

Tema 2: Introducción al CSS



Ciclo Superior DAW

Asignatura: Diseño de interfaces

Curso 20/21



Introducción

- En este capítulo veremos los siguientes conceptos:
 - Reconocer las posibilidades de modificar las etiquetas HTML
 - Definir estilos de forma directa
 - Definir y asociar estilos globales en hojas externas
 - Definir hojas de estilo alternativas
 - Identificar las distintas propiedades de cada elemento
 - Crear clases e IDs de estilos



Hojas de estilos CSS

- Las **hojas de estilos** o **CSS** (Cascading Style Sheets) son **un estándar de la W3C** para **aplicar distintos estilos, formatos y características** a los **documentos como HTML**.
- En la práctica son **ficheros de texto** en los que se van definiendo **propiedades concretas para los elementos de la página**.



Uso de hojas de estilos CSS

- Una **hoja de estilos CSS** nos **permite definir las propiedades visuales de los elementos y aplicarlos a una página web**
- Esta separación permite que el diseño esté centralizado y el mantenimiento sea mucho más sencillo



Uso de hojas de estilos CSS

- Gracias a las hojas de estilos se consigue separar totalmente el contenido (HTML) de la forma de presentarlo.
- A un mismo documento HTML se le pueden aplicar distintas hojas de estilo CSS y conseguir un aspecto totalmente distinto.



Uso de hojas de estilos CSS

- Se puede aplicar la hoja de estilos de varias formas:
 - En nuestra página HTML, usando la etiqueta `<STYLE>`

```
<style type="text/css">
  body {
    background: #7f8c8d;
    color: #fff;
    display: inline-block;
    font-size: 13px;
    margin: 5px 5px;
    padding: 4px 5px;
    text-align: center;
    border:none;
  }
</style>
```



Uso de hojas de estilos CSS

- Se puede aplicar la hoja de estilos de varias formas:
 - En un fichero de texto externo, usando la etiqueta **<LINK>**
 - Conviene definir la hoja de estilos en un fichero un fichero de texto con extensión **.css**.

```
<link rel="stylesheet" href="css/estilos.css" type="text/css">
```

- Utilizar la función **@import** en un elemento style dentro del HTML.

```
<style type="text/css">  
  @import url(css/estilos.css)  
</style>
```



Actividad 1

Las sentencias que aparecen dentro de la etiqueta <style>, mételas en un fichero CSS y aplícalo a la web con los dos métodos que hemos visto



Uso de hojas de estilos CSS. ID

- Cada elemento de una página HTML puede tener un **atributo id que lo identifique de manera inequívoca en todo el documento.**
- El ID se suele usar para acceder desde código JavaScript, y también la aplicación de estilos propios para un solo elemento.
- En ese caso se emplea el símbolo # delante del nombre de estilo.



Uso de hojas de estilos CSS. ID

- Veamos un ejemplo donde se definen varios elementos que tienen el atributo ID, y en su hoja de estilos se aplican modificaciones a través del valor de ID.

- CSS:

```
#titulo {  
    color: gray;  
    font-family: Verdana;  
    font-size: 40pt;  
}
```

- HTML:

```
<h1 id="titulo">Encabezado</h1>
```



Uso de hojas de estilos CSS. Clases

- El **atributo class** se usa para **identificar un conjunto de elementos**.
- Los estilos class se definen poniendo un punto por delante del nombre de la clase
- Por ejemplo:

```
<style type="text/css">
  .miestilo {
    color: black;
    font-size: 14pt;
  }
</style>
```



Uso de hojas de estilos CSS. Pseudo-clases

- Las pseudoclasas van más allá de los estilos y permiten controlar elementos circunstanciales de página HTML
- Por ejemplo:

```
a:link { color: blue; }           /* Enlaces no visitados */  
a:visited { color: purple; }      /* Enlaces visitados */  
a:hover { background: yellow; }  /* El ratón está sobre el enlace */  
a:active { color: lime; }        /* Enlaces activos */
```



Aplicación de estilos

- Existen distintas formas de aplicar los estilos:
 - Aplicar estilos **a elementos concretos de un documento HTML, usando <pre>**
 - Aplicar estilos **a clases definidas por nosotros, a través del atributo class**
 - Aplicar estilos **a un único elemento del documento a través del atributo ID**
 - **Sin hoja de estilos, usando el atributo style** que tienen todas las etiquetas HTML



