

Lenguaje de Marcas y Sistemas de Gestión

UA 2.12 - Introducción CSS: Posicionamiento y Desplazamiento



**Universidad
Europea**

LAUREATE INTERNATIONAL UNIVERSITIES

Raúl Rodríguez Mercado

raul.rodriguez@universidadeuropea.es / @raulrodriguezue

Dpto. Ciencias y Tecnología de la Informática y Comunicación

UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Objetivos

- ☐ Definir qué es CSS
- ☐ Conocer la estructura de las hojas de estilo y como se aplican a los documentos HTML



UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Contenidos

- Posicionamiento
- Desplazamiento



UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Posicionamiento en CSS

- ❑ La propiedad **position**, como su nombre indica, sirve para **establecer la posición de un elemento**.
- ❑ Su valor por defecto es **static** (sin posición).

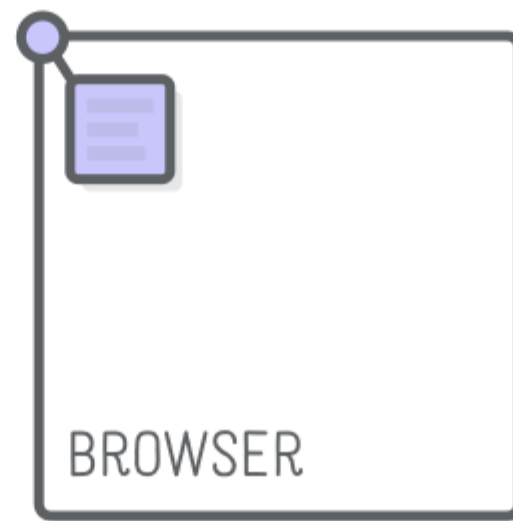
STATIC



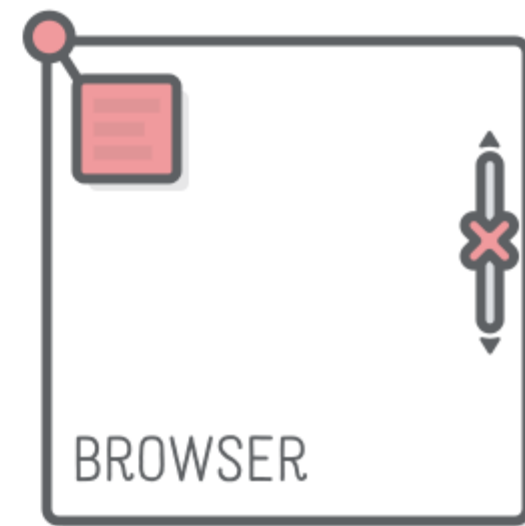
RELATIVE



ABSOLUTE



FIXED



UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento

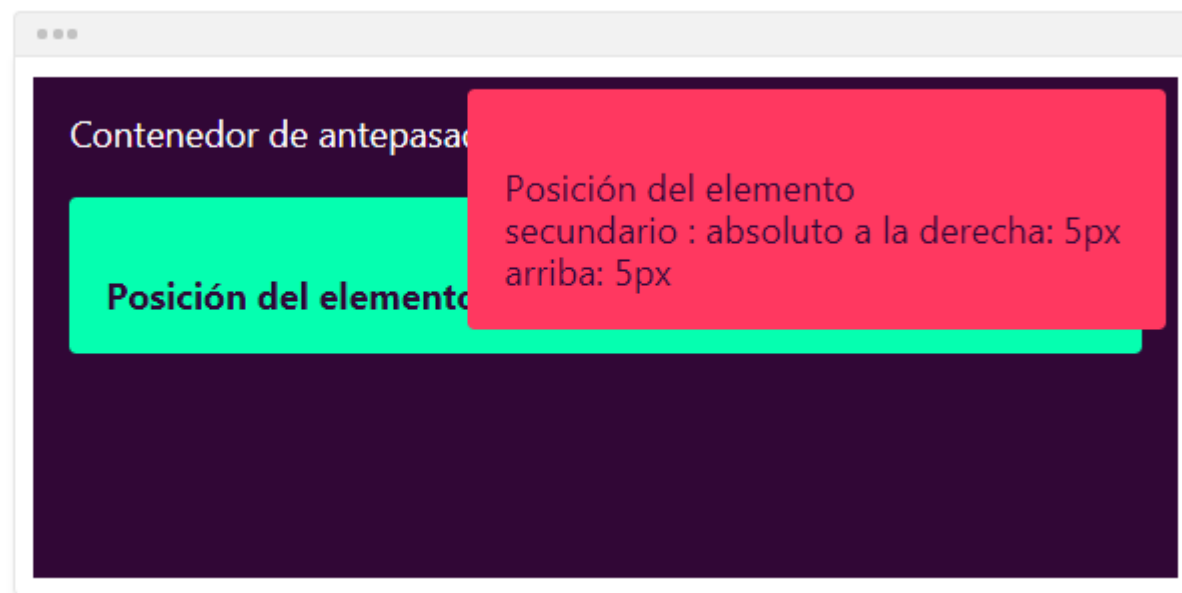


position: static;

- ☐ El elemento permanecerá en el **flujo natural** de la página.
- ☐ Como resultado, **no** actuará como punto de anclaje para el bloque rosa absolutamente posicionado.
- ☐ Además, **no** reaccionará a las siguientes propiedades:

- ✓ top
- ✓ bottom
- ✓ left
- ✓ right
- ✓ z-index

default



UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo de position: static;

```
<!Doctype html>
<html>
<head><link rel="stylesheet" type="text/css"
      href="position.css" />
</head>
<body>
<div class="static">
```

static (estático) es el valor por defecto. Un elemento con position: static; no está posicionado en ninguna forma en específico. Se dice que un elemento static, está no posicionado y un elemento con valor establecido de position está posicionado.

```
</div>
</body>
</html>
```

```
.static {
  border-style: solid;
  border-color: blue;
  position: static;
}
```



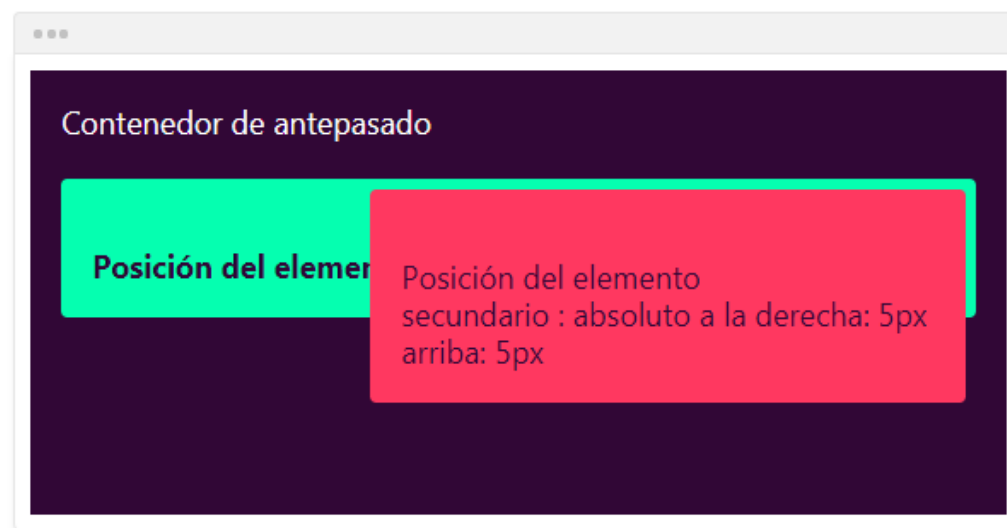
static (estático) es el valor por defecto. Un elemento con position: static; no está posicionado en ninguna forma en específico. Se dice que un elemento static, está no posicionado y un elemento con valor establecido de position está posicionado.

UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



position: relative;

- ❑ El elemento permanecerá en el **flujo natural** de la página.
- ❑ Es decir, los elementos se colocan exactamente igual que en el posicionamiento estático (permanecen en la misma posición), pero dependiendo del valor de las propiedades top, bottom, left o right variarán ligeramente la posición del elemento.
- ❑ También **posiciona** el elemento: actuará como un punto de anclaje para el bloque rosa absolutamente posicionado y **reacciona** a la propiedad z-index.



UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo de position: relative;

```
<!Doctype html>
<html>
<head><link rel="stylesheet" type="text/css"
      href="position2.css" />
</head>
<body>
<div class="relative1">
relative (relativo) se comporta de la misma manera que static a menos que le agregues otras propiedades.
</div>
<div class="relative2">
Establecer las propiedades top, right, bottom, y left de un elemento con position: relative; causará que su posición normal se
reajuste. Otro contenido no se puede ajustar para adaptarse a cualquier hueco dejado por el elemento.
</div>
</body>
</html>
```

```
.relative1 {
border-style: solid;
position: relative;
}
.relative2 {
border-style: solid;
position: relative;
top: -20px;
left: 20px;
background-color: blue;
width: 500px;
}
```



relative (relativo) se comporta de la misma manera que static a menos que le agregues otras propiedades.

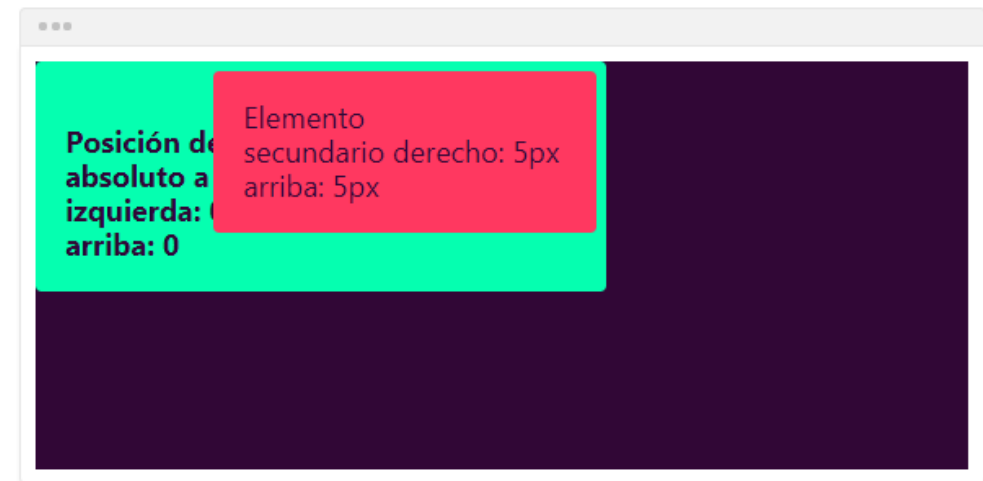
Otro contenido no se puede ajustar para adaptarse a cualquier hueco dejado por el elemento.

UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



position: absolute;

- ❑ El elemento **no** permanecerá en el flujo natural de la página. Se posicionará de acuerdo con el **ancestro posicionado más cercano**. Debido a que está **posicionado**, actuará como un punto de anclaje para el bloque rosa absolutamente posicionado.
- ❑ Es decir, absolute se comporta como fixed.
- ❑ Si un elemento con position: absolute; no tiene ancestros posicionados, usará el elemento body del documento, y se seguirá moviendo al hacer scroll en la página. Recordar que un elemento "posicionado" es aquel cuyo valor es cualquiera excepto static.
- ❑ Además, **reaccionará** a las siguientes propiedades:
 - ✓ top
 - ✓ bottom
 - ✓ left
 - ✓ right
 - ✓ z-index



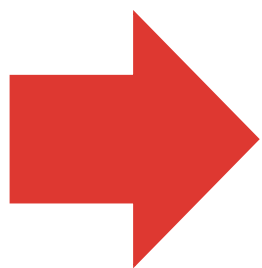
UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo de position: absolute;

```
<!Doctype html>
<html>
<head><link rel="stylesheet" type="text/css"
      href="position3.css" />
</head>
<body>
  <div class="relative">
    Este elemento está usando position: relative;. Si estuviera usando position: static; su sucesor con position: absolute;
    escaparía y se posicionaría relativamente al body del documento.
  </div>
  <div class="absolute">
    Este elemento tiene position: absolute;. Se posiciona de manera relativa a su ancestro.
  </div>
</body>
</html>
```

```
.relative {
  position: relative;
  border-style: solid;
  border-color: chartreuse;
  width: 600px;
  height: 400px;
}
.absolute {
  position: absolute;
  border-style: solid;
  border-color: yellow;
  top: 120px;
  right: 0;
  width: 300px;
  height: 200px;
}
```



Este elemento está usando position: relative;. Si estuviera usando position: static; su sucesor con position: absolute; escaparía y se posicionaría relativamente al body del documento.

