



Maquetación Web

FUNDAMENTOS DE DESARROLLO WEB

Curso académico

2023 – 2024

Maquetación: Índice

1. Capas
2. Flujo HTML
3. Propiedad DISPLAY
 - Block
 - Inline
 - Inline-block
4. Propiedades FLOAT y CLEAR
 - Right
 - Left
5. Propiedad POSITION
 - Static
 - Absolute
 - Relative
 - Fixed



Maquetación: Capas

- Son partes o secciones de la página que podemos colocar en nuestra página web de 'forma independiente'.
- La etiqueta que define una capa es **<div>** (de división). Crea secciones o agrupa distintos contenidos.
- Sirve para definir bloques (como si se crearan cajas). Es un contenedor genérico para otras etiquetas.

```
<div>
```

```
    <p>Este párrafo está dentro de una capa</p>
```

```
</div>
```

Maquetación: Capas

- Las capas se utilizan para ‘maquetar’ la página web, es decir, darle una determinada estructura.
 - Las capas son elementos de bloque y pueden contener otras capas, tablas, imágenes, etc.
- Distintas implementaciones de los navegadores.
 - Es posible que el código sea ‘incompatible’.
 - Se recomienda el uso de hojas de estilo para dar formato.
- El elemento div es un contenedor genérico **sin un significado semántico en particular**, por lo tanto **div debería usarse como último recurso**, reservado únicamente para aquellos casos en los que elementos con mayor significado, como <main>, <header>, <footer>, <nav> o <article> no son aplicables.

