

Material complementario - Float

Utilizando floats

La propiedad float permite fijar un elemento a la izquierda o a la derecha, y que el texto alrededor envuelva el elemento flotante.



Imagen 1. Ejemplo básico de floats
Fuente: Desafío Latam.



Actividad 1

- Replica la imagen 1.
- Crea la estructura de un sitio web, agrega una imagen y agrega mucho texto dentro de uno o más párrafos.
- Agrega el siguiente css (en el head o archivo externo).

```
img {  
  float: left;  
}
```

- Observa la página ¿lograste replicarlo?
 - Puedes utilizar width y height para cambiar el tamaño de la imagen y lograr un mejor efecto.
- A continuación implementa los siguientes cambios:
 - Cambia **float: left** por **float: right**
 - Agrega una segunda imagen con la clase **r**
 - Agrega el selector **.r** al css y agrega **float:left**
 - ¿Qué imagen se muestra primero? ¿Si en el HTML cambiamos el orden de las imágenes cambia el orden al ver la página?



Actividad 2

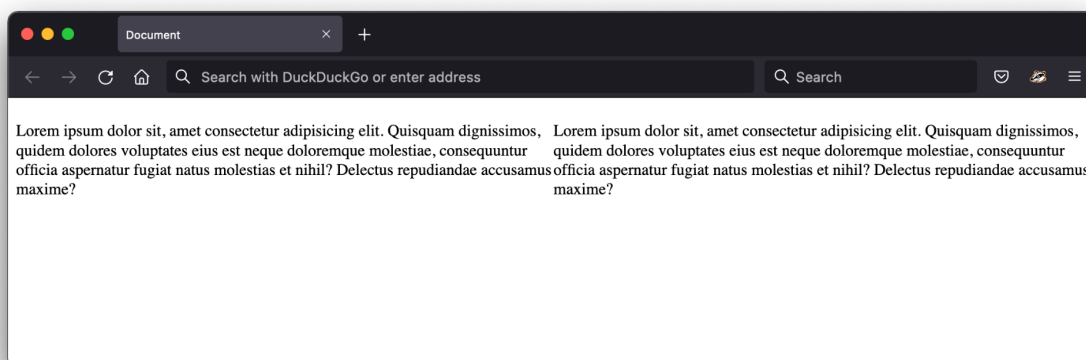


Imagen 2. Textos con float

Fuente: Desafío Latam.

- Replica la imagen 2.
- Crea la estructura de un sitio web, agrega dos párrafos con el texto Lorem Ipsum, utilizando css agrega **float: left** sobre los párrafos.

Colapso por elementos flotantes

Cuando un elemento padre contiene solo hijos flotantes, el contenedor padre colapsa y deja de ser visible.

En la siguiente actividad, podremos observar en qué consiste este fenómeno y aprender a cómo resolverlo.

