

UT2 – GUÍAS DE ESTILO

Patricia Fernández Sánchez

Diseño de Interfaces Web

DAW

2020-2021

Índice

Ejercicio 1.....3

 HTML5.....3

 CSS3.....4

Ejercicio 2.....7

 HTML.....7

 CSS.....9

Ejercicio 1

Para este ejercicio se han creado dos hojas independientes pero enlazadas: una Html5 y otra CSS3

HTML5

Al ser un documento en Html5 la estructura del documento se definirá principalmente por los siguientes campos semánticos, todo ello dentro del cuerpo del documento:

<html> Etiqueta para definir que es una estructura en el lenguaje Html

<head> Etiqueta que no se visible, que contiene el titulo de la página, en la pestaña, donde se enlaza las paginas CSS, el favicon y otros metadatos. Aquí enlazaremos nuestra hoja de CSS

```
<link rel="stylesheet" href="tabla.css">
```

```
</head>
```

<body> Etiqueta que contiene el contenido web y se muestra en el navegador

<header> Define el cabecero de la web.**</header>**

<article> Define el contenido de la web.**</article>**

<footer> Define el pie de página de la web.**</footer>**

```
</body>
```

```
</html>
```

Dentro de cada apartado declaramos las etiquetas correspondientes a lo necesario, las cuales iremos describiendo, según el campo en el que están situadas.

Dentro de **<header>** definiremos:

<h1> Título principal de la web**</h1>**

Dentro del **<article>** definiremos

<table> Es la etiqueta para definir y comenzar a crear la tabla

<tr> Es la etiqueta de las filas de la tabla

<th> Es la etiqueta de los encabezados de la tabla, se definen directamente en negrita**</th>**

<td> Es la etiqueta de las columnas/celdas, se definen habitualmente dentro de la fila**</td>**

```
</tr>
```

```
</table>
```

Dentro de la etiqueta **<footer>** definiremos

<address> Es la etiqueta definida para el autor que ya viene definido el estilo de la fuente en cursiva **</address>**

CSS3

Una vez creado el documento Html5, procedemos a crear la hoja de estilo que le proporcionará el estilo que se busca. Para ello es necesario usar los **selectores** adecuados.

Hemos definido los siguientes **selectores** y sus correspondientes **atributos**:

body (selector de la etiqueta body) {

widows: atributo para el tamaño del documento, en este caso queremos el total: 100%;

}

header, h1 (selector de la cabecera y del título principal, al ponerlos juntos con una coma, se encadenan {

text-align: atributo para alinear el texto, en este caso queremos que sea centrado: center;

font-family: atributo para la fuentes del texto, en este caso modificamos la predeterminada:

Arial, Helvetica, sans-serif; (La fuente de texto por defecto de html es Times New Roman)


color: atributo para el color de la letra del texto, en este caso queremos #333333; 

text-transform: atributo para modificar el texto, en este caso buscamos todas las primeras letras de cada palabra en mayúsculas: capitalize;

text-indent: atributo de la sangría de la primera línea, en este caso la hemos usado ya que el texto centrado no estaba a la altura de la caja y así poder alinear el principio del texto con el borde de la misma: 50px;

}

h1::first-letter (declaramos un *pseudo-elemento* para seleccionar en concreto el elemento que queremos darle un diferente estilo, en este caso la primera letra del texto del título h1) {

color: atributo para el color de la letra del texto, en este caso queremos: red; 

font-style: atributo para el estilo de la fuente del texto, en este caso cursiva: italic;

font-size: atributo para el tamaño de la fuente del texto, queremos que sea más grande: 2em;

}


article (selector del cuerpo del documento){

margin: atributo para darle margen, al poner auto lo reemplaza por un valor adecuado, en este caso lo usamos para centrar horizontalmente el contenido: auto;

display: atributo de propiedad de visualización, en este caso queremos que todo lo que se encuentre en esta etiqueta actúe como una tabla: table;