Aplicación de estilos

- Vemos algunos ejemplos:
 - Para aplicar un mismo estilo a varios elementos podemos separarlos con comas:

```
a, p, pre {  
    font-size: 12pt;  
}
```

- Para aplicar un ID a determinada etiqueta HTML:

```
a#enlaceespecial {  
    text-decoration: underline;  
    color: red;  
}
```



Aplicación de estilos

- Vemos algunos ejemplos:
 - Para aplicar una clase a determinada etiqueta HTML:

```
p.presentacion {  
    color:gray;  
    text-align:left;  
}
```



Actividad 2

Aplica un estilo a un elemento utilizando su ID.

Aplica un estilo a varios elementos utilizando la clase a la que pertenecen.



Aplicación de estilos. Fuentes CSS3

- Las fuentes CSS3 nos permiten utilizar fuentes personalizadas en un sitio web, pero no hay un formato único de fuente para todos los navegadores, así que la fuente debe aparecer en todos los formatos posibles:

FORMATO	DESCRIPCIÓN
EOT	Embedded open type
TTF	True Type Font
SVG	Scalar Vector Graphics
WOFF	Web Open Font Format



Aplicación de estilos. Fuentes CSS3

- Vemos un ejemplo en un documento con dos párrafos:

- **HTML**

```
<section>
  <div class="roman">
    Este DIV usa una fuente Times
  </div>
  <div class="especial">
    Este DIV usa una fuente Comic Sans
  </div>
</section>
```

- **CSS**

```
.roman {
  font-family: "Times New Roman", Times, serif;
}

.especial {
  font-family: "Comic Sans MS", cursive, sans-serif;
}
```



Aplicación de estilos. Fuentes CSS3

- Vemos un ejemplo en un documento con dos párrafos:
 - **Resultado**

Este DIV usa una fuente Times
Este DIV usa una fuente Comic Sans



Actividad 3

Investiga un poco acerca de las fuentes CSS3 que hay disponibles y aplica dos fuentes diferentes al fichero de la actividad anterior.



Aplicación de estilos. Listas

- Las listas también tienen su hueco entre las propiedades de las hojas de estilos.
- El ejemplo más conocido es cambiar el símbolo que marca el inicio de cada elemento de la lista.
- Veremos a continuación el código:



Aplicación de estilos. Listas

- HTML:

```
<ul class="listasespeciales">  
  <li><a>Google</a></li>  
  <li><a>Pinterest</a></li>  
  <li><a>Flickr</a></li>  
</ul>
```

- CSS:

```
.listasespeciales {  
  color: green;  
  font-family: "Arial, Verdana";  
}
```



Aplicación de estilos. Listas

- Resultado:





Aplicación de estilos. Tablas

- Para facilitar el trabajo de diseño de tablas la mejor solución pasa por aplicar un estilo a toda la tabla.
- Además, gracias a las etiquetas **<thead>**, **<tbody>** y **<tfoot>** se puede delimitar la tabla en distintas zonas y aplicar estilos específicos para cada una de ellas.

Aplicación de estilos. Tablas

En este ejemplo,
vemos cómo aplicar un sencillo
estilo a una única tabla:

Aquí tenemos el HTML:

```
<table>
  <thead>
    <tr>
      <th>Nombre</th>
      <th>Posición</th>
      <th>Altura</th>
      <th>Año nacimiento</th>
      <th>Salario</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Karim Benzema</td>
      <td>DC</td>
      <td>180cm</td>
      <td>10-11-1990</td>
      <td>23.917.000€</td>
    </tr>
    <tr>
      <td>Cristiano Ronaldo</td>
      <td>DC</td>
      <td>184cm</td>
      <td>02-07-1989</td>
      <td>47.360.400€</td>
    </tr>
    <tr>
      <td>Thibaut Courtois</td>
      <td>P</td>
      <td>198cm</td>
      <td>02-11-1993</td>
      <td>12.895.960€</td>
    </tr>
  </tbody>
</table>
```