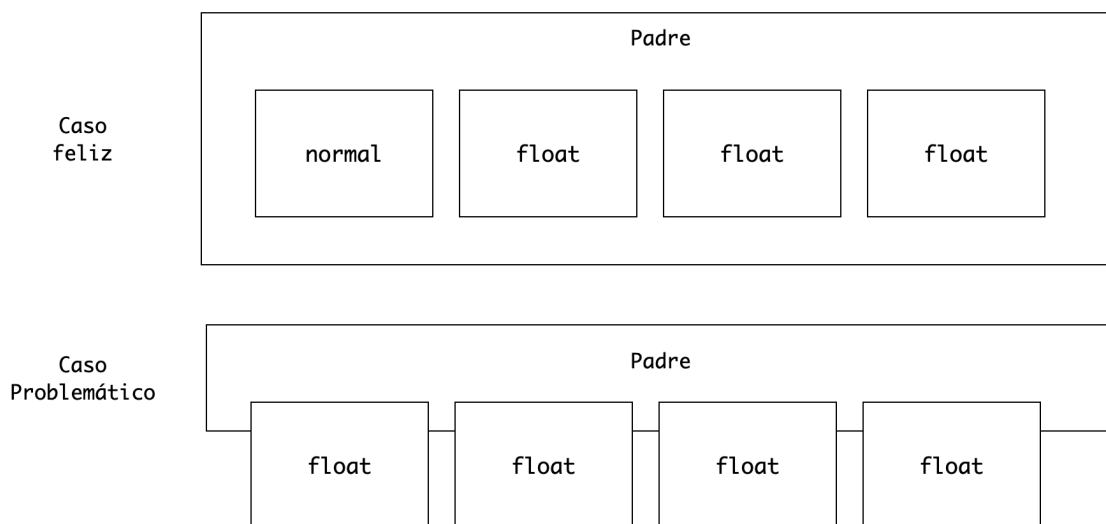


Imagen 3. Colapso de elementos flotantes
Fuente: Desafío Latam.



Actividad 3

- Crea un archivo index.html y agrega la siguiente estructura:

```
<section class="features">  
  <p> Lorem, ipsum dolor sit amet consectetur adipisicing elit. Deserunt cumque aperiam quo  
  rerum a debitis recusandae veniam error quibusdam minus tempora, ullam distinctio labore  
  voluptatibus consequatur ratione esse sed voluptatem.</p>  
  <p> Lorem, ipsum dolor sit amet consectetur adipisicing elit. Deserunt cumque aperiam quo  
  rerum a debitis recusandae veniam error quibusdam minus tempora, ullam distinctio labore  
  voluptatibus consequatur ratione esse sed voluptatem.</p>  
  <p> Lorem, ipsum dolor sit amet consectetur adipisicing elit. Deserunt cumque aperiam quo  
  rerum a debitis recusandae veniam error quibusdam minus tempora, ullam distinctio labore  
  voluptatibus consequatur ratione esse sed voluptatem.</p>  
</section>
```

- Agrega css especificando que los párrafos deben flotar a la izquierda y con un ancho de 33%.
- Agrega un color de fondo sobre la clase padre.
- Abre la página en el navegador, deberías ver los párrafos, pero no deberías ver el color de fondo.
- Agrega la propiedad display: flow-root al contenedor padre.
- Abre el navegador deberías poder ver el color de fondo.

display:flow-root

La solución de aplicar display:flow-root se conoce como clearfix, y de hecho existen otras formas de lograrla. display:flow-root es una forma nueva de hacerlo, pero si por algún extraño motivo llegáramos a necesitar compatibilidad con navegadores muy antiguos, como por ejemplo IE 6 o 7 tendríamos que ocupar tanto html como css, el siguiente ejemplo explica cómo lograrlo:

```
<div class="padre">
  <img class="float" -- >
  <img class="float" >
  <div class="clearfix">
</div>
```

```
.clearfix::after {
  content: "";
  clear: both;
  display: table;
}
```

Podemos saber que tan compatible es una propiedad de CSS con algún navegador utilizando la página caniuse <https://caniuse.com/?search=flow-root>

Problemas de tamaño y Floats

Hay un caso muy similar al anterior, donde dentro de un contenedor padre hay 2 elementos o más elementos hijos, donde uno de los elementos es float y el otro no, donde el más grande flota.

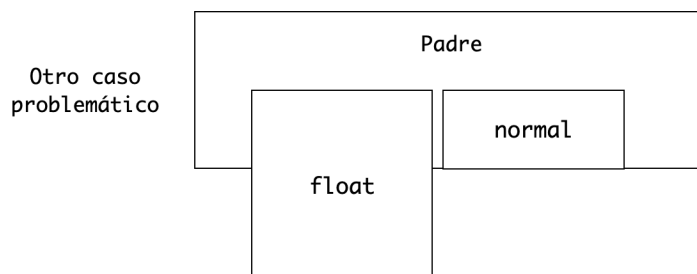


Imagen 4. Colapso de elementos flotantes (v2)

Fuente: Desafío Latam.