(Al definir aquí el atributo display, es innecesario crear un selector table, ya que no cambia nada al volver a añadir el atributo y es imprescindible declararlo en este selector)


padding: atributo para darle relleno entre en contenido y su borde, cuyo valor nos han proporcionado, al poner un único valor se le aplica a todos los lados por igual: 7em;

border: atributo para crear un borde al contenido, en este caso queremos un borde sólido, visible, de color negro y de grosor 1px: 1px solid black; 

font-family: tributo para la fuentes del texto, en este caso modificamos la predeterminada: Arial, Helvetica, sans-serif;

}


th (selector de los encabezados de la tabla){

border: atributo para crear un borde al contenido, en este caso queremos un borde sólido, visible, de color negro y de grosor 1px: 1px solid black; 

text-align: atributo para alinear el texto, en este caso queremos que sea centrado: center;

background-color: atributo para proporcionar color al fondo, en este caso queremos: #F0F0F0; 

padding: atributo para darle relleno entre en contenido y su borde, en este caso se ha aplicado para que la celda no estuviera tan cerca del texto: 7px;

color: atributo para el color de la letra del texto, en este caso queremos: #333333; 

}

td (selector de las columnas/celdas de la tabla){

border: atributo para crear un borde al contenido, en este caso queremos un borde sólido, visible, de color negro y de grosor 1px: 1px solid black;1px solid black;

text-align: atributo para alinear el texto, en este caso queremos que sea centrado: center;

}

tr:nth-child(odd) (declaramos una *pseudoclase* de las filas impares para crear el estilo demandado)
{

background-color: atributo para proporcionar color al fondo, en este caso queremos: #eeeeee; 

}

tr:nth-child(odd):hover (declaramos una *pseudoclase* de las filas impares para crear el estilo demandado al pasar el curso por encima de ella){

background-color: atributo para proporcionar color al fondo, en este caso queremos: #00FF00; 

}

tr:nth-child(even) (declaramos una *pseudoclase* de las filas pares para crear el estilo demandado){

background-color: atributo para proporcionar color al fondo, en este caso queremos: #FFFF00;

}

tr:nth-child(even):hover (declaramos una *pseudoclase* de las filas pares para crear el estilo demandado al pasar el curso por encima de ella){

background-color: atributo para proporcionar color al fondo, en este caso queremos: red;

}

footer > address (declaramos el selector del elemento hijo *address* cuyo elemento padre es *footer*, al cual le daremos las siguientes características) {

color: atributo para el color de la letra del texto, en este caso queremos: #FF00FF;

text-align: atributo para alinear el texto, en este caso queremos que sea centrado: center;

padding: atributo para darle relleno entre el contenido y su borde, en este caso para que el texto no esté pegado al borde del *article*: 10px;

margin: atributo para darle margen, al poner auto lo reemplaza por un valor adecuado, en este caso lo usamos para centrar horizontalmente el contenido: auto;

text-indent: atributo de la sangría de la primera línea, en este caso la hemos usado ya que el texto centrado no estaba a la altura de la caja y así poder alinear el principio del texto con el borde de la misma: -390px;

}

127.0.0.1:5500/EjerciciosDAW/DIW/tablas.html

Tabla De Ciclos Formativos CFP La Laboral

Ciclo	Nivel	Alumnado	Turno	Cursos
Informática	CFGS	125	DyV	4
Comercio	CFGM	85	D	3
Fabricación	CFGS	68	D	2
Madera	CFGS	120	V	2

José A. Sánchez - CFP La Laboral

127.0.0.1:5500/EjerciciosDAW/DIW/tablas.html

Tabla De Ciclos Formativos CFP La Laboral

Ciclo	Nivel	Alumnado	Turno	Cursos
Informática	CFGS	125	DyV	4
Comercio	CFGM	85	D	3
Fabricación	CFGS	68	D	2
Madera	CFGS	120	V	2

José A. Sánchez - CFP La Laboral

Ejercicio 2

Para este ejercicio se han creado dos hojas independientes pero enlazadas: una Html y otra CSS

HTML

En este caso, al ser un ejercicio basado en **Grid Layout** usaremos **HTML4**, es decir con esta estructura semántica basada en bloques **<div></div>**, que para distinguir unos de otros se coloca **class** o **id**.