Aplicación de estilos. Tablas

Aquí tenemos el CSS:

```
table {  
    border: 1px solid gray;  
}  
  
thead {  
    background-color: gray;  
    color: white;  
    font-style: italic;  
    text-align: center;  
}  
  
tbody {  
    background-color: white;  
    color: gray;  
    font-style: italic;  
    text-align: center;  
}  
  
tfoot {  
    background-color: black;  
    color: white;  
    text-align: center;  
}
```





Aplicación de estilos. Tablas

- Este sería el resultado final:

<i>Nombre</i>	<i>Posición</i>	<i>Altura</i>	<i>Año nacimiento</i>	<i>Salario</i>
<i>Karim Benzema</i>	<i>DC</i>	<i>180cm</i>	<i>10-11-1990</i>	<i>23.917.000€</i>
<i>Cristiano Ronaldo</i>	<i>DC</i>	<i>184cm</i>	<i>02-07-1989</i>	<i>47.360.400€</i>
<i>Thibaut Courtois</i>	<i>P</i>	<i>198cm</i>	<i>02-11-1993</i>	<i>12.895.960€</i>



Aplicación de estilos. Formularios

- Los formularios con sus líneas y sus colores grises pueden romper el diseño de una página HTML.
- Sin embargo, gracias a los estilos, se pueden adecuar los formularios tanto en líneas como en colores, evitando así las interfaces muy repetitivas.
- Veremos un ejemplo a continuación:



Aplicación de estilos. Formularios

- HTML:

```
<form name="Formulario" method="post" class="bordeazul" id="Formulario">
  <table class="especial">
    <tbody>
      <tr>
        <td>Login</td>
        <td><input name="Login" type="text"></td>
      </tr>
      <tr>
        <td>Password</td>
        <td><input name="Password" type="password"></td>
      </tr>
      <tr>
        <td>Descripción</td>
        <td>
          <textarea name="descripcion">
            Contenido inicial de la descripción
          </textarea>
        </td>
      </tr>
      <tr>
        <td>Login</td>
        <td><input name="Login" type="text"></td>
      </tr>
    </tbody>
  </table>
</form>
```



Aplicación de estilos. Formularios

- CSS:

```
.especial {  
    border-top: dashed;  
    border-bottom: dashed;  
    border-right: dashed;  
    border-left: dashed;  
    color: blue;  
    font-size: 8pt;  
    border-bottom: thick solid blue;  
}  
  
form input, textarea {  
    border: 1px dashed blue;  
    color: green;  
}
```



Aplicación de estilos. Formularios

- Resultado final:

Login

Password

Descripción

Contenido inicial de la descripción

Login



Actividad 4

Añadimos a la actividad anterior una tabla, una lista y un formulario, y le definimos un estilo particular para cada uno de los elementos en el fichero CSS



Maquetación con bloques <DIV> y estilos

- El **<div>** es una **etiqueta que nos sirve para agrupar contenido en un documento HTML**. Dentro de un **<div>** se puede meter cualquier cosa, incluyendo más divs anidados.
- Lo mismo con los elementos ****, con la diferencia de que el **<div>** ocupa por defecto todo el ancho del documento y provoca un salto de línea, mientras que el **** no.



Maquetación con bloques <DIV> y estilos

- **IMPORTANTE:** En lugar de utilizar tablas para el diseño de la página, se recomienda definir bloques DIV con un posicionamiento concreto.
- Resulta menos engorroso que las tablas, y es la tendencia actual en cuanto a desarrollo de páginas.



Maquetación con bloques <DIV> y estilos.

Posicionamiento

- Los elementos HTML se colocan siguiendo de izquierda a derecha y de arriba a abajo.
- Si en el documento se ha metido una tabla, una imagen, una cabecera y un párrafo, en el navegador seguirán ese flujo y se posicionarán uno debajo del otro.
- Pero esto se puede alterar con las propiedades de posicionamiento CSS.



Maquetación con bloques <DIV> y estilos.

Posicionamiento

- Para posicionar los elementos, por ejemplo, los bloques DIV, se usan las propiedades **top**, **left**, **bottom** y **right** dándoles un valor en píxeles.
- Gracias a eso se dan las coordenadas a los elementos para que se posicionen en uno u otro sitio.
- La forma de posicionarse puede variar según la propiedad **position**



Maquetación con bloques <DIV> y estilos.