Maquetación: Capas – Ejemplo

<div> anterior

<div> central

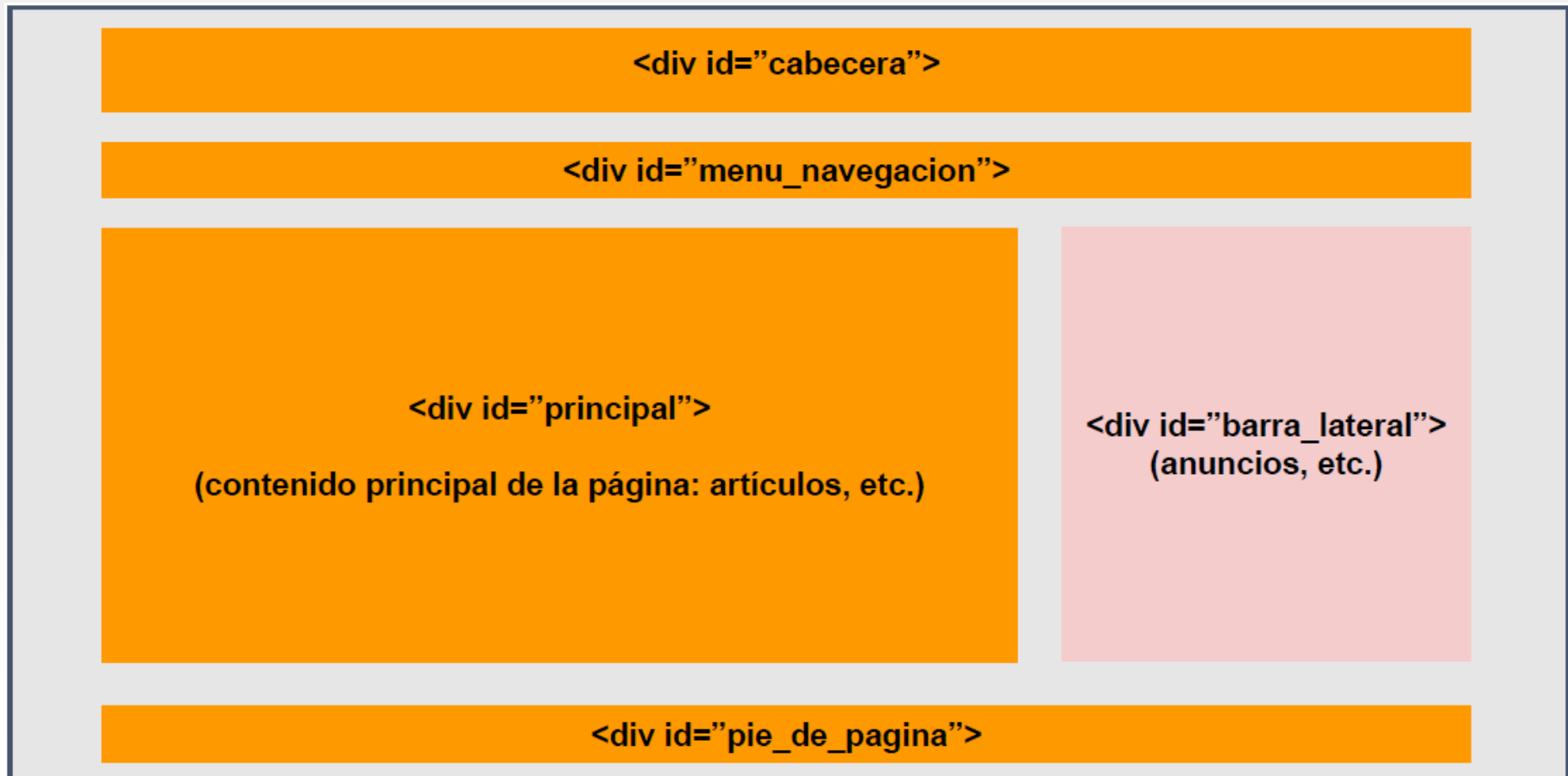
<div> posterior

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Divs</title>
    <link href="divs.css" rel="stylesheet" type="text/css">
  </head>
  <body>
    <div id="anterior"> &lt;div&gt; anterior </div>