<html> Etiqueta para definir que es una estructura en el lenguaje Html

<head> Etiqueta que no se visible, que contiene el titulo de la página, en la pestaña, donde se enlaza las paginas CSS, el favicon y otros metadatos. Aquí enlazaremos nuestra hoja de CSS

```
<link rel="stylesheet" href="grid.css">
```

```
</head>
```

<body> Etiqueta que contiene el contenido web y se muestra en el navegador

```
<div id="header">Define el cabecero de la web.</div id="header">
```

```
<div class="parent">Bloque padre y comienzo del Grid, en este caso lo dividiremos en 5 bloques de diferentes tamaños</div>
```

```
<div class="div1"> Bloque 1</div>
```

```
<div class="div2"> Bloque 2 </div>
```

```
<div class="div3"> Bloque 3 </div>
```

```
<div class="div4"> Bloque 4 </div>
```

```
<div class="div5"> Bloque 5 </div>
```

```
<div id="footer">Define el pie de página de la web.</div>
```

```
</body>
```

```
</html>
```

Para definir el div les he puesto un identificador **id** en los que serán el cabecero y pie de página ya que serán valores únicos que no se van a repetir.

Definiremos dentro del **<div id="header">**

```
<h1> Título principal de la web</h1>
```

Definiremos dentro del **<div class="div1">**

```
<table>Es la etiqueta para definir y comenzar a crear la tabla
```

<tr>Es la etiqueta de las filas de la tabla

<td>Es la etiqueta de las columnas/celdas, se definen habitualmente dentro de la fila**</td>**

</tr>

</table>

Definiremos dentro del **<div class="div2">**

<h3> Título inferior al h1 y al h2 **</h3>**

<p> Etiqueta párrafo de texto **</p>**

<p> Etiqueta párrafo de texto **</p>**

<p> Etiqueta párrafo de texto **</p>**

<p> Etiqueta párrafo de texto **</p>**

Definiremos dentro del **<div class="div3">**

<h3> Título inferior al h1 y al h2 **</h3>**

<p> Etiqueta párrafo de texto **</p>**

<p> Etiqueta párrafo de texto **</p>**

<p> Etiqueta párrafo de texto **</p>**

<p> Etiqueta párrafo de texto **</p>**

Definiremos dentro del **<div class="div4">**

<h3> Título inferior al h1 y al h2 **</h3>**

<h5> Título inferior al h1, al h2, al h3 y al h4 **</h5>**

<p class="redes"> Etiqueta párrafo de texto con clase *redes* ya que se modificaran de manera independiente **</p>**

<p class="autor"> Etiqueta párrafo de texto con clase *autor* ya que se modificaran de manera independiente **</p>**

**
** Etiqueta de salto de línea

<h5> Título inferior al h1, al h2, al h3 y al h4 **</h5>**

<p class="redes"> Etiqueta párrafo de texto con clase *redes* ya que se modificaran de manera independiente **</p>**

<p class="autor"> Etiqueta párrafo de texto con clase *autor* ya que se modificaran de manera independiente **</p>**

Definiremos dentro del **<div class="div5">**

<h3> Título inferior al h1 y al h2 **</h3>**

<p> Etiqueta párrafo de texto **</p>**

<p> Etiqueta párrafo de texto **</p>**

Definiremos dentro del **<div id="footer">**


<p class="autor"> Etiqueta párrafo de texto con clase *autor* ya que tendrá las mismas características que el del *.div4* **</p>**

Se crearon varias etiquetas **<p>** en diferentes **<div>** debido a que esta etiqueta no te respeta los saltos de párrafo, de esta manera al hacer varios independientes crea diferentes bloques consiguiendo ese efecto.