Posicionamiento

- Según el valor de la propiedad position:
 - **static**: el elemento se posiciona según el flujo normal y no hace caso de las propiedades top, left, bottom y right.
 - **absolute**, se coloca en el sitio que se le dice al margen de otros elementos.
 - **fixed**, además de ser absoluta el elemento no se mueva del sitio al desplazarse en la página
 - **relative**, el elemento sigue el flujo normal y se coloca según los valores top, left, bottom y right.



Maquetación con bloques <DIV> y estilos.

Posicionamiento

- Veamos un ejemplo:

- HTML:

```
<div id="cabecera">
  <H1>Soy el DIV de la cabecera</H1>
</div>
<div id="lateral">
  <H1>Soy el DIV lateral</H1>
</div>
<div id="principal">
  <H1>Soy el DIV principal</H1>
</div>
<div id="pie">
  <H1>Soy el DIV del pie</H1>
</div>
```



Maquetación con bloques <DIV> y estilos.

Posicionamiento

- CSS:

```
body{
  margin: 20px;
}

#cabecera {
  position: fixed;
  width: 100%;
  height: 15%;
  top: 0;
  right: 0;
  bottom: auto;
  left: 0;
  background-color: green;
}

#lateral {
  position: fixed;
  width: 10em;
  height: auto;
  top: 15%;
  right: auto;
  bottom: 100px;
  left: 0;
  background-color: yellow;
}
```

```
#principal {
  position: fixed;
  width: auto;
  height: auto;
  top: 15%;
  right: 0;
  bottom: 100px;
  left: 10em;
  background-color: maroon;
}

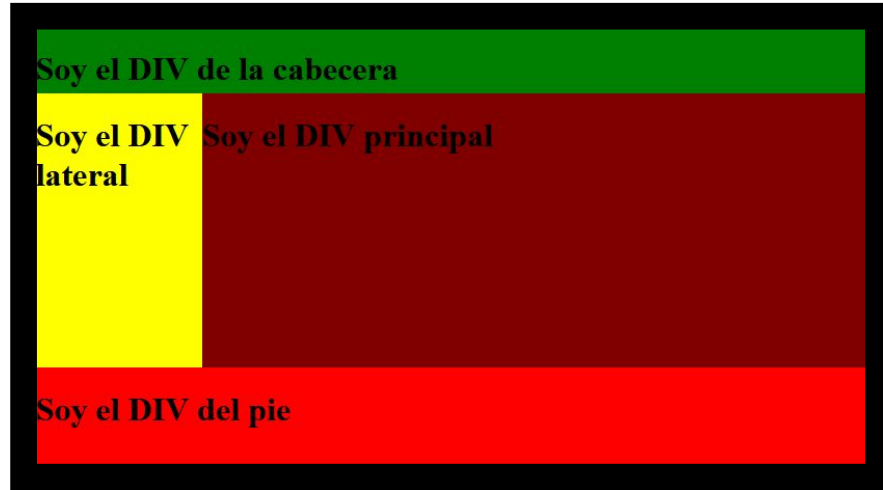
#pie {
  position: fixed;
  width: 100%;
  height: 100px;
  top: auto;
  right: 0;
  bottom: 0;
  left: 0;
  background-color: red;
}
```



Maquetación con bloques <DIV> y estilos.

Posicionamiento

- Resultado:





Actividad 5

Codifica el ejemplo anterior. Debes probar los cuatro tipos de posicionamiento que hay.

¿Te parece sencilla esta manera de posicionar los elementos?



Efectos CSS3

- La última versión del estándar CSS3 trae consigo un amplio conjunto de efectos que se pueden aplicar en el diseño de páginas
- Algunos ejemplos son:
 - Sombras: texto y cajas
 - Gradientes
 - Transiciones
 - Transformaciones
 - Animaciones
 - Formularios



Efectos CSS3. Sombras: texto y cajas

- En el texto con text-shadow y en otros elementos con box-shadow.