    <div id="centro"> &lt;div&gt; central </div>
    <div id="posterior"> &lt;div&gt; posterior </div>
  </body>
</html>
```

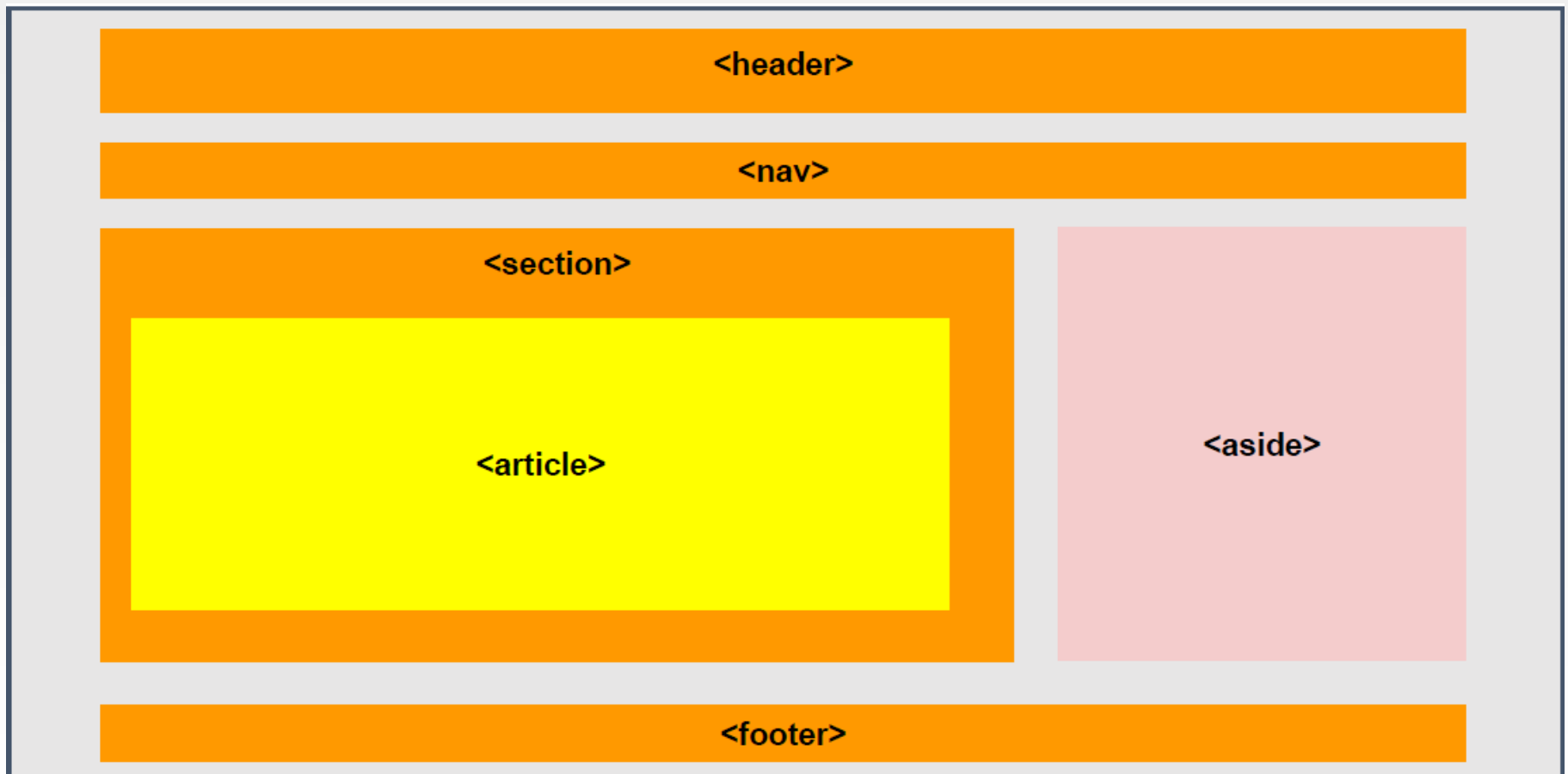
```
#centro {
  height: 200px;
  width: 300px;
  border: 5px solid #C00;
  margin: 30px;
  padding: 20px;
}

