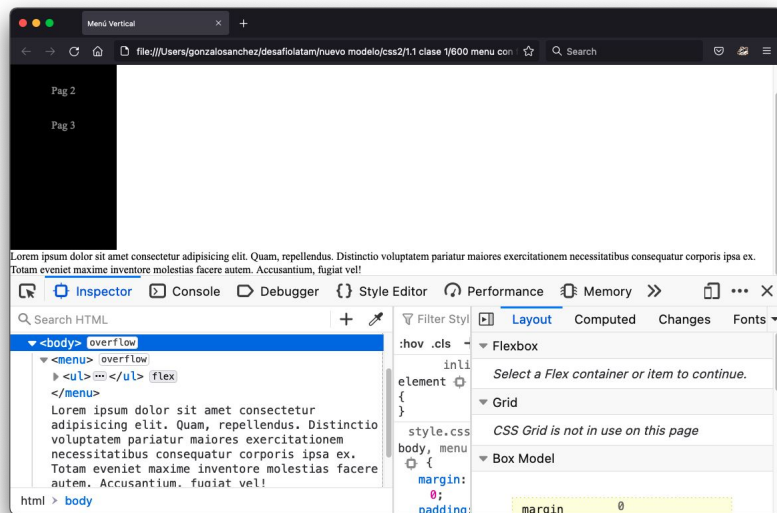
## *Alto de la columna*



- En este caso, Flex ocupa el 100% del documento, pero si queremos que ocupe todo, podemos utilizar: **min-height: 100vh** dentro del menú.
- Esto se lee como: asígnale el alto mínimo posible (**min-height**) de 100% en función del viewport vertical (**100vh**).

# Flexbox

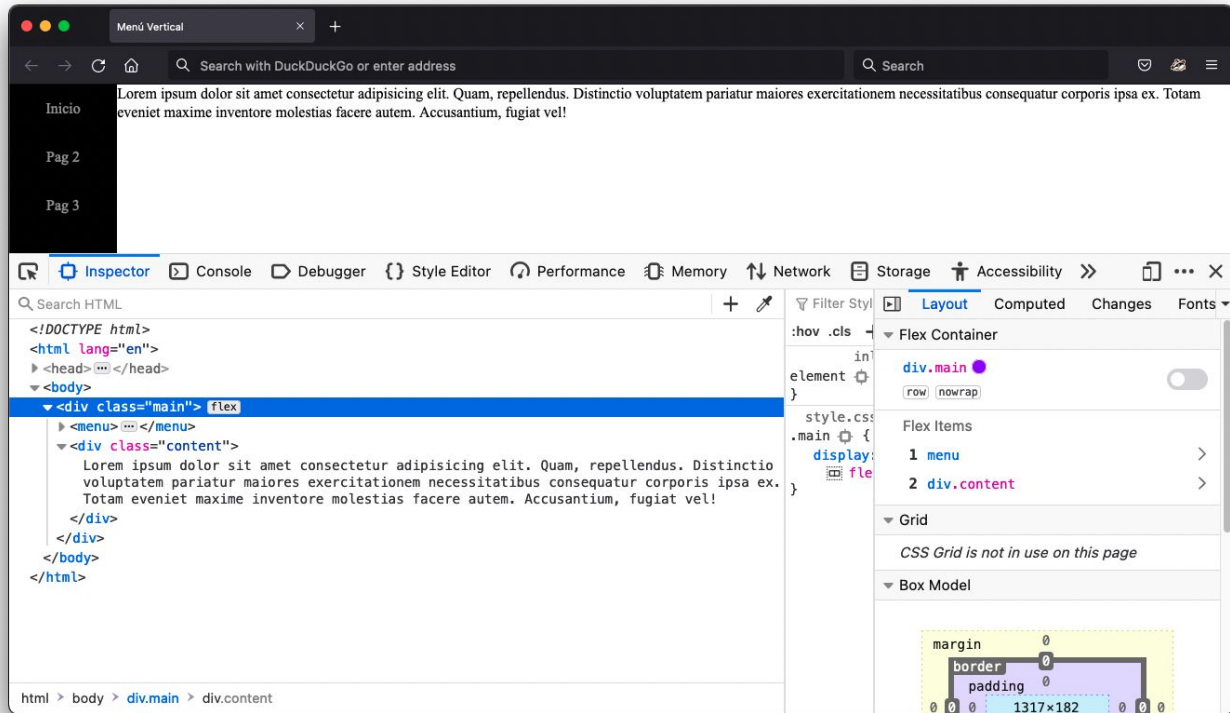
## Agregando contenido en el bloque principal



- Si intentamos agregar contenido, quedará debajo del menú, en lugar de la sección blanca.
- Para lograr dejarlo al lado del menú, nos falta agregar un contenedor padre al menú, y al contenido que agreguemos en la sección principal.
- Al contenedor padre le agregaremos la propiedad `display:flex`

# Flexbox

## Agregando contenido en el bloque principal



## Utilizando Flexbox

1. Repite los pasos del ejercicio anterior.
2. Agrega al menú un alto mínimo de 100vh (min-height: 100vh).
3. Agrega un div sobre menú.
4. Agrega un div (o sección u otro tipo de bloque) como etiqueta hermano al menú.
5. Agrega un párrafo dentro de la sección agregada.
6. Observa los cambios.

## Ejercicio ¡Manos al teclado!





`/* Alinear contenido con Flexbox */` ✓

`/* Crear un menú vertical */` ✓

`/* Crear un layout con Flexbox */` ✓

## Objetivos de la sesión



Cierre

{desafío}  
latam\_



¿Existe algún concepto que no  
hayas comprendido?

Reflexionemos

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

¿Qué sigue?



*Academia de  
talentos digitales*

[www.desafiolatam.com](http://www.desafiolatam.com)



/DesafioLatam



/DesafioLatam



/DesafioLatam



/DesafioLatam