

UT3_1.- LENGUAJES DE MARCAS EN ENTORNO WEB. CSS

BÁSICO

RESULTADOS DE APRENDIZAJE ASOCIADOS
<ol style="list-style-type: none">1. Reconoce las características de lenguajes de marcas analizando e interpretando fragmentos de código.2. Utiliza lenguajes de marcas para la transmisión de información a través de la Web analizando la estructura de los documentos e identificando sus elementos3. Establece mecanismos de validación para documentos XML utilizando métodos para definir su sintaxis y estructura.
CRITERIOS DE EVALUACIÓN
<ul style="list-style-type: none">- De RA1 – desde CEA hasta CEK- De RA2 – desde CEA hasta CEH- De RA3 – desde CEA hasta CEH

UT3_1.- LENGUAJES DE MARCAS EN ENTORNO WEB.

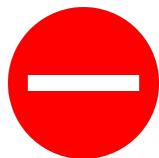
CSS BÁSICO

Índice de contenido

1.- Introducción.....	3
2.- Integración de estilos CSS.....	5
2.1.- CSS en línea.....	5
2.2.- CSS dentro del documento HTML.....	5
2.3.- CSS en un documento externo.....	6
3.- Sintaxis de las reglas CSS.....	9

1.- Introducción

Una forma de asignar estilos a nuestro documento es utilizar estilos HTML.



Esta forma de asignar estilos está obsoleta y no se debe utilizar nunca.

Ejemplo de **código obsoleto**:

```
<h1>
  <font color="red" face="Arial" size="5">Titular de la página</font>
</h1>
<p>
  <font color="blue" face="Verdana" size="2">
    Párrafo con texto.
  </font>
</p>
```

En el ejemplo podemos ver como se utiliza la etiqueta con sus atributos color, face y size para definir el color, el tipo y el tamaño de letra de cada elemento de la página.



El problema al utilizar estilos HTML es que cada vez que queramos aplicar el estilo tenemos que repetirlo y esto hace que nuestro código crezca innecesariamente debido a la repetición de código. El mantenimiento es mucho más costoso, ya que la modificación o eliminación de un valor repetido por toda nuestra página puede ser una labor muy pesada.

La solución que propone CSS es mejor, como se puede ver en el siguiente ejemplo:

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Ejemplo estilos</title>
  <style type="text/css">
    h1{
      color: red;
      font-family: Arial;
      font-size: large;
    }

    p{
      color: blue;
      font-family: Verdana;
      font-size: medium;
    }
  </style>
</head>
<body>
  <h1>Titular de la página</h1>
  <p>Párrafo con texto</p>
</body>
```

- CSS nos permite **diferenciar entre los estilos (el formato) de la página y la estructura de la página (el código HTML)** sin mezclar unos con otros.
- Utilizando CSS, se pueden establecer los mismos estilos con menos esfuerzo y sin *ensuciar* el código HTML de los contenidos con etiquetas .



En el ejemplo anterior, dentro de la etiqueta **<style>** se indica que todas las etiquetas <h1> de la página se deben ver de color rojo, con un tipo de letra Arial y con un tamaño de letra grande. Además, las etiquetas <p> de la página se deben ver de color azul, con un tipo de letra Verdana y con un tamaño de letra medio.

Definir los estilos de esta forma:

- Facilita la escritura de los estilos.

- Evita repeticiones.
- Nos permite diferenciar el estilo de la estructura de la página
- Mejora el mantenimiento.

2.- Integración de estilos CSS

Tenemos tres formas de integrar el código CSS en nuestro código HTML:

- ❖ En línea.
- ❖ Dentro del documento HTML.
- ❖ Documento externo.

2.1.- CSS en línea

Este método añade el código CSS en la etiqueta HTML, y como podemos imaginarnos, tiene los mismos problemas que nos planteaba la utilización de los atributos para estilos de HTML.



Este método no es recomendable utilizarlo.

```
<body>  
  <p style="color: yellow; font-family: Verdana">Párrafo con texto</p>  
</body>
```

2.2.- CSS dentro del documento HTML

Utilizamos la etiqueta **<style>** dentro del **<head>** de nuestro documento HTML. De esta forma los estilos se aplican a todo el documento en el que se encuentran.

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Ejemplo etiqueta style</title>
  <style>
    p{
      color: yellow;
      font-family: Verdana;
    }
  </style>
</head>
<body>
  <p>Párrafo con texto.</p>
</body>
```

Este método se emplea cuando se define un número pequeño de estilos o cuando se quieren incluir estilos específicos en una determinada página HTML que completen los estilos que se incluyen por defecto en todas las páginas del sitio web.



El principal inconveniente es que, si se quiere hacer una modificación en los estilos definidos, es necesario modificar todas las páginas que incluyen el estilo que se va a modificar.

2.3.- CSS en un documento externo

En este caso, todos los estilos CSS se incluyen en un archivo externo **.CSS**, que las páginas HTML enlazan mediante la etiqueta **<link>**.



Podemos crear todos los archivos CSS que sean necesarios y cada página HTML puede enlazar tantos archivos CSS como necesite.