- La sintaxis sería:

```
text-shadow: 2px 3px 0px hsla (140, 3%, 36%, 0.5);
```

- Donde:

- 2px: es el tamaño de la sombra que se aplica a la derecha
- 3px: es la sombra que se proyecta hacia abajo
- 0px: sería la cantidad de sombra borrosa a aplicar
- hsla(140, 3%, 36%, 0.5) sería el color de sombra



Efectos CSS3. Gradientes

- Hasta la llegada de CSS3 la única forma de conseguir un efecto de colores degradados utilizar imágenes de fondo con un degradado ya hecho.
- Ahora podemos utilizar gradientes:

```
div{  
    color: #fff;  
    text-align:center;  
    padding:100px 0;  
    width:400px;  
    height:50px;  
    background: linear-gradient(top, #3cf, #078);  
    background: -webkit-linear-gradient(top, #3cf, #078);  
}
```



Efectos CSS3. Gradientes

- Donde:
 - El primer parámetro es un valor que indica en qué dirección (en grados) sale el gradiente
 - El siguiente valor indica el color de inicio del gradiente y su punto de inicio
 - El tercer valor especifica el color al que cambia el gradiente
 - El último gradiente es el color final del mismo



Efectos CSS3. Gradientes

- Por ejemplo:





Efectos CSS3. Transiciones

- Uno de los efectos más utilizados hasta ahora era la **pseudoclase: hover**, la cual permite aplicar diferentes estilos en dos estados diferentes.
- Las transiciones permiten aplicar estilos diferentes entre estados de una manera más amplia.



Efectos CSS3. Transiciones

- Veremos un ejemplo, con esta clase que se aplicará a un enlace:

```
.grow { font-size : 12pt; }

.grow:hover {
    transition : all 1s ease 0s;
    font-size: 26pt;
    text-shadow: 3px 3px 4px gray, -3px -3px 4px gray;
}
```



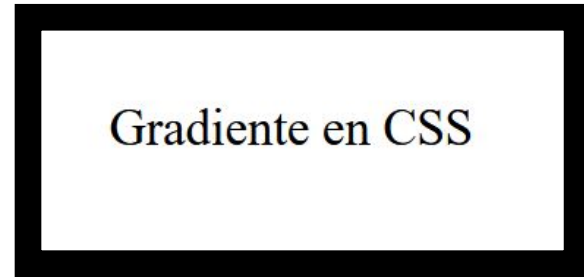

Efectos CSS3. Transiciones

- Donde:
 - **all**: se aplica a todos los estilos cambiantes.
 - **1s**: segundos que tarda la transición.
 - **ease**: modo en que se produce la transición.
 - **0s**: es un retardo que se puede aplicar para que se empiece a aplicar la transición.



Efectos CSS3. Transiciones

- Lo que se ve en el navegador a lo largo de un segundo es la transición de este enlace desde un tamaño de fuente 12pt a 26pt al que además se le añade sombra:





Efectos CSS3. Transiciones

- Son las propiedades:
 - **transition-property:** indica a qué atributos aplicamos la transición.
 - **transition-duration:** tiempo de duración de la transición.
 - **transition-timing-function:** la forma de ir cambiando con las opciones (ease, linear, ease-in, ease-out, ease-in—out, cubic—bezier)
 - **transition-delay:** puede ponerse un valor de retardo, incluso negativo, que la transición se aplique con un efecto de que ya ha progresado.



Actividad 6

Aplica al menos un tipo de transición a la página de la actividad 4



Efectos CSS3. Transformaciones

- Las transformaciones que existen son:
 - **Scale**: cambia de tamaño los elementos.
 - **Translate**: cambia de posición a izquierda, derecha, arriba o abajo.
 - **Rotate**: gira o hace rotar elementos en determinados grados.
 - **Skew**: distorsiona los elementos.
 - **Matrix**: permite mover y transformar con precisión de píxel.
 - A continuación, veremos ejemplos de cada uno

