Css: Cascading Style Sheets (hojas de estilo en cascada)

SINTAXIS:

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

SELECTORES:

* **SELECTOR UNIVERSAL \*** Sirve para agregar estilos a todos los elementos de la página. Normalmente se utiliza para hacer “reset” de estilos.
* **SELECTOR ETIQUETA.** Aplica estilos a todos los elementos de ese tipo (p, h1, header etc.)
* **SELECTOR ID.** Aplica estilos a un elemento único con ese Id, se recomienda hacer buen uso de este selector.
* **SELECTOR CLASE**. Aplica estilos a todos los elementos con esa clase (el que más vas a usar).
* **SELECTOR ANIDADO.** Aplica estilos a elementos descendientes de otros elementos (no necesario que sea hijo directo).
* **SELECTOR HIJO >.** Aplica estilos a los elementos que sean hijos directos de otros.
* **SELECTOR ADYACENTE +.** Aplica estilos al elemento adyacente.
* **SELECTOR DE ATRIBUTO INPUT[TYPE=“NUMBER”].** Aplica estilos al elemento con el atributo especificado.

La prioridad de un selector se determina por la suma de su contenido:

* **ID = 100**
* **Clase = 10**
* **Etiqueta = 1**

A mayor la suma, mayor prioridad.

* **!IMPORTANT** es un valor especial tiene un valor de un millón, nunca lo uses a menos que sea tu única opción como cuando no tienes acceso al código fuente.

PREPROCESADOR:

Herramienta que permite escribir pseudocodigo que luego se convierte a **CSS** o **HTML** estándar

¿Por qué utilizarlos?

* Salva tiempo y dinero al poder reutilizar código
* Sencillez de Código
* Modularizar los proyectos

VARIABLE:

Pedazo de memoria reservado para almacenar un valor, correspondiente a un tipo de dato (es donde se guardan y recuperan datos que se utilizan en el programa)

FUNCIONES:

Tienen la posibilidad de tener parámetros o argumentos que son variables que modifican su comportamiento

MIXIN:

Es una clase cuya finalidad es ofrecer una funcionalidad que pueda ser reutilizada en otras clases pero que no esta pensada para usarse de forma autónoma

METODOLOGIAS DE ESTRUCTURAR CODIGO:

Sistemas preestablecidos que ayudan a escribir y organizar Código escalable

Ventajas:

* Evitar redundancio para crear componentes escalables y reutilizables
* Evita el uso de !IMPORTANT
* Solucionar problemas de manejo de sistemas complejos

BEM:

* **BLOQUES.** Los bloques son nuestros contenedores más grandes que a su vez contienen elementos u otros bloques.
* **ELEMENTOS**. Los elementos siempre forman parte de un bloque, normalmente son los botones, textos, imágenes etc.
* **MODIFICADORES**. Los modificadores se usan para establecer estilos diferentes a un mismo bloque o elemento.

Ventajas:

* Menos repeticiones
* Independencia
* Mejoría en herencia múltiple

Diagrama

Descripción generada automáticamente