

Unidad 2.- Uso de estilos:

- a) Introducción a hojas de estilo en cascada (CSS, Cascading Style Sheet).
- b) Selectores: estilos en línea basados en etiquetas, en clases y en identificadores.
- c) Agrupación y anidamiento de selectores.
- d) Atributos.
- e) Superposición y precedencia de estilos.
- f) Elementos: colores de fondo, textos, enlaces, listas, tablas, cajas, posicionamiento, visibilidad, alineamiento, imágenes.
 - 1. Unidades.
 - 2. Colores.
 - 3. Estilo para textos.
 - 4. Desbordamiento.
 - 5. Alineación.
 - 6. Estilos para el fondo.
 - 7. Posicionamiento.
 - 8. Estilos para cajas.
 - 9. Colocar o Ubicación.
 - 10. Visibilidad.
 - 11. Estilos para listas.
 - 12. Estilos para columnas.
 - 13. Animaciones.
 - 14. Otros.
- g) Crear y vincular hojas de estilo.
- h) Crear y vincular hojas de estilo en cascada externa.
- i) Herramientas y test de verificación.

a) Introducción a hojas de estilo en cascada (CSS, Cascading Style Sheet).

CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML.

Una hoja de estilo es un conjunto de especificaciones que declara cómo deben mostrarse los componentes de una página web, es decir, cómo debe interpretar el navegador, a efectos de presentación visual, los distintos elementos presentes en una página web.

Cascada es la capacidad que proporcionan algunos lenguajes de hojas de estilo tales como CSS que permiten que las informaciones de estilo provenientes de varias fuentes puedan combinarse. Éstas podrían ser, por ejemplo, las guías de estilo de una empresa, los estilos comunes a un grupo de documentos, y los estilos específicos de un documento en particular. Al almacenarse independientemente, las hojas de estilo pueden reutilizarse, lo cual simplifica las tareas de diseño y hace más efectiva la utilización de la memoria caché de la red. La cascada define una secuencia ordenada de hojas de estilo en la que las reglas de las últimas hojas tienen una prioridad mayor que las de las primeras. No todos los lenguajes de hojas de estilo soportan la cascada.

El formato de cada elemento de una página web se indica mediante declaraciones de estilos denominadas reglas.

Una regla de CSS está compuesta por tres partes:

- ❖ selector
- ❖ propiedad
- ❖ valor

Una regla simple organiza los tres elementos anteriores de acuerdo a la siguiente estructura:

`selector { propiedad:valor; }`

Una regla puede contener más de una propiedad con sus respectivos valores, separados entre sí por punto y coma:

`selector{ propiedad-1:valor-1 [;propiedad-2:valor-2] ... ; }`

Una regla puede dar formato a más de un selector, separando éstos mediante comas:

`selector-1 [, selector-2]... { propiedad:valor; }`

Una regla puede dar formato a más de un selector, separando éstos mediante comas y puede contener más de una propiedad con sus respectivos valores, separados entre sí por punto y coma:

`selector-1 [, selector-2] ... { propiedad-1:valor-1 [;propiedad-2:valor-2] ... ; }`

Las reglas de las hojas de estilo se pueden indicar en tres sitios distintos:

- ❖ En la página web, dentro de una etiqueta HTML de un elemento. Lo que denominamos estilo interno.
- ❖ En la página web, dentro de la etiqueta HTML de estilo. Lo que llamamos estilo empotrado.
- ❖ En un archivo externo (*.css), lo que es la hoja de estilo. Estilo vinculado.

Para indicar un estilo interno pondremos:

```
<etiqueta style="estilos">
```

Para expresar un estilo empotrado usaremos:

```
<style type="text/css" media="medio" title="titulo" lang="idioma"  
dir="ltr|rtl">  
estilos  
</style>
```

Los atributos **media**, **title**, **dir** y **lang** son opcionales y estos dos últimos los tienen casi todas las etiquetas. Con **title** nos permite poner un título a la etiqueta, con **dir** la dirección en que aparecen las palabras y con **lang** se indica en qué idioma se va a utilizar.

Para utilizar un estilo externo u hoja de estilo deberemos tener la hoja de estilo en un fichero externo y luego en el documento HTML deberemos poner:

```
<link href="fichero-css" rel="stylesheet" type="text/css"  
media="medio">
```

El atributo **media** es opcional. El atributo **media** nos permite establecer para qué dispositivo está diseñada esa hoja de estilo. Se puede poner **and**, **or**, **not** para una lista. Los valores normales que puede tomar son:

screen para pantallas no paginadas de ordenador

tty para medios que utilicen una cuadricula de caracteres de ancho fijo (terminales).

tv para dispositivos de tipo televisión.

projection para proyectores

handheld para dispositivos de mano, pantalla pequeña.

print para material paginado, opaco y documentos que se muestran en una pantalla a modo de vista preliminar.

braille para dispositivos táctiles braille

aural para sintetizadores de voz.

all para todos los dispositivos.

además también se puede configurar a través de los siguientes elementos

width:valor; indica el ancho del área de visualización

min-width:valor; indica el ancho mínimo del área de visualización.

max-width:valor; indica el ancho máximo del área de visualización.

height:valor; indica el alto del área de visualización.

min-height:valor; indica el alto mínimo del área de visualización.

max-height:valor; indica el alto máximo del área de visualización.

device-width:valor; indica el ancho del dispositivo o papel.

min-device-width:valor; indica el ancho mínimo del dispositivo o papel.

max-device-width:valor; indica el ancho máximo del dispositivo o papel.

device-height:valor; indica el alto del dispositivo o papel.

min-device-height:valor; indica el alto mínimo del dispositivo o papel.

max-device-height:valor; indica el alto máximo del dispositivo o papel.

orientation:portrait|landscape; orientación del dispositivo vertical u horizontal.

aspect-ratio:valor; relación de aspecto del área de visualización, relación entre el valor máximo y mínimo. media='aspect-ratio:16/9'.

device-aspect-ratio:valor; relación de aspecto del dispositivo, relación entre el valor máximo y mínimo. media='device-aspect-ratio:16/9'

color:valor; indica el número de bits que se utiliza por color en la visualización.

min-color:valor; indica el número mínimo de bits que se utiliza por color en la visualización.

max-color:valor; indica el número máximo de bits que se utiliza por color en la visualización.

color-index:valor; indica el número de colores que se utilizan en la visualización.

min-color-index:valor; indica el número mínimo de colores que se utilizan en la visualización.

max-color-index:valor; indica el número máximo de colores que se utilizan en la visualización.

monochrome:valor; indica el número de bits que se utilizan por pixel en un monitor monocromo.

min-monochrome:valor; indica el número mínimo de bits que se utilizan por pixel en un monitor monocromo.

max-monochrome:valor; indica el número máximo de bits que se utilizan por pixel en un monitor monocromo.

resolution:valor; indica la densidad de los pixeles expresada en dpi o bien en dpcm(puntos por pulgada o puntos por cm)resolution: "print and (resolution:300dpi)"

min-resolution:valor; indica la densidad mínima de los pixeles expresada en dpi o bien en dpcm(puntos por pulgada o puntos por cm).

max-resolution:valor; indica la densidad máxima de los pixeles expresada en dpi o bien en dpcm(puntos por pulgada o puntos por cm).

scan:progressive | interlace; indica el método de barrido de la pantalla de televisión, progresivo o entrelazado(en el primero se mandan todos los datos a mostrar y en el segundo primero se envían las líneas impares y luego las líneas pares).

grid:0|1; indica si el dispositivo de salida es de rejilla (monitor) con 1 o de mapa de bits con 0.

Cuando queremos hacer un diseño de hojas de estilo en las cuales vamos a pensar en el diseño de móviles, como opción por defecto, entonces deberemos hacer que la hoja de estilos inicial se corresponda con las características generales y las específicas del diseño para el móvil. El resto de hojas de estilo deberán llevar el atributo media con el valor “(**min-width: tamaño**)”, las hojas de estilos se van a poner en orden ascendente (de menor a mayor) con respecto al tamaño indicado en el atributo media y en cada caso se deberá de poner en dichas hojas de estilo las características específicas para ese tamaño.

Ejemplo-02-3-001.html, tenemos varias hojas de estilo que se aplican en función de la resolución, en este orden se piensa en el diseño primero en móviles.

001	<link href="estilos/ejemplo-02-3-001.css" rel="stylesheet" type="text/css"/>
002	<link href="estilos/ejemplo-02-3-001-300.css" media="(min-width:300px)" rel="stylesheet" type="text/css"/>
003	<link href="estilos/ejemplo-02-3-001-600.css" media="(min-width:600px)" rel="stylesheet" type="text/css"/>
004	<link href="estilos/ejemplo-02-3-001-900.css" media="(min-width:900px)" rel="stylesheet" type="text/css"/>

Cuando queremos hacer un diseño de hojas de estilo en las cuales vamos a pensar en el diseño de ordenadores, como opción por defecto, entonces deberemos hacer que la hoja de estilos inicial se corresponda con las características generales y las específicas del diseño para el ordenador. El resto de hojas de estilo deberán llevar el atributo media con el valor “(**max-width: tamaño**)”, las hojas de estilos se van a poner en orden descendente (de mayor a menor) con respecto al tamaño indicado en el atributo media y en cada caso se deberá de poner en dichas hojas de estilo las características específicas para ese tamaño.

Ejemplo-02-3-002.html, tenemos varias hojas de estilo que se aplican en función de la resolución, en este orden se piensa en el diseño primero en ordenadores.

001	<link href="estilos/ejemplo-02-3-002.css" rel="stylesheet" type="text/css"/>
002	<link href="estilos/ejemplo-02-3-001-900.css" media="(max-width:900px)" rel="stylesheet" type="text/css"/>
003	<link href="estilos/ejemplo-02-3-001-600.css" media="(max-width:600px)" rel="stylesheet" type="text/css"/>
004	<link href="estilos/ejemplo-02-3-001-300.css" media="(max-width:300px)" rel="stylesheet" type="text/css"/>

Aunque hoy en día, para hacer los diseños anteriores, se utiliza una hoja de estilo.

Si vamos a hacer un diseño pensando en móviles, las características generales y las correspondientes al diseño del móvil se van a poner al inicio de la hoja de estilo y luego cuando queramos poner las características específicas de una resolución deberemos poner:

@media (min-width=tamaño) {estilos-para-esa-resolución}

Deberemos poner tantos @media como sean necesarios y los vamos a poner ordenados por el tamaño en modo ascendente, es decir de menor a mayor.

Ejemplo-02-3-004.css estilos que se aplican en función de la resolución, en este orden se piensa en el diseño primero en móviles

001	body{
002	font-size:8pt;
003	background-color:violet;
004	background-clip:content-box;
005	font-family:calibri;
006	font-size:8pt;
007	margin-top:2%;
008	margin-left:5%;
009	margin-right:3%;
010	padding: 1%;
011	color:white;
012	line-height:1.3rem;
013	}
014	img{
015	width:20%;
016	}
017	h1,h2,h3 {
018	color:#07BADA;
019	font-size:r2em;
020	}
021	@media(min-width:300px) {
022	body {
023	font-size:10pt;
024	background-color:blue;
025	}
026	}
027	}
028	@media(min-width:600px) {
029	body {
030	font-size:12pt;
031	background-color:red;
032	}
033	}
034	@media(min-width:900px) {
035	body {
036	font-size:14pt;
037	background-color:green;
038	}
039	}

También podemos utilizar:

@media (min-width=tamaño) {estilos-para-esa-resolución}

Deberemos poner tantos @media como sean necesarios y los vamos a poner ordenados por el tamaño en modo ascendente, es decir de menor a mayor.

Ejemplo-02-3-004-d.css estilos que se aplican en función de la resolución, en este orden se piensa en el diseño primero en móviles

```

001 body {
002     font-size:8pt;
003     background-color:violet;
004     background-clip:content-box;
005     font-family:calibri;
006     font-size:8pt;
007     margin-top:2%;
008     margin-left:5%;
009     margin-right:3%;
010     padding: 1%;
011     color:white;
012     line-height:1.3rem;
013 }
014 img {
015     width:20%;
016 }
017 h1, h2, h3 {
018     color:#07BADA;
019     font-size:2rem;
020 }
021 @media( min-width = 500px ) {
022     body {
023         font-size:10pt;
024         background-color:blue;
025     }
026 }
027 }
028 @media( min-width = 800px ) {
029     body {
030         font-size:12pt;
031         background-color:red;
032     }
033 }
034 @media( min-width = 1200px ) {
035     body {
036         font-size:14pt;
037         background-color:green;
038     }
039 }

```

Para hacer lo mismo que en el apartado anterior también vamos a poder utilizar

@media (width>tamaño) {estilos-para-esa-resolución}

O bien

@media (width>= tamaño) {estilos-para-esa-resolución}

Deberemos poner tantos @media como sean necesarios y los vamos a poner ordenados por el tamaño en modo ascendente, es decir de menor a mayor.

Ejemplo-02-3-004-b.css estilos que se aplican en función de la resolución, en este orden se piensa en el diseño primero en móviles

```

001 body{
002     font-size:8pt;
003     background-color:violet;
004     background-clip:content-box;
005     font-family:calibri;
006     font-size:8pt;
007     margin-top:2%;
008     margin-left:5%;
009     margin-right:3%;
010     padding: 1%;

```

```

011 color:white;
012 line-height:1.3rem;
013 }
014 img{
015 width:20%;
016 }
017 h1,h2,h3 {
018 color:#07BADA;
019 font-size:2rem;
020 }
021 @media( width> 500px ) {
022 body{
023 font-size:10pt;
024 background-color:blue;
025 }
026 }
027 @media( width> 800px ) {
028 body{
029 font-size:12pt;
030 background-color:red;
031 }
032 }
033 @media( width> 1200px ) {
034 body{
035 font-size:14pt;
036 background-color:green;
037 }
038 }

```

También podemos utilizar

@media (tamaño-1 >[=] width >[=] tamaño-2) {estilos-para-esa-resolución}

Ejemplo-02-3-004-c.css estilos que se aplican en función de la resolución, en este orden se piensa en el diseño primero en móviles

```

001 body {
002     font-size:8pt;
003     background-color:violet;
004     background-clip:content-box;
005     font-family:calibri;
006     font-size:8pt;
007     margin-top:2%;
008     margin-left:5%;
009     margin-right:3%;
010     padding: 1%;
011     color:white;
012     line-height:1.3rem;
013 }
014 img {
015 width:20%;
016 }
017 h1, h2, h3 {
018 color:#07BADA;
019 font-size:2rem;
020 }
021 @media( 800px > width >= 500px ) {
022 body {
023     font-size:10pt;
024     background-color:blue;
025 }
026 }
027 }
028 @media( 1200px >= width >= 800px ) {
029 body {
030     font-size:12pt;
031     background-color:red;
032 }
033 }
034 @media( width > 1200px ) {

```

035	body {
036	font-size:14pt;
037	background-color:green;
038	}
039	}

Si vamos a hacer un diseño pensando en ordenadores, las características generales y las correspondientes al diseño del ordenador, se van a poner al inicio de la hoja de estilo y luego cuando queramos poner las características específicas de una resolución deberemos poner:

@media (max-width=tamaño) {estilos-para-esa-resolución}

Deberemos poner tantos @media como sean necesarios y los vamos a poner ordenados por el tamaño en modo descendente, es decir de mayor a menor.

Ejemplo-02-3-003.css estilos que se aplican en función de la resolución, en este orden se piensa en el diseño primero en ordenadores.

001	body {
002	background-color:violet;
003	background-clip:content-box;
004	font-family:calibri;
005	font-size:14pt;
006	margin-top:2%;
007	margin-left:5%;
008	margin-right:3%;
009	padding: 1%;
010	color:white;
011	}
012	img{
013	width:20%;
014	}
015	h1,h2,h3 {
016	color:#07BADA;
017	font-size:2em;
018	}
019	@media(max-width:900px) {
020	body {
021	font-size:12pt;
022	background-color:green;
023	}
024	}
025	@media(max-width:600px) {
026	body {
027	font-size:10pt;
028	background-color:red;
029	}
030	}
031	@media(max-width:300px) {
032	body {
033	font-size:8pt;
034	background-color:blue;
035	color:white;
036	}
037	}

Lo mismo que hacemos en el caso anterior se puede hacer con:

@media (max-width=tamaño) {estilos-para-esa-resolución}

Deberemos poner tantos @media como sean necesarios y los vamos a poner ordenados por el tamaño en modo descendente, es decir de mayor a menor.

Ejemplo-02-3-003-d.css estilos que se aplican en función de la resolución, en este orden se piensa en el diseño primero en ordenadores

```

001 body {
002     background-color:violet;
003     background-clip:content-box;
004     font-family:calibri;
005     font-size:14pt;
006     margin-top:2%;
007     margin-left:5%;
008     margin-right:3%;
009     padding: 1%;
010     color:white;
011     line-height:1.3rem;
012 }
013 img {
014     width:20%;
015 }
016 h1, h2, h3 {
017     color:#07BADA;
018     font-size:2em;
019 }
020 @media( max-width = 1200px ) {
021     body {
022         font-size:12pt;
023         background-color:green;
024     }
025 }
026 @media( max-width=800px ) {
027     body {
028         font-size:10pt;
029         background-color:red;
030     }
031 }
032 @media( max-width=500px ) {
033     body {
034         font-size:8pt;
035         background-color:blue;
036         color:white;
037     }
038 }

```

Para hacer lo mismo que en el apartado anterior también vamos a poder utilizar

@media (width<tamaño) {estilos-para-esa-resolución}

O bien

@media (width<= tamaño) {estilos-para-esa-resolución}

Deberemos poner tantos @media como sean necesarios y los vamos a poner ordenados por el tamaño en modo ascendente, es decir de mayor a menor.

Ejemplo-02-3-003-b.css estilos que se aplican en función de la resolución, en este orden se piensa en el diseño primero en ordenadores.

```

001 body{
002     background-color:violet;
003     background-clip:content-box;
004     font-family:calibri;
005     font-size:14pt;
006     margin-top:2%;
007     margin-left:5%;
008     margin-right:3%;
009     padding: 1%;
010     color:white;
011 }
012 img{
013     width:20%;
014 }
015 h1,h2,h3 {
016     color:#07BADA;

```

```

017   font-size:2em;
018 }
019 @media( width< 1200px ) {
020   body{
021     font-size:12pt;
022     background-color:green;
023   }
024 }
025 @media( width< 800px ) {
026   body{
027     font-size:10pt;
028     background-color:red;
029   }
030 }
031 @media( width< 500px ) {
032   body{
033     font-size:8pt;
034     background-color:blue;
035     color:white;
036   }
037 }

```

O bien

@media (tamaño-1 <[=] width <[=] tamaño-2) {estilos-para-esa-resolución}

Deberemos poner tantos @media como sean necesarios y los vamos a poner ordenados por el tamaño en modo ascendente, es decir de mayor a menor.

Ejemplo-02-3-003-c.css estilos que se aplican en función de la resolución, en este orden se piensa en el diseño primero en ordenadores.

```

001 body {
002   background-color:violet;
003   background-clip:content-box;
004   font-family:calibri;
005   font-size:14pt;
006   margin-top:2%;
007   margin-left:5%;
008   margin-right:3%;
009   padding: 1%;
010   color:white;
011   line-height:1.3rem;
012 }
013 img {
014   width:20%;
015 }
016 h1, h2, h3 {
017   color:#07BADA;
018   font-size:2em;
019 }
020 @media( 600px <= width <= 900px ) {
021   body {
022     font-size:12pt;
023     background-color:green;
024   }
025 }
026 @media( 300px <= width < 600px ) {
027   body {
028     font-size:10pt;
029     background-color:red;
030   }
031 }
032 @media( width < 300px ) {
033   body {
034     font-size:8pt;
035     background-color:blue;
036     color:white;
037   }
038 }

```

Tanto si estamos utilizando diseño pensado en móviles o en ordenadores podemos utilizar **min-width** y **max-width** uniéndoles con el operador **and**.

Ejemplo-02-3-004-e.css estilos que se aplican en función de la resolución, en este orden se piensa en el diseño primero en móviles.

```

001 body {
002     font-size:8pt;
003     background-color:violet;
004     background-clip:content-box;
005     font-family:calibri;
006     font-size:8pt;
007     margin-top:2%;
008     margin-left:5%;
009     margin-right:3%;
010     padding: 1%;
011     color:white;
012     line-height:1.3rem;
013 }
014 img {
015     width:20%;
016 }
017 h1, h2, h3 {
018     color:#07BADA;
019     font-size:2rem;
020 }
021 @media( min-width:500px and max-width:800px ) {
022     body {
023         font-size:10pt;
024         background-color:blue;
025     }
026 }
027 @media( min-width:800px and max-width:1200px ) {
028     body {
029         font-size:12pt;
030         background-color:red;
031     }
032 }
033 @media( min-width:1200px ) {
034     body {
035         font-size:14pt;
036         background-color:green;
037     }
038 }
```

Ejemplo-02-3-003-e.css estilos que se aplican en función de la resolución, en este orden se piensa en el diseño primero en ordenadores.

```

001 body {
002     background-color:violet;
003     background-clip:content-box;
004     font-family:calibri;
005     font-size:14pt;
006     margin-top:2%;
007     margin-left:5%;
008     margin-right:3%;
009     padding: 1%;
010     color:white;
011     line-height:1.3rem;
012 }
013 img {
014     width:20%;
015 }
016 h1, h2, h3 {
017     color:#07BADA;
018     font-size:2em;
019 }
020 @media( max-width:900px and min-width:600px ) {
021     body {
022         font-size:12pt;
023         background-color:green;
024     }
025 }
```

026	@media(max-width:600px and min-width:300px) {
027	body {
028	font-size:10pt;
029	background-color:red;
030	}
031	}
032	@media(max-width:300px) {
033	body {
034	font-size:8pt;
035	background-color:blue;
036	color:white;
037	}
038	}

El atributo **rel** indica la relación entre el documento actual y la hoja de estilo, los valores que puede tomar son:

stylesheet dirección a una hoja de estilos para importar.

alternate para indicar que es una hoja de estilo alternativa.

next indica que el documento forma parte de una serie y que es el siguiente.

prev indica que el documento forma parte de una serie y que es el anterior.

autor enlace con el autor del documento.

help enlace a un documento de ayuda.

icon importa un icono para representar el documento.

license enlace a derechos de autor sobre el documento.

prefetch indica que el documento se debe almacenar en caché.

search enlaces a una herramienta de búsqueda para el documento.

se dejan de aplicar en html5

start indica que el documento forma parte de una serie y es el inicial.

contents documento que sirve como tabla de contenidos.

index documento que proporciona un índice

glossary documento que proporciona un glosario de términos.

copyright aviso de copyright del documento actual.

chapter documento que actúa como capítulo en un colección de documentos.

section documento que actúa como una sección en un colección de documentos.

subsection documento que actúa como una subsección en un colección de documentos.

appendix documento que actúa como un apéndice en un colección de documentos.

bookmark señal de lectura.

Para Internet Explorer podemos poner elementos opcionales para las diferentes versiones del navegador para lo cual pondremos en el head:

```
<!--[if condición]><link ..>|<style> ..</style>|<script> ..</script><[endif]-->
```

En donde la condición la vamos a expresar de la siguiente manera:

IE → para establecer que para cualquier versión de internet explorer.

IE número → para la versión de internet explorer indicada por número.

lt IE número → para las versiones de internet explorer anterior a la indicada por número.

lte IE número → para las versiones de internet explorer anteriores o igual a la indicada por número.

gt IE número → para las versiones de internet explorer posteriores a la indicada por número.

gte IE número → para las versiones de internet explorer posteriores o igual a la indicada por número.

En las condiciones también vamos a poder utilizar los operadores lógicos siguientes:

! → negación de la condición.

| → ó lógico, con que se cumpla una de las condiciones el resultado será verdadero.

& → y lógico, se deben cumplir ambas condiciones para que el resultado sea verdadero.

Vamos a ver algunos ejemplos.

001	<!--[if IE 8]><link href="css/ocho.css" rel="stylesheet" type="text/css"><! [endif]-->
002	<!--[if lte IE 6]><link href="css/seis.css" rel="stylesheet" type="text/css"><! [endif]-->
003	<!--[if gte IE 6 <e IE 8]><link href="css/entre.css" rel="stylesheet" type="text/css"><! [endif]-->
004	<!--[if !IE 8]><link href="css/no-ocho.css" rel="stylesheet" type="text/css"><! [endif]-->
005	<!--[if IE 10 IE 9]><link href="css/nueve.css" rel="stylesheet" type="text/css"><! [endif]-->

b) Selectores: estilos en línea basados en etiquetas, en clases y en identificadores.

Los selectores nos permiten indicar a que elemento HTML se le aplica el estilo correspondiente. Los selectores los vamos a utilizar cuando empleamos estilos empotrados y estilos vinculados. Así tendremos selectores que son:

- *
- etiquetas de HTML.
- identificadores.
- clases.
- mezclas del segundo con los dos siguientes

Los estilos con el selector * indica que ese estilo se va a aplicar a todos los elementos.

* { **estilo** }

Todos los elementos van a mostrar los caracteres en color azul. *ejemplo-02-001.css*.

001	* {color:blue;}
-----	-----------------

Los selectores de etiquetas nos sirven para aplicar estilo a todo el contenido de la etiqueta indicada.

etiqueta {estilo}

El contenido, el texto, de todos los párrafos div se verá de color verde. *ejemplo-02-002.css*.

001	div {color:green;}
-----	--------------------

Los selectores de identificadores nos sirven para aplicar los estilos a los elementos que tienen ese identificador.

estilo

etiqueta#identificador {estilo}
#identificador { estilo }

HTML

<etiqueta id="identificador">

El texto del párrafo div con identificador economía se muestra de color rojo y el texto del párrafo con identificador historia se muestra de color naranja. *ejemplo-02-003.css*.

001	div#economia {color:red;}
002	#historia {color:orange;}

Los selectores de clases nos permiten aplicar un estilo a todos los elementos que tienen esa clase definida. Si tenemos una lista de clases las clases deben estar separadas por blancos.

estilos

```
etiqueta.clase {estilo}  
.clase {estilo}
```

HTML

```
<etiqueta class="lista-clases">
```

El texto del párrafo, div, de la clase inicio se muestra de color oro y los párrafos de la clase ancestros se muestran de color plata. *ejemplo-02-004.css*.

001	div.inicio { color: gold; }
002	.ancestros {color:silver;}

Pseudoclasses que pueden funcionar como selectores

Para aplicar a la primera linea. Hoy se debe utilizar la correspondiente al doble dos puntos.

```
selector:first-line {estilo}  
:first-line {estilo}
```

```
selector::first-line {estilo}  
::first-line {estilo}
```

El texto de la primera línea de todos los párrafos se va a mostrar de color lima, el texto de la primera línea de los párrafos, p, se va a mostrar de color marrón, el texto de la primera línea del párrafo con identificador inicio se va a mostrar de color coral y el texto de la primera línea de todos los párrafos de la clase primeros se va a mostrar de color violeta. *ejemplo-02-005.css*.

001	::first-line{color:lime;}
002	p::first-line{color:brown;}
003	#inicio::first-line{color:coral;}
004	.primeros::first-line{color:violet;}

ejemplo-02-006.css

001	:first-line{color:lime;}
002	p:first-line{color:brown;}
003	#inicio:first-line{color:coral;}
004	.primeros:first-line{color:violet;}

Para aplicar a la primera letra de los párrafos. Hoy se debe utilizar la correspondiente al doble dos puntos.

```
selector:first-letter {estilo}  
:first-letter {estilo}
```

```
selector::first-letter {estilo}  
::first-letter {estilo}
```

La primera letra de todos los párrafos se va a mostrar de color violeta y la primera letras de los párrafos, p, se va a mostrar de color oro. *ejemplo-02-007.css*.

001	::first-letter{color:violet;}
002	p::first-letter{color:gold;}

ejemplo-02-008.css

001	:first-letter{color:violet;}
002	p:first-letter{color:gold;}

Para aplicar a cualquier enlace, se utiliza con extensiones.

```
a:any-link {estilo}
```

```
::any-link {estilo}
```

El color del texto de todos los enlaces va a ser rojo. *ejemplo-02-009.css*.

001	:any-link{color:red;}
002	:-moz-any-link{color:red;}
003	:-webkit-any-link{color:red;}

Para aplicar a todos los enlaces no visitados, si se incluye también la pseudocase anterior se aplicará el estilo del último que se pone y que será el último que se ejecuta.

```
a:link {estilo}
```

```
::link {estilo}
```

El color del texto de todos enlaces no visitados será verde. *ejemplo-02-010.css*.

001	:link{color:green;}
-----	---------------------

Para aplicar a todos los enlaces visitados

a:visited {estilo}**:visited {estilo}**

El color del texto de todos enlaces no visitados será oro y el color del texto de todos los enlaces visitados será violeta. *ejemplo-02-011.css*.

001	:link{color:gold;}
002	:visited{color:violet;}

Para aplicar a todos los enlaces activos, cuando se pulsa con el ratón y no se suelta, o bien a otros elementos, los indicados por el selector o bien a todos los elementos.

a:active {estilo}**selector:active{estilo}****:active {estilo}**

El color del texto de los enlaces no visitados será oro, el color del texto de los enlaces visitados será violeta, el color de los enlaces cuando hacemos click sobre ellos será marrón, el color del texto de la cabecera h2 cuando pulsamos sobre ella será rojo y el color del resto de elementos cuando pulsamos el ratón sobre ellos será azul. *ejemplo-02-12.css*.

001	:link{color:gold;}
002	:visited{color:violet;}
003	a:active{color:brown;}
004	h2:active{color:red;}
005	:active{color:blue;}

En los enlaces que llevan al mismo documento, el estilo se aplica al párrafo al cual se va cuando se pulsa el enlace.

a:target {estilo}**:target {estilo}**

El contenido de los párrafos, a los cuales se va mediante un hiperenlace, se va a mostrar el texto de color verde. *ejemplo-02-013.css*.

001	div:target{color:green;}
-----	--------------------------

Para aplicar cuando se sitúa el ratón sobre el elemento correspondiente

selector:hover{estilo}

Cuando nos situemos con el ratón sobre el párrafo div de identificador muerta y sobre el párrafo div de identificador linajes el color del texto será rojo. Cuando nos situemos sobre el elemento de identificador acueducto y cuando nos situemos sobre el elemento de identificador santiago el color del texto será azul. Cuando nos situemos sobre el elemento de la clase leyenda el color del texto será verde. *ejemplo-02-015.css*.

001	div#muerta:hover,div#linajes:hover{color:red;}
002	#acueducto:hover,#santiago:hover{color:blue;}
003	.leyenda:hover{color:green;}

Para aplicar a los elementos de tipo input cuando reciben el foco.

selector:focus {estilo}

Cuando las cajas de texto obtengan el foco el color de fondo será azul y el color del texto blanco. *ejemplo-02-016.css*.

001	:focus{
002	background-color:blue;
003	color:white;
004	}

Para aplicar a todos los selectores de ese idioma.

selector:lang(idioma) {estilo}

El texto definido con el lenguaje español se verá de color rojo, el texto definido con el lenguaje inglés se verá de color azul, el texto definido con el lenguaje alemán se verá de color violeta y el texto definido con el lenguaje francés se verá de color verde. *ejemplo-02-017.css*.

001	:lang(es){color:red;}
-----	-----------------------

002	:lang(en){color:blue;}
003	:lang(dt){color:violet;}
004	:lang(fr){color:green;}

Para insertar contenido antes de todos los elementos selectores o bien de todos. Hoy se debe utilizar la correspondiente al dobel dos puntos.

selector::before{estilo}
selector:::before{estilo}

Antes del párrafo div que está dentro de otro párrafo div y este a su vez dentro de otro párrafo div, le añadimos el literal "Leyenda de Segovia" con el texto en color blanco y el fondo de color azul. *ejemplo-02-018.css*.

001	div div div::before{
002	content:"Leyenda de Segovia";
003	background-color:blue;
004	color:white
005	}

Para aplicar para insertar contenido después del selector o de todos los elementos Hoy se debe utilizar la correspondiente al dobel dos puntos.

selector::after {estilo}
selector:::after {estilo}

Después del párrafo div que está dentro de otro párrafo div y este a su vez dentro de otro párrafo div, le añadimos el literal "Leyenda de Segovia" con el texto en color blanco y el fondo de color azul. *ejemplo-02-019.css*.

001	div div div::after{
002	content:"Leyenda de Segovia";
003	background-color:green;
004	color:red;
005	}

Para aplicar al primer hijo que es del selector indicado, cuidado con primer hijo de <body>

selector::first-child {estilo}

Al párrafo div que sea el primer hijo de otro elemento el color del texto va a ser rojo y el fondo de color verde. La cabecera H2 que sea el primer hijo de otro elemento el color del texto va a ser azul. *ejemplo-02-020.css*

001	div::first-child{
002	background-color:green;
003	color:red;
004	}
005	h2::first-child{
006	color:blue;
007	}

El elemento indicado que es el último hijo de otro elemento

selector::last-child {estilo}

El párrafo div que sea el último hijo de otro elemento va a tener como color de texto azul. *ejemplo-02-021.css*.

001	div:last-child{color:blue;}
-----	-----------------------------

El elemento indicado que es el hijo que ocupa la posición indicada dentro del parente

selector::nth-child(posición) {estilo}

El párrafo div que es el tercer hijo de otro elemento se va a ver con el texto en color rojo. *ejemplo-02-022.css*.

001	div:nth-child(3){color:red;}
-----	------------------------------

El elemento indicado que es el hijo y que ocupa la posición indicada, dentro del padre, empezando a contar por el final.

selector:`:nth-last-child(posición) {estilo}`

El párrafo div que es el cuarto hijo, empezando por el final, de otro elemento se va a ver con el texto en color verde. *ejemplo-02-023.css*.

```
001 div:nth-last-child(4){color:green;}
```

El elemento indicado si es el único hijo de su elemento padre

selector:`:only-child {estilo}`

El párrafo p que sea hijo único de otro elemento se va a mostrar con el texto en color marrón. *ejemplo-02-024.css*.

```
001 p:only-child{color:brown;}
```

Elemento que no tiene ningún hijo, ni texto, ni blancos <etiqueta></etiqueta>

selector:`:empty {estilo}`

El párrafo div que esté vacío se va a mostrar con el color de fondo azul, con un ancho de 200px y un alto de 300px. *ejemplo-02-025.css*. Para indicar el tamaño no se puede utilizar porcentajes.

```
001 div:empty{  
002     background-color:blue;  
003     width:200px;  
004     height:300px;  
005 }
```

Elemento que solo tiene blancos o caracteres asimilados al blanco, si está vacío también se incluye. Solo lo soporta Mozilla Firefox con una extensión: `-moz-only whitespace`.

selector:`:blank {estilo}`

El párrafo div que solo contenga espacios se va a mostrar con el color de fondo azul, con un ancho de 200px y un alto de 300px. *ejemplo-02-031.css*. Para indicar el tamaño no se puede utilizar porcentajes.

```
001 div:-moz-only whitespace{  
002     background-color:blue;  
003     width:200px;  
004     height:300px;  
005 }  
006 div:blank{  
007     background-color:blue;  
008     width:200px;  
009     height:300px;  
010 }
```

El primer elemento del tipo indicado por selector y que sea hijo

selector:`:first-of-type {estilo}`

El primer párrafo de tipo div de cualquier agrupación se va a mostrar con el fondo de color azul y los caracteres de color blanco. *ejemplo-02-026.css*

```
001 div:first-of-type{  
002     background-color:blue;  
003     color:white;  
004 }
```

El último elemento indicado y que es hijo

selector:`:last-of-type {estilo}`

El último párrafo div de cualquier agrupación se va a mostrar con color de fondo azul y color de los caracteres blancos. *ejemplo-02-027.css*.

```
001 div:last-of-type{  
002     background-color:blue;  
003     color:white;  
004 }
```

El elemento indicado que ocupa la posición indicada dentro de su padre.

selector:`:nth-of-type(posición) {estilo}`

El cuarto párrafo div de cualquier agrupación se va a mostrar con el fondo azul y el color de los caracteres blanco. *ejemplo-02-029.css*.

001	<code>div:nth-of-type(4) {</code>
002	<code>background-color:blue;</code>
003	<code>color:white;</code>
004	<code>}</code>

El elemento indicado que ocupa la posición indicada dentro de su padre, empezando a contar desde el final.

selector:`:nth-last-of-type(posición) {estilo}`

El tercer párrafo div, empezando por el final, de cualquier agrupación se va a mostrar con el fondo azul y el color de los caracteres en blanco. *ejemplo-02-028.css*.

001	<code>div:nth-last-of-type(3) {</code>
002	<code>background-color:blue;</code>
003	<code>color:white;</code>
004	<code>}</code>

El elemento indicado si el único elemento de ese tipo dentro de un padre

selector:`:only-of-type {estilo}`

El párrafo *p* que se encuentre dentro de un contenedor y que sea el único *p*, se va a mostrar con las letras en marrón. *ejemplo-02-032.css*.

001	<code>p:only-of-type{color:brown;}</code>
-----	---

Aplicado a todo el documento

selector:`:root {estilo}`

Todo el documento va a tener el fondo azul y los caracteres en color blanco. *ejemplo-02-033.css*.