#anterior, #posterior {
  height: 75px;
  width: 100px;
  border: 2px solid #30F;
  margin: 10px;
  padding: 5px;
}
```

Maquetación: Capas en HTML4



Maquetación: Capas en HTML5



Maquetación: Capas

■ Header

- No confundir con el elemento **<head>**.
 - El elemento `<header>` va dentro del `<body>`.
 - La información contenida en el `<header>` se muestra al usuario.
- Útil para contener la siguiente información.
 - Títulos principales.
 - Logo.
 - Cuadros de búsqueda.
 - Etc.

Maquetación: **Capas**

■ **Nav**

- Se utiliza para establecer la barra de navegación (menú).
- Incluye los enlaces a las diferentes páginas que forman parte del sitio web o enlaces internos en caso de que se utilice una única página.
- No forma parte del <header>.

■ **Section**

- Incluye el contenido de la página web.
 - Dentro de una sección, la información se suele dividir en varios bloques para distribuir los contenidos. Para ello utilizaremos elementos <article>.
- Se inserta dentro del <body>, normalmente después del elemento <nav> (si lo hay) o del <header>.

Maquetación: Capas

■ Article

- Su función es mostrar el contenido relevante de nuestra página web.
- Los artículos sirven para dividir el contenido en partes que comparten características similares.
- Se pueden incluir uno o varios elementos `<article>` dentro de un elemento `<section>`.

■ Aside

- Sirve para incluir otros contenidos que no son los principales de la página web.
 - Publicidad.
 - Enlaces externos.
 - Calendario.
- Se inserta dentro del `<body>` y suele ir después de los elementos `<section>`.

Maquetación: **Capas**

■ **Footer**

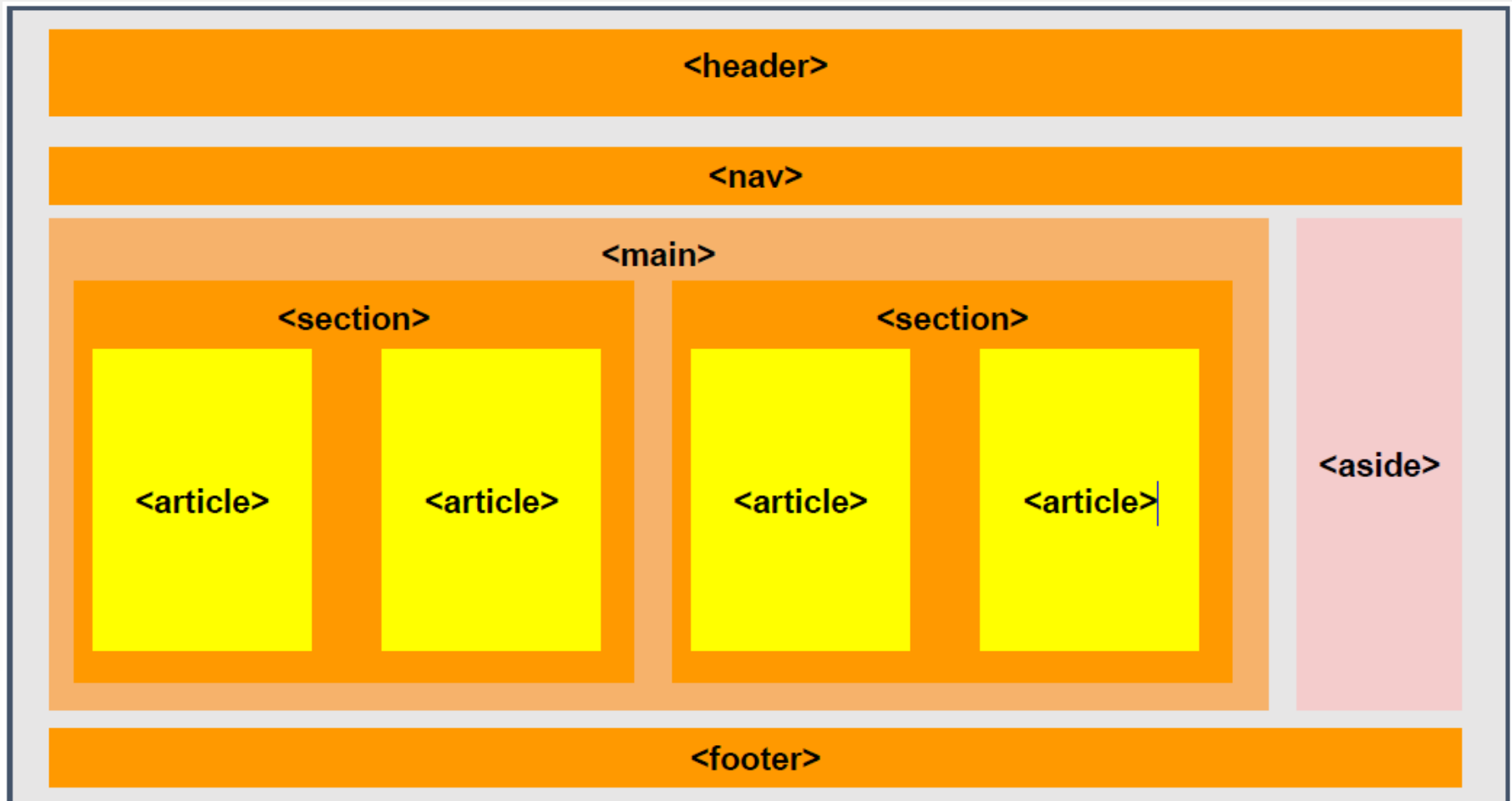
- Incluye información no relevante pero sí necesaria.
 - Autor.
 - Copyright.
 - Fecha.
 - Se puede utilizar también para mostrar otro menú con los enlaces principales.
- Se inserta justo antes del cierre del `<body>`.

Maquetación: **Capas**

■ **Main**

- Se puede utilizar para agrupar el contenido principal de una página.
 - No debe haber más de un elemento `<main>` por página.
 - No puede ser hijo de ninguno de los otros.
 - No debe contener información que se repita en diferentes páginas (logos, títulos principales, cuadros de búsqueda, menú de navegación, etc.).

Maquetación: Ejemplo HTML5



Maquetación: Ejemplo Estructura