Veamos el siguiente ejemplo:

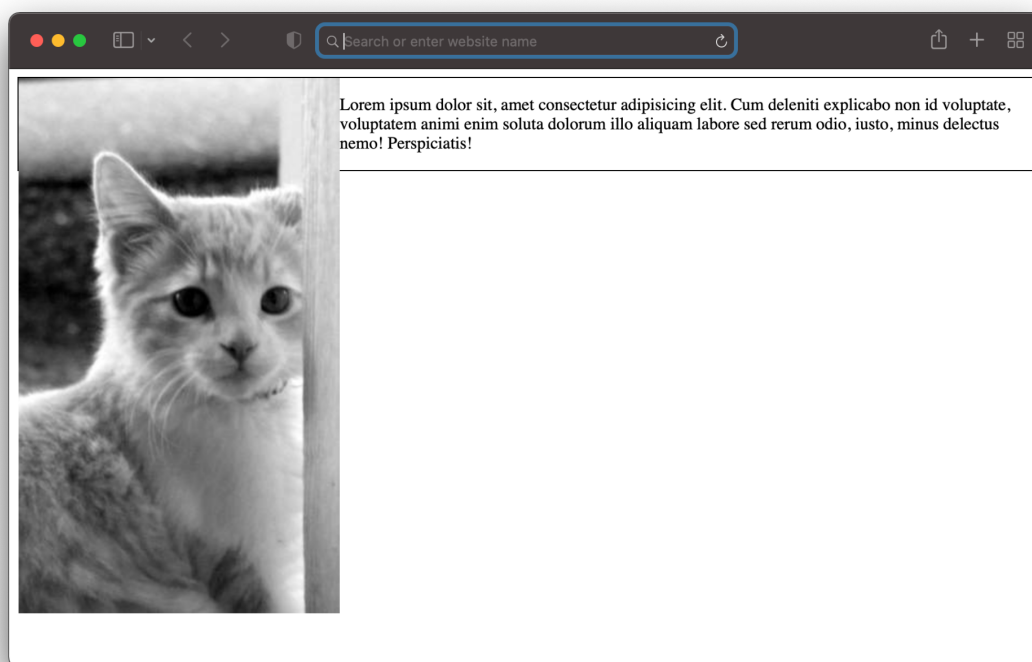


Imagen 5. Screenshot de ejemplo de un problemas con Floats

Fuente: Desafío Latam.

A partir de este, podemos recrear la página mostrada con el siguiente html:

```
<section>
  
  <p> Lorem ipsum dolor sit, amet consectetur adipisicing elit. Cum deleniti explicabo non id voluptate, voluptatem animi enim soluta dolorum illo aliquam labore sed rerum odio, iusto, minus delectus nemo! Perspiciatis! </p>
</section>
```

Y agregando el siguiente css:

```
section {
  border: solid 1px black;
}
img{
  float: left;
}
```

Podemos resolver el problema de la imagen de la misma forma que aprendimos anteriormente: agregando al css del contenedor padre (en este caso .section) display: flow-root;

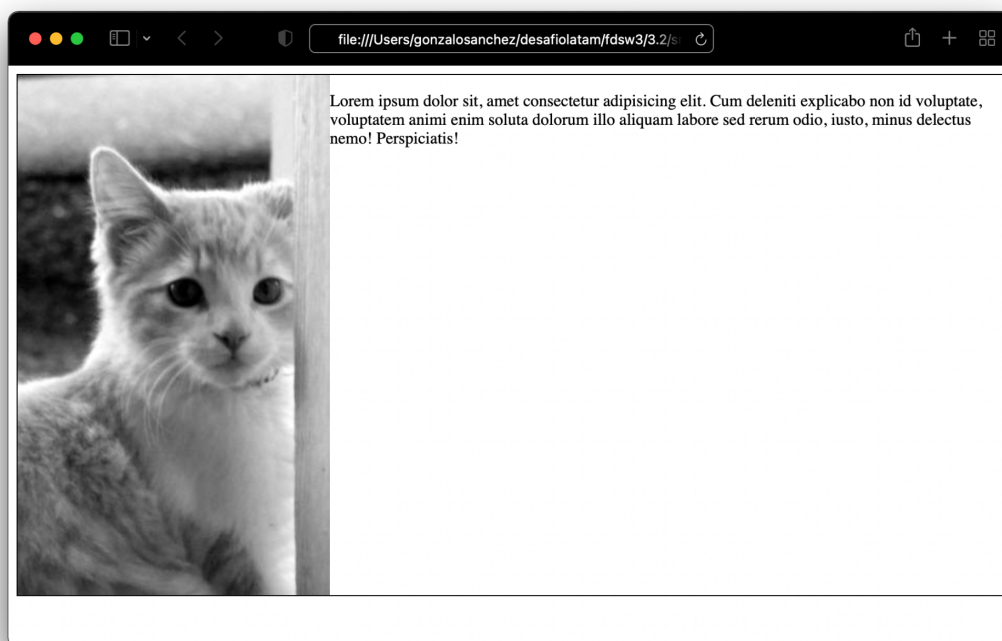


Imagen 6. Solucionando con display: flow-root
Fuente: Desafío Latam.