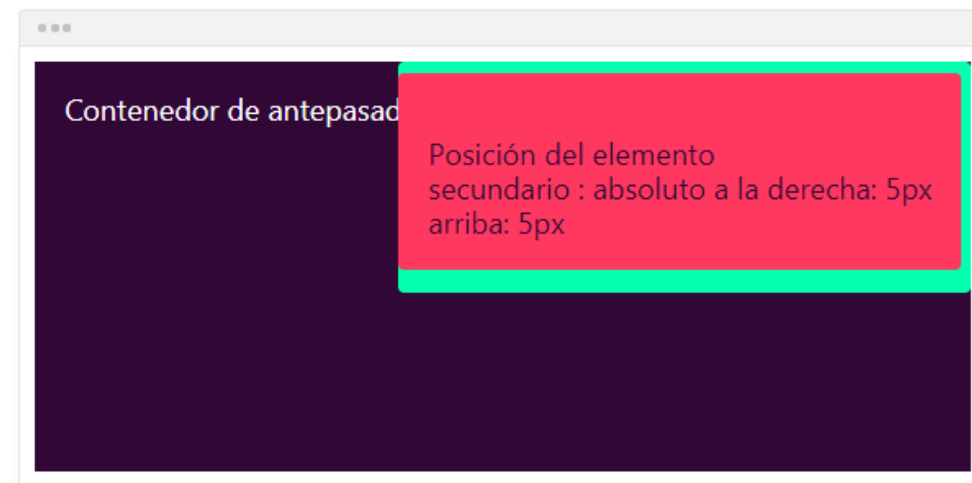
Este elemento tiene position: absolute;. Se posiciona de manera relativa a su ancestro.

UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



position: fixed;

- ❑ El elemento **no** permanecerá en el flujo natural de la página. Se posicionará de acuerdo con la **ventana gráfica** .
- ❑ Un elemento fixed (fijo) se posiciona a la ventana del navegador de manera relativa, lo que significa que se mantendrá en el mismo lugar incluso después de hacer scroll en la página
- ❑ Debido a que está **posicionado** , actuará como un punto de anclaje para el bloque rosa absolutamente posicionado.
- ❑ Además, **reaccionará** a las siguientes propiedades:
 - ✓ top
 - ✓ bottom
 - ✓ left
 - ✓ right
 - ✓ z-index



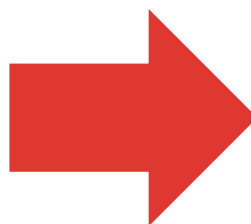
UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo de position: fixed;

```
<!Doctype html>
<html>
<head><link rel="stylesheet" type="text/css"
      href="position4.css" />
</head>
<body>
  <div class="fixed">
    Este es un elemento Fixed que no varía con el scrollola! No me prestes atención todavía
  </div>
</body>
</html>
```

```
.fixed {
  position: fixed;
  border-style: solid;
  border-color: blue;
  bottom: 0;
  right: 0;
  width: 200px;
  background-color: white;
}
```



Este es un elemento Fixed que
no varía con el scrollola! No
me prestes atención todavía

UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo de global de position;

```
<!Doctype html>
<html>
<head><link rel="stylesheet" type="text/css"
      href="position5.css" />
</head>
<body>
```



```
    <nav>
    <ul>
    <li>Inicio</li>
    <li>Contacto</li>
    <li>Servicios</li>
    <li>Quienes Somos</li>
    <li>Proyectos realizados</li>
    <li>Redes sociales</li>
    </ul>
</nav>
<section>
```

El estilo margin-left para section, es para que haya espacio para nav.

```
</section>
<section>
    Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.
</section>
```

```
<section>
    Mira lo que pasa si haces la ventana de tu navegador mas pequeña. ¡Funciona!
</section>
```

```
</body>
</html>
```

```
.container {
    position: relative;
    border-style: solid;
    border-color: blue;
}
nav {
    position: absolute;
    border-style: solid;
    border-color: red;
    left: 0px;
    width: 200px;
}
section {
    /* position is static by default */
    margin-left: 200px;
    border-style: solid;
    border-color: yellow;
}
footer {
    position: fixed;
    bottom: 0;
    left: 0;
    height: 70px;
    background-color: white;
    width: 100%;
}
body {
    margin-bottom: 120px;
    border-style: solid;
    border-color: green;
}
```



UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo de global de position;

```
<!doctype html>
<html>
<head><link rel="stylesheet" type="text/css"
      href="position5.css" />
</head>
<body>
  <nav>
    <ul>
      <li>Inicio</li>
      <li>Contacto</li>
      <li>Servicios</li>
      <li>Quienes Somos</li>
      <li>Proyectos realizados</li>
      <li>Redes sociales</li>
    </ul>
  </nav>
  <section>
    El estilo margin-left para section, es para que haya espacio para nav.
  </section>
  <section>
    Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.
  </section>
  <section>
    Mira lo que pasa si haces la ventana de tu navegador mas pequeña. ¡Funciona!
  </section>
</body>
</html>
```



```
.container {
  position: relative;
  border-style: solid;
  border-color: blue;
}
nav {
  position: absolute;
  border-style: solid;
  border-color: red;
  left: 0px;
  width: 200px;
}
section {
  /* position is static by default */
  margin-left: 200px;
  border-style: solid;
  border-color: yellow;
}
footer {
  position: fixed;
  bottom: 0;
  left: 0;
  height: 70px;
  background-color: white;
  width: 100%;
}
body {
  margin-bottom: 120px;
  border-style: solid;
  border-color: green;
}
```



- Inicio
- Contacto
- Servicios
- Quienes Somos
- Proyectos realizados
- Redes sociales

El estilo margin-left para section, es para que haya espacio para nav.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.

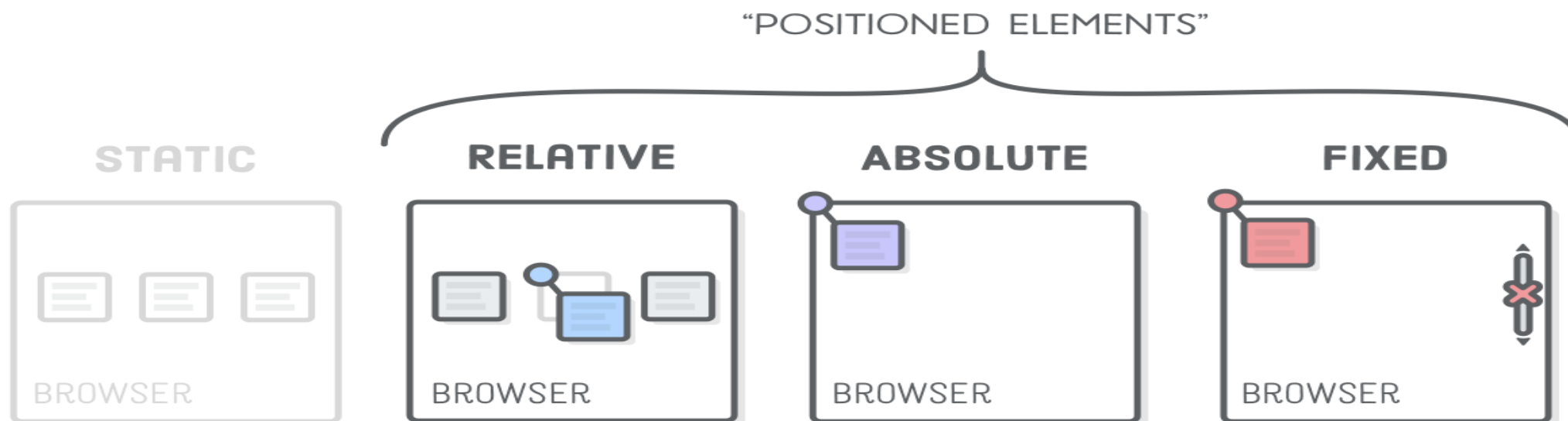
Mira lo que pasa si haces la ventana de tu navegador mas pequeña. ¡Funciona!

UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplos de Position

- ❑ <https://developer.mozilla.org/en-US/docs/Web/CSS/position>
- ❑ <http://cssreference.io/property/position/>
- ❑ <https://internetingishard.com/html-and-css/advanced-positioning/>
- ❑ <http://es.learnlayout.com/position-example.html>