001	<code>:root{</code>
002	<code>color:white;</code>
003	<code>background-color:blue;</code>
004	<code>}</code>

Aplicar a los que no sean de la lista de selectores.

selector:`:not(selector) {estilo}`

En el párrafo div con identificador general los div que no pertenezcan a la clase uno se muestran color de fondo azul y color de los caracteres blancos. En el párrafo div con identificador leyendas los div que no pertenezcan al identificador muerta se muestran con el color de fondo verde y el color de los caracteres gris. *ejemplo-02-034.css*.

001	<code>div#general div:not(.uno) {</code>
002	<code>color:white;</code>
003	<code>background-color:blue;</code>
004	<code>}</code>
005	<code>div#leyendas div:not(#muerta) {</code>
006	<code>color:gray;</code>
007	<code>background-color:green;</code>
008	<code>}</code>

Se pueden poner dos o más selectores: `:not(selector)` para que se apliquen a los que no sean ningunos de los selectores indicados.

En el párrafo div con identificador general los div que no pertenezcan a la clase uno ni a la clase dos se muestran color de fondo azul y color de los caracteres blanco. *ejemplo-02-034-2.css*

001	<code>div#general{</code>
002	<code>border:1px solid red;</code>
003	<code>}</code>

004	div#general div:not(.uno):not(.dos){
005	color:white;
006	background-color:blue;
007	}

Con una lista de selectores solo funciona en Safari, aquí tenemos un ejemplo.

En el párrafo div con identificador general los div, que no pertenezcan a la clase uno ni a la clase primero, se muestran con color de fondo azul y color de los caracteres blanco. En el párrafo div con identificador leyendas los div que no pertenezcan al identificador muerta ni al identificador santiago ni al identificador corpus se muestran con el color de fondo verde y el color de los caracteres es gris. *ejemplo-02-035.css*.

001	div#general div:not(.uno,.primero){
002	color:white;
003	background-color:blue;
004	}
005	div#leyendas div:not(#muerta,#santiago,#corpus){
006	color:gray;
007	background-color:green;
008	}

Aplicar a los que sean de la lista de selectores, tiene que tener todos los selectores. Se usa con extensiones, poniendo: **-moz-any** y **-webkit-any**. En la nueva especificación se usa **is** en lugar de matches, con lo cual se debe incluir lo anterior más **is**.

selector:matches(pseudoclase|selector){estilo}
selector:is(pseudoclase|selector){estilo}

Los elementos de la clase uno, se van a mostrar con el fondo en azul y los caracteres en blanco y el elemento de identificador santiago se va a mostrar con el fondo verde y los caracteres grises. *ejemplo-02-036.css*.

001	: -moz-any(.uno) {
002	color:white;
003	background-color:blue;
004	}
005	: -webkit-any(.uno) {
006	color:white;
007	background-color:blue;
008	}
009	: matches(.uno) {
010	color:white;
011	background-color:blue;
012	}
013	: -moz-any(#santiago) {
014	color:gray;
015	background-color:green;
016	}
017	: -webkit-any(#santiago) {
018	color:gray;
019	background-color:green;
020	}
021	: matches(#santiago) {
022	color:gray;
023	background-color:green;
024	}

Los elementos que están en un párrafo dividido la clase uno y de la clase primero se van a mostrar con el fondo en azul y los caracteres en blanco; y los elementos que están en un div con identificador leyendas con el identificador muerta o el identificador santiago o el identificador corpus se va a mostrar con el fondo verde y los caracteres grises. *ejemplo-02-037.css*.

001	div#general div:-moz-any(.uno,.primero){
002	color:white;
003	background-color:blue;
004	}
005	div#general div:-webkit-any(.uno,.primero){
006	color:white;
007	background-color:blue;
008	}

```

009 div#general div:matches(.uno,.primero) {
010     color:white;
011     background-color:blue;
012 }
013
014 div#leyendas div:-moz-any(#muerta,#santiago,#corpus) {
015     color:gray;
016     background-color:green;
017 }
018 div#leyendas div:-webkit-any(#muerta,#santiago,#corpus) {
019     color:gray;
020     background-color:green;
021 }
022 div#leyendas div:matches(#muerta,#santiago,#corpus) {
023     color:gray;
024     background-color:green;
025 }

```

Se aplica al texto seleccionado, solamente se puede aplicar estilo a color, background-color, background y text-shadow. Con extensiones para firefox.

selector::selection { estilo }
selector::-moz-selection { estilo }

Cuando se selecciona texto, éste se va a mostrar de color de fondo verde y color de los caracteres plata.
ejemplo-02-040.css.

```

001 ::-moz-selection{
002     background-color:green;
003     color:silver;
004 }
005 ::selection{
006     background-color:green;
007     color:silver;
008 }

```

Se aplica a los elementos activos

selector:enabled { estilo }

Los elementos activos se van a mostrar con el color de fondo verde y el color de los caracteres en gris. *ejemplo-02-040.css*.

```

001 :enabled{
002     background-color:green;
003     color:silver;
004 }

```

Se aplica a los elementos desactivados

selector:disabled { estilo }

Los elementos desactivados se van a mostrar con el color de fondo azul y el color de los caracteres en blanco. *ejemplo-02-040.css*.

```

001 :disabled{
002     background-color:blue;
003     color:white;
004 }

```

Se aplica a los elementos seleccionados, botones de selección y casillas de verificación.

selector:checked { estilo }

Los elementos que se seleccionan van a tener un tamaño de 25x25 pixeles. *ejemplo-02-041.css*.

```

001 :checked{
002     width:25px;
003     height:25px

```

004 }

Se aplica a los elementos cuando un elemento tiene un valor indeterminado

selector:indeterminate {estilo}
ejemplo-02-042.css.

001	:indeterminate{
002	background-color:blue;
003	color:white;
004	width:25px;
005	height:25px
006	}

Se aplica a los elementos cuando el elemento es la opción por defecto dentro de un grupo. Ejemplos lista.

selector:default {estilo}

Los elementos por defecto se van a mostrar con el color de fondo verde y el color de los caracteres en color plata. *ejemplo-02-043.css.*

001	:default{
002	background-color:green;
003	color:silver;
004	}

Se aplica a los elementos de entrada que tienen un valor valido. Después de validar el elemento, la validación se realiza por el navegador en función del tipo definido en la etiqueta input.

selector:valid {estilo}

Los elementos de tipo email cuando el valor sea blanco y correcto se muestra el color de fondo verde y el color de los caracteres en color plata. *ejemplo-02-044.css.*

001	input[type="email"]:valid{
002	background-color:green;
003	color:silver;
004	}

Se aplica a los elementos de entrada que tienen un valor no valido, después de validar el elemento, la validación se realiza por el navegador en función del tipo definido en la etiqueta input.

selector:invalid {estilo}

Los elementos de tipo email cuando el valor sea incorrecto se muestra el color de fondo amarillo y el color de los caracteres rojo. *ejemplo-02-044.css.*

001	input[type="email"]:invalid{
002	background-color:yellow;
003	color:red;
004	}

Se aplica a los elementos elemento de entrada, input, que tiene el atributo required.

selector:required {estilo}
ejemplo-02-045.css

001	:required{
002	background-color:green;
003	color:silver;
004	}

Se aplica a los elementos elemento de entrada, input, que no tiene el atributo required

selector:optional {estilo}

ejemplo-02-045.css

001	:optional{
002	background-color:yellow;
003	color:red;
004	}

Se aplica a los elementos de valor numérico que está dentro del rango de valores en un input type="number".

selector:in-range {estilo}
ejemplo-02-047.css

001	:in-range{
002	background-color:green;
003	color:silver;
004	}

Se aplica a los elementos de valor numérico que está fuera del rango de valores en un input type="number".

selector:out-of-range {estilo}
ejemplo-02-047.css

001	:out-of-range{
002	background-color:yellow;
003	color:red;
004	}

Se aplica a los elementos que no puede modificar el usuario, tiene el atributo readonly. En Mozilla Firefox se aplica con extensiones.

selector:read-only {estilo}
ejemplo-02-046.css

001	:-moz-read-only{
002	background-color:yellow;
003	color:red;
004	}
005	:read-only{
006	background-color:yellow;
007	color:red;
008	}

Se aplica a los elementos que puede modificar el usuario, no tiene el atributo readonly. En Mozilla Firefox se aplica con extensiones.

selector:read-write {estilo}
ejemplo-02-046.css

001	:-moz-read-write{
002	background-color:green;
003	color:silver;
004	}
005	:read-write{
006	background-color:green;
007	color:silver;
008	}

Se aplica a los elementos elemento de entrada que tiene un texto de ejemplo, se aplica al texto de ejemplo. Placeholder. Se suele usar con prefijos y además cambia el nombre de los elementos así vamos a tener. Se usa con extensiones:-moz-placeholder, -ms-input-placeholder. Se aplica a input de los tipos text, number, date y time.

selector:placeholder-show {estilo}
ejemplo-02-050.css

```

001 : -moz-placeholder{
002     background-color: yellow;
003     color: red;
004     text-shadow: 2px solid blue;
005 }
006 : placeholder-shown{
007     background-color: yellow;
008     color: red;
009     text-shadow: 2px solid blue;
010 }

```

Se aplica a los elementos en función de la dirección de texto, si está definido el idioma se debe corresponder con la dirección del texto.

selector:dir(ltr|rtl) {estilo}

```

001 : dir(ltr) {color: green; }
002 : dir(rtl) {color: red; }

```

c) Agrupación y anidamiento de selectores.

selector descendiente, se aplica el estilo al selector 2 que se encuentra dentro del selector1

selector-1 selector-2 {estilo}

ejemplo-02-051.css

```
001 div h2 {color: blue; }
```

selectores hijo

selector-1>selector-2 {estilo}

Se aplican los estilos al selector 2 que es hijo directo del selector 1

ejemplo-02-052.css

```
001 div>span{color: blue; }
```

selectores nietos o más

selector-1 * selector-2 {estilo}

se aplica el estilo al selector 2 que es un selector nieto o más del selector 1.

ejemplo-02-053.css

```
001 div* span {color: blue; }
```

selector hermano-adyacente, se aplica al selector 2 que está inmediatamente a continuación del selector 1.

selector-1+selector-2 {estilo}

ejemplo-02-054.css

```
001 h2 +div{color: yellow; }
```

selector predecesor, se aplica el estilo a todos los selector2 que está precedido del selector1 en el orden en que se muestra o encuentra en el documento HTML.

selector-1~selector-2 {estilo}

ejemplo-02-055.css

```
001 h2~div{color: red; }
```

d) Atributos.

Vamos a poner como seleccionar elementos de HTML en función de sus atributos, se puede poner con el selector o bien sin el selector

.se aplica al selector que tiene el atributo indicado

selector[atributo]{estilo}

ejemplo-02-056.css

```
001 | div[class]{color:salmon; }
```

se aplica al selector que tiene el atributo indicado con el valor indicado exactamente.

selector[atributo="valor"]{estilo}

ejemplo-02-057.css

```
001 | span[class="uno"]{color:salmon; }  
002 | [class="dos"]{color:brown; }
```

se aplica al selector que tiene el atributo indicado con el valor indicado, no se realiza distinción entre mayúsculas y minúsculas.

selector[atributo="valor" i]{estilo}

ejemplo-02-058.css

```
001 | span[class="UNO"]{color:salmon; }  
002 | [class="DOS" i]{color:brown; }
```

se aplica al selector que tiene el atributo indicado y al menos uno de sus valores es el indicado

selector[atributo~="valor"]{estilo}

ejemplo-02-059.css

```
001 | [class~="tres"]{color:green; }
```

se aplica al selector que tiene el atributo indicado y cuyo primer valor es el indicado (que es una palabra completa), tiene varios valores separados por guiones. Solo se suele utilizar para el atributo lang. Cuidad con poner ese atributo en la etiqueta html.

selector[atributo|= "valor"]{estilo}

ejemplo-02-060.css

```
001 | [lang|=es]{color:green; }
```

se aplica al selector que tiene el atributo indicado y que el valor empieza por el valor indicado.

selector[atributo^="valor"]{estilo}

ejemplo-02-061.css

```
001 | [class^="un"]{color:green; }
```

se aplica al selector que tiene el atributo indicado y que el valor termina por el valor indicado

selector[atributo\$="valor"]{estilo}

ejemplo-02-062.css

```
001 | [class$="o"]{color:green; }
```

se aplica al selector que tiene el atributo indicado y que el valor contiene el valor indicado.

selector[atributo *="valor"] {estilo}

ejemplo-02-063.css

```
001 [class*="o"]{color:green; }
```

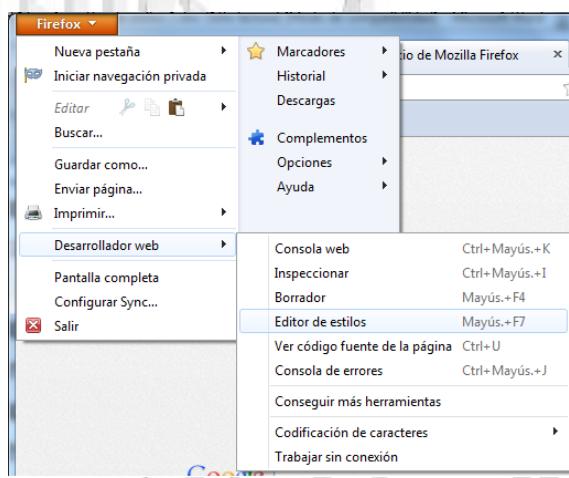
e) Superposición y precedencia de estilos.

Vamos a decir que existen tres tipos de estilos

- estilos del navegador, son aquellos estilos definidos por el propio navegador.
- estilos del usuario, son aquellos que define el usuario y que aplica a todas las páginas que visita con su navegador.
- estilos del diseñador, son aquellos definidos por el diseñador del sitio web.

El orden en que se ejecutan es el indicado anteriormente, con lo cual las últimas machacan a las anteriores.

Para crear estilos de usuario deberemos ir Firefox → Desarrollo web → Editor de estilos.



nos sale la siguiente ventana



Desde aquí podemos ver los estilos definidos por el navegador y así mismo podemos definir un nuevo estilo, pulsando sobre el botón **Nuevo**.

Dentro de los tres tipos de especificación de estilos tienen las siguientes prioridades:

1. Estilo interno.
2. Estilo empotrado.
3. Estilo vinculado.

Vamos a ver como organiza CSS las prioridades entre hojas de estilo que pudieran entrar en conflictos. CSS estipula cinco niveles de prioridad, de tal modo que las reglas que se correspondan con un nivel superior se impondrán sobre aquellas que pertenezcan a uno inferior. Estos niveles de prioridad son, ordenados de menor a mayor, los siguientes:

- Si no se aplican propiedades a un selector, se usará el valor heredado. Si no hubiera un valor heredado, se usará el valor por defecto.
- Las declaraciones marcadas con **!important** prevalecen sobre aquellas que no estén marcadas. Se aplica a las características que queremos aplicar.

ejemplo-02-3-81.css. Las declaraciones con important se van a aplicar, con la máxima prioridad.

001	#leyendas* div {
002	font-family:Calibri;
003	font-size:14pt;
004	background-color:blue;
005	color:white;
006	}
007	#acueducto{
008	font-family:"Comic Sans MS";
009	font-size:15pt;
010	background-color:green !important;
011	}
012	#muerta{
013	font-family:"Cambria";
014	font-size:16pt;
015	background-color:brown;
016	}
017	.sego{
018	font-family:"Times New Roman";
019	font-size:13pt;
020	background-color:red !important;
021	}

- Todas las hojas de estilo prevalecen sobre los valores por defecto del navegador. Las hojas de estilo del diseñador prevalecen sobre las del navegante. Las reglas incrustadas se imponen sobre las hojas vinculadas que, a su vez, tienen más peso que las externas.
- Los selectores de identificación ID se imponen ante los selectores de clase CLASS, que son más importantes que los selectores contextuales los cuales, a su vez, se imponen sobre los selectores normales.
- Las reglas de una hoja que sean posteriores a otras con las que entren en conflicto prevalecerán sobre estas.

Para el cálculo de la especificidad (prioridad) de un selector usaremos

1. se cuenta el número de atributos ID en el selector (= a)
2. se cuenta el número de los otros atributos y pseudo-clases en el selector (= b)
3. se cuenta el número de nombres de elementos en el selector (= c)
4. se ignoran los pseudo-elementos.

La concatenación de los tres números a-b-c (en un sistema numérico de base mayor) proporciona la especificidad.

Algunos ejemplos:

```
*      {} /* a=0 b=0 c=0 -> especificidad = 0 */
LI     {} /* a=0 b=0 c=1 -> especificidad = 1 */
UL LI   {} /* a=0 b=0 c=2 -> especificidad = 2 */
UL OL+LI {} /* a=0 b=0 c=3 -> especificidad = 3 */
H1 + *[REL=up]{} /* a=0 b=1 c=1 -> especificidad = 11 */
UL OL LI.red {} /* a=0 b=1 c=3 -> especificidad = 13 */
LI.red.level {} /* a=0 b=2 c=1 -> especificidad = 21 */
#x34y    {} /* a=1 b=0 c=0 -> especificidad = 100 */
```

Se aplica el de valor mayor.

ejemplo-02-3-80.css

001	#leyendas* div {
002	font-family:Calibri;
003	font-size:14pt;
004	background-color:blue;
005	color:white;
006	}
007	#acueducto{
008	font-family:"Comic Sans MS";
009	font-size:15pt;
010	background-color:green;

011	}
012	#muerta{
013	font-family:"Cambria";
014	font-size:16pt;
015	background-color:brown;
016	}
017	.sego{
018	font-family:"Times New Roman";
019	font-size:13pt;
020	background-color:red;
021	}

Si calculamos la prioridad en el caso anterior tendremos:

#leyendas* div calculo de prioridades a=1;b=0;c=1

#acueducto calculo de prioridades a=1;b=0;c=0

#muerta calculo de prioridades a=1;b=0;c=0

.sego calculo de prioridades a=0;b=1;c=0

Se aplicará la primera, la correspondiente a #leyendas* div, ya que es la que tiene mayor prioridad.

ejemplo-02-3-82.css

001	#leyendas* div {
002	font-family:Calibri;
003	font-size:14pt;
004	background-color:blue;
005	color:white;
006	}
007	#leyendas* #acueducto{
008	font-family:"Comic Sans MS";
009	font-size:15pt;
010	background-color:green;
011	}
012	#leyendas* #muerta{
013	font-family:"Cambria";
014	font-size:16pt;
015	background-color:brown;
016	}
017	#leyendas* .sego{
018	font-family:"Times New Roman";
019	font-size:13pt;
020	background-color:red;
021	}

Si calculamos la prioridad en el caso anterior tendremos:

#leyendas* div calculo de prioridades a=1;b=0;c=1

#leyendas* #acueducto calculo de prioridades a=2;b=0;c=0

#leyendas* #muerta calculo de prioridades a=2;b=0;c=0

#leyendas* .sego calculo de prioridades a=1;b=1;c=0

Se aplicarán la segunda (#leyendas* #acueducto) y la tercera (#leyendas* #muerta), luego la cuarta (#leyendas* .sego) y por último la primera (#leyendas* div), ya que este es el orden de prioridades.

- f) Elementos: colores de fondo, textos, enlaces, listas, tablas, cajas, posicionamiento, visibilidad, alineamiento, imágenes.

Vamos a ir viendo los diferentes elementos que nos encontramos a la hora de declarar estilos:

Para poner comentarios

/ comentario */*

1.- Unidades

Para expresar valores vamos a utilizar

valor% valores con porcentajes

valor unidad

calc(*expresión*)

Las unidades que se utilizan para medir o de longitud son:

- ❖ **em**: relativo a la altura de la fuente del elemento. Multiplica por el valor que le precede la altura de la fuente.
- ❖ **rem**: relativo a la altura definida para el tipo de letra del elemento principal, del body:
- ❖ **ex**: aproximadamente la altura de la letra x. Multiplicado por el valor que le precede.
- ❖ **px**: pixeles.
- ❖ **pt**: puntos. 1pt=1/72 in
- ❖ **in**: pulgadas 1in=2.54cm
- ❖ **cm**: centímetros
- ❖ **mm**: milímetros
- ❖ **q**: cuarta parte del milímetro.
- ❖ **pc**: picas 1pc=12pt
- ❖ **ch**: la anchura del carácter "0" (cero).
- ❖ **vw**: viewportwidth: 1% del ancho de visualización o de la ventana.
- ❖ **vh**: viewportheight: 1% del alto de visualización o de la ventana.
- ❖ **vmin**: valor menor entre vw y vh.
- ❖ **vmax**: valor mayor entre vw y vh.

Para ángulos

- ❖ **deg**: degrees o grados
- ❖ **grad**: gradienes
- ❖ **rad**:radianes.
- ❖ **turn**: una vuelta.

Para tiempo

- ❖ **ms**:milisegundos
- ❖ **s**:segundos

Para expresar frecuencia

- ❖ **Hz**:Hertz o hercios
- ❖ **kHz**: kilo Hertz

Para expresar resolución

- ❖ **dpi**: puntos por pulgada
- ❖ **dpcm**: puntos por centímetro.
- ❖ **dppx**: puntos por pixel.

2.- Colores

Para expresar los colores pondremos

nombre-color

blue

#rrggbb siendo rr, gg y bb dos dígitos en hexadecimal, que expresan los valores rr para proporción del color rojo, gg para proporción del color verde, bb para proporción del color azul

#A34598

#rgb siendo r, g, y b un dígito en hexadecimal. Cada dígito en hexadecimal se duplica para formar un valor para el formato anterior. Así tenemos que #F83 equivale a #FF8833.

#E45

#rrggbbooo siendo rr, gg, bb y oo dos dígitos en hexadecimal, que expresan los valores rr para proporción del color rojo, gg para proporción del color verde, bb para proporción del color azul y oo la opacidad o transparencia en donde 00 indica transparente y ff opaco.

#0000FF12

#rgbo siendo r, g, b y o un dígito en hexadecimal. Cada dígito en hexadecimal se duplica para formar un valor para el formato anterior. Así tenemos que #F835 equivale a #FF883355

#3F49

rgb(rojo,verde,azul) valores entre 0 y 255.

rgb(12, 120, 236)

rgb(%rojo,%verde,%azul)

rgb(12%, 85%, 23%)

rgba(rojo, verde, azul, transparencia)

Rojo, ver y azul son valores entre 0 y 255. Transparencia valor entre 0(trasparente) y 1 (opaco).

rgba(12, 120, 236, 0.3)

rgba(12%, 85%, 23%, 0.4)

hsl(matiz, saturación, luminosidad)

El matriz es la rueda de colores en la cual el 0 o 360 es el rojo, 120 es el verde y 240 es el azul. La saturación es un porcentaje que indica la intensidad del color el 0% es gris y el 100% es el propio color. La luminosidad es un porcentaje que en donde 0% es blanco y 100% es negro.

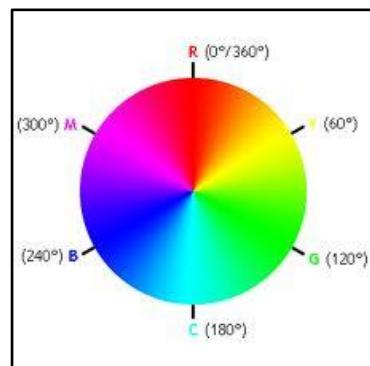
hsl(145, 75%, 35%)

hsla(matiz, saturación, luminosidad, transparencia)

El matriz es la rueda de colores en la cual el 0 o 360 es el rojo, 120 es el verde y 240 es el azul. La saturación es un porcentaje que indica la intensidad del color el 0% es gris y el 100% es el propio color. La luminosidad es un porcentaje que en donde 0% es blanco y 100% es negro. Transparencia valor entre 0(trasparente) y 1 (opaco).

hsla(145, 75%, 35%, 0.80)

Rueda colores



3.- Estilo para textos

Para las letras o fuentes tenemos los siguientes estilos:

Tipo de letra o fuente

font-family:*lista-tipos-letras;*

estilo de fuente o tipo de letra que se va a aplicar como 'times', 'courier', 'arial'. Familias de fuentes genéricas 'serif', 'sans-serif', 'cursive', 'fantasy', 'monospace'. Se aplica la primera fuente que hay en la lista y que está en el cliente.

ejemplo-02-065.css

```

001 .uno{
002     font-family: Calibri,"Times New Roman", "Courier New";
003     color:blue;
004 }
005 .dos{
006     font-family: "Comic Sans MS", Cambria,Arial;
007     color:green;
008 }
009 span {
010     font-family:Century, "Comic Sans MS", Calibri;
011     color:red;
012 }
013 #santiago,#corpus,#linajes{
014     font-family:"Courier New", Arial, Cambria;
015     color:brown;
016 }
```

Tamaño de fuente

font-size:*tamaño | xx-small | x-small | small | medium | large | x-large |
xx-large | smaller | larger;*

ejemplo-02-066.css

```

001 .uno{
002     font-family: Calibri,"Times New Roman", "Courier New";
003     font-size:14pt;
004     color:blue;
005 }
006 .dos{
007     font-family: "Comic Sans MS", Cambria,Arial;
008     font-size:12px;
009     color:green;
010 }
011 span {
012     font-family:Century, "Comic Sans MS", Calibri;
013     font-size:1.2em;
014     color:red;
015 }
016 #santiago,#corpus,#linajes{
017     font-family:"Courier New", Arial, Cambria;
018     font-size:1.5rem;
019     color:brown;
020 }
021 h2 { font-size:1.1cm; }
022 h1 {font-size:0.8in; }
023 .undecimo{
024     font-size:1.3pc;
025     color:orange;
026 }
```

ejemplo-02-067.css

```

001 .uno{
002     font-family: Calibri,"Times New Roman", "Courier New";
003     font-size:xx-small;
004     color:blue;
005 }
006 .dos{
007     font-family: "Comic Sans MS", Cambria,Arial;
008     font-size:x-samll;
```

```

009     color:green;
010 }
011 span {
012     font-family:Century, "Comic Sans MS", Calibri;
013     font-size:small;
014     color:red;
015 }
016 #santiago,#corpus,#linajes{
017     font-family:"Courier New", Arial, Cambria;
018     font-size:medium;
019     color:brown;
020 }
021 h2 {font-size:x-large; }
022 h1 {font-size:xx-large; }
023 .undecimo{
024     font-size:large;
025     color:orange;
026 }
027 #principe{
028     font-size:larger;
029     color:gold;
030 }
031 #parral{
032     font-size:smaller;
033     color:violet;
034 }

```

Tamaño máximo de la fuente

font-max-size:valor

Establece el tamaño máximo de la fuente

Tamaño mínimo de la fuente

font-min-size: valor

Establece el tamaño mínimo de la

Intensidad de la fuente

**font-weight: normal | bold | bolder | lighter | 100 | 200 | 300 | 400 | 500 | 600 |
700 | 800 | 900**

Indica el grado de intensidad de la fuente normal, negrilla. Con número a partir de 600 es la negrilla.

ejemplo-02-068.css

```

001 .uno{
002     font-family: Calibri,"Times New Roman", "Courier New";
003     font-size:xx-small;
004     color:blue;
005     font-weight:bold;
006 }
007 .dos{
008     font-family: "Comic Sans MS", Cambria,Arial;
009     font-size:x-samll;
010     color:green;
011     font-weight:bolder;
012 }
013 span {
014     font-family:Century, "Comic Sans MS", Calibri;
015     font-size:small;
016     color:red;
017     font-weight:lighter;
018 }
019 #santiago{
020     font-family:"Courier New", Arial, Cambria;
021     font-size:medium;
022     color:brown;

```

```

023     font-weight:100;
024 }
025 #corpus{
026     font-weight:200;
027     color:cyan;
028 }
029 #linajes{
030     font-weight:300;
031     color:magenta;
032 }
033 h2 {
034     font-size:x-large;
035     font-weight:800;
036 }
037 h1 {
038     font-size:xx-large;
039     font-weight:900;
040 }
041 .undecimo{
042     font-size:large;
043     color:orange;
044     font-weight:400;
045 }
046 #principe{
047     font-size:larger;
048     color:gold;
049     font-weight:500;
050 }
051 #parral{
052     font-size:smaller;
053     color:violet;
054     font-weight:600;
055 }
056 #cordon{
057     font-weight:700;
058     color:orange;
059 }
060 #acueducto{font-weight:bold; }
061 #acueducto.undecimo{font-weight:normal; }

```

Estilo que se aplica a la fuente.

font-style: normal | italic | oblique | oblique | grados;

Indica el estilo que se aplica las letras, que pueden ser: normal u oblicua o cursiva. Con oblique grados se indica los grados de inclinación de las letras, debe ser un valor comprendido entre -90deg y 90deg, deberemos tener en cuenta que si los grados son negativos la inclinación se realiza hacia el lado izquierdo en lugar de hacia el lado derecho.

ejemplo-02-069.css

```

001 #muerta{font-style:italic; }
002 #muerta.tres{font-style:normal; }
003 .uno{font-style:oblique; }

```

ejemlo-02-3-16.css

```

001 body{font-size:16pt}
002 @font-face {
003     font-family: "Source Sans Pro";
004     src: url("SourceSansPro-Regular.otf") format("opentype"), url("SourceSansPro-Regular.ttf") format("opentype");
005 }
006
007 #d1{
008     font-family: "Source Sans Pro";
009     font-style:oblique 45deg;
010 }
011 #d2{
012     font-family: "Source Sans Pro";
013     font-style:oblique;
014 }
015 #d3{

```

```

016     font-family: "Source Sans Pro";
017     font-style:oblique -45deg;
018 }

```

Decoración de texto

text-decoration-line:none | underline | overline | line-through

normal, subrayado, línea sobre el texto, una línea a través del texto. Se utiliza con las extensiones

Ejemplo-02-071-B.css

```

001 #muerta{
002     -moz-text-decoration-line:underline;
003     -webkit-text-decoration-line:underline;
004     text-decoration-line:underline;
005 }
006 #muerta.tres{
007     -moz-text-decoration-line:normal;
008     -webkit-text-decoration-line:normal;
009     text-decoration-line:normal;
010 }
011 #corpus{
012     -moz-text-decoration-line:overline;
013     -webkit-text-decoration-line:overline;
014     text-decoration-line:overline;
015 }
016 #principe{
017     -moz-text-decoration-line:line-through;
018     -webkit-text-decoration-line:line-through;
019     text-decoration-line:line-through;
020 }
021 #linajes{
022     -moz-text-decoration-line:overline underline;
023     -webkit-text-decoration-line:overline underline;
024     text-decoration-line:overline underline;
025 }
026 div.uno{
027     -moz-text-decoration-line:line-through underline;
028     -webkit-text-decoration-line:line-through underline;
029     text-decoration-line:line-through underline;
030 }
031 #parral{
032     -moz-text-decoration-line:line-through overline;
033     -webkit-text-decoration-line:line-through overline;
034     text-decoration-line:line-through overline;
035 }
036 #acueducto{
037     -moz-text-decoration-line:line-through overline underline;
038     -webkit-text-decoration-line:line-through overline underline;
039     text-decoration-line:line-through overline underline;
040 }

```

Para establecer el color de la decoración, se utiliza con extensiones

text-decoration-color:color

Ejemplo-02-72-B.css.

```

001 #muerta{
002     -moz-text-decoration-line:underline;
003     -moz-text-decoration-color:blue;
004
005     -webkit-text-decoration-line:underline;
006     -webkit-text-decoration-color:blue;
007
008     text-decoration-line:underline;
009     text-decoration-color:blue;
010 }
011
012 #corpus{
013     -moz-text-decoration-line:overline;
014     -moz-text-decoration-color:green;

```

```

015
016     -webkit-text-decoration-line:overline;
017     -webkit-text-decoration-color:green;
018
019     text-decoration-line:overline;
020     text-decoration-color:green;
021 }
022 #principe{
023     -moz-text-decoration-line:line-through;
024     -moz-text-decoration-color:red;
025
026     -webkit-text-decoration-line:line-through;
027     -webkit-text-decoration-color:red;
028
029     text-decoration-line:line-through;
030     text-decoration-color:red;
031 }

```

Para indicar el tipo de línea que se va a aplicar tenemos. Se usa con extensiones.

text-decoration-style:solid | double | dotted | dashed | wavy

Línea continua, línea doble, punteado, guiones, dientes de sierra

ejemplo-02-073-B

```

001 #muerta{
002     -moz-text-decoration-line:underline;
003     -moz-text-decoration-color:blue;
004     -moz-text-decoration-style:double;
005
006     -webkit-text-decoration-line:underline;
007     -webkit-text-decoration-color:blue;
008     -webkit-text-decoration-style:double;
009     text-decoration-line:underline;
010     text-decoration-color:blue;
011     text-decoration-style:double;
012 }
013 #corpus{
014     -moz-text-decoration-line:overline;
015     -moz-text-decoration-color:green;
016     -moz-text-decoration-style:solid;
017
018     -webkit-text-decoration-line:overline;
019     -webkit-text-decoration-color:green;
020     -webkit-text-decoration-style:solid;
021     text-decoration-line:overline;
022     text-decoration-color:green;
023     text-decoration-style:solid;
024 }
025 #principe{
026     -moz-text-decoration-line:line-through;
027     -moz-text-decoration-color:red;
028     -moz-text-decoration-style:dotted;
029
030     -webkit-text-decoration-line:line-through;
031     -webkit-text-decoration-color:red;
032     -webkit-text-decoration-style:dotted;
033     text-decoration-line:line-through;
034     text-decoration-color:red;
035     text-decoration-style:dotted;
036 }
037 #acueducto{
038     -moz-text-decoration-line:underline;
039     -moz-text-decoration-color:red;
040     -moz-text-decoration-style:dashed;
041
042     -webkit-text-decoration-line:underline;
043     -webkit-text-decoration-color:red;
044     -webkit-text-decoration-style:dashed;
045     text-decoration-line:underline;
046     text-decoration-color:red;
047     text-decoration-style:dashed;

```

```

048 }
049 #linajes{
050   -moz-text-decoration-line:underline;
051   -moz-text-decoration-color:red;
052   -moz-text-decoration-style:wavy;
053
054   -webkit-text-decoration-line:underline;
055   -webkit-text-decoration-color:red;
056   -webkit-text-decoration-style: wavy;
057   text-decoration-line:underline;
058   text-decoration-color:red;
059   text-decoration-style:wavy;
060 }

```

text-decoration:none | line style color

se aplican las características anteriores.

ejemplo-02-74.css

```

001 #muerta{text-decoration:underline double blue; }
002 #acueducto{text-decoration:overline solid green; }
003 #linajes{text-decoration:line-through wavy red; }
004 #santiago{text-decoration:underline overline dotted yellow; }
005 #salto{text-decoration:underline line-through dashed orange; }
006 #principe{text-decoration:overline line-through solid brown; }
007 #cordon{text-decoration:underline overline line-through double violet; }

```

Para indicar como se muestran letras tenemos

**font-variant-caps: normal | small-caps | all-small-caps | petite-caps | unicase |
titling-caps | all-petite-caps**

Nos indica que las letras se muestren de manera normal,

Con **small-caps**: las letras minúsculas se muestran como mayúsculas, pero con un tamaño inferior al de las mayúsculas.

Con **all-small-caps**: todas las letras en mayúsculas, pero con un tamaño algo inferior al normal.

Con **petite-caps**: permite visualizar caracteres pequeños.

Con **all-petite-caps**: todas las letras en pequeñas mayúsculas.

Con **unicase**: letras minúsculas normales y mayúsculas un poco más pequeñas que las normales.

Con **titling-caps**: permite la visualización de capitales de titulación

ejemplo-02-087.css

```

001 #muerta{ font-variant-caps:small-caps; }
002 #muerta.tres{ font-variant-caps:normal; }
003 #acueducto{ font-variant-caps:all-small-caps; }
004 #linajes{ font-variant-caps:petite-caps; }
005 #santiago{ font-variant-caps:all-petite-caps; }
006 #salto{ font-variant-caps:unicase; }
007 #principe{ font-variant-caps:titling-caps; }

```

Para tener texto en formato subíndice o exponencial

font-variant-position: normal | sub | super

Con **sub** el texto aparece en formato subíndice

Con **super** el texto aparece en formato exponencial

ejemplo-023--010.css

001	#inf{font-variant-position:sub;}
002	#arr{font-variant-position:super;}

Indica como visualizar valores numéricos

font-variant-ligatures: normal | ordinal | slashed-zero | ining-nums | oldstyle-nums |
 diagonal-fractions | stacked-fractions | proportional-nums |
 tabular-nums

Para que se visualice estas características el tipo de letra debe estar en formato opentype o bien se tiene que activar ciertas características con **font-feature-settings**.

Con **normal** se muestra de forma normal.

Con **ordinal** se muestra los números como ordinales, siempre que lleven un postfijo que indique el ordinal. Si ponemos 1st se va a mostrar como 1st. Se debe activar **ordn**.

Con **slashed-zero** se van a mostrar los ceros con una diagonal o bien con un punto en medio en función del navegador. Se debe activar **zero**.

Con **ining-nums** se va a mostrar. Se debe activar **lnum**.