```
<!DOCTYPE html>
<html>
  <head>
    <title>Ejemplo Estructura</title>
  </head>
  <body>
    <header></header>
    <nav></nav>
    <main>
      <section>
        <article></article>
        <article></article>
      </section>
      <section>
        <article></article>
        <article></article>
      </section>
    </main>
    <aside></aside>
    <footer></footer>
  </body>
</html>
```

Maquetación: Ejemplo

<header>

<nav>

<section>

<article>

<aside>

<footer>

Cabecera	THE TIMES	332,146 SUBSCRIBERS
Navegación	Business Money Sport Life Arts Puzzles Papers	Friday, August 2 London Max 26°C
Contenido	<p>Ashes: Aussies get Root and Bresnan</p>	<p>Follow reaction to The #Ashes with our live dashboard</p>
Home Office investigated over stop-and-search checks Sadie Gray Last updated at 3:49PM, August 2 2013 The Home Office is to be investigated by the UK's equality watchdog after random spot-checks for illegal immigrants were reportedly aimed only at people from ethnic minorities. The stop-and-search style checks, performed yesterday outside London railway stations, contributed to a total of 130 arrests in a series of immigration raids around the country. The Equality and Human Rights Commission will investigate the spot-checks and a controversial advertisement campaign in which posters bearing the words "illegal immigrants – go home or face arrest" are displayed on the side of vans. A spokesman said this afternoon: "The commission is writing today to the Home Office about these reported operations, confirming that it will be...".	Minimum 30 years jail for Daniel Pelka killer couple 	China 'building aircraft carrier' Satellite images of a shipyard near Shanghai have persuaded leading Western military experts that China may have begun building its first indigenous aircraft...
Destacados Fracking: drilling starts in Sussex as protests continue Berlusconi party relaunch despite looming house arrest New RBS chief: on £1m-a-year - but he won't get a bonus Zimbabwe: African observers give mixed verdict on election Hotel deaths: cyanide suicide theory for Edinburgh couple Terror alert: US to shut Mid East embassies	Stay connected by tablet, phone and web 	The Digital Pack

Maquetación: **Flujo HTML**

- Colocación de los elementos en la pantalla según su aparición en el código.
- Por defecto, los elementos se muestran por orden de aparición.
 - De arriba a abajo y de izquierda a derecha.
 - Teniendo en cuenta si el elemento es de línea o de bloque.
- Se puede “romper” este flujo mediante atributos CSS de los diferentes elementos que conforman la página.

Maquetación: **Propiedad DISPLAY**

- Valores típicos
 - block
 - inline
 - inline-block
- Podemos hacer que un elemento de bloque se muestre como si fuera un elemento de línea y viceversa.

Maquetación - **display: block**

■ Ejemplo

Elemento de bloque

Otro elemento de bloque

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>display block</title>
    <link href="display.css" rel="stylesheet" type="text/css">
  </head>
  <body>
    <p class="bloque">Elemento de bloque</p>
    <p class="bloque">Otro elemento de bloque</p>
  </body>
</html>
```

```
.bloque {
  padding: 10px;
  background: #00FFFF;
  display: block;
  text-align: center;
}
```