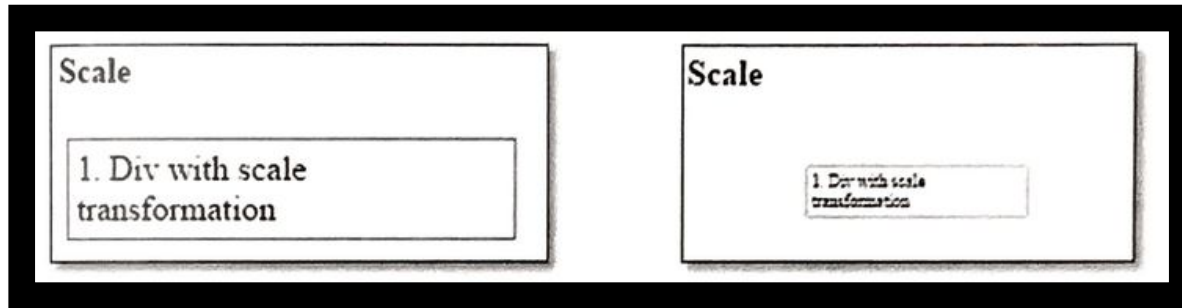


Efectos CSS3. Transformaciones. Scale

- **Scale:**

```
.scale:hover { transform: scale(0.5); }
```

- **Resultado:**



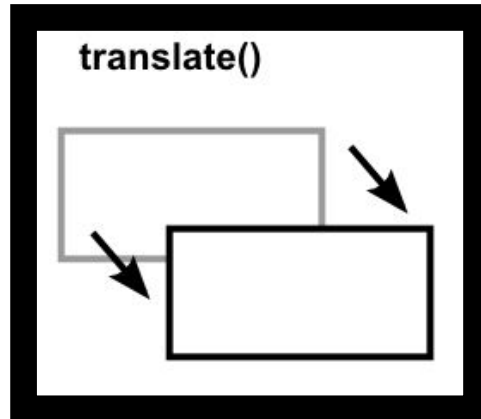


Efectos CSS3. Transformaciones. Translate

- **Translate:**

```
.translate:hover { transform:translate(10px,20px); }
```

- **Resultado:**



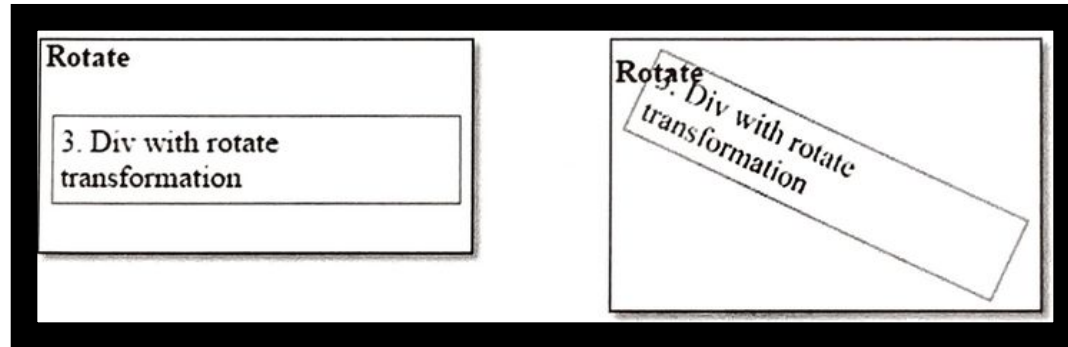


Efectos CSS3. Transformaciones. Rotate

- **Rotate:**

```
.rotate:hover { transform: rotate(25deg); }
```

- **Resultado:**



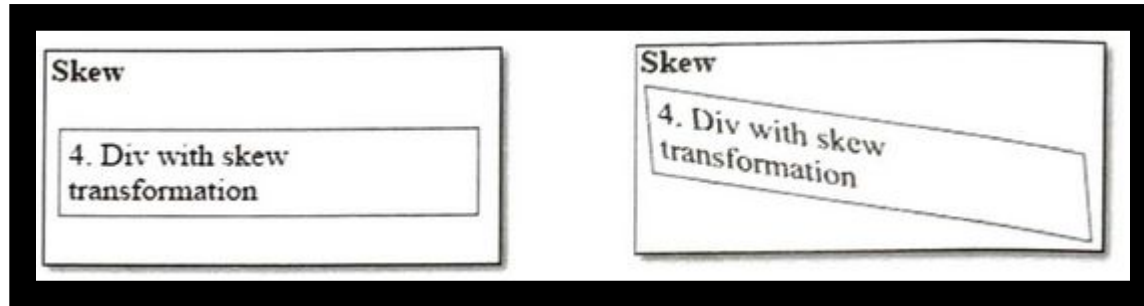


Efectos CSS3. Transformaciones. Skew

- **Skew:**

```
.skew:hover { transform: skew (5deg, 10deg); }
```

- **Resultado:**



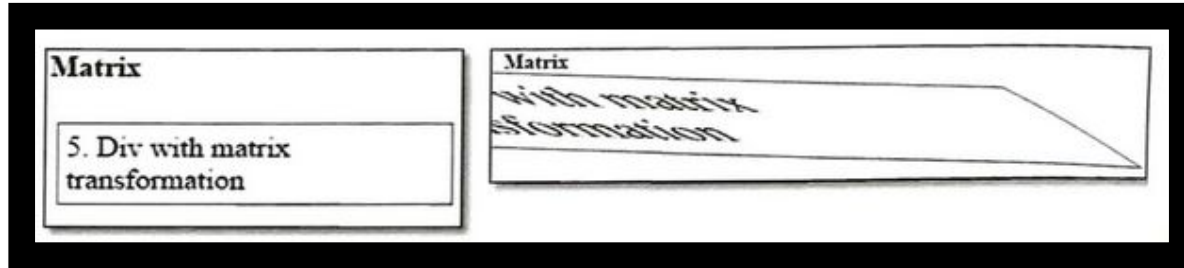


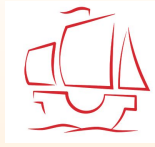
Efectos CSS3. Transformaciones. Matrix

- **Matrix:**

```
.matrix:hover { transform: matrix(1,2,3,4,5,6); }
```

- **Resultado:**





Actividad 7

*Utiliza alguna de las transformaciones que hemos visto previamente en la página
de la actividad anterior*



Efectos CSS3. Animaciones

- Las animaciones que introduce CSS3 se basan en el sistema de **keyframes**.

En ese tipo de herramientas se contaba con una línea temporal donde se podían ir aplicando distintas transformaciones a través de los distintos fotogramas

- Todo eso lo podemos hacer ahora con HTML5



Efectos CSS3. Animaciones

- Se debe definir un elemento **@keyframes**:
- Aquí se define la animación en cada porcentaje

```
h1 {  
    animation:yellowgrow 5s infinite;  
}  
  
@keyframes yellowgrow {  
    from {  
        font-size: 10pt;  
        color: black;  
    }  
    25% {  
        font-size: 15pt;  
        color: yellow;  
    }  
    50% {  
        font-size: 10pt;  
        color: green;  
    }  
    75% {  
        font-size: 15pt;  
        color: blue;  
    }  
    to {  
        font-size: 10pt;  
        color: black;  
    }  
}
```