Para poder utilizar los estilos en un documento **.CSS** externo, vamos a tener que:

1. Crear un fichero **.CSS** con los estilos que necesitamos.

- Guardaremos el fichero **.CSS**. Normalmente vamos a crear una carpeta (que se suele llamar **styles** o **css**).
- Y finalmente enlazaremos nuestro documento **.CSS** dentro de nuestro documento HTML con la etiqueta **<link>**, que añadiremos dentro del **<head>** y en el que utilizando el atributo *href* indicaremos la ruta donde se encuentra el fichero .css. De esta forma podremos acceder a todos los estilos que tengamos dentro del fichero .css

Ejemplo:

- Contenido de *index.html*:

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Ejemplo css externo</title>
  <link rel="stylesheet" href="css/estilosEjemplo5.css">
</head>
<body>
  <p>Párrafo con texto.</p>
</body>
```

- Contenido de *estilos.css* (archivo contenido en la carpeta llamada css):

```
p{
  color: #999;
  font-family: Verdana;
}
```

De todas las formas de incluir CSS en las páginas HTML, **esta es la más utilizada con mucha diferencia.**



La principal ventaja es que se puede incluir un mismo archivo CSS en multitud de páginas HTML, por lo que se garantiza la aplicación homogénea de los mismos estilos a todas las páginas que forman un sitio web.

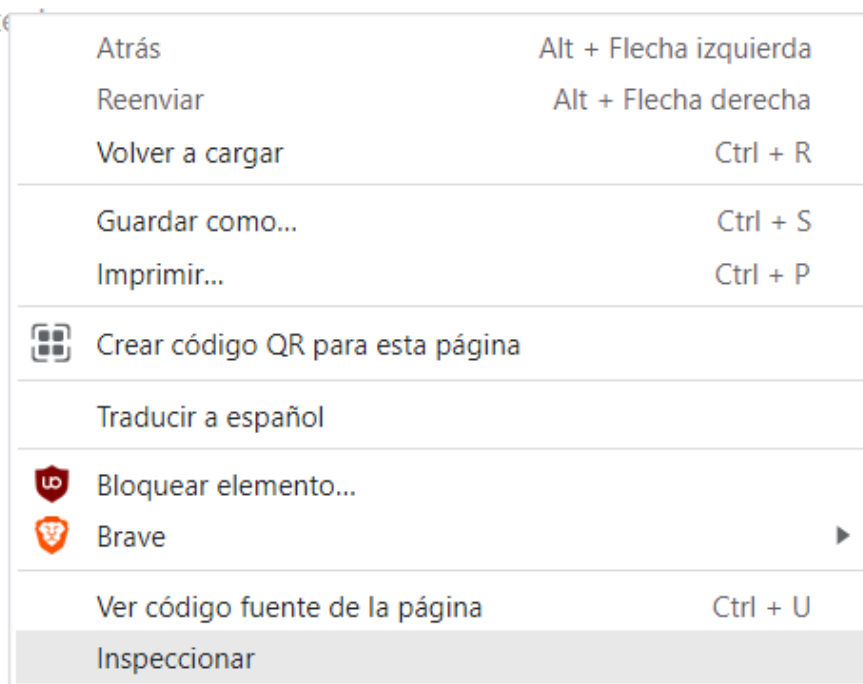
Este método es el más utilizado, ya que:

- Nos permite **reutilizar** el código CSS en todos los documentos HTML que necesitemos.
- El **mantenimiento es mucho más sencillo**, ya que un cambio en un archivo CSS permite actualizar los estilos de los documentos HTML que lo contengan.



Si inspeccionamos código en el navegador:

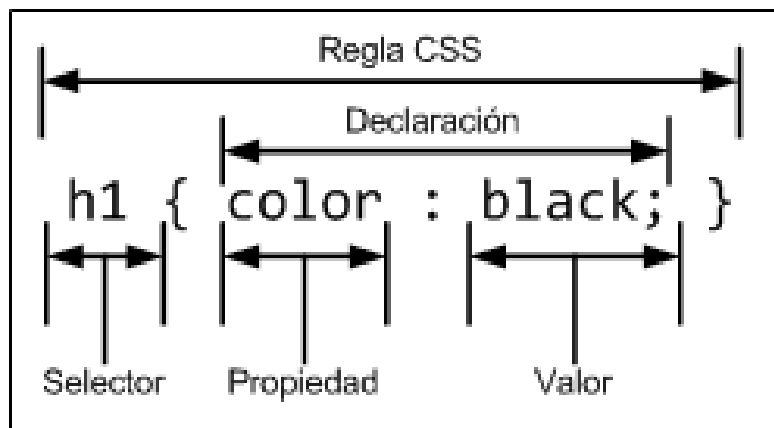
Párrafo con te...



Aquí podremos probar estilos y veremos que se actualizarán los cambios en tiempo real en nuestra página. Si actualizamos la página (pulsando la tecla F-5, por ejemplo), todo volverá a mostrarse según el código que hayamos implementado.

3.- Sintaxis de las reglas CSS

CSS utiliza una sintaxis para crear reglas de estilo, el siguiente esquema muestra las partes que forman un estilo CSS muy básico:



Los diferentes términos se definen a continuación:

- ❖ **Regla:** cada uno de los estilos que componen una hoja de estilos CSS. Cada regla está compuesta de una parte de "selectores", un símbolo de "llave de apertura" ({), otra parte denominada "declaración" y, por último, un símbolo de "llave de cierre" (}).
- ❖ **Selector:** indica el elemento o elementos HTML a los que se aplica la regla CSS.
- ❖ **Propiedad:** característica que se modifica en el elemento seleccionado, como por ejemplo su tamaño de letra, su color de fondo, etc.
- ❖ **Valor:** establece el nuevo valor de la característica modificada en el elemento.

Un archivo CSS puede contener un número ilimitado de reglas CSS, cada regla se puede aplicar a varios selectores diferentes y cada declaración puede incluir tantos pares propiedad/valor como se desee.