Otra forma de resolver el problema anterior, es utilizando otras propiedades como overflow:auto al contenedor padre, o aplicando la técnica de clearfix previamente aprendida.



Actividad 4

- Resolver el problema anterior ocupando el método antiguo de clearfix que requiere un div adicional.
- Resolver el problema agregando la propiedad overflow: auto en lugar de display: flow-root



En CSS hay varias formas de resolver un mismo problema, algunas serán más sencillas que otras, y en algunos casos especiales tendremos que recurrir a técnicas más complejas por problemas específicos de compatibilidad.

Construyendo un menú con floats

Para desarrollar menú con floats, utilizaremos la misma estructura html del menú que hicimos con display:inline-block

```
<menu>
<ul>
  <li>
    <a href=""> 1 </a>
  </li>
  <li>
    <a href=""> 2 </a>
  </li>
  <li>
    <a href=""> 3 </a>
  </li>
</ul>
</menu>
```

Luego, tendremos que flotar los elementos hermanos, en este caso los y asignar un color de fondo.

```
li {
  float: left;
}
menu {
  background-color: black;
  color: white;
  line-height: 50px;
}
```

Sin embargo, no podremos ver el color de fondo del menú por el problema de que todos los elementos internos flotan, así que agregamos el `display:flow-root`. Con esto, deberíamos poder ver el menú.

```
li {  
  float: left;  
}  
  
menu {  
  background-color: black;  
  color: white;  
  line-height: 50px;  
  display:flow-root;  
}
```

Luego, damos ajustes finales como márgenes:

```
body, menu {  
  padding: 0px;  
  margin: 0px;  
}  
  
menu ul{  
  margin: 0px;  
}  
  
menu {  
  background-color: black;  
  color: white;  
  line-height: 50px;  
  display: flow-root;  
}  
  
menu li{  
  display: float;  
  margin-right: 50px;  
}  
  
menu a {  
  color: white;  
  text-decoration: none;  
}
```


¿Tabla, Float o Inline block?

Cuando queremos poner un elemento al lado de otro, podemos utilizar tanto tablas, floats o display: inline-block, sin embargo, en general se considera una mala práctica hacer uso de tablas para realizarlo, por el motivo que requiere agregar mucho más html de lo que significa hacerlo sin tablas.

Adicionalmente, las tablas pueden romper el copiado de texto del sitio en algunas ocasiones. Considera que deberías ocupar tablas exclusivamente cuando quieras agregar datos en una tabla.

¿Cómo escoger entre float o inline blocks?

Para envolver una imagen entre texto debemos utilizar floats, para cualquier otro caso podemos escoger entre cualquiera de las dos.

En general, es un poco más sencillo utilizar inline blocks, el diagrama visto en clases puede ayudar a tomar una decisión.