Efectos CSS3. Animaciones

- El resultado es:





Efectos CSS3. Animaciones

- La animación se ha aplicado con todas las opciones con la propiedad única de animation.
- Sin embargo, se puede desglosar en otras propiedades concretas:
 - **animation-name:** nombre de la animación definida previamente.
 - **animation-duration:** tiempo de animación en segundos, permitiendo fracciones.



Efectos CSS3. Animaciones

- Sin embargo, se puede desglosar en otras propiedades concretas:
 - **animation-timing-function**: función de animación
 - **animation-iteration-count**: número de iteraciones de animación, pudiendo ser infinito
 - **animation-play-state**: estado de la animación
 - **animation-delay**: retardo para el inicio de la animación en segundos.
 - **animation-fill-mode**: modo de relleno.



Actividad 8

Crea una animación donde varíes el color de alguno de los elementos de la página utilizando el sistema de @keyframes



Formularios

- A continuación veremos cómo mezclar HTML5 con CSS3.
- En un formulario, por ejemplo, nos puede valer para indicar si los datos introducidos son correctos o no.
- Veremos un ejemplo:



Formularios

- HTML

```
<form name="Formulario" method="post" class="bordeazul" id="Formulario">
  <fieldset>
    <legend>
      Registrarse:
    </legend>
    <div>
      <label for="email">Email</label>
      <input id="email" name="email" placeholder="Introduzca su email" type="email">
    </div>
    <div>
      <label for="password">Password</label>
      <input id="password" name="password" placeholder="Introduzca su password"
        type="password" required pattern="(0-9A-Za-z){8,}">
    </div>
    <div>
      <label for="description">Descripción</label>
      <textarea id="description" name="description" placeholder="Introduzca su descripción"></textarea>
    </div>
    <div>
      <input type="submit" class="submit">
    </div>
  </fieldset>
</form>
```