Para poder modificar posteriormente en el **CSS** se les aplicó a **<p>** algunas **class** para poder distinguirlos.

CSS

body (selector de la etiqueta body) {

background-color: atributo para proporcionar color al fondo, en este caso queremos: #abcdef; 

widows: atributo para el tamaño del documento, en este caso queremos el total: 100%;

}

#header (selector para el elemento que contiene id="header"){

font-style: atributo para el estilo de la fuente del texto, en este caso cursiva: italic;


text-align: atributo para alinear el texto, en este caso queremos que sea centrado: center;

}

h1::before (declaramos un *pseudo-elemento* para insertar contenido antes, en este caso del h1) {

content: atributo que se utiliza en este selector selector para añadir contenido: "Módulo:";

font-style: atributo para el estilo de la fuente del texto, en este caso cursiva: italic;


color: atributo para el color de la letra del texto, en este caso queremos: red; 

}

h1::after(declaramos un *pseudo-elemento* para insertar contenido después, en este caso del h1) {

content: atributo que se utiliza en este selector selector para añadir contenido: "CSS Grid Layout";

font-style: atributo para el estilo de la fuente del texto, en este caso cursiva: italic;

color: atributo para el color de la letra del texto, en este caso queremos: red; 

}

h3::first-letter (declaramos un *pseudo-elemento* para seleccionar en concreto el elemento que queremos darle un diferente estilo, en este caso la primera letra del texto del título h3) {

color: atributo para el color de la letra del texto, en este caso queremos: red; 

font-style: atributo para el estilo de la fuente del texto, en este caso cursiva: italic;

font-size: atributo para el tamaño de la fuente del texto, queremos que sea más grande: 2em;

}

p (selector de todos párrafos de texto del documento) {

text-align: atributo para alinear el texto, en este caso queremos que sea justificado: justify;

}

h3 (selector de todos títulos h3 del documento){

text-transform: atributo para modificar el texto, en este caso buscamos todas las primeras letras de cada palabra en mayúsculas: capitalize;

}

h5 (selector de todos títulos h5 del documento){

text-transform: atributo para modificar el texto, en este caso buscamos todas todas las letras en mayúsculas: uppercase;

}

p.redes (selector de los párrafos de texto con clase “redes” del documento){

text-align: atributo para alinear el texto, en este caso queremos que sea a la derecha: right;

display: atributo de propiedad de visualización, en este caso queremos que todo lo que se encuentre en esta etiqueta actúe como un bloque en conjunto; block;


font-style: atributo para el estilo de la fuente del texto, en este caso cursiva: italic;

white-space: atributo que maneja el espacio en blanco dentro de un elemento, aquí mantendremos los saltos de línea y reduciremos el espacios en blanco sobrantes sin que el texto salga del contenedor: pre-line;

padding: atributo para darle relleno entre en contenido y su borde, en este caso no queremos que esté muy pegado a los bordes del grid y al ser un valor único se aplica a todo los lados: 10px

}

p.autor (selector de los párrafos de texto con clase “autor” del documento){

border-top: atributo para crear un único borde superior al contenido, en este caso queremos un borde punteado, visible, de un color específico y de grosor 1px: 1px dotted #bc548c; 

text-transform: atributo para modificar el texto, en este caso buscamos todas las primeras letras de cada palabra en mayúsculas: capitalize;

text-align: atributo para alinear el texto, en este caso queremos que sea centrado: center;

color: atributo para el color de la letra del texto, en este caso queremos: #e07262; 

font-style: atributo para el estilo de la fuente del texto, en este caso cursiva: italic;

}

.div1 > table (declaramos el selector del elemento hijo *table* cuyo elemento padre es *.div1*, al cual le daremos las siguientes características){

display: atributo de propiedad de visualización, en este caso queremos que todo lo que se encuentre en esta etiqueta actúe como una tabla: table;

margin: atributo para darle margen, al poner auto lo reemplaza por un valor adecuado, en este caso lo usamos para centrar horizontalmente el contenido: auto;


text-align: atributo para alinear el texto, en este caso queremos que sea centrado: center;

height: propiedad que establece la altura, en este caso de la altura de la tabla: 10px;

border-spacing: propiedad para establecer distancia entre las celdas: 10px;