Con **oldstyle-nums** se va a mostrar. Se debe activar **onum**.

Con **diagonal-fractions** cuando tenemos dos números separados por la barra '/' se va a mostrar una fracción de los números, si tenemos 1 / 2 se va a mostrar ½. Se debe activar **frac**.

Con **stacked-fractions** cuando tenemos dos números separados por la barra '/' se va a mostar una fracción con barra horizontal. Se debe activar **afrc**. De momento no funciona.

Con **proportional-nums** se va a mostrar. Se debe activar **pnum**.

Con **tabular-nums** se va a mostrar. Se debe activar **tnum**.

ejemplo-02-3-012.css

001	body{font-size:16pt}
002	@font-face {
003	font-family: "Source Sans Pro";
004	font-style: normal;
005	font-weight: 400;
006	src: url("SourceSansPro-Regular.otf") format("opentype"), url("SourceSansPro-Regular.ttf") format("opentype");
007	}
008	#d1{
009	font-family: "Source Sans Pro";
010	font-variant-numeric:ordinal;
011	
012	}
013	#d1A{
014	font-family: "Calibri";
015	font-feature-settings: "ordn" on;
016	font-variant-numeric:ordinal;
017	}
018	#d2{
019	font-family: "Source Sans Pro";
020	font-variant-numeric:diagonal-fractions;
021	}
022	#d3{
023	font-family: "Calibri";
024	font-feature-settings:"frac" on;
025	font-variant-numeric:diagonal-fractions;
026	}
027	#d4{
028	font-family: "Source Sans Pro";
029	font-variant-numeric:slashed-zero;
030	}
031	#d5{
032	font-family: Calibri;
033	font-feature-settings: 'zero' on;
034	font-variant-numeric:slashed-zero;
035	}
036	#d7{

```

037     font-family: "Source Sans Pro";
038     font-variant-numeric:lining-nums;
039 }
040 #d8{
041     font-family: "Source Sans Pro";
042     font-variant-numeric:oldstyle-nums;
043 }
044 #d9{
045     font-family: "Calibri";
046     font-feature-settings:"lnum" on;
047     font-variant-numeric:lining-nums;
048 }
049 #d10{
050     font-family: "Calibri";
051     font-feature-settings:"onum" on;
052     font-variant-numeric:oldstyle-nums;
053 }
054 #d12{
055     font-family: "Source Sans Pro";
056     font-variant-numeric:stacked-fractions;
057 }
058 #d13{
059     font-family: "Calibri";
060     font-feature-settings:"afrc" on;
061     font-variant-numeric:stacked-fractions;
062 }
063 #d14{
064     font-family: "Source Sans Pro";
065     font-variant-numeric:proportional-nums;
066 }
067 #d15{
068     font-family: "Source Sans Pro";
069     font-variant-numeric:tabular-nums;
070 }
071 #d16{
072     font-family: "Calibri";
073     font-feature-settings:"pnum" on;
074     font-variant-numeric:proportional-nums;
075 }
076 #d17{
077     font-family: "Calibri";
078     font-feature-settings:"tnum" on;
079     font-variant-numeric:tabular-nums;
080 }

```

Para activar ciertas características del texto vamos a poner
font-feature-settings:"valor"on|1

Deberemos tener en cuenta que al activar estas acciones la visualización de las mismas va a depender del tipo de letra y el navegador utilizado.

Los valores que puede tomar son:

- **liga**: standard ligatures
- **dlig**: discretionaryligatures
- **onum**: oldstyle figures
- **lnum**: lining figures
- **tnum**: tabular figures
- **zero**: slashedzero
- **frac**: fractions
- **sups**: los caracteres se van a mostrar como subíndice. No funciona.
- **subs**: los caracteres se van a mostrar como superíndice.
- **smcp**: las letras se van a mostrar en mayúsculas.
- **c2sc**: las letras se van a mostrar en minúsculas.
- **case**: case-sensitiveforms

- **hlig:** historicalligatures
- **calt:** contextual alternates
- **swsh:** swashes
- **hist:** historicalforms
- **ss**:** stylistic sets
- **kern:** kerning
- **locl:** localizedforms
- **rlig:** requiredligatures
- **medi:** medial forms
- **init:** initialforms
- **isol:** isolatedforms
- **fina:** final forms
- **mark:** mark
- **mkmk:** mark-to-mark positioning

Ejemplo-02-3-015.css

```
001 body{font-size:16pt}
002 @font-face {
003   font-family: "Source Sans Pro";
004   src: url("SourceSansPro-Regular.otf") format("opentype"), url("SourceSansPro-
Regular.ttf") format("opentype");
005 }
006
007 #d1{
008   font-family: "Source Sans Pro";
009   font-feature-settings: "sups" on;
010 }
011 #d2{
012   font-family: "Source Sans Pro";
013   font-feature-settings: "subs" on;
014 }
015 #d3{
016   font-family: "Source Sans Pro";
017   font-feature-settings:"smcp" on;
018 }
019 #d4{
020   font-family: "Source Sans Pro";
021   font-feature-settings:"c2sc" on;
022 }
023 #d6{
024   font-family: "Source Sans Pro";
025   font-feature-settings:"case" on;
026 }
027 #d7{
028   font-family: "Source Sans Pro";
029   font-feature-settings:"kern" on;
030 }
031 #d8{
032   font-family: "Calibri";
033   font-feature-settings: "sups" on;
034 }
035 #d9{
036   font-family: "Calibri";
037   font-feature-settings: "subs" on;
038 }
039 #dA{
040   font-family: "Calibri";
041   font-feature-settings:"smcp" on;
042 }
043 #dB{
044   font-family: "Calibri";
045   font-feature-settings:"c2sc" on;
046 }
047 #dC{
048   font-family: "Calibri";
049   font-feature-settings:"case" on;
050 }
```

```

051 #dd{
052     font-family: "Calibri";
053     font-feature-settings:"kern" on;
054 }

```

Versalitas, las letras minúsculas similares a las mayúsculas, pero de menor tamaño.
font-variant: normal | small-caps | all-small-caps | petite-caps | unicase | titling-caps | all-petite-caps | sub | super

ejemplo-02-077.css

```

001 #muerta{font-variant:small-caps; }
002 #muerta.tres{font-variant:normal; }

```

Transformaciones del texto.

text-transform: capitalize | uppercase | lowercase | none | full-width

letra capital en todas las palabras, mayúsculas, minúsculas, ninguna. El último valor no funciona.

Ejemplo-02-078.css

```

001 #muerta{text-transform:capitalize; }
002 #muerta.tres{text-transform:none; }
003 #acueducto{text-transform:uppercase; }
004 #linajes{text-transform:lowercase; }
005 #santiago{text-transform:full-width; }

```

Altura de la línea

line-height: altura-línea | none

ejemplo-02-079.css

```

001 #muerta{
002     font-size:16pt;
003     line-height:2em;
004 }
005 #acueducto{ line-height:1.5cm; }
006 #linajes{ line-height:0.7in; }
007 #santiago{ line-height:1.5rem; }
008 #salto{ line-height:32pt; }
009 #principe{ line-height:38px; }

```

Fuente en general

**font: v-font-style v-font-variant v-font-weight v-font-stretch
 v-font-size / v-line-height v-font-family | caption | icon | menu | message-box |
 small-caption | status-bar;**

ejemplo-02-080.css

```

001 #muerta{ font:1.2em/2em Calibri; }
002 #acueducto{font: normal bold 1.3rem 'Comic Sans MS'; }
003 #linajes{font:caption; }
004 #santiago{font:icon; }
005 #salto{font:menu; }
006 #principe{font:message-box; }
007 #cordón{font:small-caption; }
008 #parral{font:status-bar; }

```

Estable la relación entre los tamaños de la X mayúscula y la x minúscula. Si aumenta su relación aumenta el tamaño de toda la fuente.

Solo funciona en firefox

font-size-adjust: valor | none

ejemplo-02-081.css

```

001 #muerta{ font-size-adjust:1.75; }
002 #muerta.tres{font-size-adjust:none; }

```

```
003 #acueducto{ font-size-adjust:0.65; }
```

Color del texto

color:color-caracteres;

ejemplo-02-082.css

001	#muerta{color:coral; }
002	#acueducto{ color:#7812AB; }
003	#linajes{ color:#37C; }
004	#santiago{color:rgb(52,123,245); }
005	#salto{color:rgba(52,123,245,0.6); }
006	#principe{color:rgb(15%,20%,10%);}
007	#cordon{color:hsl(305,50%,75%);}
008	#parral{color:hsla(305,50%,75%,0.4); }

Para indicar que hacer cuando el contenedor es menor que el contenido

text-overflow:clip | ellipsis | initial | cadena

Con **clip** e **initial** no se muestra, con **ellipsis** se ponen puntos suspensivos y con **cadena** se pone la cadena en sustitución del texto, solo en firefox. Es necesario establecer un ancho y aplicarles **white-space: nowrap;** y **overflow:hidden;**

ejemplo-02-083.css

001	#muerta p {
002	width:50%;
003	white-space: nowrap;
004	overflow:hidden;
005	-o-text-overflow:ellipsis;
006	text-overflow:ellipsis;
007	}
008	#acueducto p{
009	width:60%;
010	white-space: nowrap;
011	overflow:hidden;
012	-o-text-overflow:clip;
013	text-overflow:clip;
014	}
015	#linajesdiv{
016	width:75%;
017	white-space: nowrap;
018	overflow:hidden;
019	text-overflow:' continua ...';
020	}

Texto con sombra

text-shadow:desplazamiento-horizontal desplazamiento-vertical difuminado color

Por orden de aparición, los valores que se indican en text-shadow son:

- ❖ Desplazamiento horizontal de la sombra: La sombra de un elemento suele estar un poco desplazada con respecto al elemento que la produce y su posición será en función del ángulo con el que llegue la luz. Valor positivo sombra a la derecha. Valor negativo sombra a la izquierda.
- ❖ Desplazamiento vertical de la sombra: El segundo valor que colocamos en el atributo text-shadow es el desplazamiento vertical de la sombra con respecto a la posición del elemento que la produce. Valores positivos indican que la sombra aparecerá hacia abajo del elemento. Valores negativos harán que la sombra aparezca desplazada un poco hacia arriba.
- ❖ Difuminado: El tercer valor indica cuánto queremos que esté difuminado el borde de la sombra. Si el difuminado fuera cero o negativo, querría decir que la sombra no tiene ningún difuminado y aparece totalmente definida

- ❖ Color de la sombra: El último atributo que se indica en el atributo text-shadow es el color de la sombra.

Se pueden poner varias sombras separadas por comas

ejemplo-02-084.css

001	#muerta h2 {text-shadow: 5px 9px 3px red; }
002	#acueducto h2{ text-shadow:5px -9px 3px blue;}
003	#linajes h2{ text-shadow: -5px 9px 3px orange; }
004	#santiago h2 { text-shadow: -5px -9px 3px green; }
005	#salto h2 { text-shadow:-1px 0 #090, 1px 0 #090, 0 1px #090, 0 -1px #090; }

Para poder utilizar una fuente que no está en el equipo del cliente, browser o navegador, pero si en el servidor y este en formato .eot o bien .ttf o bien .otf o bien woff o bien svgpodemos utilizar:

```
@font-face{
    font-family:nombre-fuente;
    src:fichero-1[ ,fichero-2 ]...;
    [font-weight:weight];
    [font-style:estilo];
    [font-stretch:valor];
    [unicode-range: código-1, código-2, ... ];
}
```

Tabla de tipos de letras soportadas por los navegadores.

	I.E.	Chrome	Firefox	Opera	Safari
ttf/otf	9.0	4.0	3.5	10.0	3.1
woff	9.0	5.0	3.6	11.1	5.1
svg, svgz	x	4.0	x	9.0	3.2
eot	6.0	x	x	x	x

ejemplo-02-085.css

001	@font-face{
002	font-family:Shanghai;
003	src: url('../tipo-letras/shanghai.ttf');
004	font-weight:normal;
005	font-style:normal;
006	}
007	@font-face{
008	font-family:Alameda;
009	src: url('../tipo-letras/Alameda.ttf');
010	font-weight:normal;
011	font-style:normal;
012	}
013	#muerta{
014	font-family:shanghai;
015	font-size:0.8rem;
016	font-style:italic;
017	font-weight:bold;
018	}
019	#acueducto{
020	font-family:alameda;
021	font-size:0.9rem;
022	}
023	@font-face{
024	font-family:FreeSans;
025	src: url('../tipo-letras/FreeSans.ttf','../tipo-letras/FreeSans.otf');
026	font-weight:normal;
027	font-style:normal;
028	}
029	#linajes{

030	<code>font-family:freesans;</code>
031	<code>font-size:1.1rem;</code>
032	<code>}</code>

Sangría

`text-indent:valor;`

ejemplo-02-086.css.

001	<code>div { text-indent:1.5cm; }</code>
002	<code>#muerta p { text-indent:5%; }</code>
003	<code>#acueducto p { text-indent:20mm; }</code>
004	<code>#linajes div { text-indent:1in; }</code>
005	<code>#santiago div { text-indent:2em; }</code>
006	<code>#salto div { text-indent:1.8rem; }</code>
007	<code>#principe div { text-indent:18pt; }</code>
008	<code>#cordón div { text-indent:15px; }</code>
009	<code>#parral div { text-indent:1.5cm; }</code>

Tamaño o desplazamiento tabuladores, solo se aplica a las etiquetas **pre** y **textarea** se utiliza con las extensiones.

`tab-size: valor;`

ejemplo-02-090.css

001	<code>.tabu{</code>
002	<code>color:green;</code>
003	<code>-moz-tab-size:2cm;</code>
004	<code>-webkit-tab-size:2cm;</code>
005	<code>tab-size:2cm;</code>
006	<code>}</code>

Espacio entre palabras

`word-spacing:normal|tamaño;`

ejemplo-02-095.css.

001	<code>.inicio{</code>
002	<code>color:silver;</code>
003	<code>word-spacing:20px;</code>
004	<code>}</code>
005	<code>.derecha{</code>
006	<code>color:blue;</code>
007	<code>word-spacing:3mm;</code>
008	<code>}</code>
009	<code>.izquierda{</code>
010	<code>color:orange;</code>
011	<code>word-spacing:5pt;</code>
012	<code>}</code>
013	<code>.inicio.uno{</code>
014	<code>color:silver;</code>
015	<code>word-spacing:normal;</code>
016	<code>}</code>

Espacio entre caracteres

`letter-spacing:normal|tamaño;`

ejemplo-02-096.css

001	<code>.inicio{</code>
002	<code>color:silver;</code>
003	<code>letter-spacing:5px;</code>
004	<code>}</code>
005	<code>.inicio.uno{</code>
006	<code>color:red;</code>
007	<code>letter-spacing:normal;</code>
008	<code>}</code>
009	<code>.derecha{</code>

```

010 color:blue;
011 letter-spacing:1mm;
012 }
013 .izquierda{
014     color:orange;
015     letter-spacing:2pt;
016 }

```

Comportamiento espacios en blanco

white-space: normal | pre | nowrap | pre-wrap | pre-line;

pre, pre-line y pre-wrap el navegador conserva los espacios en blanco y los saltos de línea, con nowrap todo en una línea y con **normal** se ve normal.

ejemplo-02-097.css

```

001 .inicio{
002     color:silver;
003     white-space:pre;
004 }
005 .inicio.uno{
006     color:red;
007     white-space:normal;
008 }
009 .derecha{
010     color:blue;
011     white-space:nowrap;
012 }
013 .izquierda{
014     color:orange;
015     white-space:pre-wrap;
016 }
017 .centro{
018     color:green;
019     white-space:pre-line;
020 }

```

Para partir palabras demasiado largas (cada una de ellas solas ocupan casi todo el ancho de visualización)

word-wrap: normal | break-word | break-spaces

Con **break-word** permite romper palabras.

Con **normal** rompe palabras solo en los puntos de ruptura permitidos.

Con **break-spaces** no hace nada.

ejemplo-02-098.css

```

001 .inicio{
002     color:silver;
003     word-wrap:break-word;
004 }
005 .derecha{
006     color:blue;
007     word-wrap:break-spaces;
008 }
009 .izquierda{
010     color:orange;
011     word-wrap:normal;
012 }

```

Indica si una palabra va a ser partida, cuando ocupa la totalidad del ancho de visualización.

overflow-wrap: normal | break-word

Con **normal** no parte palabras, mientras que con **break-word** si lo hace.

ejemplo-02-099.css

```

001 .inicio{
002     color:silver;
003     overflow-wrap:break-word;
004 }
005 .izquierda{
006     color:orange;
007     overflow-wrap:normal;
008 }

```

Indica donde se hace un salto de línea cuando se desborda una línea.

word-break:normal | keep-all | break-all | break-work

Con **break-all** se parte la palabra cuando se llega al final del ancho de visualización, el resto no parte palabras.

ejemplo-02-100.css

```

001 .inicio{
002     color:silver;
003     word-break:break-word;
004 }
005 .izquierda{
006     color:orange;
007     word-break:break-all;
008 }
009 .derecha{
010     color:brown;
011     word-break:keep-all;
012 }
013 .centro{
014     color:red;
015     word-break:normal;
016 }

```

Indica si se hacen salto de líneas o se rompen palabras al trabajar con acentos y símbolos. Solo soportado por chrome y opera con extensiones. no entiendo para que sirve.

line-break:auto | loose | normal | strict | anywhere

Con **loose** rompe el texto utilizando la regla de restricción de línea menos restrictiva. Se utiliza normalmente para líneas cortas, como en los periódicos

Con **strict** rompe texto utilizando la regla de salto de línea más estricta

Con **auto** rompe el texto mediante la regla de salto de línea predeterminada.

Con **normal** rompe texto utilizando la regla de interrupción de línea más común

ejemplo-02-101.css

```

001 .inicio{
002     color:silver;
003     -webkit-line-break:auto;
004     line-break:auto;
005 }
006 .izquierda{
007     color:orange;
008     -webkit-line-break:loose;
009     line-break:loose;
010 }
011 .derecha{
012     color:brown;
013     -webkit-line-break:normal;
014     line-break:normal;
015 }
016 .centro{
017     color:red;
018     -webkit-line-break:strict;
019     line-break:strict;
020 }
021 .justo{
022     color:blue;
023     -webkit-line-break:anywhere;

```

024	line-break: anywhere;
025	}

Para indicar la partición de las palabras y poner el guión. Chrome con extensiones.
hyphens:none | manual | auto

No hay partición, de forma manual, tampoco hay partición o automática.
ejemplo-02-108.css

001	.inicio{
002	color:blue;
003	width:50%;
004	-webkit-hyphens:none;
005	hyphens:none;
006	}
007	.izquierda{
008	color:orange;
009	width:50%;
010	-webkit-hyphens:manual;
011	hyphens:manual;
012	}
013	.derecha{
014	color:red;
015	width:50%;
016	-webkit-hyphens:auto;
017	hyphens:auto;
018	}

Para indicar la dirección del texto.

direction:ltr | rtl

Con **ltr** de izquierda a derecha y con **rtl** de derecha a izquierda.

ejemplo-02-102.css

001	.inicio{
002	color:silver;
003	direction:ltr;
004	}
005	.izquierda{
006	color:orange;
007	direction:rtl;
008	}

unicode-bidi:normal | embed | isolate | bidi-override | isolate-override | plaintext

Esta propiedad se utiliza junto con la propiedad **direction** para establecer o devolver si el texto debe reemplazarse para admitir varios idiomas en el mismo documento.

Con **embed** e **isolate** se muestra como si no estuviera y con **bidi-override** e **isolate-override** se cambia el sentido de escritura. Se notan los efectos con **direction:rtl**;

ejemplo-02-106.css

001	.inicio{
002	direction:rtl;
003	color:blue;
004	unicode-bidi:embed;
005	}
006	.izquierda{
007	direction:rtl;
008	color:orange;
009	unicode-bidi:isolate;
010	}
011	.derecha{
012	direction:rtl;
013	color:red;
014	unicode-bidi:bidi-override;
015	}
016	.centro{
017	direction:rtl;

```

018     color:brown;
019     unicode-bidi:isolate-override;
020 }
021 .justo{
022     direction:rtl;
023     color:gold;
024     unicode-bidi:plaintext;
025 }

```

Para indicar la dirección y el sentido del texto, chrome necesita extensiones

writing-mode:horizontal-tb | vertical-rl | vertical-lr | sideways-rl | sideways-lr

Con **horizontal-tb** horizontal de izquierda a derecha

Con **vertical-rl** vertical de arriba abajo y en horizontal de derecha a izquierda.

Con **vertical-lr** vertical de arriba abajo y en horizontal de izquierda a derecha

Con **sideways-rl** vertical de arriba abajo y en horizontal de derecha a izquierda

Con **sideways-lr** vertical de abajo a arriba y en horizontal de izquierda a derecha.

ejemplo-02-103.css

```

001 .inicio{
002     width:50%;
003     height:300px;
004     color:silver;
005     -webkit-writing-mode:horizontal-tb;
006     writing-mode:horizontal-tb;
007 }
008 .izquierda{
009     width:50%;
010     height:300px;
011     color:orange;
012     -webkit-writing-mode:vertical-rl;
013     writing-mode:vertical-rl;
014 }
015 .derecha{
016     width:50%;
017     height:300px;
018     color:red;
019     -webkit-writing-mode:vertical-lr;
020     writing-mode:vertical-lr;
021 }
022 .centro{
023     width:50%;
024     height:300px;
025     color:brown;
026     -webkit-writing-mode:sideways-rl;
027     writing-mode:sideways-rl;
028 }
029 .justo{
030     width:50%;
031     height:300px;
032     color:blue;
033     -webkit-writing-mode:sideways-lr;
034     writing-mode:sideways-lr;
035 }

```

Para definir la orientación de los caracteres en la escritura vertical, debe ir acompañada de writing-mode en modo vertical. Chrome necesita extensiones.

text-orientation:mixed | upright | sideways

Con **mixed** igual que vertical

Con **upright** las letras se ven horizontales, pero se leen de arriba abajo.

Con **sideways** igual que vertical

ejemplo-02-104.css

```

001 .inicio{

```

```

002     width:50%;
003     height:300px;
004     color:blue;
005     -webkit-writing-mode:vertical-rl;
006     writing-mode:vertical-rl;
007 }
008 .izquierda{
009     width:50%;
010     height:300px;
011     color:orange;
012     -webkit-writing-mode:vertical-rl;
013     writing-mode:vertical-rl;
014     -webkit-text-orientation:mixed;
015     text-orientation:mixed;
016 }
017 .derecha{
018     width:50%;
019     height:300px;
020     color:red;
021     -webkit-writing-mode:vertical-rl;
022     writing-mode:vertical-rl;
023     -webkit-text-orientation:upright;
024     text-orientation:upright;
025 }
026 .centro{
027     width:50%;
028     height:300px;
029     color:brown;
030     -webkit-writing-mode:vertical-rl;
031     writing-mode:vertical-rl;
032     -webkit-text-orientation:sideways;
033     text-orientation:sideways;
034 }

```

Número de dígitos en la posición de un símbolo, en chrome con extensiones.

text-combine-upright:none | all | digits numero

Indica el número de dígitos que se van a mostrar en lugar de un carácter o símbolo cuando se muestra el texto en vertical. Para lenguajes orientado a símbolos como el chino.

Esta propiedad define el tipo de énfasis que se utiliza, una línea por encima del texto. En Chrome con extensiones.

text-emphasis-style:none || [filled | open] | [dot | circle | double-circle | triangle | sesame] | cadena]

Con **none** ninguna. Con **filled** circulos negros. Con **open** circunferencias negras. Con **dot** puntos negros. Con **circle** circulos negros. Con **double-circle** botones de selección. Con **triangle** triangulos. Con **sesame** acentos.

ejemplo-02-109.css

```

001 #muerta h2 {
002     -webkit-text-emphasis-style:sesame;
003     text-emphasis-style:sesame;
004 }
005 #acueducto h2 {
006     -webkit-text-emphasis-style:filled;
007     text-emphasis-style:filled;
008 }
009 #linajes h2 {
010     -webkit-text-emphasis-style:open;
011     text-emphasis-style:open;
012 }
013 #santiago h2 {
014     -webkit-text-emphasis-style:dot;
015     text-emphasis-style:dot;
016 }

```

```

017 #salto h2 {
018     -webkit-text-emphasis-style:circle;
019     text-emphasis-style:circle;
020 }
021 #principe h2 {
022     -webkit-text-emphasis-style:double-circle;
023     text-emphasis-style:double-circle;
024 }
025 #cordón h2 {
026     -webkit-text-emphasis-style:triangle;
027     text-emphasis-style:open triangle;
028 }
029 #parral h2 {
030     -webkit-text-emphasis-style:open circle;
031     text-emphasis-style:open circle;
032 }
033 #corpus h2 {
034     -webkit-text-emphasis-style:filled circle;
035     text-emphasis-style:filled circle;
036 }

```

Para establecer el color del énfasis. Chrome con extensiones

text-emphasis-color:color

ejemplo-02-110.css

```

001 #muerta h2 {
002     -webkit-text-emphasis-style:sesame;
003     text-emphasis-style:sesame;
004     -webkit-text-emphasis-color:red;
005     text-emphasis-color:red;
006 }
007 #acueducto h2 {
008     -webkit-text-emphasis-style:filled;
009     text-emphasis-style:filled;
010     -webkit-text-emphasis-color:green;
011     text-emphasis-color:green;
012 }
013 #linajes h2 {
014     -webkit-text-emphasis-style:open;
015     text-emphasis-style:open;
016     -webkit-text-emphasis-color:blue;
017     text-emphasis-color:blue;
018 }
019 #santiago h2 {
020     -webkit-text-emphasis-style:dot;
021     text-emphasis-style:dot;
022     -webkit-text-emphasis-color:orange;
023     text-emphasis-color:orange;
024 }
025 #salto h2 {
026     -webkit-text-emphasis-style:circle;
027     text-emphasis-style:circle;
028     -webkit-text-emphasis-color:gold;
029     text-emphasis-color:gold;
030 }
031 #principe h2 {
032     -webkit-text-emphasis-style:double-circle;
033     text-emphasis-style:double-circle;
034     -webkit-text-emphasis-color:violet;
035     text-emphasis-color:violet;
036 }
037 #cordón h2 {
038     -webkit-text-emphasis-style:triangle;
039     text-emphasis-style:open triangle;
040     -webkit-text-emphasis-color:brown;
041     text-emphasis-color:brown;
042 }
043 #parral h2 {
044     -webkit-text-emphasis-style:open circle;

```

```

045     text-emphasis-style:open circle;
046     -webkit-text-emphasis-color:pink;
047     text-emphasis-color:pink;
048 }
049 #corpus h2 {
050     -webkit-text-emphasis-style:filled circle;
051     text-emphasis-style:filled circle;
052     -webkit-text-emphasis-color:magenta;
053     text-emphasis-color:magenta;
054 }

```

Para indicar donde se coloca el énfasis. Chrome con extensiones.

text-emphasis-position: [over | under] [right | left]

over arriba y under abajo.

ejemplo-02-111.css

```

001 #muerta h2 {
002     -webkit-text-emphasis-style:sesame;
003     text-emphasis-style:sesame;
004     -webkit-text-emphasis-color:red;
005     text-emphasis-color:red;
006     -webkit-text-emphasis-position:over;
007     text-emphasis-position:over;
008 }
009 #acueducto h2 {
010     -webkit-text-emphasis-style:filled;
011     text-emphasis-style:filled;
012     -webkit-text-emphasis-color:green;
013     text-emphasis-color:green;
014     -webkit-text-emphasis-position:under;
015     text-emphasis-position:under;
016 }
017 #linajes h2 {
018     -webkit-text-emphasis-style:open;
019     text-emphasis-style:open;
020     -webkit-text-emphasis-color:blue;
021     text-emphasis-color:blue;
022     -webkit-text-emphasis-position:right;
023     text-emphasis-position:right;
024 }
025 #santiago h2 {
026     -webkit-text-emphasis-style:dot;
027     text-emphasis-style:dot;
028     -webkit-text-emphasis-color:orange;
029     text-emphasis-color:orange;
030     -webkit-text-emphasis-position:left;
031     text-emphasis-position:left;
032 }
033 #salto h2 {
034     -webkit-text-emphasis-style:circle;
035     text-emphasis-style:circle;
036     -webkit-text-emphasis-color:gold;
037     text-emphasis-color:gold;
038     -webkit-text-emphasis-position:over right;
039     text-emphasis-position:over right;
040 }
041 #principe h2 {
042     -webkit-text-emphasis-style:double-circle;
043     text-emphasis-style:double-circle;
044     -webkit-text-emphasis-color:violet;
045     text-emphasis-color:violet;
046     -webkit-text-emphasis-position:under left;
047     text-emphasis-position:under left;
048 }
049 #cordón h2 {
050     -webkit-text-emphasis-style:triangle;
051     text-emphasis-style:open triangle;
052     -webkit-text-emphasis-color:brown;
053     text-emphasis-color:brown;

```

```

054     -webkit-text-emphasis-position:over left;
055     text-emphasis-position:over left;
056 }
057 #parral h2 {
058     -webkit-text-emphasis-style:open circle;
059     text-emphasis-style:open circle;
060     -webkit-text-emphasis-color:pink;
061     text-emphasis-color:pink;
062     -webkit-text-emphasis-position:under right;
063     text-emphasis-position:under right;
064 }
065 #corpus h2 {
066     -webkit-text-emphasis-style:filled circle;
067     text-emphasis-style:filled circle;
068     -webkit-text-emphasis-color:magenta;
069     text-emphasis-color:magenta;
070     -webkit-text-emphasis-position:over under right left ;
071     text-emphasis-position:over under right left ;
072 }

```

Define el énfasis mediante el estilo y el color. En chrome con extensiones.

text-emphasis: estilo color

ejemplo-02-112.css.

```

001 #muerta h2 {
002     -webkit-text-emphasis:sesame red;
003     text-emphasis:sesame red;
004 }
005 #acueducto h2 {
006     -webkit-text-emphasis:filled green;
007     text-emphasis:filled green;
008 }
009 #linajes h2 {
010     -webkit-text-emphasis:open blue;
011     text-emphasis:open blue;
012 }
013 #santiago h2 {
014     -webkit-text-emphasis:dot orange;
015     text-emphasis:dot orange;
016 }
017 #salto h2 {
018     -webkit-text-emphasis:circle gold;
019     text-emphasis:circle gold;
020 }
021 #principe h2 {
022     -webkit-text-emphasis:double-circle violet;
023     text-emphasis:double-circle violet;
024 }
025 #cordón h2 {
026     -webkit-text-emphasis:triangle brown;
027     text-emphasis:triangle brown;
028 }
029 #parral h2 {
030     -webkit-text-emphasis:open circle pink;
031     text-emphasis:open circle pink;
032 }
033 #corpus h2 {
034     -webkit-text-emphasis:filled circle magenta;
035     text-emphasis:filled circle magenta;
036 }

```

4.- Desbordamiento

Para indicar que hacer cuando el contenido de un párrafo sobrepasa sus límites tenemos

overflow:visible | hidden | scroll | auto
overflow-x:visible | hidden | scroll | auto | no-display | no-content
overflow-y:visible | hidden | scroll | auto | no-display | no-content

- ❖ **visible**: Esto hace que el contenido que no cabía en la capa se muestre igualmente en el navegador. Es el comportamiento predeterminado.
- ❖ **hidden**: Sirve para decirle al navegador que no muestre cualquier contenido que se salga de las dimensiones especificadas en el contenedor.
- ❖ **scroll**: Permite mostrar unas barras de desplazamiento para que el usuario pueda hacer scroll sobre el contenido y ver áreas que quedarían fuera del contenedor. En el caso que se aplique este atributo, las barras de desplazamiento aparecerían siempre en el contenedor, independientemente de si el contenido sobrepasa o no las dimensiones del contenedor.
- ❖ **auto**: Este valor indica que las barras de desplazamiento deben aparecer solo en el caso que el contenido supere los límites del contenedor. Es decir, es lo mismo que scroll, pero no aparecerían siempre las barras de desplazamiento, sino solamente cuando sean necesarias.
- ❖ **no-display**: Este comportamiento a día de hoy no está implementado en los navegadores, pero serviría para que, en caso que un contenido sobrepase el límite asignado al contenedor, la capa completa contenedora sea eliminada de la página. El efecto sería el mismo que si hubiésemos colocado **display:none** en el contenedor (si es que había contenido que saliese de sus dimensiones, claro).
- ❖ **no-content**: Esto provocaría que cualquier contenido, que no cupiese en las dimensiones del contenedor, fuera eliminado como si le hubiésemos colocado el atributo **visibility:hidden**. Osea, en diferencia del atributo anterior, lo que se elimina es el contenido y no el contenedor entero.

ejemplo-02-113.css

```
001 #muerta{  
002     width:50%;  
003     height:150px;  
004     overflow:visible;  
005 }  
006 #acueducto{  
007     width:50%;  
008     height:150px;  
009     overflow:hidden;  
010 }  
011 #linajes{  
012     width:50%;  
013     height:150px;  
014     overflow:scroll;  
015 }  
016 #santiago{  
017     width:50%;  
018     height:150px;  
019     overflow:auto;  
020 }  
021 #salto{  
022     width:25%;  
023     height:150px;  
024     overflow-x:auto;  
025 }  
026 #principe{  
027     width:25%;  
028     height:150px;  
029     overflow-x:visible;  
030 }  
031 #cordon{  
032     width:50%;  
033     height:150px;  
034     overflow-x:hidden;  
035 }  
036 #parral{
```

```

037     width:50%;  

038     height:150px;  

039     overflow-x:scroll;  

040 }  

041 #corpus{  

042     width:25%;  

043     height:150px;  

044     overflow-y:auto;  

045 }  

046 #vida{  

047     width:25%;  

048     height:150px;  

049     overflow-y:visible;  

050 }  

051 .izquierda{  

052     width:25%;  

053     height:150px;  

054     overflow-y:hidden;  

055 }  

056 .derecha{  

057     width:25%;  

058     height:150px;  

059     overflow-y:scroll;  

060 }

```

5.- Alineación

Alineación horizontal del texto

text-align:start | end | left | right | center | justify | match-parent | justify-all

Alinea el texto al inicio, al final, a la izquierda, a la derecha, centrado, justificado, el resto no son ven resultados

ejemplo-02-091.css

```

001 .inicio{  

002     color:silver;  

003     text-align:strat;  

004 }  

005 .izquierda{  

006     color:red;  

007     text-align:left;  

008 }  

009 .derecha{  

010     color:blue;  

011     text-align:right;  

012 }  

013 .centro{  

014     color:orange;  

015     text-align:center;  

016 }  

017 .justo{  

018     color:green;  

019     text-align:justify;  

020 }  

021 .final{  

022     color:brown;  

023     text-align:end;  

024 }  

025 .nuevo{  

026     color:pink;  

027     text-align:match-parent;  

028 }  

029 .undecimo{  

030     color:magenta;  

031     text-align:justify-all;  

032 }

```

Indica la alineación de la última línea del texto

text-align-last:auto | start | end | left | right | center | justify

La alineación de la última línea será, automática, al inicio, al final, izquierda, derecha, centrada y justificada. Se usa con extensiones en firefox.

ejemplo-02-092.css

```
001 .inicio{  
002     color:silver;  
003     text-align:strat;  
004     text-align-last:end;  
005 }  
006 .izquierda{  
007     color:red;  
008     text-align:left;  
009     -moz-text-align-last:right;  
010     text-align-last:right;  
011 }  
012 .derecha{  
013     color:blue;  
014     text-align:right;  
015     -moz-text-align-last:left;  
016     text-align-last:left;  
017 }  
018 .centro{  
019     color:orange;  
020     text-align:center;  
021     -moz-text-align-last:justify;  
022     text-align-last:justify;  
023 }  
024 .justo{  
025     color:green;  
026     text-align:justify;  
027     -moz-text-align-last:center;  
028     text-align-last:center;  
029 }  
030 .final{  
031     color:brown;  
032     text-align:end;  
033     -moz-text-align-last:start;  
034     text-align-last:start;  
035 }
```

Para indicar como se lleva a cabo el justificado. solo internet explorer, edge y safari.

text-justify:auto | none | inter-word | inter-character

Con **auto** el navegador determina el algoritmo de justificado

Con **none** se elimina el justificado.

Con **inter-word** aumenta disminuye espacio entre palabras.