Formularios

- CSS

```
html {
    font-family: Arial, Helvetica, sans-serif;
}
legend{
    font-size: 2em;
    margin-bottom: 20px;
    float: left;
    width: 100%;
}
fieldset{
    width: 80%;
    padding: 3%;
    border: 1px solid #aaa;
    background-color: #c2c2c2;
    margin-bottom: 20px;
    box-shadow: 2px 2px 5px gray;
    border-radius: 3px;
}
label{
    width: 40%;
    float: left;
}
input[type=submit]{
    margin-bottom: 10px;
    display: block;
    width: 50%;
}
input, textarea {
    height: 20px;
    width: 50%;
    float: right;
    border: 1px solid #bfbfbf;
    padding: 0.2em;
    border-radius: 3px;
}
form div{
    clear: both;
}
```



Formularios



- **Resultado:**

Registrarse:

Email	<input type="text" value="Introduzca su email"/>
Password	<input type="password" value="Introduzca su password"/>
Descripción	<input type="text" value="Introduzca su descripción"/>
<input type="button" value="Enviar consulta"/>	



Actividad 9

Implementa un formulario que solicite:

- *Nombre*
- *Apellidos*
- *DNI*
- *Email*
- *Teléfono*

Asegúrate de que si el campo está en blanco o es incorrecto, aparezca una señal de error



Media queries

- **Mecanismo introducido en CSS3 que permiten aplicar estilos según el tipo de medio en el que se muestran los documentos**
- Se utiliza el selector **@media** y, aplicando distintas condiciones como el ancho del medio, el soporte para colores, etc., se pueden variar los estilos que se utilizan en el documento.



Media queries

- Veamos un ejemplo, donde necesitamos aplicar un estilo para pantallas con un ancho máximo de 800 px, en la que se hace desaparecer un menú lateral:

```
@media (max-width: 800px){  
  sidebar {display: none;}  
  .estilo1{  
    border:4px;font-size:15px;  
  }  
}
```




Media queries

- El navegador detecta si la condición es verdadera, y si es así aplica el estilo que contenga. También puede aplicar distintos estilos según el tipo de medio genérico, como pantallas, impresión, televisión, etc.



Media queries. Tipos de medios

- Los tipos de medios reconocidos son:

VALOR	DESCRIPCIÓN
All	Todos los tipos de medios
print	Impresoras
screen	Pantallas de ordenadores, tablets, etc.
speech	Para lectores de pantallas



Media queries. Condiciones

- Algunas de las condiciones que se pueden aplicar son:

FORMATO	DESCRIPCIÓN
Color	Indica el número de bits por componente de color del dispositivo de salida.
color-index	Indica el número de entradas en la tabla de colores para el dispositivo de salida.
device-aspect-ratio	Ratio entre anchura y altura del dispositivo de salida
device-height	Altura del dispositivo de salida
device-width	Anchura del dispositivo de salida
grid	Si el dispositivo es grid o bitmap



Media queries. Operadores lógicos

- Las media queries permiten el uso de expresiones similares a las utilizadas para crear expresiones condicionales complejas:
 - **Operadores AND**
 - Aplica los estilos si todas las condiciones son verdaderas. Por ejemplo:

```
@media (max-width: 800px) and (orientation: portrait) {}
```



Media queries. Operadores lógicos

- Las media queries permiten el uso de expresiones similares a las utilizadas para crear expresiones condicionales complejas:
 - **Operadores OR**
 - No existe un operador explícito, pero cuando se necesita aplicar estilos a un medio o a otro simplemente se ponen selectores separados por comas.

```
@media (max-width: 800px), handheld and (orientation: portrait) {}
```



Media queries. Operadores lógicos

- Las media queries permiten el uso de expresiones similares a las utilizadas para crear expresiones condicionales complejas:
 - **Operadores NOT**
 - Sirve para aplicar estilos precisamente a medios que no cumplan las propiedades específicas.

```
@media not screen and (color) {}
```



Actividad 10

Con el formulario de la actividad anterior, asegúrate que:

- *Si se usa en un ordenador, el fondo sea amarillo*
- *Si se usa en un móvil, el color de las letras sea azul*
- *Si el tamaño es mayor a 1080px, el fondo debe ser rojo Y las letras negras*

Configura las media queries que hagan falta

Tema 2: Introducción al CSS



Ciclo Superior DAW

Asignatura: Diseño de interfaces

Curso 20/21