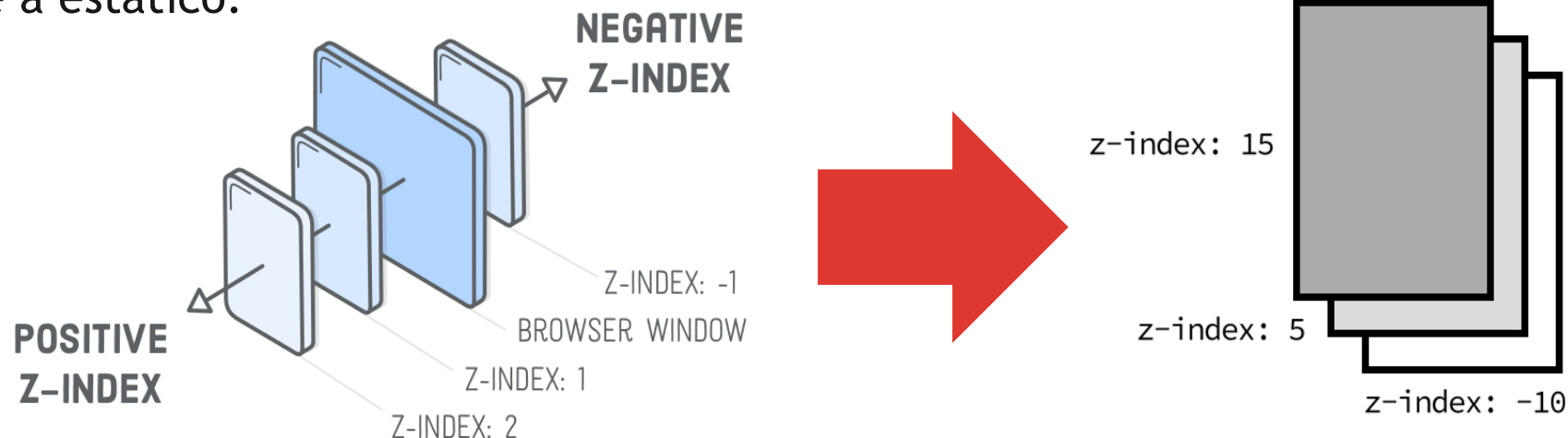


UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Profundidad (Niveles)

- ❑ La propiedad **z-index**, establece el **nivel de profundidad** en el que está un elemento sobre los demás.
- ❑ De esta forma, podemos hacer que un elemento se coloque encima o debajo de otro.
- ❑ Su funcionamiento es muy sencillo, sólo hay que indicar un número que representará el nivel de profundidad del elemento: los números más altos estarán por encima de otros números más bajos, que permanecerán ocultos.
- ❑ Los niveles **z-index**, así como las propiedades **top**, **left**, **bottom** y **right** no funcionan con elementos que estén utilizando posicionamiento estático. Deben tener un tipo de posicionamiento diferente a estático.



UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo de Profundidad (Niveles)

```
<!DOCTYPE html>
<html>
<head>
<title>Ejemplo Z-Index</title>
<link rel="stylesheet" type="text/css" href="zindex.css"/>
</head>
<body>
  <h1>Escalera de color</h1>

  <div id="ten_of_diamonds">
    
  </div>

  <div id="jack_of_diamonds">
    
  </div>

  <div id="queen_of_diamonds">
    
  </div>

  <div id="king_of_diamonds">
    
  </div>

  <div id="ace_of_diamonds">
    
  </div>
</body>
</html>
```



```
#ten_of_diamonds {
  position: absolute;
  left: 100px;
  top: 100px;
  z-index: 1;
}
#jack_of_diamonds {
  position: absolute;
  left: 115px;
  top: 115px;
  z-index: 2;
}
#queen_of_diamonds {
  position: absolute;
  left: 130px;
  top: 130px;
  z-index: 3;
}
#king_of_diamonds {
  position: absolute;
  left: 145px;
  top: 145px;
  z-index: 4;
}
#ace_of_diamonds {
  position: absolute;
  left: 160px;
  top: 160px;
  z-index: 5;
}
```

UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo de Profundidad (Niveles)

```
<!DOCTYPE html>
<html>
<head>
<title>Ejemplo Z-Index</title>
<link rel="stylesheet" type="text/css" href="zindex.css"/>
</head>
<body>
  <h1>Escalera de color</h1>

  <div id="ten_of_diamonds">
    
  </div>

  <div id="jack_of_diamonds">
    
  </div>

  <div id="queen_of_diamonds">
    
  </div>

  <div id="king_of_diamonds">
    
  </div>

  <div id="ace_of_diamonds">
    
  </div>
</body>
</html>
```



```
#ten_of_diamonds {
  position: absolute;
  left: 100px;
  top: 100px;
  z-index: 1;
}
#jack_of_diamonds {
  position: absolute;
  left: 115px;
  top: 115px;
  z-index: 2;
}
#queen_of_diamonds {
  position: absolute;
  left: 130px;
  top: 130px;
  z-index: 3;
}
#king_of_diamonds {
  position: absolute;
  left: 145px;
  top: 145px;
  z-index: 4;
}
#ace_of_diamonds {
  position: absolute;
  left: 160px;
  top: 160px;
  z-index: 5;
}
```