Con **inter-character** aumenta disminuye espacio entre caracteres.

ejemplo-02-094.css

```
001 .nuevo{  
002     color:silver;  
003     text-align:justify;  
004     text-justify:auto;  
005 }  
006 .derecha{  
007     color:blue;  
008     text-align:justify;  
009     text-justify:inter-word;  
010 }  
011 .centro{  
012     color:orange;  
013     text-align:justify;  
014     text-justify:inter-character;  
015 }
```

alineación vertical texto, con respecto al elemento que tiene al lado.

vertical-align:baseline|sub|super|top|text-top|middle|bottom|text-bottom|porcentaje|valor|inherit;

- ❖ **baseline** (línea de base) alinea la línea de base del elemento (o la parte inferior si el elemento no tiene línea de base) con la línea de base del parente
- ❖ **middle** (medio)alinea el punto medio vertical del elemento (normalmente una imagen) con la línea de base más la mitad de la altura x del parente
- ❖ **sub** representa el elemento como un subíndice
- ❖ **super** representa el elemento como un superíndice
- ❖ **text-top** (texto superior)alinea la parte superior del elemento con la parte superior de la fuente del elemento parente
- ❖ **text-bottom** (texto inferior)alinea la parte inferior del elemento con la parte inferior de la fuente del elemento parente
- ❖ **top** (superior)alinea la parte superior del elemento con el elemento más alto de la linea.
- ❖ **bottom** (inferior)alinea la parte inferior del elemento con el elemento más bajo de la linea.

ejemplo-02-114.css

001	:root{font-size:24pt;}
002	img{width:10%;}
003	#uno{vertical-align:baseline;}
004	#dos{vertical-align:sub;}
005	#tres{vertical-align:super;}
006	#cuatro{vertical-align:top;}
007	#cinco{vertical-align:text-top;}
008	#seis{vertical-align:middle;}
009	#siete{vertical-align:bottom;}
010	#ocho{vertical-align:text-bottom;}
011	#nueve{vertical-align:75%;}
012	#diez{vertical-align:2cm;}
013	#once{vertical-align:top;}
014	#doce{font-size:48pt;}

6.- Estilos para el fondo

Color de fondo

background-color:color|transparentt

ejemplo-02-115.css

001	#muerta{background-color:blue;}
002	#acueducto{ background-color:#FF0000; }
003	#linajes{background-color:#0F0;}
004	#santiago{background-color:rgb(120,0,120);}
005	#salto{background-color:rgba(70,70,70,0.7)}
006	#principe{background-color:hsl(180,20%,30%);}
007	#cordón{background-color:hsla(90,30%,70%)}

Punto a partir del que se muestra el color, no se aplica a body ni a :root. Si no se indica background-origin también se aplica a la imagen.

background-clip: border-box| padding-box | content-box

Con **border-box** el color se muestra a partir del margen.

Con **padding-box** el color se muestra a partir del relleno

Con **content-box** el color se muestra a partir del contenido.

ejemplo-02-116.css

001	#muerta{
002	margin:40px 30px 20px 10px;
003	border:5px solid transparent;

```

004     outline:gold solid 10px;
005     padding:40px 30px 20px 40px;
006     background-color:blue;
007     background-clip:border-box;
008 }
009 #acueducto{
010     margin:40px 30px 20px 10px;
011     border:5px solid transparent;
012     outline:gold solid 10px;
013     padding:40px 30px 20px 40px;
014     background-color:red;
015     background-clip:padding-box;
016 }
017 #linajes{
018     margin:40px 30px 20px 10px;
019     border:5px solid transparent;
020     outline:gold solid 10px;
021     padding:40px 30px 20px 40px;
022     background-color:green;
023     background-clip:content-box;
024 }

```

Para indicar el grado de opacidad o transparencia tenemos

opacity:valor|inherit

Indicamos mediante el valor, valor entre 0 y 1, el grado de opacidad valor 1 o transparencia valor 0

ejemplo-02-3-008.css

```

001 article{
002     background-color:green;
003     height:600px;
004     width:80%;
005 }
006 div{
007     height:600px;
008     width:80%;
009     background-color:blue;
010     opacity:0.2;
011 }

```

Imagen de fondo, se pueden poner varias imágenes de fondo

background-image:none | url(*fichero*)

ejemplo-02-117.css

```

001 #muerta{background-image:url(..../imagenes/acueducto.jpg);}
002 #acueducto{background-image:url(..../imagenes/alcazar.jpg);}
003 #linajes{background-image:url(..../imagenes/catedral.jpg);}

```

Para establecer el origen de coordenadas para la imagen de fondo tenemos, no se aplica a body:

background-origin: border-box | padding-box | content-box

- ❖ **border-box:** Este valor significa que el origen de coordenadas de la imagen será el borde del elemento. `background-origin: border-box;`
- ❖ **padding-box:** Este valor es el utilizado por defecto en CSS 3 y es tal como se posiciona la imagen en navegadores que sólo entienden CSS 2 o inferior. Cuando indicamos `background-origin: padding-box;` queremos que el eje de coordenadas donde se va a colocar la imagen sea el espacio destinado al elemento, incluyendo su posible padding y sin contar el posible borde.

- ❖ **content-box:** El tercero de los posibles valores de background-origin sirve para que el origen de coordenadas para la posición de la imagen de fondo sea el lugar donde empieza el contenido del elemento, osea, sin tener en cuenta sus posibles bordes y padding.

Nota: Si background-attachment tiene el valor "fixed" este atributo se ignorará. La imagen no se debe repetir.

ejemplo-02-118.css

001	#muerta{
002	margin:40px 30px 20px 10px;
003	border:5px solid transparent;
004	outline:gold solid 10px;
005	padding:40px 30px 20px 40px;
006	background-image:url(..../imagenes/acueducto.jpg);
007	background-repeat:no-repeat;
008	background-origin:border-box;
009	}
010	#acueducto{
011	margin:40px 30px 20px 10px;
012	border:5px solid transparent;
013	outline:gold solid 10px;
014	padding:40px 30px 20px 40px;
015	background-image:url(..../imagenes/alcazar.jpg);
016	background-repeat:no-repeat;
017	background-origin:padding-box;
018	}
019	#linajes{
020	margin:40px 30px 20px 10px;
021	border:5px solid transparent;
022	outline:gold solid 10px;
023	padding:40px 30px 20px 40px;
024	background-image:url(..../imagenes/catedral.jpg);
025	background-repeat:no-repeat;
026	background-origin:content-box;
027	}

Repetición de la imagen de fondo

background-repeat:repeat|repeat-x|repeat-y|no-repeat|space|round

Con space añade espacios para ajustar las imágenes al tamaño del contenedor. Con round redimensiona todas las imágenes para ajustarlas al contenedor.

ejemplo-02-119.css

001	#muerta{
002	background-image:url(..../imagenes/acueducto.jpg);
003	background-repeat:repeat;
004	}
005	#acueducto{
006	background-image:url(..../imagenes/acueducto.jpg);
007	background-repeat:no-repeat;
008	}
009	#linajes{
010	background-image:url(..../imagenes/acueducto.jpg);
011	background-repeat:repeat-x;
012	}
013	#santiago{
014	background-image:url(..../imagenes/acueducto.jpg);
015	background-repeat:repeat-y;
016	}
017	#salto{
018	background-image:url(..../imagenes/acueducto.jpg);
019	background-repeat:space;
020	}
021	#principe{
022	background-image:url(..../imagenes/acueducto.jpg);
023	background-repeat:round;
024	}

Establecer el tamaño de la imagen de fondo

background-size: cover | contain | tamaño | lista-tamaños

Con **contain**: la caja contiene la imagen. La imagen aparece integralmente en la caja y es redimensionada proporcionalmente, al tamaño máximo: o bien es tan alta como la caja, o es tan ancha como la caja.

Con **cover**: la imagen es redimensionada proporcionalmente y cubre toda la caja. Parte de la imagen puede no aparecer en pantalla.

ejemplo-02-120.css

```

001 #muerta{
002     background-image:url(..../imagenes/acueducto.jpg);
003     background-size:100% 100%;
004 }
005 #acueducto{
006     background-image:url(..../imagenes/alcazar.jpg);
007     background-size:50% 50%;
008 }
009 #linajes{
010     background-image:url(..../imagenes/catedral.jpg);
011     background-size:75% 75%;
012     background-repeat:no-repeat;
013 }
014 #santiago{
015     background-image:url(..../imagenes/san-esteban.jpg);
016     background-size:cover;
017     background-repeat:no-repeat;
018 }
019 #salto{
020     background-image:url(..../imagenes/san-millan.jpg);
021     background-size:contain;
022     background-repeat:no-repeat;
023 }
```

acción imagen de fondo

background-attachment:scroll | fixed | local

realiza scroll o permanece fija

ejemplo-02-121.css

```

001 :root{
002     background-image:url(..../imagenes/alcazar.jpg);
003     background-size:100% 100%;
004     background-repeat:no-repeat;
005     background-attachment:scroll;
006 }
```

ejemplo-02-122.css

```

001 :root{
002     background-image:url(..../imagenes/alcazar.jpg);
003     background-size:100% 100%;
004     background-repeat:no-repeat;
005     background-attachment:fixed;
006 }
```

Posición imagen de fondo

background-position: posición-imagen-esquina-superior-izquierda | [top | center | bottom] [left | center | right]
ejemplo-02-123.css

```

001 #muerta{
002     background-image:url(..../imagenes/acueducto.jpg);
003     background-size:75% 75%;
004     background-repeat:no-repeat;
005     background-position:top left;
006 }
007 #acueducto{
```

```

008 background-image:url(..../imagenes/alcazar.jpg);
009 background-size:75% 75%;
010 background-repeat:no-repeat;
011 background-position:center center;
012 }
013 #linajes{
014 background-image:url(..../imagenes/catedral.jpg);
015 background-size:75% 75%;
016 background-repeat:no-repeat;
017 background-position:bottom right;
018 }
019 #santiago{
020 background-image:url(..../imagenes/san-esteban.jpg);
021 background-size:75% 75%;
022 background-repeat:no-repeat;
023 background-position:20% 60%;
024 }
025 #salto{
026 background-image:url(..../imagenes/san-millan.jpg);
027 background-size:75% 75%;
028 background-repeat:no-repeat;
029 background-position:40px 15px;
030 }

```

Fondo general, no se tienen porque poner todos los valores

**background: vbackground-color | vbackground-image
 vbackground-position | vbackground-size | vbackground-repeat
 vbackground-attachment**

Podemos poner varias imágenes de fondo con background, separadas por comas o las correspondientes etiquetas.

ejemplo-02-124.css

```

001 #muerta{background:url(..../imagenes/acueducto.jpg) top left/100% 100% no-repeat ;}
002 #acueducto{background:url(..../imagenes/alcazar.jpg) center center/75% 75% no-repeat ;
}
003 #linajes{background:url(..../imagenes/catedral.jpg) bottom right/85% 85% no-repeat ;}
004 #santiago{background:url(..../imagenes/san-esteban.jpg) 20% 60%/65% 65% no-repeat ;}
005 #salto{background:url(..../imagenes/san-millan.jpg) 40px 15px/80% 80% no-repeat ;}
006 #principe{background: green;}

```

ejemplo-02-125.css

```

001 :root{
002     background: url(..../imagenes/acueducto.jpg) top left/50% 50% no-repeat,
url(..../imagenes/alcazar.jpg) top right/50% 50% no-repeat,
url(..../imagenes/catedral.jpg) bottom left/50% 50% no-repeat ,
url(..../imagenes/san-esteban.jpg) bottom right/50% 50% no-repeat;
003     background-attachment:fixed;
004 }

```

Se puede hacer esto último de con las propiedades individuales

ejemplo-02-126.css

```

001 :root{
002     background-image:url(..../imagenes/acueducto.jpg) ,
url(..../imagenes/alcazar.jpg) , url(..../imagenes/catedral.jpg) ,
url(..../imagenes/san-esteban.jpg) ;
003     background-size:50% 50%;
004     background-position:top left, top right, bottom left, bottom right;
005     background-repeat:no-repeat;
006     background-attachment:fixed;
007 }

```

Para utilizar los degradados tenemos:

**background: linear-gradient(*parámetros*);
 background-image: linear-gradient(*parámetros*);**

- ❖ Origen-y/o-angulo del degradado: El primer parámetro sería el origen desde donde comenzará el degradado y/o el ángulo de disposición del gradiente de color. Podemos decir que el degradado comience desde arriba, abajo o desde una esquina cualquiera. Por defecto los degradados serán distribuidos en un gradiente en línea recta, pero además podemos indicar un ángulo distinto con el que se vaya produciendo el gradiente de color.
- ❖ lista-de-colores y opcionalmente, el lugar hasta donde se debe mostrar cada uno: Luego colocaremos los colores, todos los que queramos, que deben utilizarse en el degradado, separados por comas. Además, si lo deseamos, podemos definir las paradas de color "color stops", que consiste en declarar el lugar desde donde debe empezar el gradiente del color.

ejemplo-02-127.css

```

001 #muerta{
002     background: -webkit-linear-gradient(orange, pink);
003     background: linear-gradient(orange, pink);
004 }
005 #acueducto{
006     background: -webkit-linear-gradient(top left, #fff, #f66);
007     background: linear-gradient(top left, #fff, #f66);
008 }
009 #linajes{
010     background: -webkit-linear-gradient(180deg, #f0f, #f66);
011     background: linear-gradient(180deg, #f0f, #f66);
012 }
013 #santiago{
014     background-image: -webkit-linear-gradient(#00f 50%, #000);
015     background-image: linear-gradient(#00f 50%, #000);
016 }
017 #salto{
018     background-image: -webkit-linear-gradient(45deg, #66f 10%, #f80 30%, #ffc 60%);
019     background-image: linear-gradient(45deg, #66f 10%, #f80 30%, #ffc 60%);
020 }

```

Repetirdegradados

background: repeating-linear-gradient(*parámetros*);
background-image: repeating-linear-gradient(*parámetros*);

ejemplo-02-129.css

```

001 #muerta{
002     background: -webkit-repeating-linear-gradient(#fff, #666 25%);
003     background: repeating-linear-gradient(#fff, #666 25%);
004 }
005 #acueducto{
006     background: -webkit-repeating-linear-gradient(#fff, #666 25%);
007     background: repeating-linear-gradient(#fff, #666 25%);
008 }
009 #linajes{
010     background: -webkit-repeating-linear-gradient(left, #ffc, #f96 30%, #963 40%, #630 51%);
011     background: repeating-linear-gradient(left, #ffc, #f96 30%, #963 40%, #630 51%);
012 }
013 #santiago{
014     background-image: -webkit-repeating-linear-gradient(45deg, red, blue, red 50px);
015     background-image: repeating-linear-gradient(45deg, red, blue, red 50px);
016 }
017 #salto{
018     background-image: -webkit-repeating-linear-gradient(left, #f00, #f80, #ff0, #0f0, #00f, #60f, #c0f, #f00 100px);
019     background-image: repeating-linear-gradient(left, #f00, #f80, #ff0, #0f0, #00f, #60f, #c0f, #f00 100px);
020 }

```

Degradados radiales (se utilizan con extensiones)

background: radial-gradient(*parámetros*);
background-image: radial-gradient(*parámetros*);

- ❖ Posición inicial del gradiente circular: Los degradados radiales comienzan en un punto cualquiera del fondo de un elemento y se extienden hacia fuera de ese punto con formas circulares o de elipse. Luego, para definirlos, necesitaremos una forma de especificar dicho punto de inicio del degradado. El punto se especifica con una o dos coordenadas, que pueden tener distintas unidades CSS. Si se omite, se entiende que el degradado tiene que comenzar en el punto central del fondo del elemento.
- ❖ Forma y/o dimensión: La forma puede ser circular o elipse, para lo cual especificamos las palabras "circle" o "ellipse". El tamaño lo expresamos con otra serie de palabras clave, que indican hasta donde debe crecer el círculo o elipse: closest-side | closest-corner | farthest-side | farthest-corner | contain | cover. Por ejemplo, closest-side indica que el círculo o elipse debe crecer hasta el lado más cercano. La palabra farthest-corner indicaría que debe crecer hasta la esquina más lejana. Contain sería lo mismo que decir closest-side y cover sinónimo de farthest-corner.
- ❖ Alternativa a B) Tamaño: De manera alternativa a especificar la forma y dimensión del degradado -punto B) anterior-, podemos indicar un par de medidas en cualquier unidad CSS o porcentajes. Esas medidas se utilizarían para generar un círculo o una elipse del tamaño deseado para nuestro gradiente. La primera medida sería para la anchura de la elipse y la segunda sería para la altura (si ambas son iguales se mostraría una forma circular en el degradado. Si son distintas, sería una elipse. El tamaño debe ser siempre positivo.
- ❖ Colores del degradado: Para acabar, se deben indicar cuantos colores se deseen, separados por comas, con la posibilidad de indicar las paradas de color que se deseen.

ejemplo-02-128.css

```

001 #muerta{
002     background: radial-gradient(ellipse cover, #66f, #f80, #ffc);
003     background: -webkit-radial-gradient(ellipse cover, #66f, #f80, #ffc);
004 }
005 #acueducto{
006     background: -webkit-radial-gradient(#0f0, #06f);
007     background: radial-gradient(#0f0, #06f);
008 }
009 #linajes{
010     background: -webkit-radial-gradient(top left, #fff, #f66);
011     background: radial-gradient(top left, #fff, #f66);
012 }
013 #santiago{
014     background-image: -webkit-radial-gradient(200px 30px, #f0f, #000);
015     background-image: radial-gradient(200px 30px, #f0f, #000);
016 }
017 #salto{
018     background-image: -webkit-radial-gradient(center, #00f, #000 50%);
019     background-image: radial-gradient(center, #00f, #000 50%);
020 }
021 #principe{
022     background-image: -webkit-radial-gradient(circle, #66f, #f80, #ffc);
023     background-image: radial-gradient(circle, #66f, #f80, #ffc);
024 }
025 #cordon{
026     background: -webkit-radial-gradient(10%, ellipse farthest-corner, #66f 60%, #f80 85%, #ffc);

```

```

027 background: radial-gradient(10%, ellipse farthest-corner, #66f 60%, #f80 85%, #ffc);
028 }
029 #parral{
030     background-image: -webkit-radial-gradient( 30% 80%, #fff, #666, #66f);
031     background-image: radial-gradient( 30% 80%, #fff, #666, #66f);
032 }
033 #corpus{
034     background: -webkit-radial-gradient(top left, #ffc, #f96, #963, #630);
035     background: radial-gradient(top left, #ffc, #f96, #963, #630);
036 }
037 #vida{
038     background-image: -webkit-radial-gradient(100% 50%, red, blue 50px, red);
039     background-image: radial-gradient( 100% 50%, red, blue 50px, red);
040 }
041 .izquierda{
042     background: -webkit-radial-gradient(left, #f00 20%, #f80, #ff0, #0f0, #00f, #60f, #c0f);
043     background: radial-gradient(left, #f00 20%, #f80, #ff0, #0f0, #00f, #60f, #c0f);
044 }
045 .justo{
046     background-image: -webkit-radial-gradient(10%, ellipse closest-side, #66f 60%, #f80 85%, #ffc);
047     background-image: radial-gradient(10%, ellipse closest-side, #66f 60%, #f80 85%, #ffc);
048 }

```

Degrados circulares con repetición

background: repeating-radial-gradient(*parámetros*);

background-image: repeating-radial-gradient(*parámetros*);

Tiene los mismos parámetros que los anteriores

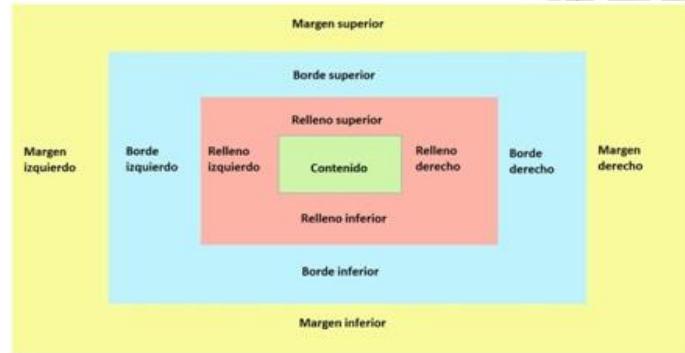
ejemplo-02-130.css

```

001 #muerta{
002     background: -webkit-repeating-radial-gradient(circle, #fff, #666 25%);
003     background: repeating-radial-gradient(circle, #fff, #666 25%);
004 }
005 #acueducto{
006     background: -webkit-repeating-radial-gradient(left, circle, #ffc, #f96 30%, #963 40%, #630 51%);
007     background: repeating-radial-gradient(left, circle, #ffc, #f96 30%, #963 40%, #630 51%);
008 }
009 #linajes{
010     background: -webkit-repeating-radial-gradient(left, ellipse farthest-side, #f00, #f80, #ff0, #0f0, #00f, #60f, #c0f, #f00);
011     background: repeating-radial-gradient(left, ellipse farthest-side, #f00, #f80, #ff0, #0f0, #00f, #60f, #c0f, #f00);
012 }

```

7.- Posicionamiento



Margenes

margin: superior derecho inferior izquierdo | inherit;

Margen superior

margin-top: valor | auto | inherit;

Margen derecho

margin-right: valor | auto | inherit;

Margen inferior

margin-bottom: valor | auto | inherit;

Margen izquierdo

margin-left: valor | auto | inherit;

ejemplo-02-132.css

001	#muerta{
002	margin-top:20px;
003	margin-bottom:10px;
004	margin-left:15px;
005	margin-right:12px;
006	}
007	#acueducto{ margin:1cm 1.5cm 1.3cm 2.5cm; }

Relleno

padding:superior derecho inferior izquierdo | inherit;

Relleno superior

padding-top: valor | inherit;

Relleno derecho

padding-right: valor | inherit;

Relleno inferior

padding-bottom: valor | inherit;

Relleno derecho

padding-left: valor | inherit;

ejemplo-02-133.css

001	#muerta{
002	margin-top:20px;
003	margin-bottom:10px;
004	margin-left:15px;
005	margin-right:12px;
006	}
007	#acueducto{
008	margin-top:20px;
009	margin-bottom:10px;
010	margin-left:15px;
011	margin-right:12px;
012	padding-top:5px;
013	padding-bottom:3px;
014	padding-left:14px;
015	padding-right:10px;
016	}
017	#linajes{
018	margin:1cm 1.5cm 1.3cm 2.5cm;
019	}
020	#santiago{
021	margin:1cm 1.5cm 1.3cm 2.5cm;

022	<code>padding:0.2cm 0.3cm 0.4cm 0.5cm;</code>
023	}

Ancho del borde

`border-width: superior derecho inferior izquierdo;`

Ancho del borde superior

`border-top-width:thin|medium|thick|valor;`

`thin' <= 'medium' <= 'thick'.`

Ancho del borde derecho

`border-right-width:thin|medium|thick|valor;`

Ancho del borde inferior

`border-bottom-width:thin|medium|thick|valor;`

Ancho del borde izquierdo

`border-left-width:thin|medium|thick|valor;`

ejemplo-02-134.css

001	<code>#acueducto{</code>
002	<code>border-top-width:2px;</code>
003	<code>border-bottom-width:1px;</code>
004	<code>border-left-width:4px;</code>
005	<code>border-right-width:2px;</code>
006	<code>border-style:solid;</code>
007	}
008	<code>#linajes{</code>
009	<code>border-top-width:medium;</code>
010	<code>border-bottom-width:medium;</code>
011	<code>border-left-width:thick;</code>
012	<code>border-right-width:thin;</code>
013	<code>border-style:solid;</code>
014	}
015	<code>#santiago{</code>
016	<code>border-width:4px 2px 1px 5px;</code>
017	<code>border-style:solid;</code>
018	}
019	<code>#salto{</code>
020	<code>border-top-width:2px;</code>
021	<code>border-bottom-width:1px;</code>
022	<code>border-left-width:4px;</code>
023	<code>border-right-width:2px;</code>
024	<code>border-style:solid;</code>
025	}
026	<code>#cordón{</code>
027	<code>border-width:4px 2px 1px 5px;</code>
028	<code>border-style:solid;</code>
029	}

Color del borde

`border-color:color-superior color-derecho color-inferior color-izquierdo;`

Color del borde superior

`border-top-color:color|transparent|inherit;`

Color del borde inferior

`border-bottom-color:color|transparent|inherit;`

Color del borde izquierdo

border-left-color:color|transparent|inherit;

Color del borde derecho

border-right-color:color|transparent|inherit;
ejemplo-02-135.css

001	#muerta{
002	border-width:4px 2px;
003	border-color:orange;
004	border-style:solid;
005	}
006	#acueducto{
007	border-top-width:2px;
008	border-bottom-width:1px;
009	border-left-width:4px;
010	border-right-width:2px;
011	border-top-color:brown;
012	border-bottom-color:yellow;
013	border-left-color:orange;
014	border-right-color:gold;
015	border-style:solid;
016	}
017	#linajes{
018	border-top-width:medium;
019	border-bottom-width:medium;
020	border-left-width:thick;
021	border-right-width:thin;
022	border-style:solid;
023	}
024	#santiago{
025	border-width:4px 2px 1px 5px;
026	border-style:solid;
027	border-color:orange blue red green;
028	}
029	#salto{
030	border-top-width:2px;
031	border-bottom-width:1px;
032	border-left-width:4px;
033	border-right-width:2px;
034	border-style:solid;
035	border-color:red;
036	}
037	#cordón{
038	border-width:4px 2px 1px 5px;
039	border-style:solid;
040	border-color:blue red;
041	}

Estilo del borde

border-style:none|dotted|dashed|solid|double|groove|ridge|inset|outset;

- ❖ **dotted** (punteado) el borde es una línea punteada dibujada encima del fondo del elemento
- ❖ **dashed** (a trazos) el borde es una línea a trazos dibujada encima del fondo del elemento
- ❖ **solid** (continuo) el borde es una línea continua
- ❖ **double** (doble) el borde es una línea doble dibujada encima del fondo del elemento. La suma de cada una de las dos líneas más el espacio entre ellas es igual al valor <ancho-de-borde>.
- ❖ **groove** (canal) se dibuja un canal o acanaladura 3D con colores basados en el valor <color>.

- ❖ **ridge** (cresta) se dibuja una cresta o saliente 3D con colores basados en el valor <color>.
- ❖ **inset** (bajorrelieve) se dibuja un bajorrelieve 3D con colores basados en el valor <color>.
- ❖ **outset** (alторrelieve) se dibuja un alторrelieve 3D con colores basados en el valor <color>.

Estilo del borde superior

```
border-top-style:none|hidden|dotted|dashed|solid|double|groove|ridge|
inset|outset|inherit;
```

Estilo del borde inferior

```
border-bottom-style:none|hidden|dotted|dashed|solid|double|groove
|ridge|inset|outset|inherit;
```

Estilo del borde izquierdo

```
border-left-style: none|hidden|dotted|dashed|solid|double|groove|ridge|
inset|outset|inherit;
```

Estilo del borde derecho

```
border-right-style: none|hidden|dotted|dashed|solid|double|groove|
ridge|inset|outset|inherit;
```

Ejemplo-02-136.css

001	#muerta{
002	border-width:4px 2px;
003	border-color:orange;
004	border-style:solid;
005	}
006	#acueducto{
007	border-top-width:2px;
008	border-bottom-width:1px;
009	border-left-width:4px;
010	border-right-width:2px;
011	border-top-color:brown;
012	border-bottom-color:yellow;
013	border-left-color:orange;
014	border-right-color:gold;
015	border-top-style:dotted;
016	border-bottom-style:dashed;
017	border-left-style:solid;
018	border-right-style:hidden;
019	}
020	#linajes{
021	border-top-width:medium;
022	border-bottom-width:medium;
023	border-left-width:thick;
024	border-right-width:thin;
025	border-style:double groove inset outset;
026	border-color:red;
027	}
028	#santiago{
029	border-width:4px 2px 1px 5px;
030	border-style:inset outset;
031	border-color:orange blue red green;
032	}
033	#salto{
034	border-top-width:2px;
035	border-bottom-width:1px;
036	border-left-width:4px;
037	border-right-width:2px;

038	<code>border-style:ridge groove;</code>
039	<code>border-color:red;</code>
040	}
041	<code>#cordón{</code>
042	<code>border-width:4px 2px 1px 5px;</code>
043	<code>border-style:solid double;</code>
044	<code>border-color:blue red;</code>
045	}

Características del borde solo se pone uno que se aplica a todos los bordes.

border: ancho estilo color

Características del borde superior

border-top: ancho estilo color

Características del borde derecho

border-right: ancho estilo color

Características del borde inferior

border-bottom: ancho estilo color

Características del borde izquierdo

border-left: ancho estilo color

Ejemplo-02-137.css

001	<code>#muerta{border:4px solid orange ; }</code>
002	<code>#acueducto{</code>
003	<code>border-top:2px inset brown;</code>
004	<code>border-bottom:1px outset yellow;</code>
005	<code>border-left:4px ridge orange;</code>
006	<code>border-right-width:2px groove gold;</code>
007	}

Colores para el borde

Para firefox, es una extensión

El tamaño del borde tiene que coincidir con el número de colores que se van a mostrar en el borde. Solo en Firefox y con extensiones.

Colores para el borde superior

border-top-colors: lista-colores

Colores para el borde derecho

border-right-colors: lista-colores

Colores para el borde inferior

border-bottom-colors: lista-colores

Colores para el borde izquierdo

border-left-colors: lista-colores

Ejemplo-02-138.css

001	<code>#muerta{</code>
002	<code>border-style: solid;</code>
003	<code>border-width: 10px;</code>
004	<code>-moz-border-top-colors: #ffcc99 #ffbb88 #ffaa77 #ff9966 #ff8855 #ff7744 #ff6633 #ff5522 #ff4411 #ff3300;</code>
005	<code>-moz-border-right-colors: #ffcc99 #ffbb88 #ffaa77 #ff9966 #ff8855 #ff7744</code>

```

006 #ff6633 #ff5522 #ff4411 #ff3300;
      -moz-border-bottom-colors: #ffcc99 #ffbb88 #ffaa77 #ff9966 #ff8855 #ff7744
      #ff6633 #ff5522 #ff4411 #ff3300;
007      -moz-border-left-colors: #ffcc99 #ffbb88 #ffaa77 #ff9966 #ff8855 #ff7744
      #ff6633 #ff5522 #ff4411 #ff3300;
008 border-top-colors: #ffcc99 #ffbb88 #ffaa77 #ff9966 #ff8855 #ff7744 #ff6633
      #ff5522 #ff4411 #ff3300;
009 border-right-colors: #ffcc99 #ffbb88 #ffaa77 #ff9966 #ff8855 #ff7744 #ff6633
      #ff5522 #ff4411 #ff3300;
010 border-bottom-colors: #ffcc99 #ffbb88 #ffaa77 #ff9966 #ff8855 #ff7744 #ff6633
      #ff5522 #ff4411 #ff3300;
012 border-left-colors: #ffcc99 #ffbb88 #ffaa77 #ff9966 #ff8855 #ff7744 #ff6633
      #ff5522 #ff4411 #ff3300;
013 }
014 #acueducto{
015     border-style: solid;
016     border-width: 4px;
017     -moz-border-top-colors: blue navy;
018     -moz-border-right-colors: red magenta;
019     -moz-border-bottom-colors: orange red;
020     -moz-border-left-colors: yellow gold;
021     border-top-colors: blue navy;
022     border-right-colors: red magenta;
023     border-bottom-colors: orange red;
024     border-left-colors: yellow gold;
025 }

```

Borde redondeado, se utiliza con extensiones:

border-radius: radio | lista-radio-x / lista-radio-y

Borde redondeado esquina inferior izquierda

border-bottom-left-radius:radio | radio-x / radio-y

Borde redondeado esquina inferior derecha

border-bottom-right-radius:radio | radio-x / radio-y

Borde redondeado esquina superior izquierda

border-top-left-radius:radio | radio-x / radio-y

Borde redondeado esquina superiorderecha

border-top-right-radius:radio | radio-x / radio-y

Ejemplo-02-139.css

```

001 #muerta{
002     margin:30px;
003     border:2px solid orange;
004     border-radius:7.5em 5em;
005 }
006 #acueducto{
007     margin:30px;
008     border:2px solid red;
009     border-top-left-radius:4em;
010     border-top-right-radius:4em;
011     border-bottom-right-radius:4em;
012     border-bottom-left-radius:4em;
013 }
014 }
015 #linajes{
016     margin:30px;
017     border:2px solid blue;
018     border-radius:2em 1em 4em / 0.5em 3em;
019 }
020 #santiago{
021     margin:30px;
022     border:2px solid green;

```

```

023 border-top-left-radius:2em 0.5em;
024 border-top-right-radius:1em 3em;
025 border-bottom-right-radius:4em 0.5em;
026 border-bottom-left-radius:1em 3em;
027
028 }
029 #salto{
030 margin:30px;
031 border:2px solid gold;
032 border-radius:0.5em 2em 0.5em 2em;
033 }
034 #cordón{
035 margin:30px;
036 border:2px solid brown;
037 border-radius:0.4em 1.6em 0.4em 1.6em;
038 }

```

Bordes con imágenes, se utiliza con extensiones

border-image: url(fichero) desplazamiento / ancho / excede-frontera / repetición

Fichero del borde con imagen

border-image-source: url(fichero)

Para indicar los valores de los desplazamiento de los bordes

border-image-slice: superior derecho inferior izquierdo

Para establecer en ancho del borde

border-image-width: superior derecho inferior izquierdo

Para establecer cuanto se sobrepasa el borde

border-image-outset: superior derecho inferior izquierdo

Para indicar como se escala la imagen de forma horizontal y vertical

border-image-repeat:streich|repeat|round|space [streich|repeat|round|space]

La primera aparición se aplica a la escala horizontal y la segunda a la vertical, si ésta no se indica se toma la primera.

Con **streich** se estira la imagen

Con **repeat** se repite la imagen

Con **round**se escala para que la imagen se muestre un número entero de veces.

Con **space** la imagen se muestra un número entero de veces y se ajustan los espacios.

Ejemplo-02-140.css

```

001 #muerta{
002 border: 45px solid transparent;
003 margin:30px;
004 border-image-source:url("../imagenes/borde.png");
005 border-image-width:auto;
006 border-image-repeat:round;
007 border-image-outset:1px;
008 border-image-slice:37% 32% 37% 32%
009 }
010 #acueducto{
011 margin:30px;
012 border: 30px solid transparent;
013 -moz-border-image:url("../imagenes/borde04.png") 27;
014 -webkit-border-image:url("../imagenes/borde04.png") 27;
015 -o-border-image:url("../imagenes/borde04.png") 27;
016 border-image:url("../imagenes/borde04.png") 27;
017 }
018 #linajes{
019 margin:30px;

```

```

020 border: 30px solid transparent;
021 -moz-border-image:url("../imagenes/borde.png") 27 repeat;
022 -webkit-border-image:url("../imagenes/borde.png") 27 repeat;
023 -o-border-image:url("../imagenes/borde.png") 27 repeat;
024 border-image:url("../imagenes/borde.png") 27 repeat;
025 }
026 #santiago{
027 margin:30px;
028 border: 30px solid transparent;
029 -moz-border-image:url("../imagenes/borde.png") 27 round;
030 -webkit-border-image:url("../imagenes/borde.png") 27 round;
031 -o-border-image:url("../imagenes/borde.png") 27 round;
032 border-image:url("../imagenes/borde.png") 27 round;
033 }
034 }
035 #salto{
036 margin:30px;
037 border: 30px solid transparent;
038 -moz-border-image:url("../imagenes/borde.png") 27 space;
039 -webkit-border-image:url("../imagenes/borde.png") 27 space;
040 -o-border-image:url("../imagenes/borde.png") 27 space;
041 border-image:url("../imagenes/borde.png") 27 space;
042 }
043 #cordón{
044 margin:30px;
045 border: 40px solid transparent;
046 margin:30px;
047 border-image-source:linear-gradient(to top, red, yellow);
048 border-image-width:29px;
049 border-image-repeat:round;
050 border-image-outset:100%;
051 border-image-slice:40% 40% 42% 40%
052 }
053 #parral{
054 margin:30px;
055 border: 40px solid transparent;
056 margin:30px;
057 border-image-source:repeating-linear-gradient(to top, red, yellow);
058 border-image-width:29px;
059 border-image-repeat:round;
060 border-image-outset:100%;
061 border-image-slice:40% 40% 42% 40%
062 }

```

Border con sombra

box-shadow: *desplazamiento-horizontal desplazamiento-vertical difuminado color;*

Por orden de aparición, los valores que se indican en box-shadow son:

- ❖ Desplazamiento horizontal de la sombra: La sombra de un elemento suele estar un poco desplazada con respecto al elemento que la produce y su posición será en función del ángulo con el que llegue la luz. Valor positivo sombra a la derecha. Valor negativo sombra a la izquierda.
- ❖ Desplazamiento vertical de la sombra: El segundo valor que colocamos en el atributo box-shadow es el desplazamiento vertical de la sombra con respecto a la posición del elemento que la produce. Valores positivos indican que la sombra aparecerá hacia abajo del elemento. Valores negativos harán que la sombra aparezca desplazada un poco hacia arriba.
- ❖ Difuminado: El tercer valor indica cuánto queremos que esté difuminado el borde de la sombra. Si el difuminado fuera cero o negativo, querría decir que la sombra no tiene ningún difuminado y aparece totalmente definida
- ❖ Color de la sombra: El último atributo que se indica en el atributo box-shadow es el color de la sombra.