Maquetación - **display: inline**

■ Ejemplo

Elemento en línea

Otro elemento en línea

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>display inline</title>
    <link href="display.css" rel="stylesheet" type="text/css">
  </head>
  <body>
    <p class="linea">Elemento en línea</p>
    <p class="linea">Otro elemento en línea</p>
  </body>
</html>
```

```
.linea {
  margin: 10px;
  padding: 10px;
  background: #80FF00;
  display: inline;
}
```

Maquetación - **display: inline-block**

■ Ejemplo

Elemento con
comportamiento en
línea y en bloque

Otro elemento que se
comporta en línea y en
bloque

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>display inline-block</title>
    <link href="display.css" rel="stylesheet" type="text/css">
  </head>
  <body>
    <p class="lineabloque">Elemento con comportamiento en línea y en bloque</p>
    <p class="lineabloque">Otro elemento que se comporta en línea y en bloque</p>
  </body>
</html>
```

```
.lineabloque {
  padding: 10px;
  background: magenta;
  display: inline-block;
  width: 150px;
}
```

Maquetación: **Propiedades FLOAT y CLEAR**

■ Propiedad Float

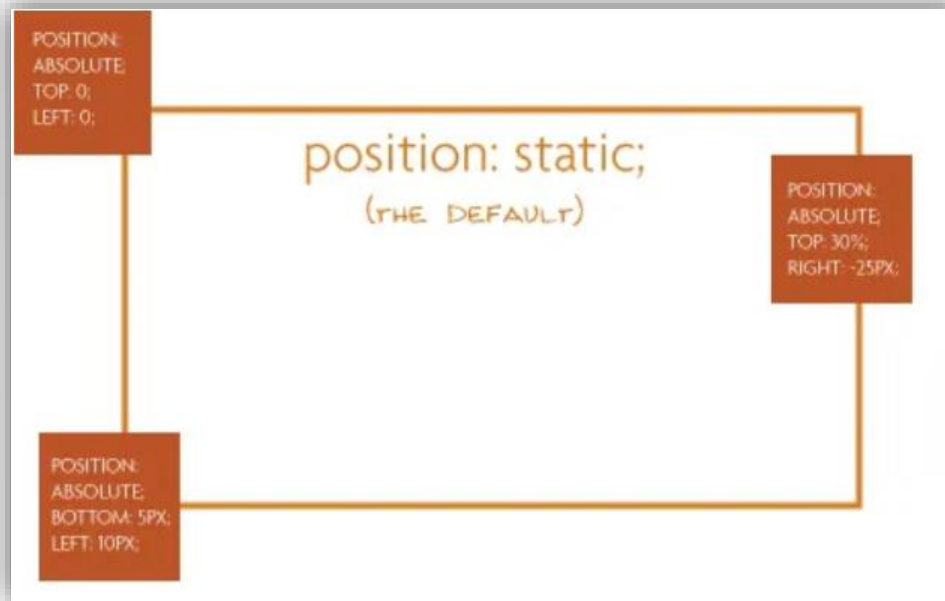
- Sirve para quitar al elemento del flujo normal de la página y colocarlo a la izquierda o derecha del resto del contenido de su elemento padre.
- Valores típicos
 - left
 - right
- El elemento se coloca, 'flota', en la posición indicada desplazando al resto de elementos.
- Se utiliza conjuntamente junto con la propiedad CLEAR.
 - Especifica que un elemento *NO* admite un flotante sobre su izquierda, su derecha o a ambos lados. También se puede usar en un elemento flotante para impedir que otros flotantes puedan quedar sobre alguno de sus lados.
- Ejemplos
 - https://www.w3schools.com/css/css_float.asp

Maquetación: **Propiedad POSITION**

- Permite establecer la posición de un elemento.
- Valores típicos
 - static
 - absolute
 - relative
 - fixed
- Tutorial en castellano y ejemplos
 - <https://desarrolloweb.com/articulos/atributo-position-css.html>
- Otros ejemplos
 - <https://uniwebsidad.com/libros/css/capitulo-5/posicionamiento?from=librosweb>

Maquetación - **position: static**

- Es el valor predeterminado del atributo y el posicionamiento normal de los elementos en la página.



Maquetación - **position: static**

■ Ejemplo



Maquetación - **position: static**

■ Ejemplo

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Posicionamiento static</title>
  </head>
  <body>
    <div style="position: static;
              background-color: #ff9;
              padding: 10px;
              width: 460px;">
      Esto es una capa con posicionamiento estático.</div>

    <div style="position: static;
              background-color: #f9f;
              padding: 10px;
              width: 460px;">
      Posicionamiento static, predeterminado.</div>

    <h1>Posicionamiento CSS Static</h1>

    <div style="background-color: #9ff;
              padding: 10px;
              width: 460px;">
      Posicionamiento static, aunque en este caso no se indicó
      el atributo position static, pues no hace falta.</div>
  </body>
</html>
```

Maquetación - **position: absolute**

- Permite a un elemento desplazarse respecto al origen de coordenadas del primer elemento contenedor o respecto a la esquina superior izquierda.
- Posiciona los elementos de manera absoluta, esto es, de manera definida por valores de los **atributos *top*, *left*, *bottom* y *right***, que indican la distancia con respecto a un punto. Las capas o elementos con posicionamiento absoluto **quedan fuera del flujo normal de HTML**.
- Los valores *top*, *left*, *bottom* y *right* indican la distancia con respecto al primer elemento contenedor que tenga un valor de *position* distinto de *static*.
- Si todos los contenedores donde esté la capa posicionada con *position: absolute* son *static*, simplemente se posiciona con respecto al lado superior de la página, para el caso de *top*, el inferior para *bottom*, del lado izquierdo para *left* o el derecho, en el caso de usar *right*.

Maquetación - **position: absolute**

■ Ejemplo



Maquetación - **position: absolute**

■ Ejemplo

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Pos.absolute</title>
    <link rel="stylesheet" href="estilo.css"/>
  </head>
  <body>
    <div id = "caja1">
      Esta capa tiene posicionamiento absoluto. <br><br>
      Hay que especificar top y left para colocarla con respecto
      a la esquina superior izquierda. z-index = 2 !!!
    </div>

    <div id = "caja2">
      Posicionamiento absoluto con z-index menor, la capa aparece
      por debajo de las otras que se superponen ya que tiene el
      z-index más bajo con z-index 1.</div>

    <div id="caja3">
      Posicionamiento absoluto con atributos bottom y right</div>

    <h1>Posicionamiento CSS Absoluto</h1>
  </body>
</html>
```