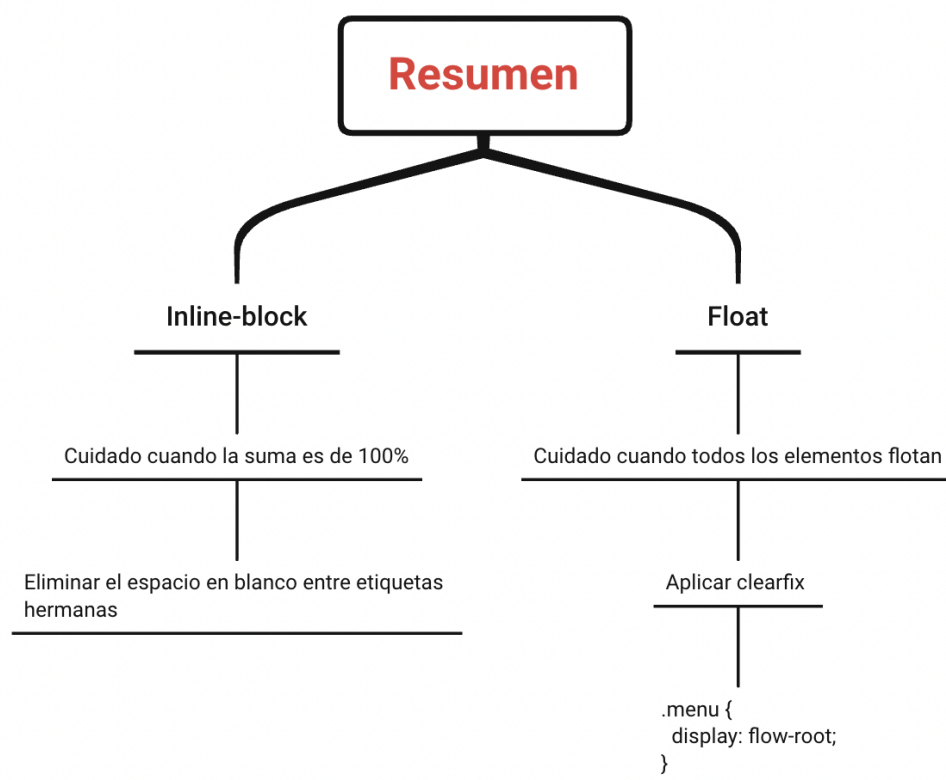


Imagen 7. Resumen de problemas y soluciones de float e inline-block

Fuente: Desafío Latam.



Actividad 5: Revisando los aprendizajes

Revisa que puedas realizar las siguientes acciones:

- Utilizar etiquetas de html semántico.
- Crear un menú utilizando float.
- Crear un menú utilizando inline-block.
- Resolver el problema de colapso de floats utilizando un clearfix.
- Agregar íconos a un sitio web como por ejemplo los de font-awesome (esta información está exclusivamente en la ppt de la clase).
- Crear una sección de características como la siguiente.

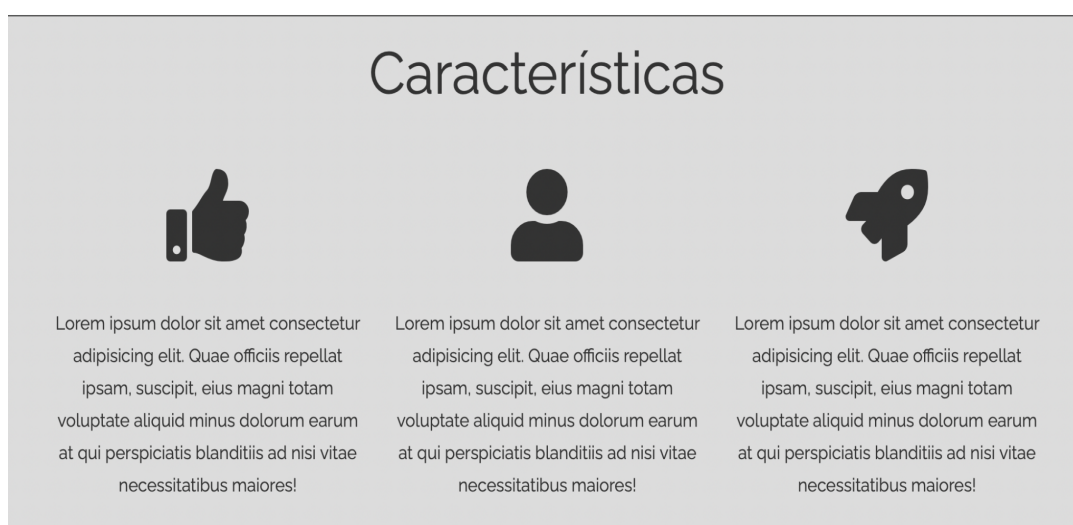


Imagen 8. Actividad 5.
Fuente: Desafío Latam.

¡Continúa aprendiendo y practicando!