Si el fondo es oscuro y los bordes y la sombra clara parecen un resplandor.

se pueden poner varias sombras separadas por comas

ejemplo-02-141.css

001	#muerta{
002	margin:30px;
003	border:2px solid orange;
004	box-shadow: 5px 9px 3px #F00;
005	}
006	#acueducto{
007	margin:30px;
008	border:2px solid orange;
009	box-shadow: -5px -9px 3px #F00;
010	}
011	#linajes{
012	margin:30px;
013	border:2px solid orange;
014	box-shadow: 5px -9px 3px #F00;
015	}
016	#santiago{
017	margin:30px;
018	border:2px solid orange;
019	box-shadow: -5px 9px 3px #F00;
020	}
021	#salto{
022	margin:30px;
023	border:2px solid orange;
024	box-shadow: 5px 9px 3px #F00, -5px -9px 3px #F00;
025	}
026	#principe{
027	margin:30px;
028	border:2px solid orange;
029	box-shadow: 5px 9px 2px #F00, 15px 19px 8px #0F0;
030	}

Borde al inicio del texto, solo en Firefox

border-inline-start: tamaño estilo color

Ancho del borde al inicio del texto

border-inline-start-width:tamaño

Estilo del borde al inicio del texto, se le aplican los estilos visto anteriormente

border-inline-start-style:estilo

Color del borde al inicio del texto

border-inline-start-color:color

Borde al final del texto

border-inline-end: tamaño estilo color

Ancho del borde al final del texto

border-inline-end-width: tamaño

Estilo del borde al final del texto, se le aplican los estilos visto anteriormente

border-inline-end-style:estilo

Color del borde al final del texto

border-inline-end-color:color

001	#muerta{
002	margin:30px;

```

003 border-inline-end: 5px dashed blue;
004 }
005 #acueducto{
006 margin:30px;
007 border-inline-start: 5px dotted green;
008 }
009 #linajes{
010 margin:30px;
011 border-inline-end-width: 5px;
012 border-inline-end-style:ridge ;
013 border-inline-end-color:red;;
014 }
015 #santiago{
016 margin:30px;
017 border-inline-start-width: 5px;
018 border-inline-start-style: groove;
019 border-inline-start-color:brown;
020 }

```

Color de la sobrelínea

outline-color:color

Estilo de la sobrelínea

outline-style: none | solid | hidden | dotted | dashed | double | groove | inset | outset

Ancho de la sobrelínea

outline-width:ancho

Características de la sobrelínea

outline: color estilo ancho

ejemplo-02-143.css

```

001 #muerta{
002 margin:30px;
003 border:5px solid orange;
004 outline:13px dotted red;
005 }
006 #acueducto{
007 margin:30px;
008 border:5px solid green;
009 outline-width:13px;
010 outline-style:groove;
011 outline-color:blue;
012 }

```

8.- Estilos para Cajas

a.- El **flexbox** caja flexible

Vamos a indicar como se visualizan los elementos hijos.

display:flex | inline-flex

Con **flex** se genera un cuadro contenedor flexible a nivel de bloque.

Con **inline-flex** se genera un cuadro contenedor de nivel en línea.

ejemplo-02-159.css

```

001 .inicio{display:flex;}
002 .inicio>div{border:2px solid blue;}
003 .izquierda{display:inline-flex;}
004 .izquierda>div{border:2px solid red;}

```

Alineación en una caja de elementos secundarios, alineación vertical del texto si no tiene elementos secundarios. Se utiliza con extensiones en chrome.

Alinea verticalmente el contenido de un elemento al inicio, al final, a la base, a la base, al centro.

ejemplo-02-148.css para elementos secundarios

```

001 #primero>div {
002     display:flex;
003     width:700px;
004     height:300px;
005     border:4px solid blue;
006 }
007 #primero* div {
008     width:300px;
009     height:200px;
010     border:2px solid red;
011 }
012 .inicio{
013     -webkit-align-items:flex-start;
014     align-items:flex-start;
015 }
016 .izquierda{
017     -webkit-align-items:center;
018     align-items:center;
019 }
020 .derecha{
021     -webkit-align-items:flex-end;
022     align-items:flex-end;
023 }
024 .centro{
025     -webkit-align-items:baseline;
026     align-items:baseline;
027 }
028 .justo{
029     -webkit-align-items:stretch;
030     align-items:stretch;
031 }

```

ejemplo-02-149.css para texto

```

001 #primero>div {
002     display:flex;
003     width:700px;
004     height:300px;
005     border:4px solid blue;
006 }
007 .inicio{
008     -webkit-align-items:flex-start;
009     align-items:flex-start;
010 }
011 .izquierda{
012     -webkit-align-items:center;
013     align-items:center;
014 }
015 .derecha{
016     -webkit-align-items:flex-end;
017     align-items:flex-end;
018 }
019 .centro{
020     -webkit-align-items:baseline;
021     align-items:baseline;
022 }
023 .justo{
024     -webkit-align-items:stretch;
025     align-items:stretch;
026 }

```

Alineación para modificar align-items, se aplica a los elementos hijos.

ejemplo-02-153.css para elementos secundarios

```

001 #primero>div {
002     display:flex;
003     width:700px;
004     height:300px;
005     border:4px solid blue;
006     -webkit-align-items:center;
007     align-items:center;
008 }
009 #primero* div {
010     display:flex;
011     width:300px;
012     height:200px;
013     border:2px solid red;
014 }
015
016 #primero.inicio div {
017     -webkit-align-self:flex-start;
018     align-self:flex-start;
019 }
020 #primero.izquierdadiv{
021     -webkit-align-self:center;
022     align-self:center;
023 }
024 #primero.derecha div {
025     -webkit-align-self:flex-end;
026     align-self:flex-end;
027 }
028
029 #primero.centro div {
030     -webkit-align-self:baseline;
031     align-self:baseline;
032 }
033 #primero.justodiv {
034     -webkit-align-self:stretch;
035     align-self:stretch;
036 }

```

Para indicar el sentido en que se visualizan los hijos. Con extensiones para chrome

flex-direction:row | row-reverse | column | column-reverse

Se debe poner previamente display:flex.

Con **row** se muestran en fila

Con **row-reverse** se muestran en fila en sentido inverso

Con **column** se muestran en columnas

Con **column-reverse** se muestran en columnas en sentido inverso.

Para texto se alinean todos a la parte superior excepto con column-reverse que se alinea a la parte inferior.

ejemplo-02-147.css

```

001 #primero > div {
002     display:flex;
003     width:700px;
004     height:300px;
005     border:4px solid blue;
006 }
007 #primero* div {
008     width:300px;
009     height:200px;
010     border:2px solid red;
011 }
012 .inicio{
013     -webkit-flex-direction:row;

```

```

014     flex-direction:row;
015 }
016 .izquierda{
017     -webkit-flex-direction:row-reverse;
018     flex-direction:row-reverse;
019 }
020 .derecha{
021     -webkit-flex-direction:column-reverse;
022     flex-direction:column-reverse;
023 }
024 .justo{
025     -webkit-flex-direction:column;
026     flex-direction:column;
027 }

```

ejemplo-02-155.css para texto, igual que anterior pero quitando las líneas de la 7 a la 11, ambas incluidas.

Para indicar si se muestra en línea o no. Extensiones para chrome

flex-wrap:nowrap | wrap | wrap-reverse

Con **wrapse** muestran en bloques alineado a la parte superior.

Con **nowrap** se muestran en bloques alineado a la parte inferior

Con **wrap-reverse** como wrap.

ejemplo-02-160.css

```

001 #primero>div {
002     display:flex;
003     width:700px;
004     height:250px;
005     border:4px solid blue;
006 }
007 #primero* div {
008     width:300px;
009     height:200px;
010     border:2px solid red;
011 }
012 .inicio{
013     -webkit-flex-wrap:wrap;
014     flex-wrap:wrap;
015 }
016 .izquierda{
017     -webkit-flex-wrap:wrap-reverse;
018     flex-wrap:wrap-reverse;
019 }
020 .derecha{
021     -webkit-flex-wrap:nowrap;
022     flex-wrap:nowrap;
023 }

```

Resumen de las dos anteriores. Se utiliza con extensiones para chrome

flex-flow: v-flex-direction v-flex-wrap

ejemplo-02-161c.css

```

001 #primero>div {
002     display:flex;
003     width:700px;
004     height:500px;
005     border:4px solid blue;
006 }
007 #primero* div {
008     width:300px;
009     height:200px;
010     border:2px solid red;
011 }

```

```

011 }
012 .inicio{
013     -webkit-flex-flow:row wrap;
014     flex-flow:row warp;
015 }
016 .izquierda{
017     -webkit-flex-flow:row-reverse wrap-reverse;
018     flex-flow:row-reverse wrap-reverse;
019 }
020 .derecha{
021     -webkit-flex-flow:column-reversenowrap;
022     flex-flow:column-reversenowrap;
023 }
024 .justo{
025     -webkit-flex-flow:columnnowrap;
026     flex-flow:columnnowrap;
027 }

```

Para indicar en qué orden se visualizan los elementos.

order:número

ejemplo-02-162.css

```

001 #primero>div {
002     display:flex;
003     width:1300px;
004     height:250px;
005     border:4px solid blue;
006 }
007 #primero* div {
008     width:300px;
009     height:200px;
010     border:2px solid red;
011 }
012 .inicio{
013     -webkit-flex-wrap:wrap;
014     flex-wrap:wrap;
015 }
016 .derecha{
017     -webkit-flex-wrap:wrap-reverse;
018     flex-wrap:wrap-reverse;
019 }
020 .justo{
021     -webkit-flex-wrap:nowrap;
022     flex-wrap:nowrap;
023 }
024 #ini-1{order:3;}
025 #ini-2{order:4;}
026 #ini-3{order:1;}
027 #ini-4{order:2;}
028 #drch-1{order:2;}
029 #drch-2{order:3;}
030 #drch-3{order:1;}
031 #jst-1{order:3;}
032 #jst-2{order:1;}
033 #jst-3{order:2;}

```

La propiedad **flex-basis** especifica la base flexible, la cual es el **tamaño inicial** de un elemento flexible, prevalece sobre la propiedad **width**. Esta propiedad determina el tamaño de una caja de contenidos a no ser que se haya especificado de otra forma usando **box-sizing**. El tamaño de visualización puede variar al aplicar las propiedades **flex-grow** y **flex-shrink**. Con extensiones en chrome.

flex-basic:content|auto|ancho

ejemplo-02-164.css

```

001 #primero>div {
002     display:flex;
003     width:750px;
004     height:300px;
005     border:4px solid blue;
006 }
007 #primero* div {
008     width:100px;
009     height:250px;
010     border:2px solid red;
011 }
012 #ini-1{
013     -webkit-flex-basis:200px;
014     flex-basis:200px;
015 }
016 #ini-2{
017     -webkit-flex-basis:150px;
018     flex-basis:150px;
019 }
020 #ini-3{
021     -webkit-flex-basis:150px;
022     flex-basis:150px;
023 }
024 #ini-4{
025     -webkit-flex-basis:250px;
026     flex-basis:250px;
027 }
028 #drch-1{
029     -webkit-flex-basis:200px;
030     flex-basis:200px;
031 }
032 #drch-2{
033     -webkit-flex-basis:250px;
034     flex-basis:250px;
035 }
036 #drch-3{
037     -webkit-flex-basis:300px;
038     flex-basis:300px;
039 }
040 #jst-1{
041     -webkit-flex-basis:150px;
042     flex-basis:150px;
043 }
044 #jst-2{
045     -webkit-flex-basis:200px;
046     flex-basis:200px;
047 }
048 #jst-3{
049     -webkit-flex-basis:400px;
050     flex-basis:400px;
051 }

```

La propiedad `flex-grow` especifica el factor de crecimiento de un elemento flexible, cuando la suma de los tamaños de los elementos flexibles es menor que el tamaño del contenedor. Se especifica qué cantidad de espacio debe ocupar el elemento dentro del contenedor flexible. Con extensiones en chrome.

flex-grow:número

El número indica en cuantas unidades se incrementa el tamaño del elemento. Sabiendo que la unidad es el espacio sobreante del padre dividido entre la suma de los números de las propiedades `flex-grow` de los hijos.

ejemplo-02-163.css

```

001 #primero>div {
002     display:flex;
003     width:750px;
004     height:300px;

```

```

005     border:4px solid blue;
006 }
007 #primero* div {
008     width:100px;
009     height:250px;
010     border:2px solid red;
011 }
012 #ini-1{
013     -webkit-flex-grow:1;
014     flex-grow:1;
015 }
016 #ini-2{
017     -webkit-flex-grow:0;
018     flex-grow:0;
019 }
020 #ini-3{
021     -webkit-flex-grow:2;
022     flex-grow:2;
023 }
024 #ini-4{
025     -webkit-flex-grow:1;
026     flex-grow:1;
027 }
028 #drch-1{
029     -webkit-flex-grow:1;
030     flex-grow:1;}
031
032 #drch-2{
033     -webkit-flex-grow:2;
034     flex-grow:2;
035 }
036 #drch-3{
037     -webkit-flex-grow:1;
038     flex-grow:1;
039 }
040 #jst-1{
041     -webkit-flex-grow:2;
042     flex-grow:2;
043 }
044 #jst-2{
045     -webkit-flex-grow:3;
046     flex-grow:3;
047 }
048 #jst-3{
049     -webkit-flex-grow:2;
050     flex-grow:2;
051 }

```

Ejemplo-02-163b.css

```

001 #primero>div {
002     display:flex;
003     width:750px;
004     height:300px;
005     border:4px solid blue;
006 }
007 #primero* div {
008     width:100px;
009     height:250px;
010     border:2px solid red;
011 }
012 #ini-1{
013     -webkit-flex-basis:100px;
014     flex-basis:100px;
015     -webkit-flex-grow:2;
016     flex-grow:2;
017 }
018 }
019 #ini-2{
020     -webkit-flex-basis:150px;
021     flex-basis:150px;
022     -webkit-flex-grow:1;
023     flex-grow:1;

```

024	}
025	
026	#ini-3{
027	-webkit-flex-basis:100px;
028	flex-basis:100px;
029	-webkit-flex-grow:0;
030	flex-grow:0;
031	
032	}
033	#ini-4{
034	-webkit-flex-basis:250px;
035	flex-basis:250px;
036	-webkit-flex-grow:1;
037	flex-grow:1;
038	}
039	#drch-1{
040	-webkit-flex-basis:200px;
041	flex-basis:200px;
042	-webkit-flex-grow:0;
043	flex-grow:0;
044	}
045	#drch-2{
046	-webkit-flex-basis:250px;
047	flex-basis:250px;
048	-webkit-flex-grow:2;
049	flex-grow:2;
050	}
051	#drch-3{
052	-webkit-flex-basis:200px;
053	flex-basis:200px;
054	-webkit-flex-grow:1;
055	flex-grow:1;
056	}
057	#jst-1{
058	-webkit-flex-basis:150px;
059	flex-basis:150px;
060	-webkit-flex-grow:3;
061	flex-grow:3;
062	}
063	#jst-2{
064	-webkit-flex-basis:200px;
065	flex-basis:200px;
066	-webkit-flex-grow:1;
067	flex-grow:1;
068	}
069	#jst-3{
070	-webkit-flex-basis:400px;
071	flex-basis:400px;
072	-webkit-flex-grow:0;
073	flex-grow:0;
074	}

La propiedad de CSS `flex-shrink` especifica el factor de contracción flexible de un elemento flexible. Los artículos flexibles se contraerán para llenar el contenedor de acuerdo con el número de contracción flexible, cuando el ancho predeterminado de los artículos flexibles es más ancho que el contenedor flexible. Con extensiones en chrome.

flex-shrink:*número*

El número indica en cuantas unidades se decrementa el tamaño del elemento. Sabiendo que la unidad es el espacio deficitario del padre dividido entre la suma de los números de las propiedades `flex-shrink` de los hijos.

ejemplo-02-165.css

001	#primero>div {
002	display:flex;

```

003     width:750px;
004     height:300px;
005     border:4px solid blue;
006 }
007 #primero* div {
008     width:300px;
009     height:250px;
010     border:2px solid red;
011 }
012 #ini-1{
013     -webkit-flex-shrink:1;
014     flex-shrink:1;
015 }
016 #ini-2{
017     -webkit-flex-shrink:0;
018     flex-shrink:0;
019 }
020 #ini-3{
021     -webkit-flex-shrink:2;
022     flex-shrink:2;
023 }
024 #ini-4{
025     -webkit-flex-shrink:1;
026     flex-shrink:1;
027 }
028 #drch-1{
029     -webkit-flex-shrink:1;
030     flex-shrink:1;};
031
032 #drch-2{
033     -webkit-flex-shrink:2;
034     flex-shrink:2;
035 }
036 #drch-3{
037     -webkit-flex-shrink:1;
038     flex-shrink:1;
039 }
040 #jst-1{
041     -webkit-flex-shrink:2;
042     flex-shrink:2;
043 }
044 #jst-2{
045     -webkit-flex-shrink:3;
046     flex-shrink:3;
047 }
048 #jst-3{
049     -webkit-flex-shrink:2;
050     flex-shrink:2;
051 }

```

ejemplo-02-165b.css

```

001 #primero>div {
002     display:flex;
003     width:750px;
004     height:300px;
005     border:4px solid blue;
006 }
007 #primero* div {
008     width:100px;
009     height:250px;
010     border:2px solid red;
011 }
012 #ini-1{
013     -webkit-flex-basis:200px;
014     flex-basis:200px;
015     -webkit-flex-shrink:2;
016     flex-shrink:2;
017 }
018 }
019 #ini-2{
020     -webkit-flex-basis:300px;
021     flex-basis:300px;

```

022	-webkit-flex-shrink:1;
023	flex-shrink:1;
024	
025	}
026	#ini-3{
027	-webkit-flex-basis:250px;
028	flex-basis:250px;
029	-webkit-flex-shrink:0;
030	flex-shrink:0;
031	
032	}
033	#ini-4{
034	-webkit-flex-basis:250px;
035	flex-basis:250px;
036	-webkit-flex-shrink:1;
037	flex-shrink:1;
038	}
039	#drch-1{
040	-webkit-flex-basis:400px;
041	flex-basis:400px;
042	-webkit-flex-shrink:0;
043	flex-shrink:0;
044	}
045	#drch-2{
046	-webkit-flex-basis:250px;
047	flex-basis:250px;
048	-webkit-flex-shrink:2;
049	flex-shrink:2;
050	}
051	#drch-3{
052	-webkit-flex-basis:300px;
053	flex-basis:300px;
054	-webkit-flex-shrink:1;
055	flex-shrink:1;
056	}
057	#jst-1{
058	-webkit-flex-basis:350px;
059	flex-basis:350px;
060	-webkit-flex-shrink:3;
061	flex-shrink:3;
062	}
063	#jst-2{
064	-webkit-flex-basis:200px;
065	flex-basis:200px;
066	-webkit-flex-shrink:1;
067	flex-shrink:1;
068	}
069	#jst-3{
070	-webkit-flex-basis:400px;
071	flex-basis:400px;
072	-webkit-flex-shrink:0;
073	flex-shrink:0;
074	}

La propiedad flex es una agrupación de las propiedades flex-grow, flex-shrink, y flex-basis. Con extensiones en chrome.

flex:[valor-crece valor-disminuye [longitud|auto | content]]| none | initial | auto | número

flex:initial equivale a **flex:0 1 auto;**
flex:auto equivale a **flex:1 1 auto**
flex:nº equivale a **flex:nº 1 0**

001	#primero>div {
002	display:flex;
003	width:750px;
004	height:300px;

```

005     border:4px solid blue;
006 }
007 #primero* div {
008     width:300px;
009     height:250px;
010     border:2px solid red;
011 }
012 #ini-1{
013     -webkit-flex:1 0 100px;
014     flex:1 0 100px;
015 }
016 #ini-2{
017     -webkit-flex:0 0 200px;
018     flex:0 0 200px;
019 }
020 #ini-3{
021     -webkit-flex:2 0 150px;
022     flex:2 0 150px;
023 }
024 #ini-4{
025     -webkit-flex:1 0 175px;
026     flex:1 0 175px;
027 }
028 #drch-1{
029     -webkit-flex:0 1 400px;
030     flex:0 3 400px;
031 }
032 #drch-2{
033     -webkit-flex:0 2 300px;
034     flex:0 2 300px;
035 }
036 #drch-3{
037     -webkit-flex:0 1 200px;
038     flex:0 1 200px;
039 }
040 #jst-1{
041     -webkit-flex:2 0 100px;
042     flex:2 0 100px;
043 }
044 #jst-2{
045     -webkit-flex:3 0 110px;
046     flex:3 0 110px;
047 }
048 #jst-3{
049     -webkit-flex:2 0 150px;
050     flex:2 0 150px;
051 }

```

ejemplo-02-157.css.

```

001 #primero>div {
002     width:900px;
003     height:400px;
004     display:flex;
005     border:4px solid blue;
006 }
007 #primero* div {
008 /*display:flex;*/
009     border:4px solid red;
010     width:150px;
011 }
012 #ini-1{
013     -webkit-flex:none;
014     -ms-flex:none;
015     flex:none;
016 }
017 #ini-2{
018     -webkit-flex:none;
019     -ms-flex:none;
020     flex:none;
021 }
022 #izq-1{
023     -webkit-flex:1 0 auto;

```

```

024     -ms-flex:1 0 auto;
025     flex:1 0 auto;
026 }
027 #izq-2{
028     -webkit-flex:1 0 auto;
029     -ms-flex:1 0 auto;
030     flex:1 0 auto;
031 }
032 #drch-1{
033     -webkit-flex:0 1 content;
034     -ms-flex:0 1 content;
035     flex:0 1 content;
036 }
037 #drch-2{
038     -webkit-flex:0 1 content;
039     -ms-flex:0 1 content;
040     flex:0 1 content;
041 }
042 #jst-1{
043     -webkit-flex:2 0 2em;
044     -ms-flex:2 0 2em;
045     flex:2 0 2em;
046 }
047 #jst-2{
048     -webkit-flex:2 0 3em;
049     -ms-flex:2 0 3em;
050     flex:2 0 3em;
051 }

```

La propiedad **justify-content** define cómo el navegador distribuye el espacio entre los elementos de contenido a lo largo del eje principal de su contenedor. Con extensiones para Chrome

justify-content: flex-start | flex-end | center | space-between | space-around

Con **flex-start** al inicio de la línea.

Con **flex-end** al final de la línea

Con **center** centrado en la línea

Con **space-between** añade espacios en medio para completar la línea.

Con **space-around** añade espacios para completar la línea.

ejemplo-02-167.css

```

001 #primero>div {
002     display:flex;
003     width:700px;
004     height:250px;
005     border:4px solid blue;
006 }
007 #primero* div {
008     width:300px;
009     height:200px;
010     border:2px solid red;
011 }
012 .inicio{
013     -webkit-justify-content:flex-start;
014     justify-content:flex-start;
015 }
016 .izquierda{
017     -webkit-justify-content:flex-end;
018     justify-content:flex-end;
019 }
020 .derecha{
021     -webkit-justify-content:center;
022     justify-content:center;
023 }
024 .centro{
025     -webkit-justify-content:space-between;

```

```

026     justify-content:space-between;
027 }
028 .justo{
029     -webkit-justify-content:space-around;
030     justify-content:space-around;
031 }

```

La propiedad align-content define cómo el navegador distribuye el espacio entre los elementos de contenido a lo largo del eje transversal de su contenedor, que sirve como un contenedor de caja flexible. Con extensiones para Chrome

align-content:flex-start | flex-end | center | space-between | space-around | strech

Con **flex-start** a la parte superior

Con **flex-end** a la parte inferior

Con **center** centrado

Con **space-between** los elementos aparecen distribuidos uniformemente: al principio, en el centro y al final del contenedor flex.

Con **space-around** los elementos aparecen distribuidos uniformemente, y con un espacio igual entre ellos.

Con **strech** los elementos aparecen estirados para ocupar el espacio restante

ejemplo-02-168.css

```

001 #primero>div {
002     display:flex;
003     width:700px;
004     height:250px;
005     border:4px solid blue;
006 }
007 #primero* div {
008     width:300px;
009     height:200px;
010     border:2px solid red;
011 }
012 .inicio{
013     -webkit-flex-wrap: wrap;
014     flex-wrap: wrap;
015     -webkit-justify-content:flex-start;
016     justify-content:flex-start;
017     -webkit-align-content:flex-start;
018     align-content:flex-start;
019 }
020 .izquierda{
021     -webkit-flex-wrap: wrap;
022     flex-wrap: wrap;
023     -webkit-justify-content:flex-end;
024     justify-content:flex-end;
025     -webkit-align-content:flex-end;
026     align-content:flex-end;
027 }
028 .derecha{
029     -webkit-flex-wrap: wrap;
030     flex-wrap: wrap;
031     -webkit-justify-content:center;
032     justify-content:center;
033     -webkit-align-content:center;
034     align-content:center;
035 }
036 .centro{
037     -webkit-flex-wrap: wrap;
038     flex-wrap: wrap;
039     -webkit-justify-content:space-between;
040     justify-content:space-between;
041     -webkit-align-content:space-between;
042     align-content:space-between;

```

```

043 }
044 .justo{
045     -webkit-flex-wrap: wrap;
046     flex-wrap: wrap;
047     -webkit-justify-content: space-around;
048     justify-content: space-around;
049     -webkit-align-content: space-around;
050     align-content: space-around;
051 }
052 .final{
053     -webkit-flex-wrap: wrap;
054     flex-wrap: wrap;
055     -webkit-align-content: strech;
056     align-content: strech;
057 }

```

b.- El **grid** es una forma de enmaquetar las capas a través de una rejilla, se utiliza cuando tengamos varias filas, para una única fila utilizar mejor flexbox.

Para indicar que vamos a poner la estructura del grid deberemos poner:

display:grid|grid-inline

Los elementos se van a visualizar en modo bloque o caja o en línea.

Para establecer el tamaño de cada una de las columnas Con extensiones para Microsoft edge y explorer

grid-template-columns:none| lista-tamaños

Para indicar el tamaño se puede poner un porcentaje, un número con las unidades, **max-content** tamaño mayor, **min-content** para tamaño menor, **minmax(rango)** para indicar el rango de valores entre los que estará el tamaño, **repeat(número-veces, lista-tamaños)** para tener varias filas o columnas con las mismas características. Se puede utilizar como unidad **fr** que representa una unidad flexible con espacio libre.

Para establecer el tamaño de cada una de las filas. Con extensiones para Microsoft edge y explorer

grid-template-rows:none| lista-tamaños

Para establecer una lista de nombre que nos permitan identificar mejor las áreas de la rejilla.

grid-template-areas:lista-nombres

Resumen de las propiedades anteriores en una sola.

**grid-template: v-grid-template-rows / v-grid-template-columns
v-grid-template-areas**

Para indicar la fila en que se muestra el elemento. Con extensiones para explorer.

grid-row:número-fila

Para indicar la columna en la que se muestra el elemento. Con extensiones para explorer.

grid-column:número-columna

ejemplo-02-170.css

```

001 #primero{
002     display:grid;
003     width:750px;
004     border:4px solid blue;
005     -ms-grid-template-columns:250px 250px 250px ;
006     grid-template-columns:250px 250px 250px;
007     -ms-grid-template-rows:1fr min-content;
008     grid-template-rows:1fr min-content;
009 }
010 #primero>div {
011     border:2px solid red;
012 }
013 .inicio{
014     -ms-grid-column:1;
015     grid-column:1;
016     -ms-grid-row:1;
017     grid-row:1;
018 }
019 .izquierda{
020     -ms-grid-column:2;
021     grid-column:2;
022     -ms-grid-row:1;
023     grid-row:1;
024 }
025 .derecha{
026     -ms-grid-column:3;
027     grid-column:3;
028     -ms-grid-row:1;
029     grid-row:1;
030 }
031 .centro{
032     -ms-grid-column:1;
033     grid-column:1;
034     -ms-grid-row:2;
035     grid-row:2;
036 }
037 .justo{
038     -ms-grid-column:2;
039     grid-column:2;
040     -ms-grid-row:2;
041     grid-row:2;
042 }
043 .final{
044     -ms-grid-column:3;
045     grid-column:3;
046     -ms-grid-row:2;
047     grid-row:2;
048 }

```

Para indicar el tamaño de las columnas, prevalece sobre grid-template-columns. Con extensiones para explorer.

grid-auto-columns:tamaño

Para indicar el tamaño de las filas, prevalece sobre grid-template-rows. Con extensiones para explorer

grid-auto-rows:tamaño

Para indicar como se colocan los elementos para los que no se ha indicado su posición en la rejilla.

grid-auto-flow:row | column | dense

Con **row** se completan las filas y si es necesario se añaden más filas.

Con **column** se completan las columnas y si es necesario se añaden más columnas.

Con **dense** se rellenan los primeros huecos existentes.

ejemplo-02-171.css

```

001 #leyendas{
002     display:grid;
003     width:1200px;
004     border:4px solid blue;
005     -ms-grid-auto-columns:300px;
006     grid-auto-columns:300px;
007     -ms-grid-auto-rows:650px;
008     grid-auto-rows:650px;
009     -ms-grid-auto-flow:row;
010     grid-auto-flow:row;
011 }
012 #leyendas>div {
013     border:2px solid red;
014 }
015 #muerta{
016     -ms-grid-column:1;
017     grid-column:1;
018     -ms-grid-row:1;
019     grid-row:1;
020 }
021 #acueducto{
022     -ms-grid-column:4;
023     grid-column:4;
024     -ms-grid-row:1;
025     grid-row:1;
026 }
027 #linajes{
028     -ms-grid-column:3;
029     grid-column:3;
030     -ms-grid-row:4;
031     grid-row:4;
032 }
033 #santiago{
034     -ms-grid-column:1;
035     grid-column:1;
036     -ms-grid-row:3;
037     grid-row:3;
038 }
039 #salto{
040     -ms-grid-column:2;
041     grid-column:2;
042     -ms-grid-row:2;
043     grid-row:2;
044 }
045 #principe{
046     -ms-grid-column:3;
047     grid-column:3;
048     -ms-grid-row:5;
049     grid-row:5;
050 }

```

Para indicar en que fila empieza a mostrarse un elemento, cuando ocupa más de una celda

grid-row-start:*número-fila*

Para indicar en que columna empieza a mostrarse un elemento, cuando ocupa más de una celda

grid-column-start:*número-columna*

Para indicar antes de que fila termina de mostrarse un elemento, cuando ocupa más de una celda.

grid-row-end:*número-fila*

Para indicar antes de que columnas termina de mostrarse un elemento, cuando ocupa más de una celda.

grid-column-end:*número-columna*

Resumen de las cuatro anteriores.

**grid-area:*v-grid-row-star* / *v-grid-column-start* / *v-grid-row-end* /
*v-grid-column-end***

También podemos utilizar las propiedades anteriores poniendo

Para indicar la posición de la fila y el número de filas que se ocupa

grid-row: *número-fila* / *span número-filas-ocupa*

Para indicar la posición de la columna y el número de columnas que se ocupa

grid-column: *número-columna* / *span número-columnas-ocupa*

ejemplo-02-172.css

```
001 #leyendas{  
002     display:grid;  
003     width:1200px;  
004     border:4px solid blue;  
005     -ms-grid-auto-columns:300px;  
006     grid-auto-columns:300px;  
007     -ms-grid-auto-rows:650px;  
008     grid-auto-rows:650px;  
009     -ms-grid-auto-flow:column;  
010     grid-auto-flow:column;  
011 }  
012 #leyendas>div {  
013     border:2px solid red;  
014 }  
015 #muerta{  
016     -ms-grid-column-start:1;  
017     grid-column-start:1;  
018     -ms-grid-row:1;  
019     grid-row:1;  
020     -ms-grid-column-end:4;  
021     grid-column-end:4;  
022 }  
023 #acueducto{  
024     -ms-grid-column:4;  
025     grid-column:4;  
026     -ms-grid-row:1;  
027     grid-row:1;  
028 }  
029 #linajes{ grid-area:4 / 3 / 6 /5; }  
030 #santiago{  
031     -ms-grid-column:1;  
032     grid-column:1;  
033     -ms-grid-row:3;  
034     grid-row:3;  
035 }  
036 #salto{  
037     -ms-grid-column:2;  
038     grid-column:2;  
039     -ms-grid-row:2;  
040     grid-row:2;  
041     grid-row-end:5;  
042 }  
043 #principe{  
044     -ms-grid-column:1;  
045     grid-column:1;  
046     -ms-grid-row:5;
```

047	grid-row:5;
048	}

Para establecer el espacio de separación entre las filas

grid-row-gap:*ancho-separacion-filas*

Para establecer el espacio de separación entre las columnas

grid-column-gap:*ancho-separación-columnas*

Resumen de las dos anteriores.

grid-gap: *ancho-separacion-filas* *ancho-separación-columnas*

ejemplo-02-173.css

001	#leyendas {
002	display:grid;
003	width:1400px;
004	border:4px solid blue;
005	-ms-grid-template:500px 480px 320px 350px 320px / 300px 250px 350px 400px;
006	grid-template:500px 480px 320px 350px 320px / 300px 250px 350px 400px;
007	-ms-grid-auto-flow:dense;
008	grid-auto-flow:dense;
009	grid-column-gap:20px;
010	grid-row-gap:30px;
011	}
012	#leyendas>div {
013	border:2px solid red;
014	}
015	#muerta {
016	-ms-grid-column-start:1;
017	grid-column-start:1;
018	-ms-grid-row:1;
019	grid-row:1;
020	-ms-grid-column-end:4;
021	grid-column-end:4;
022	}
023	#acueducto {
024	-ms-grid-column:4;
025	grid-column:4;
026	-ms-grid-row:1;
027	grid-row:1;
028	}
029	#linajes {
030	-ms-grid-area:4 / 3 / 6 /5;
031	grid-area:4 / 3 / 6 /5;
032	}
033	#santiago {
034	-ms-grid-column:1;
035	grid-column:1;
036	-ms-grid-row:3;
037	grid-row:3;
038	}
039	#salto {
040	-ms-grid-column:2;
041	grid-column:2;
042	-ms-grid-row:2;
043	grid-row:2;
044	grid-row-end:5;
045	}
046	#principe {
047	-ms-grid-column:1;
048	grid-column:1;
049	-ms-grid-row:5;
050	grid-row:5;

051 }

Resumen de algunas de las anteriores

**grid:v-grid-template | v-grid-template-row / [auto-flow dense] v-grid-auto-columns
 v-grid-auto-rows [auto-flow dense]/ grid-template-columns**

ejemplo-02-174.css

```

001 #leyendas{
002     display:grid;
003     width:1400px;
004     border:4px solid blue;
005     -ms-grid:500px 480px 320px 350px 320px / 300px 250px 350px 400px;
006     grid:500px 480px 320px 350px 320px / 300px 250px 350px 400px;
007     -ms-grid-auto-flow:dense;
008     grid-auto-flow:dense;
009     grid-gap:30px 20px;
010 }
011 #leyendas>div {
012     border:2px solid red;
013 }
014 #muerta{
015     -ms-grid-column-start:1;
016     grid-column-start:1;
017     -ms-grid-row:1;
018     grid-row:1;
019     -ms-grid-column-end:4;
020     grid-column-end:4;
021 }
022 #acueducto{
023     -ms-grid-column:4;
024     grid-column:4;
025     -ms-grid-row:1;
026     grid-row:1;
027 }
028 #linajes{
029     -ms-grid-area:4 / 3 / 6 /5;
030     grid-area:4 / 3 / 6 /5;
031 }
032 #santiago{
033     -ms-grid-column:1;
034     grid-column:1;
035     -ms-grid-row:3;
036     grid-row:3;
037 }
038 #salto{
039     -ms-grid-column:2;
040     grid-column:2;
041     -ms-grid-row:2;
042     grid-row:2;
043     grid-row-end:5;
044 }
045 #principe{
046     -ms-grid-column:1;
047     grid-column:1;
048     -ms-grid-row:5;
049     grid-row:5;
050 }
```

Para establecer la alineación de los elementos de la rejilla vamos a utilizar las mismas propiedades que hemos visto para el flexbox, **justify-content** y **align-content**.

9.- Colocación o Ubicación

Respecto a que elemento se establece la posición.

position:absolute | fixed | relative | static | sticky

Permite establecer a partir de que elemento se establecer la posición de un elemento, a través de top, bottom, left, right.

Con **absolute** se toma como punto inicial la esquina superior izquierda del elemento padre (de todos).

Con **fixed** se toma como punto inicial la esquina superior izquierda del área del navegador.

Con **relative** se toma como punto inicial la posición que le corresponde inicialmente al elemento (esquina inferior izquierda del elemento anterior). Con **static** se toma su posición normal ignorando los elementos de posicionamiento.

Con **sticky** se toma como punto de partida el correspondiente al flujo normal, no se aplica.