Maquetación - **position: absolute**

■ Ejemplo

```
#caja1{  
  position: absolute;  
  width: 300px;  
  height: 140px;  
  top: 100px;  
  left: 30px;  
  background-color: blue;  
  color: #ffff;  
  padding: 15px;  
  margin-left: 35px;  
  z-index: 2;  
}
```

```
#caja2{  
  position: absolute;  
  width: 620px;  
  height: 50px;  
  padding: 10px;  
  background-color: green;  
  color: #ffff;  
  top: 150px;  
  left: 10px;  
  z-index: 1;  
}
```

```
#caja3{  
  position: absolute;  
  width: 200px;  
  height: 80px;  
  padding: 10px;  
  background-color: red;  
  bottom: 100px;  
  right: 100px;  
  color: #ffff";  
}
```

Maquetación - **position: relative**

- Permite a un elemento desplazarse respecto a su posición normal.
- Atributos: *top*, *left*.
 - Indica que la capa **sí forma parte del flujo normal de elementos de la página**, por lo que su posición dependerá del lugar donde esté en el código y el flujo HTML.
 - Los valores *top* y *left* definen cuánto se desplaza el elemento respecto a lo que sería su posición normal.
 - El resto de elementos continúan en su posición lo que puede crear superposiciones.

Maquetación - **position: relative**

■ Ejemplo



Maquetación - **position: relative**

■ Ejemplo

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Pos.relative</title>
  </head>
  <body>
    <h1>Posicionamiento CSS Relative</h1>

    <div style="background-color: #606;
      color:#ffc;
      padding:10px;
      text-align: center;
      width: 500px;">
      Esto es una capa o div!!!!</div>
    <div style="position: relative;
      width: 300px;
      padding: 20px;
      background-color: #066;
      color:#ffc;
      top:135px;
      left: 50px;">
      Capa de posicionamiento relative.
      <br>Se tiene en cuenta esta capa para posicionar las siguientes.
    </div>
    <h2>¡¡¡Esto es un H2!!!</h2>
  </body>
</html>