Escalera de color



UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Posicionamiento: float y clear

- ❑ La propiedad **float** nos permite realizar cambios en el flujo de ordenación de elementos, es decir, podemos conseguir que un elemento «flote» a la izquierda o a la derecha de otro elemento.
- ❑ La propiedad **clear** se encarga de impedir elementos flotantes en la zona indicada, a la izquierda del elemento (left), a la derecha (right) o en ambos lados (both).

Propiedad	Valor	Significado
float:	none left right	Cambia el flujo para que el elemento flote a la izquierda o derecha.
clear:	none left right both	Impide que los elementos puedan flotar en la orientación indicada.

UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo Posicionamiento: float

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
img {  
  float: right; CSS  
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<p>In this example, the image will float to the right in the  
paragraph, and the text in the paragraph will wrap around the image.  
</p>
```

```
<p>
```

```
  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus  
  imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae  
  scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices  
  nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue  
  ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in  
  odio. Praesent convallis urna a lacus interdum ut hendrerit risus  
  congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim  
  ac. In at libero sed nunc venenatis imperdiet sed ornare turpis.  
  Donec vitae dui eget tellus gravida venenatis. Integer fringilla  
  congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta.  
  Cras ac leo purus. Mauris quis diam velit.</p>
```

```
</body>
```

```
</html>
```

In this example, the image will float to the right in the paragraph, and the text in the paragraph will wrap around the image.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.



UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo Posicionamiento: float

```
<!DOCTYPE html>
<html>
<head>
<style>
img {
  float: left;
}
</style>
</head>
<body>
```

CSS

```
<p>In this example, the image will float to the left in the
paragraph, and the text in the paragraph will wrap around the image.
</p>
```

```
<p>
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus
imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae
scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices
nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut
aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in
odio. Praesent convallis urna a lacus interdum ut hendrerit risus
congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim
ac. In at libero sed nunc venenatis imperdiet sed ornare turpis.
Donec vitae dui eget tellus gravida venenatis. Integer fringilla
congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta.
Cras ac leo purus. Mauris quis diam velit.</p>
```

```
</body>
</html>
```

In this example, the image will float to the left in the paragraph, and the text in the paragraph will wrap around the image.



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis imperdiet sed ornare turpis.

Donec vitae dui eget tellus gravida venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.

UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo Posicionamiento: clear: left

```
<div class="wrapper">
  <p class="black">Lorem ipsum dolor sit amet,
consectetuer adipiscing elit. Phasellus sit amet diam.
Duis mattis varius dui. Suspendisse eget dolor.</p>
  <p class="red">Lorem ipsum dolor sit amet,
consectetuer adipiscing elit.</p>
  <p class="left">This paragraph clears left.</p>
</div>
```



```
.wrapper{
  border:1px solid black;
  padding:10px;
}
.left {
  border: 1px solid black;
  clear: left;
}
.black {
  float: left;
  margin: 0;
  background-color: black;
  color: #fff;
  width: 20%;
}
.red {
  float: left;
  margin: 0;
  background-color: pink;
  width:20%;
}
p {
  width: 50%;
}
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Phasellus sit amet diam. Duis mattis varius dui. Suspendisse eget dolor.	Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
This paragraph clears left.	



UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo Posicionamiento: clear: right

```
<div class="wrapper">
  <p class="black">Lorem ipsum dolor sit
  amet, consectetur adipiscing elit.
  Phasellus sit amet diam. Duis mattis
  varius dui. Suspendisse eget dolor.</p>
  <p class="red">Lorem ipsum dolor sit
  amet, consectetur adipiscing elit.</p>
  <p class="right">This paragraph clears
  right.</p>
</div>
```



```
.wrapper{
  border:1px solid black;
  padding:10px;
}
.right {
  border: 1px solid black;
  clear: right;
}
.black {
  float: right;
  margin: 0;
  background-color: black;
  color: #fff;
  width:20%;
}
.red {
  float: right;
  margin: 0;
  background-color: pink;
  width:20%;
}
p {
  width: 50%;
}
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit.	Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus sit amet diam. Duis mattis varius dui. Suspendisse eget dolor.
----------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------

This paragraph clears right.

UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo Posicionamiento: clear: both