Posición respecto a la parte superior

top:valor

Posición respecto a la parte izquierda.

left:valor

Posición respecto a la parte derecha.

right:valor

Posición respecto a la parte inferior.

bottom:valor
ejemplo-2-180.css

001	#primero{
002	position:relative;
003	top:20px;
004	left:20px;
005	border:3px solid green;
006	}
007	#leyendas{
008	position:relative;
009	top:0px;
010	left:0px;
011	
012	}
013	#primero>div {
014	width:700px;
015	height:450px;
016	border:4px solid blue;
017	position:relative;
018	top:5%;
019	left:5%;
020	}
021	#primero* div {
022	width:300px;
023	height:200px;
024	border:2px solid red;
025	
026	}
027	#ini-1{
028	position:absolute;
029	top:40px;
030	left:40px;
031	}
032	#izq-1{
033	position:absolute;
034	top:5%;
035	left:5%;

```

036 }
037 #izq-2{
038     position:absolute;
039     top:5%;
040     left:50%;
041 }
042 #drch-1{
043     position:relative;
044     top:5%;
045     left:5%;
046 }
047 #drch-2{
048     position:relative;
049     top:5%;
050     left:50%;
051 }
052 #jst-1{
053     position:relative;
054     top:0%;
055     left:5%;
056 }
057 #jst-2{
058     position:relative;
059     top:-200px;
060     left:50%;
061 }
062 #central{
063     position:sticky;
064     top:5%;
065     left:25%;
066 }
067 .final div {
068     position:fixed;
069     top:5%;
070     left:60%;
071 }

```

Para establecer el orden de visualización

z-index: valor

Cuando tenemos elementos superpuestos nos permite indicar cual se va a mostrar en primer término, el que tenga el valor mayor en esta propiedad.

ejemplo-02-181.css

```

001 #primero>div {
002     width:700px;
003     height:450px;
004     border:4px solid blue;
005 }
006 #primero* div {
007     width:300px;
008     height:200px;
009     border:2px solid red;
010 }
011 .inicio>div:first-child{
012     position:absolute;
013     top:40px;
014     left:40px;
015     background-color:yellow;
016     z-index:3;
017 }
018 .inicio>div:nth-child(2){
019     position:absolute;
020     top:190px;
021     left:140px;
022     background-color:silver;

```

```

023    z-index:2;
024 }
025 .inicio>div:last-child{
026     position:absolute;
027     top:210px;
028     left:15px;
029     background-color:brown;
030     z-index:4;
031 }
032 .inicio>div:active{
033     z-index:8;
034 }

```

Tamaños

Ancho

width:valor|auto|inherit

Ancho mínimo

min-width:valor|inherit

Ancho máximo

max-width:valor|none|inherit

Altura

height:valor|auto|inherit

Altura mínima

min-height:valor|inherit

Altura máxima

max-height:valor|none|inherit

ejemplo-02-182.css

```

001 #primero>div {
002     max-width:900px;
003     min-width:500px;
004     max-height:600px;
005     min-height:300px;
006     border:4px solid blue;
007 }
008 #primero* div {
009     width:300px;
010     height:200px;
011     border:2px solid red;
012 }

```

10.-Visibilidad

Indica de visibilidad de los elementos

visibility:visible|hidden|collapse|inherit

La propiedad 'visibility' especifica si las cajas generadas por un elemento son procesadas.

- ❖ **visible:** La caja generada es visible.

- ❖ **hidden:** La caja generada es invisible (completamente transparente, no se dibuja nada), pero sigue afectando la composición. Además, los descendientes del elemento serán visibles si tienen 'visibility: visible'.
- ❖ **collapse:** Si se usa en otros elementos que no sean filas, grupos de filas, columnas o grupos de columnas, 'collapse' tiene el mismo significado que 'hidden'.

ejemplo-02-183.css

001	#primero>div {
002	width:500px;
003	height:650px;
004	border:4px solid blue;
005	}
006	#primero* div {
007	width:300px;
008	height:200px;
009	border:2px solid red;
010	}
011	.inicio{visibility:visible;}
012	.izquierda{visibility:hidden;}
013	#izq-1{visibility:visible;}
014	.derecha{visibility:collapse;}

Elementos flotantes

float:left|right|none

Con el valor **none** el elemento será mostrado donde aparezca en el texto. Con un valor **left** el elemento será movido a la izquierda y el texto continuará por el lado derecho del elemento. Con un valor **right** el elemento será movido a la derecha y el texto cambiará de línea por el lado izquierdo del elemento

ejemplo-02-184.css

001	#primero>div {
002	width:700px;
003	height:650px;
004	border:4px solid blue;
005	}
006	#primero* div {
007	width:300px;
008	height:200px;
009	border:2px solid red;
010	}
011	#ini-1{float:right;}
012	#izq-2{float:left;}

ejemplo-02-185.css

001	#asturias{
002	float:right;
003	}
004	#cudillero{
005	float:left;
006	}

Lados no acepta elementos flotantes:

clear:none|left|right|both

Con **none** se permiten elementos flotantes a ambos lados. Con **left** no se admiten elementos flotantes a la izquierda. Con **right** no se admiten elementos flotantes a la derecha. Con **both** no se admiten elementos flotantes ni a la izquierda ni a la derecha. Se aplica a los elementos que están junto a los elementos flotantes. Los elementos flotantes se declaran con float.

ejemplo-02-186.css

001	h2{float:right;}
002	.siniestro>h2 {float:left;}
003	.nuevo{clear:left;}
004	.undecimo{clear:both;}
005	.dos{clear:left;}
006	.uno{clear:none;}

Esta propiedad describe como se muestra un elemento o los elementos que contiene.

**display:block | inline | inline-block | inline-table | list-item | run-in | table |
table-caption | table-column-group | table-header-group |
table-footer-group | table-row-group | table-cell | table-column |
table-row | none**

block⇒ muestra los elementos como un bloque

inline⇒ muestra los elementos en línea

inline-block⇒ su contenido son bloques en línea

inline-table⇒ se muestra como una table de nivel de línea

list-item⇒ muestra el elemento como un

run-in⇒ muestra el elemento dependiendo del contexto.

table⇒ el elemento se comporta como una tabla<table>

table-caption⇒ el elemento se comporta como un caption

table-column-group⇒ el elemento se comporta como un colgroup

table-header-group⇒ el elemento se comporta como un thead

table-footer-group⇒ el elemento se comporta como un tfoot

table-row-group⇒ el elemento se comporta como un tbody.

table-cell⇒ el elemento se comporta como una celda (td)

table-column⇒ el elemento se comporta como una columna (col)

table-row⇒ el elemento se comporta como una fila(tr)

none⇒ sin efecto

Flow ⇒ Nuevo contexto de bloque en el flujo, si el padre está en línea el también.

flow-root⇒ Nuevo contexto de bloque en el flujo

ruby⇒ genera un cuadro contenedor ruby.

ruby-text⇒ elemento de texto en ruby

ruby-base⇒ elemento base en ruby.

ruby-base-container⇒ elemento base contendor en ruby

ruby-text-container⇒ elemento contendor de texto en ruby

display-outside⇒

display-inside⇒

display-listitem⇒

display-internal⇒

display-box⇒

display-legancy⇒

contents⇒

11.- Estilos para listas

Tipo de estilo de la lista ordenadas con números

```
list-style-type:none | decimal | decimal-leading-zero | arabic-indic | armenian |
upper-armenian | lower-armenian | bengali | cambodian | khmer |
cjk-decimal | devanagari | georgian | gujarati | gurmukhi | hebrew |
kannada | lao | malayalam | mongolian | myanmar | oriya | persian |
lower-roman | upper-roman | tamil | telugu | thai | tibetan
```

Nos indica el tipo de carácter que se muestra delante de los valores de las listas que pueden ser: ninguno o números en diversos idiomas, de momento no funcionan gujarati ni malayalam.

Tipo de estilo de la lista ordenas con letras

```
list-style-type:none | lower-alpha | lower-latin | upper-alpha | upper-latin |
cjk-earthly-branch | cjk-heavenly-stem | lower-greek | hiragana |
hiragana-iroha | katakana | katakana-iroha
```

Tipo de estilo de la lista ordenas con caracteres por países

```
list-style-type:none | japanese-informal | japanese-formal | korean-hangul-formal |
korean-hanja-informal | korean-hanja-formal | simp-chinese-informal |
simp-chinese-formal | trad-chinese-informal | trad-chinese-formal |
cjk-ideographic | ethiopic-numeric
```

ejemplo-02-200.css

001	#listas{
002	display:grid;
003	width:100%;
004	border:4px solid blue;
005	-ms-grid:1fr 1fr 1fr auto auto 1fr 1fr1fr1fr1fr1fr 1fr/ auto autoautoauto;
006	grid:1fr 1fr 1fr auto auto 1fr 1fr1fr1fr1fr1fr1fr / auto autoautoauto;
007	-ms-grid-auto-flow:dense;
008	grid-auto-flow:dense;
009	}
011	#listas>div {
012	border:2px solid red;
013	}
014	#uno>ol {list-style-type:decimal;}
015	#dos>ol {list-style-type:decimal-leading-zero;}
016	#tres>ol {list-style-type:arabic-indic;}
017	#cuatro>ol {list-style-type:armenian;}
018	#cinco>ol {list-style-type:upper-armenian;}
019	#seis>ol {list-style-type:lower-armenian;}
020	#siete>ol {list-style-type:bengali;}
021	#ocho>ol {list-style-type:cambodian;}
022	#nueve>ol {list-style-type:khmer;}
023	#diez>ol {list-style-type:cjk-decimal;}
024	#once>ol {list-style-type:devanagari;}
025	#doce>ol {list-style-type:georgian;}
026	#trece>ol {list-style-type:gujarati;}
027	#catorce>ol {list-style-type:gurmukhi;}
028	#quince>ol {list-style-type:hebrew;}
029	#diecisiete>ol {list-style-type:kannada;}
030	#diecisiete>ol {list-style-type:lao;}
031	#dieciocho>ol {list-style-type:malayalam;}
032	#diecinueve>ol {list-style-type:mongolian;}
033	#veinte>ol {list-style-type:myanmar;}
034	#veinte01>ol {list-style-type:oriya;}
035	#veinte02>ol {list-style-type:persian;}
036	#veinte03>ol {list-style-type:lower-roman;}
037	#veinte04>ol {list-style-type:upper-roman;}
038	#veinte05>ol {list-style-type:tamil;}
039	#veinte06>ol {list-style-type:telugu;}
040	#veinte07>ol {list-style-type:thai;}
041	#veinte08>ol {list-style-type:tibetan;}
042	#veinte09>ol {list-style-type:lower-alpha;}

```

043 #treinta>ol {list-style-type:lower-latin;}
044 #treinta01>ol {list-style-type:upper-alpha;}
045 #treinta02>ol {list-style-type:upper-latin;}
046 #treinta03>ol {list-style-type:cjk-earthly-branch;}
047 #treinta04>ol {list-style-type:cjk-heavenly-stem;}
048 #treinta05>ol {list-style-type:lower-greek;}
049 #treinta06>ol {list-style-type:hiragana;}
050 #treinta07>ol {list-style-type:hiragana-iroha;}
051 #treinta08>ol {list-style-type:katakana;}
052 #treinta09>ol {list-style-type:katakana-iroha;}
053 #cuarenta>ol {list-style-type:japanese-informal;}
054 #cuarenta01>ol {list-style-type:japanese-formal;}
055 #cuarenta02>ol {list-style-type:korean-hangul-formal;}
056 #cuarenta03>ol {list-style-type:korean-hanja-informal;}
057 #cuarenta04>ol {list-style-type:korean-hanja-formal;}
058 #cuarenta05>ol {list-style-type:simp-chinese-informal;}
059 #cuarenta06>ol {list-style-type:simp-chinese-formal;}
060 #cuarenta07>ol {list-style-type:trad-chinese-informal;}
061 #cuarenta08>ol {list-style-type:trad-chinese-formal;}
052 #cuarenta09>ol {list-style-type:cjk-ideographic;}
063 #cincuenta>ol {list-style-type:ethiopic-numeric;}

```

Tipo de estilo de la lista NO ordenadas

list-style-type:none|disc|circle|square|disclosure-open|disclosure-closed

ejemplo-02-201.css

```

001 #listas{
002     display:grid;
003     width:100%;
004     border:4px solid blue;
005     -ms-grid:1fr 1fr / auto autoauto ;
006     grid:1fr 1fr 1fr / auto autoauto ;
007     -ms-grid-auto-flow:dense;
008     grid-auto-flow:dense;
009 }
010 #listas>div {
011     border:2px solid red;
012 }
013 #uno>ul {list-style-type:disc;}
014 #dos>ul {list-style-type:circle;}
015 #tres>ul {list-style-type:square;}
016 #cuatro>ul {list-style-type:disclosure-open;}
017 #cinco>ul {list-style-type:disclosure-closed;}
018 #seis>ul {list-style-type:none;}

```

Imagen como indicador de lista

list-style-image: url(*fichero*)|none

Estable la imagen que se muestra delante de un elemento de una lista, normalmente para listas no ordenadas.

ejemplo-02-202.css

```

001 #listas{
002     display:grid;
003     width:100%;
004     border:4px solid blue;
005     -ms-grid:1fr 1fr auto/ auto auto ;
006     grid:1fr 1fr 1fr auto/ auto auto ;
007     -ms-grid-auto-flow:dense;
008     grid-auto-flow:dense;
009 }
010 #listas>div {
011     border:2px solid red;
012 }
013 #uno>ul {list-style-image:url("../imagenes/lapiz_2.jpg");}
014 #dos>ul {list-style-image:url("../imagenes/pluma2.jpg");}
015 #tres>ul {list-style-image:url("../imagenes/lapiz01.png");}

```

```

016 #cuatro>ul {list-style-image:url("../imagenes/avion01.png");}
017 #cinco>ul {list-style-image:url("../imagenes/avion02.png");}
018 #seis>ul {list-style-image:none;}

```

Posición del carácter indicador de lista

list-style-position:inside|outside

Indica la posición de los elementos, si están dentro del flujo de contenido o fuera

ejemplo-02-203.css

```

001 #listas{
002     display:grid;
003     width:100%;
004     border:4px solid blue;
005     -ms-grid:1fr 1fr auto/ auto auto ;
006     grid:1fr 1fr 1fr auto/ auto auto ;
007     -ms-grid-auto-flow:dense;
008     grid-auto-flow:dense;
009 }
010 #listas>div {
011     border:2px solid red;
012 }
013 #uno>ol {list-style-position:inside;}
014 #dos>ul {list-style-position:inside;}
015 #tres>ol {list-style-position:outside;}
016 #cuatro>ul {list-style-position:outside;}

```

Tipo de estilo

list-style:tipo posición imagen

ejemplo-02-204.css

```

001 #listas{
002     display:grid;
003     width:100%;
004     border:4px solid blue;
005     -ms-grid:1fr 1fr auto/ auto auto ;
006     grid:1fr 1fr 1fr auto/ auto auto ;
007     -ms-grid-auto-flow:dense;
008     grid-auto-flow:dense;
009 }
010 #listas>div {
011     border:2px solid red;
012 }
013 #uno>ol {list-style:lower-roman inside;}
014 #dos>ul {list-style:square inside;}
015 #tres>ol {list-style:decimal-leading-zero outside;}
016 #cuatro>ul {list-style:disclosure-closed outside;}
017 #cinco>ol {list-style:none;}
018 #seis>ul {list-style:insideurl("../imagenes/lapiz01.png");}

```

Para definir un estilo vamos a utilizar: De momento solo en Firefox

@counter-style nombre-estilo {características}

El nombre de estilo, es el que se va a poner cuando se utilice la propiedad list-style o list-style-type.

Para definir las características vamos a utilizar,

**system:cyclic | numeric | alphabetic | symbolic | [fixed número-entero] |
 [extends nombre-tipo-contador] | additive número carácter [, número carácter] ...**

Con **cyclic** vamos a poner una serie de símbolos que se van a repetir de manera cíclica. Si se solo se pone uno se va a poner en todas, es el modelo a utilizar en las listas no ordenadas.

Con **numeric** los símbolos se interpretan como dígitos.

Con **alphabetic** los símbolos indicados se asocian como si fueran letras, siendo el alfabeto el conjunto de símbolos.

Con **symbolic** se indican una serie de símbolos, que se van a ir duplicando, triplicando, ...

Con **additive** se asocian símbolos con números para construir luego los números.

Con **fixed** recorre los símbolos y luego continúa con los números.

Con **extendsnos** permite indicar cualquier otro sistema de contador para añadirle características.

También se puede utilizar para asignar valor a list-style o list-style-type

Para indicar la lista de caracteres que se utilizan para realizar la representación cuando el valor de system es distinto de additive

symbols:lista-caracteres

Para indicar la lista de caracteres que se utilizan cuando se utiliza additive en lugar de symbol

additive-symbols:número carácter [, número carácter] ...

Se debe colocar ordenados por el número de mayor a menor.

ejemplo-02-205.css

```

001 #listas{
002     display:grid;
003     width:100%;
004     border:4px solid blue;
005     -ms-grid:1fr auto/ auto auto ;
006     grid:1fr 1fr auto / auto auto ;
007     -ms-grid-auto-flow:dense;
008     grid-auto-flow:dense;
009 }
010 #listas>div {
011     border:2px solid red;
012 }
013 @counter-style primo {
014     system:cyclic;
015     symbols:'\00B6'\00A2'\00F8'\01c2'\01fe'\03b8'\03d5';
016 }
017 @counter-style pares {
018     system:numeric;
019     symbols:'0'1';
020 }
021 @counter-style octavo {
022     system: cyclic;
023     symbols: ①;
024 }
025 #uno>ol {list-style:primo;}
026 #dos>ol {list-style-type:pares;}
027 #ocho>ul {list-style-type:octavo;}

```

ejemplo-02-206.css

```

001 #listas{
002     display:grid;
003     width:100%;
004     border:4px solid blue;
005     -ms-grid:1fr 1fr auto 1fr/ auto auto ;
006     grid:1fr 1fr 1fr auto 1fr/ auto auto ;
007     -ms-grid-auto-flow:dense;
008     grid-auto-flow:dense;
009 }
010 #listas>div {
011     border:2px solid red;
012 }
013 @counter-style primo {
014     system:symbolic;
015     symbols: © ® ¶ §;
016 }
017 @counter-style pares {

```

```

018     system: fixed;
019     symbols:'\00B6''\00A2''\00F8''\01c2''\01fe''\03b8''\03d5';
020 }
021 #uno>ol {list-style:primo;}
022 #dos>ol {list-style-type:pares;}
ejemplo-02-207.css
001 #listas{
002     display:grid;
003     width:100%;
004     border:4px solid blue;
005     -ms-grid:1fr 1fr auto 1fr/ auto auto ;
006     grid:1fr 1fr 1fr auto 1fr/ auto auto ;
007     -ms-grid-auto-flow:dense;
008     grid-auto-flow:dense;
009 }
010 #listas>div {
011     border:2px solid red;
012 }
013 @counter-style primo {
014     system: additive;
015     additive-symbols:56 '\00B6', 28 '\00A2', 14 '\00F8', 7 '\01c2', 3 '\01fe', 2
' \03b8', 1 '\03d5';
016 }
017 @counter-style pares {
018     system:alphabetic;
019     symbols: "\00B6""\00A2""\00F8""\01c2""\01fe""\03b8""\03d5";
020 }
021 #uno>ol {list-style:primo;}
022 #dos>ol {list-style-type:pares;}

```

Carácter que se usa delante del valor del índice

prefix:carácter

Carácter que se usa detrás del valor del índice

suffix:carácter

Rango de valores para los cuales se ha definido el estilo del contador fuera del rango el índice continúa con su valor normal.

range:número-1 número-2| auto

Cuando el número de caracteres que se muestran es inferior al número índice se rellenan con el símbolo que se indica.

pad:número carácter

Se indica un segundo estilo de contador, que será el que se aplique cuando se supere el valor indicado en un rango.

fallback:nombre-estilo-contador

ejemplo-02-208.css

```

001 #listas{
002     display:grid;
003     width:100%;
004     border:4px solid blue;
005     -ms-grid:1fr auto/ auto auto ;
006     grid:1fr 1fr auto / auto auto ;
007     -ms-grid-auto-flow:dense;
008     grid-auto-flow:dense;
009 }
010 #listas>div {
011     border:2px solid red;
012 }
013 @counter-style primo {
014     system:cyclic;

```

```

015     symbols:'\00B6''\00A2''\00F8''\01c2''\01fe''\03b8''\03d5';
016     prefix:'#';
017     suffix:". - ";
018 }
019 @counter-style pares {
020     system:numeric;
021     symbols:'0' '1';
022     pad:8 '0';
023     range:1 8;
024     fallback:upper-alpha;
025 }
026 #uno>ol {list-style:primo;}
027 #dos>ol {list-style-type:pares;}
028 #tres>ol {list-style:secuolla;}
ejemplo-02-209.css
001 #listas{
002     display:grid;
003     width:100%;
004     border:4px solid blue;
005     -ms-grid:1fr auto/ auto auto ;
006     grid:1fr 1fr auto / auto auto ;
007     -ms-grid-auto-flow:dense;
008     grid-auto-flow:dense;
009 }
010 #listas>div {
011     border:2px solid red;
012 }
013 @counter-style primo {
014     system:cyclic;
015     symbols:'\00B6''\00A2''\00F8''\01c2''\01fe''\03b8''\03d5';
016 }
017 @counter-style pares {
018     system:numeric;
019     symbols:'0' '1';
020 }
021 @counter-style secuolla{
022     system:symbolic;
023     symbols: © ® ¶ §;
024 }
025 @counter-style pinar{
026     system: fixed;
027     symbols:'\00B6''\00A2''\00F8''\01c2''\01fe''\03b8''\03d5';
028 }
029 @counter-style promesa{
030     system: additive;
031     additive-symbols:56 '\00B6', 28 '\00A2', 14 '\00F8', 7 '\01c2', 3 '\01fe', 2
     '\03b8', 1 '\03d5';
032 }
033 @counter-style pamesano{
034     system:alphabetic;
035     symbols: "\00B6""\00A2""\00F8""\01c2""\01fe""\03b8""\03d5";
036 }
037 @counter-style paris{
038     system:cyclic;
039     symbols:'\00B6''\00A2''\00F8''\01c2''\01fe''\03b8''\03d5';
040     prefix:'#';
041     suffix:". - ";
042 }
043 @counter-style palencia{
044     system:numeric;
045     symbols:'0' '1';
046     pad:8 '0';
047     range:1 8;
048     fallback:upper-alpha;
049 }
050 @counter-style octavo {
051     system: cyclic;
052     symbols: ①;
053 }
054 #uno>ol {list-style:primo;}
055 #dos>ol {list-style-type:pares;}
056 #tres>ol {list-style:secuolla}

```

```

057 #cuatro>ol {list-style-type:pinar}
058 #cinco>ol {list-style-type:promesa}
059 #seis>ol {list-style-type:pamesano}
060 #siete>ol {list-style-type:paris}
061 #ocho>ol {list-style-type:palencia}
062 #nueve>ol {list-style-type:octavo}

ejemplo-02-210.css
001 #listas{
002     display:grid;
003     width:100%;
004     border:4px solid blue;
005     -ms-grid:1fr auto/ auto auto ;
006     grid:1fr 1fr auto / auto auto ;
007     -ms-grid-auto-flow:dense;
008     grid-auto-flow:dense;
009 }
010 #listas>div {
011     border:2px solid red;
012 }
013 @counter-style promesa{
014     system: additive;
015     additive-symbols:56 '\00B6', 28 '\00A2', 14 '\00F8', 7 '\01c2', 3 '\01fe', 2
      '\03b8', 1 '\03d5';
016 }
017 @counter-style paris{
018     system:cyclic;
019     symbols:'\00B6'\00A2'\00F8'\01c2'\01fe'\03b8'\03d5';
020     prefix:'#';
021     suffix:". - ";
022 }
023 @counter-style palencia{
024     system:numeric;
025     symbols:'0''1';
026     pad:8 '0';
027     range:1 8;
028     fallback:upper-alpha;
029 }
030 @counter-style octavo {
031     system: cyclic;
032     symbols: ①;
033 }
034 @counter-style diezmo{
035     system:alphabetic;
036     symbols: \24D0 \24D1 \24D2 \24D3 \24D4 \24D5 \24D6 \24D7 \24D8 \24D9
      \24DA \24DB \24DC \24DD \24DE \24DF\24E0 \24E1 \24E2 \24E3 \24E4 \24E5
      \24E6 \24E7 \24E8 \24E9;
037 }
038 #uno>ol {list-style-type:symbols(cyclic '\00B6'\00A2'\00F8'\01c2'\01fe' '\03b8'
      '\03d5');}
039 #dos>ol {list-style-type:symbols(numeric '0''1');}
040 #tres>ol {list-style-type:symbols(symbolic '©''®''™''$');}
041 #cuatro>ol {list-style-type:symbols(fixed '\00B6'\00A2'\00F8'\01c2'\01fe'
      '\03b8'\03d5);}
042 #cinco>ol {list-style-type:promesa;}
043 #seis>ol {list-style-type:symbols(alphabetic "\00B6""\00A2""\00F8""\01c2" "\01fe"
      "\03b8""\03d5");}
044 #siete>ol {list-style-type:paris;}
045 #ocho>ol {list-style-type:octavo;}
046 #nueve>ol {list-style-type:symbols(cyclic ①);}
047 #diez>ol {list-style-type:diezmo;}

```

Para indicar el tipo de contador que se utiliza cuando se utilizan los lectores de pantalla, se usa cuando se utilizan símbolos que no se pueden pronunciar.

speak-as:auto | bullets | numbers | words | spell-out | nombre-estilo-contador

Para indicar los caracteres que vamos a poner con el índice cuando éste es negativo
negative:carácter-anterior [carácter-posterior]

Si queremos establecer unos contadores con unos valores iniciales, un incremento y el valor que se muestra distinto de los normales vamos a utilizar.

En primer lugar, deberemos quitar el estilo de la lista mediante

list-style-type:none

Luego deberemos poner para el elemento que define la lista, ol.

counter-reset:*nombre-contador* *valor-inicial*

Luego en los elementos de la lista deberemos poner

counter-increment:*nombre-contador* *valor-incremento*

Luego para indicar que es lo que queremos que se visualize deberemos poner que delante de los elementos de la lista

content: [*cadena*] counter(*nombre-contador*) [*cadena*]

counter(*nombre*[, *estilo-contador*])

Cuando tenemos elementos anidados y queremos tener contadores, podemos tener varios contadores con el mismo nombre y simplificar la puesta en marcha de los contadores.

counters(*nombre* , *cadena* [, *estilo-contador*])

ejemplo-02-212.css

```
001 #listas{  
002     display:grid;  
003     width:100%;  
004     border:4px solid blue;  
005     -ms-grid:1fr auto/ auto auto ;  
006     grid:1fr 1fr auto / auto auto ;  
007     -ms-grid-auto-flow:dense;  
008     grid-auto-flow:dense;  
009 }  
010 #listas>div {  
011     border:2px solid red;  
012 }  
013 #uno>ol {  
014     list-style-type:none;  
015     counter-reset:primero;  
016 }  
017 #uno>ol >li::before{  
018     counter-increment:primero;  
019     content: counter(primero) ". ";  
020 }  
021 #dos>ol {  
022     list-style-type:none;  
023     counter-reset:segundo 10;  
024 }  
025 #dos>ol >li{counter-increment:segundo 2;}  
026 #dos>ol >li::before{content: counter(segundo) ". ";}  
027 #tres>ol {  
028     list-style-type:none;  
029     counter-reset:tercero 10;  
030 }  
031 #tres>ol >li::before{  
032     counter-increment:tercero -3;  
033     content: counter(tercero) ". ";  
034 }  
035 #cuatro>ol {
```

```

036     list-style-type:none;
037     counter-reset:cuarto;
038 }
039 #cuatro>ol >li::before{
040     counter-increment:cuarto;
041     content: counter(cuarto, upper-alpha) ". ";
042 }
043 #cinco>ol {
044     list-style-type:none;
045     counter-reset:quinto 10;
046 }
047 #cinco>ol >li::before{
048     counter-increment:quinto -3;
049     content: counter(quinto) ". ";
050 }
051 #cinco>ol >ol{
052     list-style-type:none;
053     counter-reset:subquinto 100;
054 }
055 #cinco>ol >ol>li::before{
056     counter-increment:subquinto 5;
057     content: counter(quinto) "." counter(subquinto) " - ";
058 }
059 #seisol{
060     list-style-type:none;
061     counter-reset:sexto;
062 }
063 #seisol li {counter-increment:sexto;}
064 #seisol li::before{
065     content: counters(sexo, ".", lower-roman) " - ";
066 }
067 #siete>ol {
068     list-style:none;
069     counter-reset:septimo;
070 }
071 #siete>ol >li::before{
072     counter-increment:septimo;
073     content: counter(septimo) " ";
074 }
075 #siete>ol >li >ol{
076     list-style:none;
077     counter-reset:subseptimo;
078 }
079 #siete>ol >li >ol>li::before{
080     counter-increment:subseptimo;
081     content: counter(septimo) ". " counter(subseptimo) " ";
082 }
083 #nueve>ol {
084     list-style-type:none;
085     counter-reset:noveno 0;
086 }
087 #nueve>ol >li::before{
088     counter-increment:noveno 2;
089     content: counter(noveno, upper-alpha) ". ";
090 }
091 @counter-style numeros-circulo{
092     system:numeric;
093     symbols:"\24EA'\2460'\2461'\2462'\2463'\2464'\2465'\2466'\2467'\2468';
094 }
095 #ocho>ol {
096     list-style-type:none;
097     counter-reset:octavo 0;
098 }
099 #ocho>ol >li::before{
100     counter-increment:octavo;
101     content: counter(octavo, numeros-circulo) ". ";
102 }

```

12.- Estilos de columnas

Para mostrar la página en varias columnas tenemos (se usan con extensiones)

Número de columnas

column-count: *número de columnas*

Nos indica el número de columnas que vamos a tener.

Ancho de las columnas

column-width: *ancho*

Servirá para definir el ancho aproximado de las columnas a crear, todas tienen el mismo ancho. En función del ancho se determina el número de columnas.

Se debe poner o bien el número de columnas o bien el ancho de las columnas, si se ponen los dos prevalece el número de columnas y no se tiene en cuenta el ancho de las columnas.

Las dos opciones juntas

columns: *número-columnas ancho-columnas*

Se puede poner uno de los dos valores o bien los dos valores, cuando se ponen los dos valores, va a prevalecer el número de columnas sobre el ancho de las columnas, para ajustar las columnas al ancho de visualización

Espacio entre columnas

column-gap: *tamaño*

Nos permitirá definir el espacio en blanco entre columnas.

Servirá para establecer el ancho de una línea divisoria entre las columnas.

column-rule-width: *tamaño*

Servirá para establecer el tipo de una línea divisoria entre las columnas.

column-rule-style: *none | solid | hidden | dotted | dashed | double | groove | inset | outset*

Servirá para establecer el color de una línea divisoria entre las columnas.

column-rule-color: *color*

Línea divisoria entre columnas

column-rule: *ancho tipo color*

Número de columnas en las que se expande un elemento. En Firefox no funciona

column-span: *all | none*

ejemplo-2-215.css

001	.inicio{
002	-moz-column-count:2;
003	-webkit-column-count:2;
004	column-count:2;
005	border:2px solid green;
006	}
007	.izquierda{
008	-moz-column-width:300px;
009	-webkit-column-width:300px;
010	column-width:300px;
011	border:2px solid blue;

```

012 }
013 .derecha{
014     -moz-column-width:250px;
015     -webkit-column-width:250px;
016     column-width:250px;
017     border:2px solid red;
018 }
019 .centro{
020     -moz-column-count:4;
021     -webkit-column-count:4;
022     column-count:4;
023     -moz-column-width:250px;
024     -webkit-column-width:250px;
025     column-width:200px;
026     border:2px solid yellow;
027 }
028 #muerta{
029     -moz-column-count:2;
030     -webkit-column-count:2;
031     column-count:2;
032     -moz-column-gap:50px;
033     -webkit-column-gap:50px;
034     column-gap:50px;
035     border:2px solid green;
036 }
037 #acueducto{
038     -moz-column-width:250px;
039     -webkit-column-width:250px;
040     column-width:250px;
041     -moz-column-gap:50px;
042     -webkit-column-gap:50px;
043     column-gap:50px;
044     -moz-column-rule-width:5px;
045     -webkit-column-rule-width:5px;
046     column-rule-width:5px;
047     -moz-column-rule-style: solid ;
048     -webkit-column-rule-style: solid ;
049     column-rule-style: solid ;
050     -moz-column-rule-color:orange;
051     -webkit-column-rule-color:orange;
052     column-rule-color:orange;
053     border:2px solid gold;
054 }
055 #linajes{
056     -moz-column-count:2;
057     -webkit-column-count:2;
058     column-count:2;
059     -moz-column-gap:50px;
060     -webkit-column-gap:50px;
061     column-gap:50px;
062     -moz-column-rule:5px dotted violet;
063     -webkit-column-rule:5px dotted violet;
064     column-rule:5px dotted violet;
065     border:2px solid salmon;
066 }
067 #linajes h2 {
068     -webkit-column-span:all;
069     column-span:all;
070 }

```

Para indicar como se rellenan las columnas tenemos. Con extensiones para Firefox.

column-fill:auto | balance | balance-all

Con **auto** las columnas se rellenan de forma secuencial

Con **balance** y **balance-all** las columnas se rellenan de forma igual.

ejemplo-2-216.css

001	.inicio{
-----	----------

002	-moz-column-count:2;
003	-webkit-column-count:2;
004	column-count:2;
005	-moz-column-fill:auto;
006	column-fill:auto;
007	border:2px solid green;
008	}
009	.izquierda{
010	-moz-column-width:300px;
011	-webkit-column-width:300px;
012	column-width:300px;
013	-moz-column-fill:balance;
014	column-fill:balance;
015	border:2px solid blue;
016	}
017	.derecha{
018	-moz-column-width:250px;
019	-webkit-column-width:250px;
020	column-width:250px;
021	-moz-column-fill:balance-all;
022	column-fill:balance-all;
023	border:2px solid red;
024	}

Añade un salto de página después del elemento al que se le aplica

page-break-after: auto | avoid | left | right | always

Añade un salto de página antes del elemento al que se le aplica

page-break-before: auto | avoid | left | right | always

Añade un salto de página dentro del elemento al que se le aplica

page-break-inside: auto | avoid

Con **always** fuerza el salto de página siempre.

Con **avoid** evita los saltos de páginas

Con **left** fuerza el salto de página siempre para que la página tenga formato de página izquierda.

Con **right** fuerza el salto de página siempre para que la página tenga formato de página derecha.

Con **auto** hace saltos de página automáticos.

Número mínimo de líneas de un párrafo. No soportada por Firefox

orphans: entero

Número mínimo de líneas en un elemento de tipo bloque. No soportada por Firefox

windows: entero

Para indicar como se muestra la decoración de ciertos elementos cuando se dividen en líneas o columnas.

text-decoration-break: slice | clone

Con **clone** se cierra al final de la división.

Con **slice** no se cierra al final de la división.

ejemplo-2-220css

001	.ejemplo{
002	background: linear-gradient(to bottom right, yellow, green);
003	box-shadow: 8px 8px 10px 0px deeppink, -5px -5px 5px 0px blue, 5px 5px 15px 0px yellow;
004	padding: 0em 1em;

```

005 border-radius: 16px;
006 border-style: solid;
007 margin-left: 10px;
008 font: 24px sans-serif;
009 line-height: 2;
010 }
011 #ini-2{
012 -webkit-box-decoration-break: clone;
013 -o-decoration-break: clone;
014 box-decoration-break: clone;
015 }
016 #izq-2{
017 -webkit-box-decoration-break: clone;
018 -o-decoration-break: clone;
019 box-decoration-break: clone;
020 }
021 #centro{
022 column-count:3;
023 }
024 #cen{
025 display:block;
026 }
027 #fin{
028 display:block;
029 -webkit-box-decoration-break: clone;
030 -o-decoration-break: clone;
031 box-decoration-break: clone;
032 }
033 #final{
034 column-count:3;
035 }

```

13.-Animaciones

Los atributos de estilo para esta capa que se ven en el código anterior son los siguientes, se utilizan con extensiones, ahora solo es necesaria para chrome, pero para versiones anteriores de los navegadores es necesario para Opera y Firefox:

El nombre del fotograma clave.

animation-name: *nombre*

Para definir el fotograma clave deberemos poner:

**@keyframes *nombre* {
 porcentaje% { *estilo* }
 from {*estilo*} /*equivale 0%*/
 to {*estilo*} /*equivale 100%*/
 }**

La duración de la animación.

animation-duration:*duración*

Las veces que se repite.

animation-iteration-count:*veces* | infinite

**animation-timing-function:linear | ease | ease-in | ease-out | ease-in-out |
 cubic-bezier(x1,x2,x3,x4) | steps(*entero* [, start | end])**

Se aplica entre los fotogramas clave, no sobre toda la animación y describe como progresó la animación a lo largo de un ciclo.

animation-direction: alternate | reverse | normal | alternate-reverse

Esta propiedad define el sentido de la animación. Si especificamos “alternate” y los ciclos de iteración son impares, la animación irá en la dirección normal, si no, se realizará en la dirección inversa

animation-delay: *retraso***ejemplo-02-230.css**

```
001 #inicio{  
002     background-color:blue;  
003     color:white;  
004     border:2px solid yellow;  
005     width:32%;  
006     height:160px;  
007     -o-animation-name:agrandar;  
008     -moz-animation-name:agrandar;  
009     -webkit-animation-name:agrandar;  
010     animation-name:agrandar;  
011     -o-animation-duration:5s;  
012     -moz-animation-duration:5s;  
013     -webkit-animation-duration:5s;  
014     animation-duration:5s;  
015     -moz-animation-iteration-count:5;  
016     -o-animation-iteration-count:5;  
017     -webkit-animation-iteration-count:5;  
018     animation-iteration-count:5;  
019     -o-animation-direction:alternate;  
020     -moz-animation-direction:alternate;  
021     -webkit-animation-direction:alternate;  
022     animation-direction:alternate;  
023     -o-animation-delay:10s;  
024     -moz-animation-delay:10s;  
025     -webkit-animation-delay:10s;  
026     animation-delay:10s;  
027     -o-animation-timing-function:ease-in-out;  
028     -moz-animation-timing-function:ease-in-out;  
029     -webkit-animation-timing-function:ease-in-out;  
030     animation-timing-function:ease-in-out;  
031 }  
032 @-o-keyframes agrandar{  
033     from {  
034         width:32%;  
035         height:160px;  
036         background-color:blue;  
037         font-size:1em;  
038     }  
039     50% {  
040         background-color:red;  
041     }  
042     to {  
043         width:85%;  
044         height:800px;  
045         background-color:blue;  
046         font-size:2.5em;  
047     }  
048 }  
049 @-moz-keyframes agrandar{  
050     from {  
051         width:32%;  
052         height:160px;  
053         background-color:blue;  
054         font-size:1em;  
055     }  
056     50% {  
057         background-color:red;  
058     }  
059     to {  
060         width:85%;  
061         height:800px;  
062         background-color:blue;  
063         font-size:2.5em;
```

```

064 }
065 }
066 @-webkit-keyframes agrandar{
067   from {
068     width:32%;
069     height:160px;
070     background-color:blue;
071     font-size:1em;
072   }
073   50% {
074     background-color:red;
075   }
076   to {
077     width:85%;
078     height:800px;
079     background-color:blue;
080     font-size:2.5em;
081   }
082 }
083 @keyframes agrandar{
084   from {
085     width:32%;
086     height:160px;
087     background-color:blue;
088     font-size:1em;
089   }
090   50% {
091     background-color:red;
092   }
093   to {
094     width:85%;
095     height:800px;
096     background-color:blue;
097     font-size:2.5em;
098   }
099 }

```

Para indicar como se inicia o queda la animación tenemos, estado.

animation-fill-mode:none | forwards | backwards | both

Con **none** no se aplica nada.