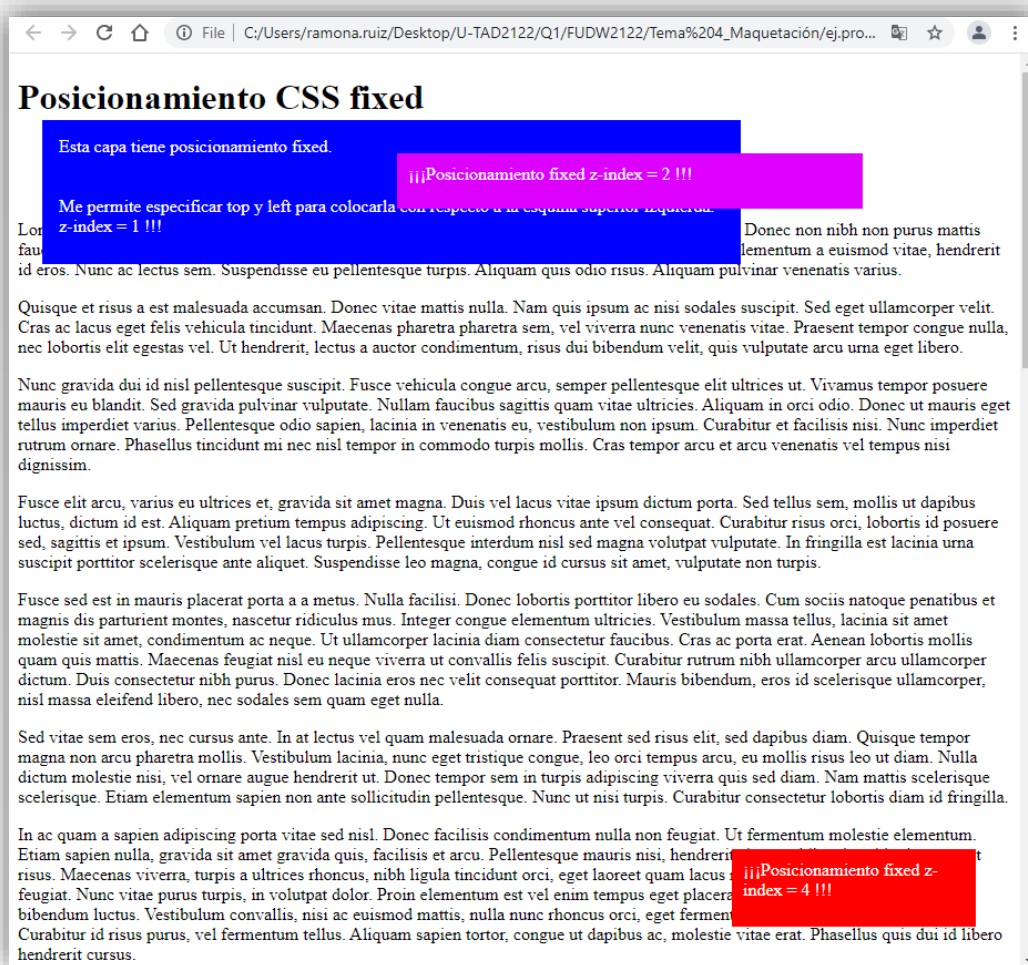
```


Maquetación - **position: fixed**

- La capa tiene **posicionamiento absoluto**, pero su posición final es **fija**, es decir, aunque se desplace el documento con las barras de desplazamiento del navegador, siempre aparecerá en la misma posición.
- El lugar donde se ‘anclará’ la capa siempre es relativo al <body>.
- Con los atributos *top* y *left*, se establece su posición con respecto a la esquina superior izquierda, y con *bottom* y *right* su posición será relativa a la esquina inferior derecha.

Maquetación - position: fixed

■ Ejemplo



Maquetación - position: fixed

■ Ejemplo

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Pos.absolute</title>
    <link rel="stylesheet" href="fixed.css" />
  </head>
  <body>
    <div id="caja1">
      Esta capa tiene posicionamiento fixed.<br><br><br>
      Me permite especificar top y left para colocarla con respecto
      a la esquina superior izquierda.  z-index = 1 !!!
    </div>
    <div id="caja2">¡¡¡Posicionamiento fixed z-index = 2 !!!</div>
    <h1>Posicionamiento CSS fixed</h1>
    <div id="caja3">¡¡¡Posicionamiento fixed z-index = 4 !!!</div>
    <br><br><br>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque bibendum porta neque eget
      ultricies. Donec non nibh non purus mattis faucibus. Pellentesque magna turpis, venenatis vel
      vehicula in, dapibus eget tellus. Integer leo enim, elementum a euismod vitae, hendrerit id eros.
      Nunc ac lectus sem. Suspendisse eu pellentesque turpis. Aliquam quis odio risus. Aliquam pulvinar
      venenatis varius.</p>
      . . .
    </body>
  </html>

```

Maquetación - **position: fixed**

■ Ejemplo

```
#caja1{  
  position: fixed;  
  width: 600px;  
  height: 100px;  
  top: 60px;  
  left: 30px;  
  background-color: blue;  
  color: #fff;  
  padding: 15px;  
  z-index: 1;  
}
```

```
#caja2{  
  position: fixed;  
  width: 400px;  
  height: 30px;  
  padding: 10px;  
  background-color: #d0f;  
  color: #ffff;  
  top: 90px;  
  left: 350px;  
  z-index: 2;  
}
```

```
#caja3{  
  position: fixed;  
  width: 200px;  
  height: 50px;  
  padding: 10px;  
  background-color: red;  
  color: #ffff;  
  bottom: 40px;  
  right: 40px;  
  z-index: 4  
}
```

Grado en Ingeniería del Software

Asignatura: Fundamentos de Desarrollo Web

Curso: 2023 – 2024

ramona.ruiz@u-tad.com