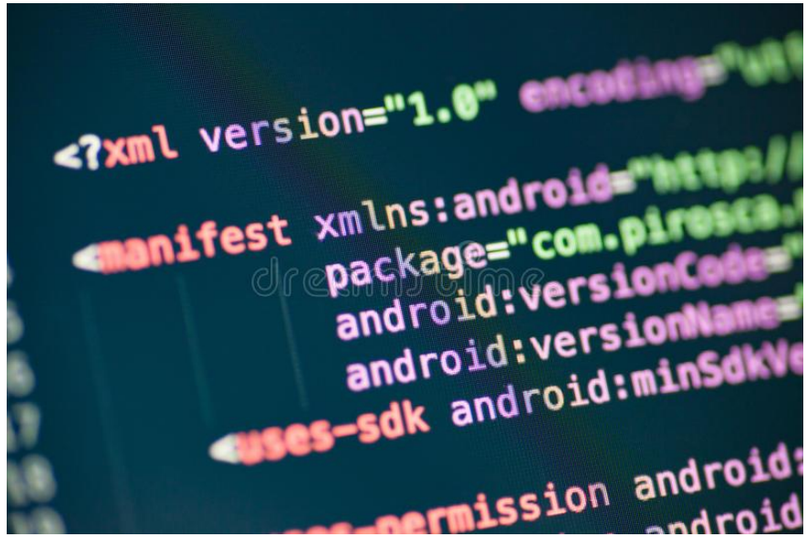
Lenguajes de marcas y sistemas de gestión de la información



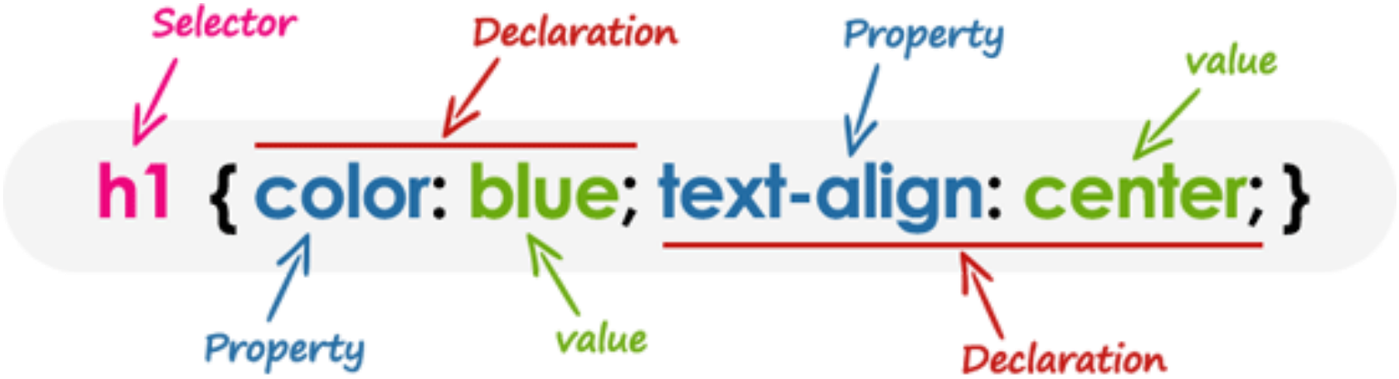
UT03 – CSS

3 – Variables y funciones

IES Clara del Rey – Madrid

Sintaxis CSS

Reglas CSS: selector y una o más declaraciones



Selector indica a que elemento o elementos del documento HTML se debe aplicar el estilo.

Cada una de las declaraciones tienen la forma “propiedad: valor”. Las declaraciones pueden estar en la misma línea o en varias.

Las declaraciones indican qué estilos queremos aplicar a los elementos seleccionados.

Variables CSS (CSS Custom Properties)

A veces en CSS tenemos que repetir un valor numérico muchas veces.

Por ejemplo, imaginemos que queremos utilizar el ancho 200px (200 píxeles) en múltiples reglas.

Tendremos una hoja de estilos con el valor en muchos sitios. Si tenemos que cambiarlo, por ejemplo, por 225px, tendremos que reemplazarlo en múltiples reglas. Pero teniendo cuidado de no reemplazar otros 200px que se refieran a otra cosa.

O podemos usar variables CSS, que permiten:

- Definir un valor en un único punto, y reutilizarlo muchas veces.
- Cambiar el valor siempre que queramos sin miedo a efectos laterales.
- Tener un código CSS más semántico, más fácil de leer

Variables CSS – Definición

Para definir una variable CSS, tenemos que definirla:

- Dentro de una regla para el elemento que queremos que aplique. En muchas ocasiones este elemento será toda la página (html o :root).
- Con el nombre que queramos, pero siempre precedido de dos guiones (--)

Ejemplo: variables para toda la página que definen el ancho de los botones medianos, y el fondo de todos los botones:

```
html {
    --btn-back-color: teal;
    --btn-width-mediano: 200px;
}
```

Los nombres de las variables los hemos elegido, no son propiedades CSS existentes, aunque pueda parecerlo.

Variables CSS – Uso

Para usar una variable CSS se utiliza la función "var".

Esta función recibe como parámetro el nombre de la variable CSS (incluidos los guiones --) y devuelve el valor asignado.

Ejemplos:

```
button {
    background-color: var(--btn-back-color);
}

.mediano {
    width: var(--btn-width-mediano);
}
```

El navegador usará los valores asignados previamente a las variables cuando tenga que calcular los estilos de los distintos elementos.

Variables CSS – Ámbito (scope)

Al definir la variable, se puede establecer su ámbito, aquellos elementos en los que tendrá valor.

Ejemplo: variables definidas para distintos ámbitos:

- Para los elementos de tipo blockquote

```
blockquote { --blockquote-right-margin: 1em; }
```

- Para los elementos con class "cita-autor"

```
.cita-autor { --margin-left: 10px; }
```

- Para cualquier elemento de la página (todo el html)

```
html { --back-color: teal; }
```

Variables CSS – Ámbito (scope) – Aplicación

Esto permite sobrescribir valores de variables. Por ejemplo, con este HTML y CSS, podemos tener valores distintos para una misma variable en cada una de las secciones de un documento HTML.

```
<section class="sect1">...</section>
<section class="sect2">...</section>
<section class="sect3">...</section>

-----------------------------------

section { --section-padding: 100px; }
.sect1 { --section-padding: 80px; }
.sect3 { --section-padding: 80px; }
```

La variable "--section-padding" valdrá 100px para todas las secciones, pero para la sección 1 y 3 su valor será de 80px, por la cascada.

Veremos esto de la cascada más adelante.

Funciones CSS

Una función CSS nos permiten realizar cálculos dentro de nuestro código CSS, como si de un lenguaje de programación se tratara.

Hay funciones para múltiples usos: matemáticas, control de tiempo en animaciones, para transformar elementos, para aplicar filtros gráficos a los elementos, para calcular colores, etc.

Ejemplos de alguna función matemática:

- calc(). Operaciones con dimensiones. Ejemplo: calc(0.2rem + 50px)
- max() y min(). Máximo o mínimo de dos valores.
- mod(). Módulo, resto de la división entera.
- attr(). Extrae el valor de un atributo del elemento en que se aplica la regla. Sólo se puede usar en ciertas situaciones que ya veremos.