Con **forwards** el objeto quedara con las características que aplica el último keyframes.

Con **backwards** al objeto se le aplica las características del primer keyframes en cuanto se cargue y mientras dura el retardo.

Con **both** se aplican los dos anteriores.

ejemplo-2-231.css

```

001 #inicio{
002   background-color:blue;
003   color:white;
004   border:2px solid yellow;
005   width:32%;
006   height:160px;
007   -o-animation-name:agrandar;
008   -moz-animation-name:agrandar;
009   -webkit-animation-name:agrandar;
010   animation-name:agrandar;
011   -o-animation-duration:5s;
012   -moz-animation-duration:5s;
013   -webkit-animation-duration:5s;
014   animation-duration:5s;
015   -o-animation-iteration-count:5;
016   -moz-animation-iteration-count:5;
017   -webkit-animation-iteration-count:5;
018   animation-iteration-count:5;
019   -o-animation-direction:alternate;
020   -moz-animation-direction:alternate;

```

```
021    -webkit-animation-direction:alternate;  
022    animation-direction:alternate;  
023    -o-animation-delay:10s;  
024    -moz-animation-delay:10s;  
025    -webkit-animation-delay:10s;  
026    animation-delay:10s;  
027    -o-animation-timing-function:ease;  
028    -moz-animation-timing-function:ease;  
029    -webkit-animation-timing-function:ease;  
030    animation-timing-function:ease;  
031    -o-animation-fill-mode:forwards;  
032    -moz-animation-fill-mode:forwards;  
033    -webkit-animation-fill-mode:forwards;  
034    animation-fill-mode:forwards;  
035    }  
036 @-o-keyframes agrandar {  
037     from {  
038         width:32%;  
039         height:160px;  
040         background-color:blue;  
041         font-size:1em;  
042     }  
043     50% {  
044         background-color:red;  
045     }  
046     to {  
047         width:85%;  
048         height:800px;  
049         background-color:blue;  
050         font-size:2.5em;  
051     }  
052 }  
053 @-moz-keyframes agrandar{  
054     from {  
055         width:32%;  
056         height:160px;  
057         background-color:blue;  
058         font-size:1em;  
059     }  
060     50% {  
061         background-color:red;  
062     }  
063     to {  
064         width:85%;  
065         height:800px;  
066         background-color:blue;  
067         font-size:2.5em;  
068     }  
069 }  
070 @-webkit-keyframes agrandar{  
071     from {  
072         width:32%;  
073         height:160px;  
074         background-color:blue;  
075         font-size:1em;  
076     }  
077     50% {  
078         background-color:red;  
079     }  
080     to {  
081         width:85%;  
082         height:800px;  
083         background-color:blue;  
084         font-size:2.5em;  
085     }  
086 }  
087 @keyframes agrandar{  
088     from {  
089         width:32%;  
090         height:160px;  
091         background-color:blue;
```

```

092     font-size:1em;
093   }
094   50% {
095     background-color:red;
096   }
097   to {
098     width:85%;
099     height:800px;
100     background-color:blue;
101     font-size:2.5em;
102   }
103 }

ejemplo-02-232.css
001 #inicio{
002   background-color:blue;
003   color:white;
004   border:2px solid yellow;
005   width:32%;
006   height:160px;
007   -o-animation-name:agrandar;
008   -moz-animation-name:agrandar;
009   -webkit-animation-name:agrandar;
010   animation-name:agrandar;
011   -o-animation-duration:5s;
012   -moz-animation-duration:5s;
013   -webkit-animation-duration:5s;
014   animation-duration:5s;
015   -o-animation-iteration-count:5;
016   -moz-animation-iteration-count:5;
017   -webkit-animation-iteration-count:5;
018   animation-iteration-count:5;
019   -o-animation-direction:alternate;
020   -moz-animation-direction:alternate;
021   -webkit-animation-direction:alternate;
022   animation-direction:alternate;
023   -o-animation-delay:10s;
024   -moz-animation-delay:10s;
025   -webkit-animation-delay:10s;
026   animation-delay:10s;
027   -o-animation-timing-function:ease-in;
028   -moz-animation-timing-function:ease-in;
029   -webkit-animation-timing-function:ease-in;
030   animation-timing-function:ease-in;
031   -o-animation-fill-mode:backwards ;
032   -moz-animation-fill-mode:backwards ;
033   -webkit-animation-fill-mode:backwards ;
034   animation-fill-mode:backwards ;
035 }
036 @-o-keyframes agrandar{
037   from{
038     width:40%;
039     height:300px;
040     background-color:green;
041     font-size:1.2em;
042   }
043   50% {
044     background-color:red;
045   }
046   to {
047     width:85%;
048     height:800px;
049     background-color:blue;
050     font-size:2.5em;
051   }
052 }
053 @-moz-keyframes agrandar {
054   from{
055     width:40%;
056     height:300px;
057     background-color:green;
058     font-size:1.2em;

```



```

059 }
060 50% {
061     background-color:red;
062 }
063 to {
064     width:85%;
065     height:800px;
066     background-color:blue;
067     font-size:2.5em;
068 }
069 }
070 @-webkit-keyframes agrandar{
071     from {
072         width:40%;
073         height:300px;
074         background-color:green;
075         font-size:1.2em;
076     }
077     50% {
078         background-color:red;
079     }
080     to {
081         width:85%;
082         height:800px;
083         background-color:blue;
084         font-size:2.5em;
085     }
086 }
087 @keyframes agrandar{
088     from {
089         width:40%;
090         height:300px;
091         background-color:green;
092         font-size:1.2em;
093     }
094     50% {
095         background-color:red;
096     }
097     to {
098         width:85%;
099         height:800px;
100         background-color:blue;
101         font-size:2.5em;
102     }
103 }

```

ejemplo-02-233.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solidyellow;
005     width:32%;
006     height:160px;
007     -o-animation-name:agrandar;
008     -moz-animation-name:agrandar;
009     -webkit-animation-name:agrandar;
010     animation-name:agrandar;
011     -o-animation-duration:5s;
012     -moz-animation-duration:5s;
013     -webkit-animation-duration:5s;
014     animation-duration:5s;
015     -o-animation-iteration-count:5;
016     -moz-animation-iteration-count:5;
017     -webkit-animation-iteration-count:5;
018     animation-iteration-count:5;
019     -o-animation-direction:alternate;
020     -moz-animation-direction:alternate;
021     -webkit-animation-direction:alternate;
022     animation-direction:alternate;
023     -o-animation-delay:10s;
024     -moz-animation-delay:10s;
025     -webkit-animation-delay:10s;

```

```

026     animation-delay:10s;
027     -o-animation-timing-function:ease-out;
028     -moz-animation-timing-function:ease-out;
029     -webkit-animation-timing-function:ease-out;
030     animation-timing-function:ease-out;
031     -o-animation-fill-mode:both ;
032     -moz-animation-fill-mode:both ;
033     -webkit-animation-fill-mode:both ;
034     animation-fill-mode:both ;
035 }
036 @-o-keyframes agrandar{
037     from {
038         width:40%;
039         height:300px;
040         background-color:green;
041         font-size:1.2em;
042     }
043     50% {
044         background-color:red;
045     }
046     to {
047         width:85%;
048         height:800px;
049         background-color:blue;
050         font-size:2.5em;
051     }
052 }
053 @-moz-keyframes agrandar{
054     from {
055         width:40%;
056         height:300px;
057         background-color:green;
058         font-size:1.2em;
059     }
060     50% {
061         background-color:red;
062     }
063     to {
064         width:85%;
065         height:800px;
066         background-color:blue;
067         font-size:2.5em;
068     }
069 }
070 @-webkit-keyframes agrandar{
071     from {
072         width:40%;
073         height:300px;
074         background-color:green;
075         font-size:1.2em;
076     }
077     50% {
078         background-color:red;
079     }
080     to {
081         width:85%;
082         height:800px;
083         background-color:blue;
084         font-size:2.5em;
085     }
086 }
087 @keyframes agrandar{
088     from {
089         width:40%;
090         height:300px;
091         background-color:green;
092         font-size:1.2em;
093     }
094     50% {
095         background-color:red;
096     }

```

```

097   to {
098     width:85%;
099     height:800px;
100     background-color:blue;
101     font-size:2.5em;
102   }
103 }

```

Esta propiedad combina las anteriores de una forma resumida

animation: nombre duración function-timing retardo veces dirección estado

ejemplo-02-234.css

```

001 #inicio{
002   background-color:blue;
003   color:white;
004   border:2px solid yellow;
005   width:32%;
006   height:160px;
007   -o-animation:agrandar 5s linear 10s 5 alternate both;
008   -moz-animation:agrandar 5s linear 10s 5 alternate both;
009   -webkit-animation:agrandar 5s linear 10s 5 alternate both;
010   animation:agrandar 5s linear 10s 5 alternate both;
011 }
012 @-o-keyframes agrandar{
013   from {
014     width:40%;
015     height:300px;
016     background-color:green;
017     font-size:1.2em;
018   }
019   50% {
020     background-color:red;
021   }
022   to {
023     width:85%;
024     height:800px;
025     background-color:blue;
026     font-size:2.5em;
027   }
028 }
029 @-moz-keyframes agrandar{
030   from {
031     width:40%;
032     height:300px;
033     background-color:green;
034     font-size:1.2em;
035   }
036   50% {
037     background-color:red;
038   }
039   to {
040     width:85%;
041     height:800px;
042     background-color:blue;
043     font-size:2.5em;
044   }
045 }
046 @-webkit-keyframes agrandar{
047   from {
048     width:40%;
049     height:300px;
050     background-color:green;
051     font-size:1.2em;
052   }
053   50% {
054     background-color:red;
055   }
056   to {
057     width:85%;

```

```

058     height:800px;
059     background-color:blue;
060     font-size:2.5em;
061 }
062 }
063 @keyframes agrandar{
064     from {
065         width:40%;
066         height:300px;
067         background-color:green;
068         font-size:1.2em;
069     }
070     50% {
071         background-color:red;
072     }
073     to {
074         width:85%;
075         height:800px;
076         background-color:blue;
077         font-size:2.5em;
078     }
079 }

```

Propiedad que nos indica el momento en el que comenzará la animación. Si el valor es 0 se ejecuta en cuanto se carga la página.

animation-play-state:paused|running

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     -o-animation:agrandar 5s linear 2s 5 alternate forwards ;
008     -moz-animation:agrandar 5s linear 2s 5 alternate forwards ;
009     -webkit-animation:agrandar 5s linear 2s 5 alternate forwards ;
010     animation:agrandar 5s linear 2s 5 alternate forwards ;
011     -o-animation-play-state:paused;
012     -moz-animation-play-state:paused;
013     -webkit-animation-play-state:paused;
014     animation-play-state:paused;
015 }
016 #inicio:hover{
017     -o-animation-play-state:running;
018     -moz-animation-play-state:running;
019     -webkit-animation-play-state:running;
020     animation-play-state:running;
021 }
022 @-o-keyframes agrandar {
023     from {
024         width:40%;
025         height:300px;
026         background-color:green;
027         font-size:1.2em;
028     }
029     50% {
030         background-color:red;
031     }
032     to {
033         width:85%;
034         height:800px;
035         background-color:blue;
036         font-size:2.5em;
037     }
038 }
039 @-moz-keyframes agrandar{
040     from {
041         width:40%;
042         height:300px;

```

```

043     background-color:green;
044     font-size:1.2em;
045   }
046   50% {
047     background-color:red;
048   }
049   to {
050     width:85%;
051     height:800px;
052     background-color:blue;
053     font-size:2.5em;
054   }
055 }
056 @-webkit-keyframes agrandar{
057   from {
058     width:40%;
059     height:300px;
060     background-color:green;
061     font-size:1.2em;
062   }
063   50% {
064     background-color:red;
065   }
066   to {
067     width:85%;
068     height:800px;
069     background-color:blue;
070     font-size:2.5em;
071   }
072 }
073 @keyframes agrandar{
074   from {
075     width:40%;
076     height:300px;
077     background-color:green;
078     font-size:1.2em;
079   }
080   50% {
081     background-color:red;
082   }
083   to {
084     width:85%;
085     height:800px;
086     background-color:blue;
087     font-size:2.5em;
088   }
089 }
```

En todos los elementos que hemos visto de animación, los que empiezan por animation, se pueden poner varias animaciones, siempre y cuando separemos los valores de las animaciones por comas.

ejemplo-02-236.css

```

001 #inicio{
002   background-color:blue;
003   color:white;
004   border:2px solid yellow;
005   width:32%;
006   height:160px;
007   -o-animation:agrandar 5s linear 2s 5 alternate forwards, borde 10s ease 3s 3
alternate-reverse both ;
008   -moz-animation:agrandar 5s linear 2s 5 alternate forwards, borde 10s ease 3s 3
alternate-reverse both ;
009   -webkit-animation:agrandar 5s linear 2s 5 alternate forwards, borde 10s ease 3s
3 alternate-reverse both ;
010   animation:agrandar 5s linear 2s 5 alternate forwards, borde 10s ease 3s 3
alternate-reverse both ;
011 }
012 @-o-keyframes agrandar{
013   from {
```

```
014     width:40%;  
015     height:300px;  
016     background-color:green;  
017     font-size:1.2em;  
018   }  
019   50% {  
020     background-color:red;  
021   }  
022   to {  
023     width:85%;  
024     height:800px;  
025     background-color:blue;  
026     font-size:2.5em;  
027   }  
028 }  
029 @-moz-keyframes agrandar{  
030   from {  
031     width:40%;  
032     height:300px;  
033     background-color:green;  
034     font-size:1.2em;  
035   }  
036   50% {  
037     background-color:red;  
038   }  
039   to {  
040     width:85%;  
041     height:800px;  
042     background-color:blue;  
043     font-size:2.5em;  
044   }  
045 }  
046 @-webkit-keyframes agrandar{  
047   from {  
048     width:40%;  
049     height:300px;  
050     background-color:green;  
051     font-size:1.2em;  
052   }  
053   50% {  
054     background-color:red;  
055   }  
056   to {  
057     width:85%;  
058     height:800px;  
059     background-color:blue;  
060     font-size:2.5em;  
061   }  
062 }  
063 @keyframes agrandar{  
064   from {  
065     width:40%;  
066     height:300px;  
067     background-color:green;  
068     font-size:1.2em;  
069   }  
070   50% {  
071     background-color:red;  
072   }  
073   to {  
074     width:85%;  
075     height:800px;  
076     background-color:blue;  
077     font-size:2.5em;  
078   }  
079 }  
080 @-o-keyframes borde{  
081   from {  
082     border-radius:0px;  
083     border:1px solid red;  
084   }
```

```

085     to {
086         border-radius:30px;
087         border:5px solid red orange yellow green blue;
088     }
089 }
090 @-moz-keyframes borde{
091     from {
092         border-radius:0px;
093         border:1px solid red;
094     }
095     to {
096         border-radius:30px;
097         border:5px solid red orange yellow green blue;
098     }
099 }
100 @-webkit-keyframes borde{
101     from {
102         border-radius:0px;
103         border:1px solid red;
104     }
105     to {
106         border-radius:30px;
107         border:5px solid red orange yellow green blue;
108     }
109 }@keyframes borde{
110     from {
111         border-radius:0px;
112         border:1px solid red;
113     }
114     to {
115         border-radius:30px;
116         border:5px solid red orange yellow green blue;
117     }
118 }

```

ejemplo-02-237.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     -o-animation-name:agrandar,borde;
008     -moz-animation-name:agrandar,borde;
009     -webkit-animation-name:agrandar,borde;
010     animation-name:agrandar,borde;
011     -o-animation-duration:5s,10s;
012     -moz-animation-duration:5s,10s;
013     -webkit-animation-duration:5s,10s;
014     animation-duration:5s,10s;
015     -o-animation-iteration-count:5,3;
016     -moz-animation-iteration-count:5,3;
017     -webkit-animation-iteration-count:5,3;
018     animation-iteration-count:5,3;
019     -o-animation-direction:alternate, alternate-reverse;
020     -moz-animation-direction:alternate, alternate-reverse;
021     -webkit-animation-direction:alternate, alternate-reverse;
022     animation-direction:alternate, alternate-reverse;
023     -o-animation-delay:10s,3s;
024     -moz-animation-delay:10s,3s;
025     -webkit-animation-delay:10s,3s;
026     animation-delay:10s,3s;
027     -o-animation-timing-function:linear, ease;
028     -moz-animation-timing-function:linear, ease;
029     -webkit-animation-timing-function:linear, ease;
030     animation-timing-function:linear, ease;
031     -o-animation-fill-mode:forwards, both;
032     -moz-animation-fill-mode:forwards, both;
033     -webkit-animation-fill-mode:forwards, both;
034     animation-fill-mode:forwards, both;
035 }
036 @-o-keyframes agrandar{

```

```

037    from {
038        width:40%;
039        height:300px;
040        background-color:green;
041        font-size:1.2em;
042    }
043    50% {
044        background-color:red;
045    }
046    to {
047        width:85%;
048        height:800px;
049        background-color:blue;
050        font-size:2.5em;
051    }
052}
053 @-moz-keyframes agrandar{
054    from {
055        width:40%;
056        height:300px;
057        background-color:green;
058        font-size:1.2em;
059    }
060    50% {
061        background-color:red;
062    }
063    to {
064        width:85%;
065        height:800px;
066        background-color:blue;
067        font-size:2.5em;
068    }
069}
070 @-webkit-keyframes agrandar{
071    from {
072        width:40%;
073        height:300px;
074        background-color:green;
075        font-size:1.2em;
076    }
077    50% {
078        background-color:red;
079    }
080    to {
081        width:85%;
082        height:800px;
083        background-color:blue;
084        font-size:2.5em;
085    }
086}
087 @keyframes agrandar{
088    from {
089        width:40%;
090        height:300px;
091        background-color:green;
092        font-size:1.2em;
093    }
094    50% {
095        background-color:red;
096    }
097    to {
098        width:85%;
099        height:800px;
100        background-color:blue;
101        font-size:2.5em;
102    }
103}
104 @keyframes borde{
105    from {
106        border-radius:0px;
107        border:1px solid red;

```

```

108 }
109 to {
110 border-radius:30px;
111 border:5px solid red orange yellow green blue;
112 }
113 }
114 @-o-keyframes borde{
115 from {
116 border-radius:0px;
117 border:1px solid red;
118 }
119 to {
120 border-radius:30px;
121 border:5px solid red orange yellow green blue;
122 }
123 }
124 @-moz-keyframes borde{
125 from {
126 border-radius:0px;
127 border:1px solid red;
128 }
129 to {
130 border-radius:30px;
131 border:5px solid red orange yellow green blue;
132 }
133 }
134 @-webkit-keyframes borde{
135 from {
136 border-radius:0px;
137 border:1px solid red;
138 }
139 to{
140 border-radius:30px;
141 border:5px solid red orange yellow green blue;
142 }
143 }@keyframes borde{
144 from {
145 border-radius:0px;
146 border:1px solid red;
147 }
148 to{
149 border-radius:30px;
150 border:5px solid red orange yellow green blue;
151 }
152 }

```

Cuando se cambia una propiedad se puede indicar que se desarrolle una transición, esto que el cambio en lugar de realizarse de una forma brusca, se establezca un tiempo para llevar a cabo el cambio. Se utiliza con extensiones.

Define las propiedades a modificar

transition-property: propiedad | lista-propiedades | none | all

Tiempo empleado en desarrollarse la transición

transition-duration: tiempo | lista-tiempo

Function de la transición

transition-timing-function: linear | ease | ease-in | ease-out | ease-in-out | cubic-bezier(x₁,x₂,x₃,x₄) | steps(entero [, start | end])

Indica cómo se desarrolla la transición en el tiempo. Puede tomar los valores: linear (igual todo el rato), ease (inicio y final lento), ease-in (inicio lento), ease-out (final lento), ease-in-out (inicio y final lento), cubic-bezier(x₁,x₂,x₃,x₄) (según la fórmula de la curva cubica de bezier).

Retraso en iniciarse la transición

transition-delay: retraso

ejemplo-02-240.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     -moz-transition-property:width;
008     -webkit-transition-property:width;
009     -o-transition-property:width;
010     transition-property:width;
011     -moz-transition-duration:5s;
012     -webkit-transition-duration:5s;
013     -o-transition-duration:5s;
014     transition-duration:5s;
015     -moz-transition-timing-function:linear;
016     -webkit-transition-timing-function:linear;
017     -o-transition-timing-function:linear;
018     transition-timing-function:linear;
019     -moz-transition-delay:2s;
020     -webkit-transition-delay:2s;
021     -o-transition-delay:2s;
022     transition-delay:2s;
023 }
024 #inicio:hover{
025     width:80%;
026 }

```

ejemplo-02-242.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     -moz-transition-property:width,height;
008     -webkit-transition-property:width,height;
009     -o-transition-property:width,height;
010     transition-property:width,height;
011     -moz-transition-duration:5s,3s;
012     -webkit-transition-duration:5s,3s;
013     -o-transition-duration:5s,3s;
014     transition-duration:5s,3s;
015     -moz-transition-timing-function:linear,ease-in;
016     -webkit-transition-timing-function:linear,ease-in;
017     -o-transition-timing-function:linear,ease-in;
018     transition-timing-function:linear,ease-in;
019     -moz-transition-delay:2s,1s;
020     -webkit-transition-delay:2s,1s;
021     -o-transition-delay:2s,1s;
022     transition-delay:2s,1s;
023 }
024 #inicio:hover{
025     width:80%;
026     height:400px;
027 }

```

Datos de la transición

transition: propiedad duración desarrollo retraso

Se pueden poner varias, separadas por comas

ejemplo-02-241.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;

```

```

007     -moz-transition: width 5s linear 2s;
008     -webkit-transition: width 5s linear 2s;
009     -o-transition: width 5s linear 2s;
010     transition: width 5s linear 2s;
011 }
012 #inicio:hover{
013     width:80%;
014 }

```

ejemplo-02-243.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     -moz-transition: width 5s linear 2s, height 3s ease-in 1s;
008     -webkit-transition: width 5s linear 2s, height 3s ease-in 1s;
009     -o-transition: width 5s linear 2s, height 3s ease-in 1s;
010     transition: width 5s linear 2s, height 3s ease-in 1s;
011 }
012 #inicio:hover{
013     width:80%;
014     height:400px;
015 }

```

La lista de propiedades para transition son:

background-color	color	grid-*	various
background-position	percentage, length	height	length, percentage
border-bottom-color	color	left	length, percentage
border-bottom-width	length	letter-spacing	length
border-color	color	line-height	number, length, percentage
border-left-color	color	margin-bottom	length
border-left-width	length	margin-left	length
border-right-color	color	margin-right	length
border-right-width	length	margin-top	length
border-spacing	length	max-height	length, percentage
border-top-color	color	max-width	length, percentage
border-top-width	length	min-height	length, percentage
border-width	length	min-width	length, percentage
bottom	length, percentage	opacity	number
clip	rectangle	outline-color	color
color	color	outline-offset	integer
crop	rectangle	outline-width	length
font-size	length, percentage	padding-bottom	length
font-weight	number	padding-left	length
		padding-right	length
		padding-top	length
		right	length, percentage
		text-indent	length, percentage
		text-shadow	shadow
		top	length, percentage

vertical-align	keywords, length, percentage	word-spacing	length, percentage
visibility	visibility	z-index	integer
width	length, percentage	zoom	number

Permite realizar modificaciones en el elemento. Se utiliza con extensiones
transform:none | transformación

Translada a posición x,y
translate(x ,y)

ejemplo-02-251.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-transform:translate(10%,20%);
013     -webkit-transform:translate(10%,20%);
014     -o-transform:translate(10%,20%);
015     transform:translate(10%,20%);
016 }
```

ejemplo-02-252.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-animation:llevar 5s 2 linear alternate;
013     -webkit-animation:llevar 5s 2 linear alternate;
014     -o-animation:llevar 5s 2 linear alternate;
015     animation:llevar 5s 2 linear alternate;
016 }
017 @-moz-keyframes llevar{
018     from {
019         top:40%;
020         left:40%;
021     }
022     to {
023         -moz-transform:translate(10%,20%);
024     }
025 }
026 @-webkit-keyframes llevar {
027     from{
028         top:40%;
029         left:40%;
030     }
031     to{
032         -webkit-transform:translate(10%,20%);
033     }
034 }
035 @-o-keyframes llevar{
036     from {
037         top:40%;
```

```

038     left:40%;  

039     }  

040     to {  

041         -o-transform:translate(10%,20%);  

042     }  

043 }  

044 @keyframes llevar{  

045     from {  

046         top:40%;  

047         left:40%;  

048     }  

049     to{  

050         transform:translate(10%,20%);  

051     }  

052 }

```

Translada a posición x,y,z

translate3d(x , y , z)

ejemplo-02-253.css

```

001 #inicio{  

002     background-color:blue;  

003     color:white;  

004     border:2px solid yellow;  

005     width:32%;  

006     height:160px;  

007     position:absolute;  

008     top:40%;  

009     left:40%;  

010 }  

011 #inicio:hover{  

012     -moz-transform:translate3d(10%,20%,50px);  

013     -webkit-transform: translate3d(10%,20%,50px);  

014     -o-transform:translate3d(10%,20%,50px);  

015     transform: translate3d(10%,20%,50px);  

016 }

```

Translado horizontal a posición x

translateX(x)

ejemplo-02-254.css

```

001 #inicio{  

002     background-color:blue;  

003     color:white;  

004     border:2px solid yellow;  

005     width:32%;  

006     height:160px;  

007     position:absolute;  

008     top:40%;  

009     left:40%;  

010 }  

011 #inicio:hover{  

012     -moz-transform:translateX(80%);  

013     -webkit-transform: translateX(80%);  

014     -o-transform:translateX(80%);  

015     transform: translateX(80%);  

016 }

```

ejemplo-02-255.css

```

001 #inicio{  

002     background-color:blue;  

003     color:white;  

004     border:2px solid yellow;  

005     width:32%;  

006     height:160px;  

007     position:absolute;  

008     top:40%;  

009     left:40%;  

010 }

```

```

011 #inicio:hover{
012     -moz-animation:llevar 5s 2 linear alternate;
013     -webkit-animation:llevar 5s 2 linear alternate;
014     -o-animation:llevar 5s 2 linear alternate;
015     animation:llevar 5s 2 linear alternate;
016 }
017 @-moz-keyframes llevar{
018     from {
019         top:40%;
020         left:40%;
021     }
022     to {
023         -moz-transform:translateX(80%);
024     }
025 }
026 @-webkit-keyframes llevar {
027     from {
028         top:40%;
029         left:40%;
030     }
031     to {
032         -webkit-transform:translateX(80%);
033     }
034 }
035 @-o-keyframes llevar{
036     from {
037         top:40%;
038         left:40%;
039     }
040     to {
041         -o-transform:translateX(80%);
042     }
043 }
044 @keyframes llevar {
045     from{
046         top:40%;
047         left:40%;
048     }
049     to{
050         transform:translateX(80%);
051     }
052 }
```

Translado vertical a posición y
translateY(y)

ejemplo-02-256.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-transform:translateY(80%);
013     -webkit-transform: translateY(80%);
014     -o-transform:translateY(80%);
015     transform: translateY(80%);
016 }
```

ejemplo-02-257.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
```

```

006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-animation:llevar 5s 2 linear alternate;
013     -webkit-animation:llevar 5s 2 linear alternate;
014     -o-animation:llevar 5s 2 linear alternate;
015     animation:llevar 5s 2 linear alternate;
016 }
017 @-moz-keyframes llevar{
018     from {
019         top:40%;
020         left:40%;
021     }
022     to {
023         -moz-transform:translateY(80%);
024     }
025 }
026 @-webkit-keyframes llevar {
027     from {
028         top:40%;
029         left:40%;
030     }
031     to {
032         -webkit-transform:translateY(80%);
033     }
034 }
035 @-o-keyframes llevar{
036     from {
037         top:40%;
038         left:40%;
039     }
040     to {
041         -o-transform:translateY(80%);
042     }
043 }
044 @keyframes llevar {
045     from{
046         top:40%;
047         left:40%;
048     }
049     to{
050         transform:translateY(80%);
051     }
052 }

```

Redimensiona multiplicando el tamaño por el valor de x,y

scale(x, y)

ejemplo-02-258.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-transform:scale(2,1.5);
013     -webkit-transform: scale(2,1.5);
014     -o-transform:scale(2,1.5);
015     transform: scale(2,1.5);
016 }

```

ejemplo-02-259.css

```

001 #inicio{

```

```

002    background-color:blue;
003    color:white;
004    border:2px solid yellow;
005    width:32%;
006    height:160px;
007    position:absolute;
008    top:40%;
009    left:40%;
010   }
011  #inicio:hover{
012    -moz-animation:llevar 5s 2 linear alternate;
013    -webkit-animation:llevar 5s 2 linear alternate;
014    -o-animation:llevar 5s 2 linear alternate;
015    animation:llevar 5s 2 linear alternate;
016   }
017  @-moz-keyframes llevar{
018    from {
019      top:40%;
020      left:40%;
021    }
022    to {
023      -moz-transform:scale(2,1.5);
024    }
025  }
026  @-webkit-keyframes llevar {
027    from {
028      top:40%;
029      left:40%;
030    }
031    to {
032      -webkit-transform:scale(2,1.5);
033    }
034  }
035  @-o-keyframes llevar{
036    from {
037      top:40%;
038      left:40%;
039    }
040    to {
041      -o-transform:scale(2,1.5);
042    }
043  }
044  @keyframes llevar {
045    from{
046      top:40%;
047      left:40%;
048    }
049    to {
050      transform:scale(2,1.5);
051    }
052  }

```

scale3d(*x*,*y*,*z*)

ejemplo-02-260.css

```

001  #inicio{
002    background-color:blue;
003    color:white;
004    border:2px solid yellow;
005    width:32%;
006    height:160px;
007    position:absolute;
008    top:40%;
009    left:40%;
010   }
011  #inicio:hover{
012    -moz-transform:scale3d(2,1.5,2);
013    -webkit-transform:scale3d(2,1.5,2);
014    -o-transform:scale3d(2,1.5,2);
015    transform:scale3d(2,1.5,2);

```

016 }

Redimensiona del ancho

scaleX(x)
ejemplo-02-261.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-transform:scaleX(2.5);
013     -webkit-transform: scaleX(2.5);
014     -o-transform:scaleX(2.5);
015     transform: scaleX(2.5);
016 }
```

ejemplo-02-262.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-animation:llevar 5s 2 linear alternate;
013     -webkit-animation:llevar 5s 2 linear alternate;
014     -o-animation:llevar 5s 2 linear alternate;
015     animation:llevar 5s 2 linear alternate;
016 }
017 @-moz-keyframes llevar{
018     from {
019         top:40%;
020         left:40%;
021     }
022     to {
023         -moz-transform:scaleX(2.5);
024     }
025 }
026 @-webkit-keyframes llevar {
027     from {
028         top:40%;
029         left:40%;
030     }
031     to {
032         -webkit-transform:scaleX(2.5);
033     }
034 }
035 @-o-keyframes llevar{
036     from {
037         top:40%;
038         left:40%;
039     }
040     to {
041         -o-transform:scaleX(2.5);
042     }
043 }
044 @keyframes llevar{
045     from {
046         top:40%;
```

```

047     left:40%;  

048   }  

049   to{  

050     transform:scaleX(2.5);  

051   }  

052 }

```

Redimensionado del alto

scaleY(y)

ejemplo-02-263.css

```

001 #inicio{  

002   background-color:blue;  

003   color:white;  

004   border:2px solid yellow;  

005   width:32%;  

006   height:160px;  

007   position:absolute;  

008   top:40%;  

009   left:40%;  

010 }  

011 #inicio:hover{  

012   -moz-transform:scaleY(3.25);  

013   -webkit-transform: scaleY(3.25);  

014   -o-transform:scaleY(3.25);  

015   transform: scaleY(3.25);  

016 }

```

ejemplo-02-264.css

```

001 #inicio{  

002   background-color:blue;  

003   color:white;  

004   border:2px solid yellow;  

005   width:32%;  

006   height:160px;  

007   position:absolute;  

008   top:40%;  

009   left:40%;  

010 }  

011 #inicio:hover{  

012   -moz-animation:llevar 5s 2 linear alternate;  

013   -webkit-animation:llevar 5s 2 linear alternate;  

014   -o-animation:llevar 5s 2 linear alternate;  

015   animation:llevar 5s 2 linear alternate;  

016 }  

017 @-moz-keyframes llevar{  

018   from {  

019     top:40%;  

020     left:40%;  

021   }  

022   to {  

023     -moz-transform:scaleY(3.25);  

024   }  

025 }  

026 @-webkit-keyframes llevar {  

027   from{  

028     top:40%;  

029     left:40%;  

030   }  

031   to {  

032     -webkit-transform:scaleY(3.25);  

033   }  

034 }  

035 @-o-keyframes llevar{  

036   from {  

037     top:40%;  

038     left:40%;  

039   }  

040   to {  

041     -o-transform:scaleY(3.25);

```

```

042     }
043 }
044 @keyframes llevar{
045   from {
046     top:40%;
047     left:40%;
048   }
049   to{
050     transform:scaleY(3.25);
051   }
052 }
```

Redimensión de la profundidad

scaleZ(z)

ejemplo-02-270.css

```

001 #inicio{
002   background-color:blue;
003   color:white;
004   border:2px solidyellow;
005   width:32%;
006   height:160px;
007   position:absolute;
008   top:40%;
009   left:40%;
010 }
011 #inicio:hover{
012   -moz-transform:scaleZ(3.25);
013   -webkit-transform: scaleZ(3.25);
014   -o-transform:scaleZ(3.25);
015   transform: scaleZ(3.25);
016 }
```

rotar

rotate(ángulo)

ejemplo-02-265.css

```

001 #inicio{
002   background-color:blue;
003   color:white;
004   border:2px solidyellow;
005   width:32%;
006   height:160px;
007   position:absolute;
008   top:40%;
009   left:40%;
010 }
011 #inicio:hover{
012   -moz-transform:rotate(45deg);
013   -webkit-transform: rotate(45deg);
014   -o-transform:rotate(45deg);
015   transform: rotate(45deg);
016 }
```

ejemplo-02-266.css

```

001 #inicio{
002   background-color:blue;
003   color:white;
004   border:2px solid yellow;
005   width:32%;
006   height:160px;
007   position:absolute;
008   top:40%;
009   left:40%;
010 }
011 #inicio:hover{
012   -moz-animation:llevar 5s 2 linear alternate;
013   -webkit-animation:llevar 5s 2 linear alternate;
014   -o-animation:llevar 5s 2 linear alternate;
```

```

015     animation:llevar 5s 2 linear alternate;
016 }
017 @-moz-keyframes llevar{
018     from {
019         top:40%;
020         left:40%;
021     }
022     to {
023         -moz-transform:rotate(45deg);
024     }
025 }
026 @-webkit-keyframes llevar {
027     from {
028         top:40%;
029         left:40%;
030     }
031     to {
032         -webkit-transform:rotate(45deg);
033     }
034 }
035 @-o-keyframes llevar{
036     from {
037         top:40%;
038         left:40%;
039     }
040     to {
041         -o-transform:rotate(45deg);
042     }
043 }
044 @keyframes llevar{
045     from {
046         top:40%;
047         left:40%;
048     }
049     to{
050         transform:rotate(45deg);
051     }
052 }