}

td (selector de las columnas/celdas de la tabla){

border-style: atributo para crear un borde al contenido, en este caso queremos un borde sólido, visible, de un color específico y de grosor 1px: 1px solid #e0e0e0; 

text-align: atributo para alinear el texto, en este caso queremos que sea centrado: center;

align-content: atributo para alinear el contenido, en este caso centrado: center;

background-color: atributo para proporcionar color al fondo, en este caso queremos: #e0e0e0; 

margin: atributo para darle margen, al poner auto lo reemplaza por un valor adecuado, en este caso lo usamos para centrar horizontalmente el contenido: auto;


padding-left: atributo para darle relleno entre en contenido y su borde, en este caso unicamente el lado izquierdo: 5px;


padding-right: atributo para darle relleno entre en contenido y su borde, en este caso unicamente el lado derecho: 5px;

color: atributo para el color de la letra del texto, en este caso queremos: #333333; 

}

td:hover (declaramos una pseudoclase de las columnas/celdas para crear el estilo demandado al pasar el curso por encima de cada una de ellas){

background-color: atributo para proporcionar color al fondo, en este caso queremos: #333333; 

color: atributo para el color de la letra del texto, en este caso queremos: #eeeeee; 

border-color: atributo para proporcionar color al borde, en este caso queremos: #333333; 

}

.parent (selector a la que pertenece la clase *.parent*) {

display: atributo de propiedad de visualización, en este caso queremos que todo lo que se encuentre en esta etiqueta actúe como un grid: grid;

grid-template-columns: propiedad que determina el número de las columnas y el ancho del grid repeat(5, 1fr);

grid-template-rows: propiedad que determina el número de las filas y la altura del grid: repeat(1, 1fr);


grid-column-gap: espacio entre columnas de la grid: 20px;


grid-row-gap: espacio entre filas de la grid: 5px;

}

.div1 (selector a la que pertenece la clase *.div1*) {

grid-area: 1 / 1 / 1 / 8;

border-bottom: atributo para crear un único borde inferior al contenido, en este caso queremos un borde punteado, visible, de un color específico y de grosor 1px: 1px dotted #bc548c; 

border-top: atributo para crear un único borde superior al contenido, en este caso queremos un borde punteado, visible, de un color específico y de grosor 1px: 1px dotted #bc548c; 

text-align: atributo para alinear el texto, en este caso queremos que sea centrado: center;

padding: atributo para darle relleno entre en contenido y su borde, en este caso para crear más espaciado entre la tabla y el borde, al ser valor único se aplica a todos los lados: 25px;

}

.div2 (selector a la que pertenece la clase *.div2*){

grid-area: atributo que define el área del grid (grid donde empieza la fila/grid donde empieza la columna/grid donde acaba la fila/grid donde acaba la columna): 2 / 1 / 2 / 3;

margin-left: atributo para darle margen, en este caso unicamente al lado izquierdo: 7px;

}

.div3 (selector a la que pertenece la clase `.div3`) {

grid-area: atributo que define el área del grid (grid donde empieza la fila/grid donde empieza la columna/grid donde acaba la fila/grid donde acaba la columna): 2 / 3 / 2 / 5;

}

.div4 (selector a la que pertenece la clase `.div4`){

grid-area: atributo que define el área del grid (grid donde empieza la fila/grid donde empieza la columna/grid donde acaba la fila/grid donde acaba la columna): 2 / 5 / 2 / 8;

}

.div5 (selector a la que pertenece la clase `.div5`){

grid-area: atributo que define el área del grid (grid donde empieza la fila/grid donde empieza la columna/grid donde acaba la fila/grid donde acaba la columna): 5 / 1 / 3 / 8;

}