```
<div class="wrapper">
  <p class="black">Lorem ipsum dolor sit amet,
consectetuer adipiscing elit. Phasellus sit amet diam.
Duis mattis varius dui. Suspendisse eget dolor. Fusce
pulvinar lacus ac dui.</p>
  <p class="red">Lorem ipsum dolor sit amet,
consectetuer adipiscing elit. Phasellus sit amet diam.
Duis mattis varius dui. Suspendisse eget dolor.</p>
  <p class="both">This paragraph clears both.</p>
</div>
```

Lorem ipsum
dolor sit amet,
consectetuer
adipiscing elit.
Phasellus sit
amet diam.
Duis mattis
varius dui.
Suspendisse
eget dolor.
Fusce pulvinar
lacus ac dui.

This paragraph clears both.

Lorem ipsum
dolor sit amet,
consectetuer
adipiscing elit.
Phasellus sit
amet diam.
Duis mattis
varius dui.
Suspendisse
eget dolor.



```
.wrapper{
  border:1px solid black;
  padding:10px;
}
.both {
  border: 1px solid black;
  clear: both;
}
.black {
  float: left;
  margin: 0;
  background-color: black;
  color: #fff;
  width:20%;
}
.red {
  float: right;
  margin: 0;
  background-color: pink;
  width:20%;
}
p {
  width: 45%;
}
```


UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo Posicionamiento: float y clear

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
<div class="header">
  <h1>Chania</h1>
</div>
<div class="clearfix">
  <div class="column menu">
    <ul>
      <li>The Flight</li>
      <li>The City</li>
      <li>The Island</li>
      <li>The Food</li>
    </ul>
  </div>
  <div class="column content">
    <h1>The City</h1>
    <p>Chania is the capital of the Chania region on the island of Crete. The city can be divided in two parts, the old town and the modern city.</p>
    <p>You will learn more about web layout and responsive web pages in a later chapter.</p>
  </div>
</div>
<div class="footer">
  <p>Footer Text</p>
</div>
</body>
</html>
```

```
* {
  box-sizing: border-box;
}
.header, .footer {
  background-color: grey;
  color: white;
  padding: 15px;
}
.column {
  float: left;
  padding: 15px;
}
.clearfix::after {
  content: "";
  clear: both;
  display: table;
}
.menu {
  width: 25%;
}
.content {
  width: 75%;
}
.menu ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
}
.menu li {
  padding: 8px;
  margin-bottom: 8px;
  background-color: #33b5e5;
  color: #ffffff;
}
.menu li:hover {
  background-color: #0099cc;
}
```

UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



Ejemplo Posicionamiento: float y clear

Chania

The Flight

The City

The Island

The Food

The City

Chania is the capital of the Chania region on the island of Crete. The city can be divided in two parts, the old town and the modern city.

You will learn more about web layout and responsive web pages in a later chapter.

Footer Text

UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



box-sizing

- ❑ Define cómo se calculan el ancho y la altura del elemento: si se tienen que incluir el relleno y los bordes o no.

```
box-sizing: content-box;
```

default

El **ancho** y la **altura** del elemento solo se aplican al **contenido** del elemento.

Por ejemplo, este elemento tiene

- `border-width: 12px`
- `padding: 30px`
- `width: 200px`

El *ancho total* es $24px + 60px + 200px = 284px$.

El contenido tiene el ancho definido. La caja se adapta a esas dimensiones.



UA 2.12: Introducción CSS: Posicionamiento y Desplazamiento



box-sizing

```
box-sizing: border-box;
```

El **ancho** y la **altura** del elemento se aplican a todas las partes del elemento: el **contenido**, el **relleno** y los **bordes**.

Por ejemplo, este elemento tiene

- `border-width: 12px`
- `padding: 30px`
- `width: 200px`

El *ancho total* es **200px**, no importa qué.

La caja tiene el ancho definido. El contenido se adapta a esas dimensiones, y termina siendo $200\text{px} - 60\text{px} - 24\text{px} = \mathbf{116\text{px}}$.



Más información sobre box-sizing: <http://cssreference.io/property/box-sizing/>
https://www.w3schools.com/cssref/css3_pr_box-sizing.asp



**Universidad
Europea**

LAUREATE INTERNATIONAL UNIVERSITIES

Madrid

Valencia

Canarias