```

rotateX(ángulo)

ejemplo-02-267.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-transform:rotateX(45deg);
013     -webkit-transform: rotateX(45deg);
014     -o-transform:rotateX(45deg);
015     transform: rotateX(45deg);
016 }

```

ejemplo-02-68css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{

```

```

012 -moz-animation:llevar 5s 2 linear alternate;
013 -webkit-animation:llevar 5s 2 linear alternate;
014 -o-animation:llevar 5s 2 linear alternate;
015 animation:llevar 5s 2 linear alternate;
016 }
017 @-moz-keyframes llevar{
018     from {
019         top:40%;
020         left:40%;
021     }
022     to {
023         -moz-transform:rotateX(45deg);
024     }
025 }
026 @-webkit-keyframes llevar {
027     from{
028         top:40%;
029         left:40%;
030     }
031     to{
032         -webkit-transform:rotateX(45deg);
033     }
034 }
035 @-o-keyframes llevar{
036     from {
037         top:40%;
038         left:40%;
039     }
040     to {
041         -o-transform:rotateX(45deg);
042     }
043 }
044 @keyframes llevar{
045     from {
046         top:40%;
047         left:40%;
048     }
049     to{
050         transform:rotateX(45deg);
051     }
052 }

```

rotateY(ángulo)

ejemplo-02-269.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-transform:rotateY(45deg);
013     -webkit-transform: rotateY(45deg);
014     -o-transform:rotateY(45deg);
015     transform: rotateY(45deg);
016 }

```

ejemplo-02-271.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;

```

```

009    left:40%;  

010 }  

011 #inicio:hover{  

012     -moz-animation:llevar 5s 2 linear alternate;  

013     -webkit-animation:llevar 5s 2 linear alternate;  

014     -o-animation:llevar 5s 2 linear alternate;  

015     animation:llevar 5s 2 linear alternate;  

016 }  

017 @-moz-keyframes llevar {  

018     from {  

019         top:40%;  

020         left:40%;  

021     }  

022     to {  

023         -moz-transform:rotateY(45deg);  

024     }  

025 }  

026 @-webkit-keyframes llevar {  

027     from {  

028         top:40%;  

029         left:40%;  

030     }  

031     to {  

032         -webkit-transform:rotateY(45deg);  

033     }  

034 }  

035 @-o-keyframes llevar {  

036     from {  

037         top:40%;  

038         left:40%;  

039     }  

040     to {  

041         -o-transform:rotateY(45deg);  

042     }  

043 }  

044 @keyframes llevar {  

045     from{  

046         top:40%;  

047         left:40%;  

048     }  

049     to {  

050         transform:rotateY(45deg);  

051     }  

052 }

```

rotateZ(ángulo)

ejemplo-02-272.css

```

001 #inicio{  

002     background-color:blue;  

003     color:white;  

004     border:2px solid yellow;  

005     width:32%;  

006     height:160px;  

007     position:absolute;  

008     top:40%;  

009     left:40%;  

010 }  

011 #inicio:hover{  

012     -moz-transform:rotateZ(60deg);  

013     -webkit-transform: rotateZ(60deg);  

014     -o-transform:rotateZ(60deg);  

015     transform: rotateZ(60deg);  

016 }

```

ejemplo-02-273.css

```

001 #inicio{  

002     background-color:blue;  

003     color:white;  

004     border:2px solid yellow;  

005     width:32%;  


```

```

006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-animation:llevar 5s 2 linear alternate;
013     -webkit-animation:llevar 5s 2 linear alternate;
014     -o-animation:llevar 5s 2 linear alternate;
015     animation:llevar 5s 2 linear alternate;
016 }
017 @-moz-keyframes llevar{
018     from {
019         top:40%;
020         left:40%;
021     }
022     to {
023         -moz-transform:rotateZ(60deg);
024     }
025 }
026 @-webkit-keyframes llevar {
027     from {
028         top:40%;
029         left:40%;
030     }
031     to {
032         -webkit-transform:rotateZ(60deg);
033     }
034 }
035 @-o-keyframes llevar{
036     from {
037         top:40%;
038         left:40%;
039     }
040     to {
041         -o-transform:rotateZ(60deg);
042     }
043 }
044 @keyframes llevar{
045     from {
046         top:40%;
047         left:40%;
048     }
049     to{
050         transform:rotateZ(60deg);
051     }
052 }

```

rotate3d(x, y, z, ángulo)

Siendo x, y, z los ejes de rotación.

ejemplo-02-281.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-transform:rotate3d(1,1,1,45deg);
013     -webkit-transform: rotate3d(1,1,1,45deg);
014     -o-transform:rotate3d(1,1,1,45deg);
015     transform: rotate3d(1,1,1,45deg);
016 }

```

Inclinación

skew(ángulo-x , ángulo-y)
ejemplo-02-274.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-transform:skew(15deg,20deg);
013     -webkit-transform: skew(15deg,20deg);
014     -o-transform:skew(15deg,20deg);
015     transform: skew(15deg,20deg);
016 }
```

ejemplo-02-275.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-animation:llevar 5s 2 linear alternate;
013     -webkit-animation:llevar 5s 2 linear alternate;
014     -o-animation:llevar 5s 2 linear alternate;
015     animation:llevar 5s 2 linear alternate;
016 }
017 @-moz-keyframes llevar{
018     from {
019         top:40%;
020         left:40%;
021     }
022     to {
023         -moz-transform:skew(15deg,20deg);
024     }
025 }
026 @-webkit-keyframes llevar{
027     from {
028         top:40%;
029         left:40%;
030     }
031     to {
032         -webkit-transform:skew(15deg,20deg);
033     }
034 }
035 @-o-keyframes llevar{
036     from {
037         top:40%;
038         left:40%;
039     }
040     to {
041         -o-transform:skew(15deg,20deg);
042     }
043 }
044 @keyframes llevar{
045     from {
046         top:40%;
047         left:40%;
048     }
049     to {
```

050	transform:skew(15deg,20deg);
051	}
052	}

Inclinación X

skewX(ángulo)

ejemplo-02-276.css

001	#inicio{
002	background-color:blue;
003	color:white;
004	border:2px solid yellow;
005	width:32%;
006	height:160px;
007	position:absolute;
008	top:40%;
009	left:40%;
010	}
011	#inicio:hover{
012	-moz-transform:skewX(30deg);
013	-webkit-transform: skewX(30deg);
014	-o-transform:skewX(30deg);
015	transform: skewX(30deg);
016	}

ejemplo-02-277.css

001	#inicio{
002	background-color:blue;
003	color:white;
004	border:2px solid yellow;
005	width:32%;
006	height:160px;
007	position:absolute;
008	top:40%;
009	left:40%;
010	}
011	#inicio:hover{
012	-moz-animation:llevar 5s 2 linear alternate;
013	-webkit-animation:llevar 5s 2 linear alternate;
014	-o-animation:llevar 5s 2 linear alternate;
015	animation:llevar 5s 2 linear alternate;
016	}
017	@-moz-keyframes llevar{
018	from {
019	top:40%;
020	left:40%;
021	}
022	to {
023	-moz-transform:skewX(30deg);
024	}
025	}
026	@-webkit-keyframes llevar {
027	from {
028	top:40%;
029	left:40%;
030	}
031	to {
032	-webkit-transform:skewX(30deg);
033	}
034	}
035	@-o-keyframes llevar{
036	from {
037	top:40%;
038	left:40%;
039	}
040	to {
041	-o-transform:skewX(30deg);
042	}
043	}
044	@keyframes llevar{

```

045 from {
046   top:40%;
047   left:40%;
048 }
049 to{
050   transform:skewX(30deg);
051 }
052 }
```

Inclinación Y

skewY(ángulo)

ejemplo-02-278.css

```

001 #inicio{
002   background-color:blue;
003   color:white;
004   border:2px solid yellow;
005   width:32%;
006   height:160px;
007   position:absolute;
008   top:40%;
009   left:40%;
010 }
011 #inicio:hover{
012   -moz-transform:skewY(45deg);
013   -webkit-transform: skewY(45deg);
014   -o-transform:skewY(45deg);
015   transform: skewY(45deg);
016 }
```

ejemplo-02-279.css

```

001 #inicio{
002   background-color:blue;
003   color:white;
004   border:2px solid yellow;
005   width:32%;
006   height:160px;
007   position:absolute;
008   top:40%;
009   left:40%;
010 }
011 #inicio:hover{
012   -moz-animation:llevar 5s 2 linear alternate;
013   -webkit-animation:llevar 5s 2 linear alternate;
014   -o-animation:llevar 5s 2 linear alternate;
015   animation:llevar 5s 2 linear alternate;
016 }
017 @-moz-keyframes llevar{
018   from {
019     top:40%;
020     left:40%;
021   }
022   to {
023     -moz-transform:skewY(45deg);
024   }
025 }
026 @-webkit-keyframes llevar {
027   from {
028     top:40%;
029     left:40%;
030   }
031   to {
032     -webkit-transform:skewY(45deg);
033   }
034 }
035 @-o-keyframes llevar{
036   from {
037     top:40%;
038     left:40%;
039   }
}
```

```

040     to {
041         -o-transform: skewY(45deg);
042     }
043 }
044 @keyframes llevar{
045     from {
046         top:40%;
047         left:40%;
048     }
049     to{
050         transform:skewY(45deg);
051     }
052 }

```

En donde transformación puede tomar uno de los siguientes valores:

matrix(*a, b, c, d, tx, ty*)

ejemplo-02-280.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-transform:matrix(1,2,-1,1,80,80);
013     -webkit-transform: matrix(1,2,-1,1,80,80);
014     -o-transform:matrix(1,2,-1,1,80,80);
015     transform: matrix(1,2,-1,1,80,80);
016 }

```

Define una transformación 2d usando una matriz de dieciseis valores (entre 0 y 1).

matrix3d(*a₁, a₂, a₃, a₄, b₁, b₂, b₃, b₄, c₁, c₂, c₃, c₄, d₁, d₂, d₃, d₄*)

ejemplo-02-282.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:32%;
006     height:160px;
007     position:absolute;
008     top:40%;
009     left:40%;
010 }
011 #inicio:hover{
012     -moz-transform:matrix3d(0.8535533905932737, 0.4999999999999999,
013     0.14644660940672619, 0, -0.4999999999999999, 0.7071067811865476,
014     0.4999999999999999, 0, 0.14644660940672619, -0.4999999999999999,
015     0.8535533905932737, 0, 22.62994231491119, -20.3223304703363, 101.3700576850888, 1);
016     -webkit-transform: matrix3d(0.8535533905932737, 0.4999999999999999,
017     0.14644660940672619, 0, -0.4999999999999999, 0.7071067811865476,
018     0.4999999999999999, 0, 0.14644660940672619, -0.4999999999999999,
019     0.8535533905932737, 0, 22.62994231491119, -20.3223304703363, 101.3700576850888, 1);
020     -o-transform:matrix3d(0.8535533905932737, 0.4999999999999999,
021     0.14644660940672619, 0, -0.4999999999999999, 0.7071067811865476,
022     0.4999999999999999, 0, 0.14644660940672619, -0.4999999999999999,
023     0.8535533905932737, 0, 22.62994231491119, -20.3223304703363, 101.3700576850888, 1);
024     transform: matrix3d(0.8535533905932737, 0.4999999999999999,
025     0.14644660940672619, 0, -0.4999999999999999, 0.7071067811865476,
026     0.4999999999999999, 0, 0.14644660940672619, -0.4999999999999999,
027     0.8535533905932737, 0, 22.62994231491119, -20.3223304703363, 101.3700576850888, 1);
028 }

```

Para cambiar la perspectiva tenemos

perspective(*n*)

ejemplo-02-283.css

```

001 .face{
002     position: absolute;
003     width: 100px;
004     height: 100px;
005     line-height: 100px;
006     font-size: 100px;
007     text-align: center;
008 }
009 .no-perspective-box{
010     width: 100px;
011     height: 100px;
012     transform-style: preserve-3d;
013     transform: rotateX(-15deg) rotateY(15deg);
014     margin-left: 100px;
015 }
016 .perspective-box{
017     width: 100px;
018     height: 100px;
019     transform-style: preserve-3d;
020     transform: perspective(7.5cm) rotateX(-15deg) rotateY(15deg);
021     margin-left: 100px;
022 }
023 .top{
024     background-color: skyblue;
025     transform: rotateX(90deg) translate3d(0, 0, 50px);
026 }
027 .left{
028     background-color: pink;
029     transform: rotateY(-90deg) translate3d(0, 0, 50px);
030 }
031 .front{
032     background-color: limegreen;
033     transform: translate3d(0, 0, 50px);
034 }
```

transform-origin: eje-x eje-y eje-z

Cambiar la posición del elemento, para el eje-x podemos poner uno de los siguientes valores:

left | center | right | valor

Para el eje-y podemos poner uno de los siguientes valores:

top | center | bottom | valor

Para el eje-z podemos poner

valor

ejemplo-02-286.css

```

001 #inicio{
002     background-color:blue;
003     color:white;
004     border:2px solid yellow;
005     width:20%;
006     height:160px;
007     position:absolute;
008     top:20%;
009     left:60%;
010     -moz-transform:matrix(1,2,-1,1,80,80);
011     -webkit-transform: matrix(1,2,-1,1,80,80);
```

```

012 -o-transform:matrix(1,2,-1,1,80,80);
013 transform: matrix(1,2,-1,1,80,80);
014 -moz-transform-origin: 20% 40%;
015 -webkit-transform-origin: 20% 40%;
016 -o-transform-origin: 20% 40%;
017 transform-origin: 20% 40%;
018 }
019 #final{
020 background-color:blue;
021 color:white;
022 border:2px solid yellow;
023 width:20%;
024 height:160px;
025 position:absolute;
026 top:20%;
027 left:20%;
028 }

```

Para indica si la transformación es en dos dimensiones o en tres.

transform-style:flat|preserve-3d

ejemplo-02-284.css

```

001 #padre{
002     width: 100px;
003     height: 100px;
004     transform-style: preserve-3d;
005     transform: perspective(7.5cm) rotateX(-15deg) rotateY(15deg);
006     margin-left: 100px;
007 }
008 .cubo{
009     position: absolute;
010     width: 100px;
011     height: 100px;
012 }
013 #inicio{
014     background-color:blue;
015     color:white;
016     border:2px solid yellow;
017     transform: translate3d(0, 0, 50px);
018 }
019 #izquierda,#izquierdal{
020     background-color:blue;
021     color:white;
022     border:2px solid yellow;
023     transform: rotateY(-90deg) translate3d(0, 0, 50px);
024 }
025 #superior,#superior1{
026     background-color:blue;
027     color:white;
028     border:2px solid yellow;
029     transform: rotateX(90deg) translate3d(0, 0, 50px);
030 }
031 #nuevo{
032     width: 100px;
033     height: 100px;
034     transform-style: preserve-3d;
035     transform: rotateX(-15deg) rotateY(15deg);
036     margin-left: 100px;
037 }
038 #seguido{
039     background-color:blue;
040     color:white;
041     border:2px solid yellow;
042     transform: translate3d(0, 0, 50px);
043 }

```

14.-Otros.

Espacio de nombre, namespace

@nanmespaces [prefijo] url | cadena

@namespace url

Toda la especificación es para esa url

001	<code>@namespace "http://www.mipagina.com";</code>
002	<code>@namespace "http://www.pagina.com/felix";</code>

@namespace prefijo url

Toda la especificación que lleve el prefijo indicado es para la url indicada

001	<code>@namespace fl "http://www.paginainterfaces.com/felix";</code>
002	<code>@namespace ng "http://www.paginainterfaces.com/angel";</code>

Para aplicarlos

prefijo | selector { estilo }

001	<code>f1 a{color:orange;}</code>
002	<code>ng a{color:brown;}</code>

Para establecer el cursor del ratón que se va a mostrar

cursor:nombre-cursor

cursor:url(fichero) , auto

Si no poneis al final auto no funciona.

Los nombres de cursores que podemos utilizar son:

Nombre cursor	Tipo de cursor
all-scroll	Se puede realizar scroll en cualquier dirección
auto	El cursor por defecto
cell	Una celda puede ser seleccionada
context-menu	El menú contextual está disponible.
col-resize	Una columna puede ser redimensionada horizontalmente.
copy	Cursor de copia y pegado
crosshair	Una cruz
default	El cursor por defecto
e-resize	Una caja se mueve a la derecha
ew-resize	Redimensionado bidireccional.
grab	Algo se puede arrastrar
grabbing	Algo se puede arrastrar
help	Ayuda
move	Elemento que se mueve.
n-resize	Moviéndose hacia arriba
ne-resize	Moviéndose hacia arriba
nesw-resize	Redimensionado en ambos sentidos
ns-resize	Redimensionado en ambos sentidos
nw-resize	Moviéndose hacia arriba y a la izquierda

Nombre cursor	Tipo de cursor
nwse-resize	Redimensionado en ambos sentidos
no-drop	El elemento arrastrado no puede moverse
none	No hay cursor
not-allowed	La acción solicitada no se ejecutará
pointer	Un puntero indica un enlace
progress	Programa ocupado o en proceso
row-resize	Fila redimensionada verticalmente
s-resize	Moviendo abajo
se-resize	Moviendo abajo a la derecha
sw-resize	Moviendo abajo a la izquierda
text	El texto puede ser seleccionado
vertical-text	El texto vertical puede ser seleccionado
w-resize	Moviendo a la izquierda
wait	Programa ocupado
zoom-in	zoom in
zoom-out	zoom out

ejemplo-02-300.css

```

001 #listas{
002     display:grid;
003     width:100%;
004     border:4px solid blue;
005     -ms-grid:1fr 1fr 1fr auto auto 1fr 1fr1fr1fr1fr1fr 1fr/ auto autoautoauto;
006     grid:1fr 1fr 1fr auto auto 1fr 1fr1fr1fr1fr1fr / auto autoautoauto;
007     -ms-grid-auto-flow:dense;
008     grid-auto-flow:dense;
009
010 }
011 #listas>div {
012     border:2px solid red;
013 }
014 #uno>ol {cursor:copy;}
015 #dos>ol {cursor:help;}
016 #tres>ol {cursor:move;}
017 #cuatro>ol {cursor:all-scroll;}
018 #seis>ol {cursor:auto;}
019 #siete>ol {cursor:cell;}
020 #ocho>ol {cursor:context-menu;}
021 #nueve>ol {cursor:col-resize;}
022 #diez>ol {cursor:crosshair;}
023 #once>ol {cursor:default;}
024 #doce>ol {cursor:e-resize;}
025 #trece>ol {cursor:ew-resize;}
026 #catorce>ol {cursor:grab;}
027 #quince>ol {cursor:grabbing;}
028 #diecisiete>ol {cursor:n-resize;}
029 #diecisiete>ol {cursor:ne-resize;}
030 #dieciocho>ol {cursor:nesw-resize;}
031 #diecinueve>ol {cursor:ns-resize;}
032 #veinte>ol {cursor:nw-resize;}
033 #veinte01>ol {cursor:nwse-resize;}
034 #veinte02>ol {cursor:no-drop;}
035 #veinte03>ol {cursor:none;}
036 #veinte04>ol {cursor:not-allowed;}
037 #veinte05>ol {cursor:pointer;}
038 #veinte06>ol {cursor:progress;}
039 #veinte07>ol {cursor:row-resize;}

```

040	#veinte08>ol {cursor:s-resize;}
041	#veinte09>ol {cursor:se-resize;}
042	#treinta>ol {cursor:sw-resize;}
043	#treinta01>ol {cursor:text;}
044	#treinta02>ol {cursor:vertical-text;}
045	#treinta03>ol {cursor:w-resize;}
046	#treinta04>ol {cursor:wait;}
047	#treinta05>ol {cursor:zoom-in;}
048	#treinta06>ol {cursor:zoom-out;}
049	#treinta07{cursor:url("../Cursos/MANO.CUR"),auto;}
050	#treinta08{cursor:url(..../Cursos/ALBERT.CUR),auto;}

Para controlar eventos del ratón

pointer-events: auto | none | inherit

La propiedad pointer-events define si un elemento reacciona o no a eventos de puntero.

Con **none**: el elemento se comporta como lo haría si no se especificara la propiedad de eventos de puntero, evita todas las opciones de clic, estado y cursor en el elemento HTML especificado

Con **auto**:restaura la funcionalidad predeterminada.

Con **inherit**: se utiliza los valores del elemento padre.

Uso de variables en css

Para declarar una variable lo que debemos hacer es asignarla un valor, sabiendo que el nombre de la variable va a empezar por dos guiones consecutivos y seguidos del nombre asignado, cuando se hace referencia a la misma también vamos a utilizar los guiones.

--**nombre:valor**

También podemos poner una expresión.

--**nombre:calc(expresión)**

Luego para utilizar la variable vamos a utilizar

var(--nombre)

La variable también se puede incluir en una expresión en la que se incluya la variable.

calc(expresión)

Podemos utilizar una variable e indicar que si no está declarada que utilice un valor, pero solo esa vez.

var(--nombre , valor)

Podemos tener declarado el mismo nombre de variable para varios elementos y aplicarla para todos ellos y tomará el valor de la variable en cada elemento.

ejemplo-02-301.css

001	:root{
002	--micolor:blue;
003	--tamLetra:10;
004	--lcolor:red;
005	--ltam:14;
006	font-size:calc(var(--ltam,16)*1pt);

007	
008	}
009	#primero{
010	background-color:var(--micolor);
011	font-size:var(--tamLetra);
012	width:calc(var(--tamLetra)*50px);
013	--lcolor:white;
014	}
015	#muerta{
016	--lcolor:green;
017	--litam:calc(var(--ltam)*0.5+3);
018	font-size:calc(var(--litam)*1pt);
019	}
020	#acueducto{
021	--lcolor:yellow;
022	}
023	* {
024	color: var(--lcolor);
025	}

g) Crear y vincular hojas de estilo.

Para establecer un estilo vinculado utilizaremos:

`<link href="hoja-estilo" rel="stylesheet" type="text/css">`

Así mismo podemos utilizar dentro de la etiqueta de estilos STYLE:

`@import url("hoja-estilo");
@import url(hoja-estilo);
@import "hoja-estilo";`

Podemos poner varias hojas de estilo en un mismo documento HTML, si queremos que una de ellas sea la alternativa a otra deberemos poner en la etiqueta LINK dentro del atributo REL y al principio del mismo la palabra **alternate** y además deberemos incluir el atributo TITLE, en Firefox, Internet Explorer tiene el efecto de que en el menú ver se puede seleccionar uno de los dos estilos. Si queremos tener estilos alternativos en el resto de navegadores deberemos incluir código en JavaScript (se verá más adelante).

h) Crear y vincular hojas de estilo en cascada externa.

Para incluir dentro de una hoja de estilos otra hoja de estilos usaremos:

`@import url("hoja-estilo");
@import url(hoja-estilo);
@import "hoja-estilo";`

i) Herramientas y test de verificación.

Para validar CSS tenemos la siguiente dirección web.

<http://jigsaw.w3.org/css-validator/>

<http://www.delosprimeros.com/validador-css.htm>

http://translate.google.es/translate?hl=es&langpair=en%7Ces&u=http://validator.w3.org/&ei=GryqULiKE-HM0QXH_4H4BQ

Browser Support

CSS Analyzer

CSSCheck

Anexo I .-Nombres de colores:

Color Name	HEX
AliceBlue	#F0F8FF
AntiqueWhite	#FAEBD7
Aqua	#00FFFF
Aquamarine	#7FFFDD
Azure	#FOFFFF
Beige	#F5F5DC
Bisque	#FFE4C4
Black	#000000
BlanchedAlmond	#FFEBBC
Blue	#0000FF
BlueViolet	#8A2BE2
Brown	#A52A2A
BurlyWood	#DEB887
CadetBlue	#5F9EA0
Chartreuse	#7FFF00
Chocolate	#D2691E
Coral	#FF7F50
CornflowerBlue	#6495ED
Cornsilk	#FFF8DC
Crimson	#DC143C
Cyan	#00FFFF
DarkBlue	#00008B
DarkCyan	#008B8B
DarkGoldenRod	#B8860B
DarkGray	#A9A9A9
DarkGreen	#006400
DarkKhaki	#BDB76B
DarkMagenta	#8B008B
DarkOliveGreen	#556B2F
DarkOrange	#FF8C00
DarkOrchid	#9932CC
DarkRed	#8B0000
DarkSalmon	#E9967A
DarkSeaGreen	#8FB88F
DarkSlateBlue	#483D8B
DarkSlateGray	#2F4F4F
DarkTurquoise	#00CED1
DarkViolet	#9400D3
DeepPink	#FF1493
DeepSkyBlue	#00BFFF
DimGray	#696969
DodgerBlue	#1E90FF
FireBrick	#B22222
FloralWhite	#FFFFA0
ForestGreen	#228B22
Fuchsia	#FF00FF
Gainsboro	#DCDCDC
GhostWhite	#F8F8FF

Color Name	HEX
Gold	#FFD700
GoldenRod	#DAA520
Gray	#808080
Green	#008000
GreenYellow	#ADFF2F
HoneyDew	#FOFFF0
HotPink	#FF69B4
IndianRed	#CD5C5C
Indigo	#4B0082
Ivory	#FFFFFF
Khaki	#F0E68C
Lavender	#E6E6FA
LavenderBlush	#FFF0F5
LawnGreen	#7CFC00
LemonChiffon	#FFFACD
LightBlue	#ADD8E6
LightCoral	#F08080
LightCyan	#E0FFFF
LightGoldenRodYellow	#FAFAD2
LightGray	#D3D3D3
LightGreen	#90EE90
LightPink	#FFB6C1
LightSalmon	#FFA07A
LightSeaGreen	#20B2AA
LightSkyBlue	#87CEFA
LightSlateGray	#778899
LightSteelBlue	#B0C4DE
LightYellow	#FFFFFF
Lime	#00FF00
LimeGreen	#32CD32
Linen	#FAF0E6
Magenta	#FF00FF
Maroon	#800000
MediumAquaMarine	#66CDAA
MediumBlue	#0000CD
MediumOrchid	#BA55D3
MediumPurple	#9370DB
MediumSeaGreen	#3CB371
MediumSlateBlue	#7B68EE
MediumSpringGreen	#00FA9A
MediumTurquoise	#48D1CC
MediumVioletRed	#C71585
MidnightBlue	#191970
MintCream	#F5FFFA
MistyRose	#FFE4E1
Moccasin	#FFE4B5
NavajoWhite	#FFDEAD
Navy	#000080

Color Name	HEX
OldLace	#FDF5E6
Olive	#808000
OliveDrab	#6B8E23
Orange	#FFA500
OrangeRed	#FF4500
Orchid	#DA70D6
PaleGoldenRod	#EEE8AA
PaleGreen	#98FB98
PaleTurquoise	#AFEEEE
PaleVioletRed	#DB7093
PapayaWhip	#FFEFBD
PeachPuff	#FFDAB9
Peru	#CD853F
Pink	#FFC0CB
Plum	#DDAODD
PowderBlue	#B0E0E6
Purple	#800080
Red	#FF0000
RosyBrown	#BC8F8F
RoyalBlue	#4169E1
SaddleBrown	#8B4513
Salmon	#FA8072
SandyBrown	#F4A460
SeaGreen	#2E8B57
SeaShell	#FFF5EE
Sienna	#A0522D
Silver	#C0C0C0
SkyBlue	#87CEEB
SlateBlue	#6A5ACD
SlateGray	#708090
Snow	#FFFAFA
SpringGreen	#00FF7F
SteelBlue	#4682B4
Tan	#D2B48C
Teal	#008080
Thistle	#D8BFD8
Tomato	#FF6347
Turquoise	#40E0D0
Violet	#EE82EE
Wheat	#F5DEB3
White	#FFFFFF
WhiteSmoke	#F5F5F5
Yellow	#FFFF00
YellowGreen	#9ACD32

Anexo II -Código de Idiomas y países.

Código	Idioma y País
ar	árabe
ar-ae	árabe (Emiratos Árabes Unidos)
ar-bh	árabe (Bahráin)
ar-dz	árabe (Argelia)
ar-eg	árabe (Egipto)
ar-iq	árabe (Irak)
ar-jo	árabe (Jordania)
ar-kw	árabe (Kuwait)
ar-lb	árabe (Líbano)
ar-ly	árabe (Libia)
ar-ma	árabe (Marruecos)
ar-om	árabe (Omán)
ar-qa	árabe (Qatar)
ar-sa	árabe (Arabia Saudita)
ar-sd	árabe (Sudán)
ar-sy	árabe (Siria)
ar-tn	árabe (Túnez)
ar-ye	árabe (Yemen)
be	bielorruso
be-by	bielorruso (Bielorrusia)
bg	búlgaro
bg-bg	búlgaro (Bulgaria)
ca	catalán
ca-es	catalán (España)
cs	checo
cs-cz	checo (Chequia)
da	danés
da-dk	danés (Dinamarca)
de	alemán
de-at	alemán (Austria)
de-ch	alemán (Suiza)
de-de	alemán (Alemania)
de-lu	alemán (Luxemburgo)
el	griego
el-cy	griego (Chipre)
el-gr	griego (Grecia)
en	inglés
en-au	inglés (Australia)
en-ca	inglés (Canadá)
en-gb	inglés (Reino Unido)
en-ie	inglés (Irlanda)
en-in	inglés (India)
en-mt	inglés (Malta)
en-nz	inglés (Nueva Zelanda)
en-ph	inglés (Filipinas)
en-sg	inglés (Singapur)

Código	Idioma y País
en-us	inglés (Estados Unidos)
en-za	inglés (Sudáfrica)
es	español
es-ar	español (Argentina)
es-bo	español (Bolivia)
es-cl	español (Chile)
es-co	español (Colombia)
es-cr	español (Costa Rica)
es-do	español (República Dominicana)
es-ec	español (Ecuador)
es-es	español (España)
es-gt	español (Guatemala)
es-hn	español (Honduras)
es-mx	español (México)
es-ni	español (Nicaragua)
es-pa	español (Panamá)
es-pe	español (Perú)
es-pr	español (Puerto Rico)
es-py	español (Paraguay)
es-sv	español (El Salvador)
es-us	español (Estados Unidos)
es-uy	español (Uruguay)
es-ve	español (Venezuela)
et	estonio
et-ee	estonio (Estonia)
fi	finés
fi-fi	finés (Finlandia)
fr	francés
fr-be	francés (Bélgica)
fr-ca	francés (Canadá)
fr-ch	francés (Suiza)
fr-fr	francés (Francia)
fr-lu	francés (Luxemburgo)
ga	irlandés
ga-ie	irlandés (Irlanda)
hi-in	hindú (India)
hr	croata
hr-hr	croata (Croacia)
hu	húngaro
hu-hu	húngaro (Hungria)
in	indonesio
in-id	indonesio (Indonesia)
is	islandés
is-is	islandés (Islandia)
it	italiano
it-ch	italiano (Suiza)

Código	Idioma y País
it-it	italiano (Italia)
iw	hebreo
iw-il	hebreo (Israel)
ja	japonés
ja-jp	japonés (Japón)
ja-jp-jp	japonés (Japón,JP)
ko	coreano
ko-kr	coreano (Corea del Sur)
lt	lituano
lt-lt	lituano (Lituania)
lv	letón
lv-lv	letón (Letonia)
mk	macedonio
mk-mk	macedonio (Macedonia)
ms	malayo
ms-my	malayo (Malasia)
mt	maltés
mt-mt	maltés (Malta)
nl	neerlandés
nl-be	neerlandés (Bélgica)
nl-nl	neerlandés (Holanda)
no	noruego
no-no	noruego (Noruega)
no-no-ny	noruego (Noruega,Nynorsk)
pl	polaco
pl-pl	polaco (Polonia)
pt	portugués
pt-br	portugués (Brasil)
pt-pt	portugués (Portugal)
ro	rumano
ro-ro	rumano (Rumania)

Código	Idioma y País
ru	ruso
ru-ru	ruso (Rusia)
sk	eslovaco
sk-sk	eslovaco (Eslovaquia)
sl	eslovenio
sl-si	eslovenio (Eslovenia)
sq	albanés
sq-al	albanés (Albania)
sr	serbio
sr-ba	serbio (Bosnia y Hercegovina)
sr-cs	serbio (Serbia y Montenegro)
sr-me	serbio (Montenegro)
sr-rs	serbio (Serbia)
sv	sueco
sv-se	sueco (Suecia)
th	tailandés
th-th	tailandés (Tailandia)
th-th-th	tailandés (Tailandia,TH)
tr	turco
tr-tr	turco (Turquía)
uk	ucranio
uk-ua	ucranio (Ucrania)
vi	vietnamita
vi-vn	vietnamita (Vietnam)
zh	chino
zh-cn	chino (China)
zh-hk	chino (Hong Kong)
zh-sg	chino (Singapur)
zh-tw	chino (Taiwán)

Anexo III. Lista de Códigos de Idiomas.

Idioma	Código Idioma
abjaso (o abjasiano)	ab
afar	aa
afrikaans	af
aimara	ay
akano	ak
albanés	sq
alemán	de
amárico	am
árabe	ar
ragonés	an
armenio	hy
asamés	as
avar	av
avéstico	ae
azerí	az
bambara	bm
baskir	ba
bengalí	bn
bhojpuri	bh
bielorruso	be
birmano	my
bislama	bi
bosnio	bs
bretón	br
búlgaro	bg
cachemiro	ks
camboyano (o jemer)	km
canarés	kn
catalán	ca
chamorro	ch
checheno	ce
checo	cs
chichewa	ny
chino	zh
chuan (o zhuang)	za
chuvasio	cv
cingalés	si
coreano	ko
córnico	kw
corso	co
cree	cr
croata	hr
danés	da
dzongkha	dz
eslavo eclesiástico antiguo	cu
eslovaco	sk
esloveno	sl

Idioma	Código Idioma
español (o castellano)	es
esperanto	eo
estonio	et
ewe	ee
feroés	fo
fijiano (o fidji)	fj
finés (o finlandés)	fi
francés	fr
frisón (o frisio)	fy
fula	ff
gaélico escocés	gd
galés	cy
gallego	gl
georgiano	ka
griego (moderno)	el
groenlandés (o kalaallisut)	kl
guaraní	gn
guyaratí (o guyaratí)	gu
haitiano	ht
hausa	ha
hebreo	he
herero	hz
hindí (o hindú)	hi
iri motu	ho
húngaro	hu
ido	io
igbo	ig
indonesio	id
inglés	en
interlingua	ia
inuktitut	iu
inupiaq	ik
irlandés (o gaélico)	ga
islandés	is
italiano	it
japonés	ja
javanés	jv
kanuri	kr
kazajo (o kazajio)	kk
kikuyu	ki
kirguís	ky
kirundi	rn
komi	kv
kongo	kg
kuanyama	kj
kurdo	ku
lao	lo

Idioma	Código Idioma
latín	la
letón	lv
limburgués	li
lingala	ln
lituano	lt
luba-katanga	lu
luganda	lg
luxemburgoés	lb
macedonio	mk
malayalam	ml
malayo	ms
maldivo	dv
malgache (o malagasy)	mg
maltés	mt
manés (gaélico manés o de Isla de Man)	gv
maorí	mi
maratí	mr
marshalés	mh
moldavo	mo
mongol	mn
nauruano	na
navajo	nv
ndebele del norte	nd
ndebele del sur	nr
ndonga	ng
neerlandés (u holandés)	nl
nepalí	ne
noruego	no
noruego bokmål	nb
nyorsk	nn
occidental	je
occitano	oc
ojibwa	oj
oriya	or
oromo	om
osético	os
pali	pi
panyabí (o penyabi)	pa
pastú (o pashto)	ps
persa	fa
polaco	pl
portugués	pt
quechua	qu
retorrománico	rm
ruandés	rw
rumano	ro

Idioma	Código Idioma
ruso	ru
sami septentrional	se
samoano	sm
sango	sg
sánskrito	sa
sardo	sc
serbio	sr
sesotho	st
setsuana	tn
shona	sn
sindhi	sd
somalí	so
suajili	sw
suazi (swati o siSwati)	ss
sueco	sv
sundanés	su
tagalo	tl
tahitiano	ty
tailandés	th
tamil	ta
tártaro	tt
tayiko	tg
telugú	te
tibetano	bo
tigríña	ti
tonganó	to
tsonga	ts
turco	tr
turcomano	tk
twi	tw
ucraniano	uk
uigur	ug
urdu	ur
uzbeko	uz
valón	wa
vascuence (o euskera)	eu
venda	ve
vietnamita	vi
volapük	vo
wolof	wo
xhosa	xh
yi de Sichuán	ii
yídish (o yiddish)	yi
yoruba	yo
zulú	zu