

**CSS**

-Se puede dar estilos por etiqueta, class o id

Etiqueta= img, body, h1{}

Class= .nombreClase{}

Id= #nombreId{}

border-radius: 9px;

    box-shadow: 1px 1px 10px rgb(168, 135, 167), 0px 0px 25px 0px rgb(255, 182, 251);

    width:150px;

    height:100px;

**PSEUDOCLASES**

BEM (para nombrar clases en CSS)

Se asigna clase al contenedor que contenga las etiquetas que se requiere dar estilo

[Metodología \*\*BEM](https://www.youtube.com/watch?v=wDUwGo98JTA&feature=youtu.be) cómo nombrar correctamente las clases.

Solo usa **clases**. Las nombra siguiendo el siguiente patrón: **BLOQUE\_\_ELEMENTO—MODIFICADOR** (son 2 guiones, se usa 1 guión para separar palabras).

* **Bloque:** sección que puede funcionar por sí sola.
* **Elemento:** forma una de las partes del bloque.
* **Modificador:** variaciones de un mismo bloque/elemento.

Ejemplo BEM:

<header>

        <nav>

            <ul *class*="main-nav">

                <li *class*="main-nav\_\_item"><a *href*="#">Home</a></li>

                <li *class*="main-nav\_\_item"><a *href*="#">Cursos</a></li>

                <li *class*="main-nav\_\_item"><a *href*="#">Instructores</a></li>

                <li *class*="main-nav\_\_item"><a *href*="#">Blog</a></li>

            </ul>

        </nav>

    </header>

**PSEUDOCLASES Y PSEUDOELEMENTOS**

[**https://developer.mozilla.org/es/docs/Web/CSS/Pseudo-classes**](https://developer.mozilla.org/es/docs/Web/CSS/Pseudo-classes)

[**https://developer.mozilla.org/es/docs/Web/CSS/Pseudo-elements**](https://developer.mozilla.org/es/docs/Web/CSS/Pseudo-elements)

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

youtube- [BEM: Simplifica tu CSS](https://youtu.be/wDUwGo98JTA)

youtube- [Que rayos es la Metodología BEM](https://youtu.be/bvnzyXGkNY4)

Documentacion- [sobre BEM](http://getbem.com/introduction/)

Documentación- [Las Pseudo-classes hacen uso de un ‘:’](https://developer.mozilla.org/es/docs/Web/CSS/Pseudo-classes)

Documentacion- [Los Pseudoelementos hacen uso de dos ‘::’](https://developer.mozilla.org/es/docs/Web/CSS/Pseudoelementos)

.main-nav{

*/\* Margen superior \*/*

    margin-top: 10px;

*/\* Eliminar puntos que aparecen en listas desordenadas \*/*

    list-style: none;

*/\* Elimina espacio a la izquierda dentro de la caja \*/*

    padding-left: 0;

    background-color: #13a4a4;

}

.main-nav\_\_item{

*/\* Para que el listado quede horizontal \*/*

    display: inline-block;

}

.main-nav\_\_item a{

    color: white;

*/\* Espacio entre los elementos \*/*

    padding: 5px;

*/\* Le agrega borde \*/*

    border-radius: 2px;

*/\* Elimina subrayado que genera la etiqueta "a" \*/*

    text-decoration: none;

}

*/\* PSEUDOCLASE \*/*

*/\* Efecto de oscurecer texto cuando se pasa el cursor por encima \*/*

.main-nav\_\_item a:hover{

    color: rgba(255, 255, 255, 0.5);

}

*/\* Cambia el color del elemento al cual se dió click \*/*

.main-nav\_\_item a:active{

    color: rgba(255, 255, 255, 0.5);

}

*/\* PSEUDOELEMENTOS \*/*

*/\* Agrega un caracter después del texto de la etiqueta \*/*

.main-nav\_\_item a::after{

    content: " | ";

}

**ANATOMIA DE UNA REGLA DE CSS**

**Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media**

**MODELO DE CAJA**

**Interfaz de usuario gráfica

Descripción generada automáticamente**

En la web los elementos se piensan en forma de rectángulos. A esto se le conoce como Modelo de caja.

**Margin**

Espacio que hay del borde de la caja hacia afuera. Visualmente, permite separar a la caja de elementos externos.

**Padding**

Espacio que hay del borde de la caja hacia adentro. Visualmente, permite separar a la caja de elementos internos.

**Border**

Es el borde de la caja. Por defecto, a partir de ahí medimos las distancias de nuestra caja con elementos internos o externos.

Imagen que contiene Código QR

Descripción generada automáticamente

**Width**

Largo del contenido

**Height**

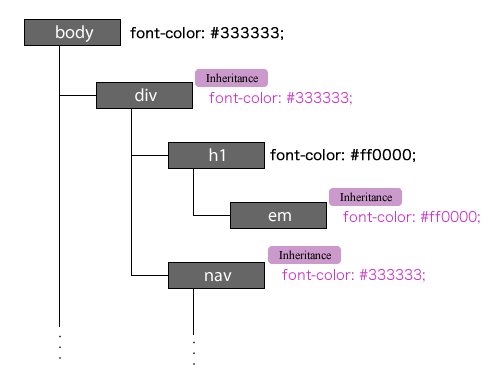
Alto del contenido

**HERENCIA**

**Inherit**. Este es un valor por medio de una *keyword* que especifica que, a la propiedad que se la apliquemos debe de heredar los valores de su elemento padre. Podemos decir que la palabra **Inherit** significa “*Usa el valor de mi padre*”, si el elemento padre no tiene definido dicho valor el navegador seguirá el DOM hasta que encuentre un elemento superior que lo contenga, y en ultima instancia de no tenerlo ningún elemento superior se aplicara el valor por defecto.

**Initial**. Este valor pertenece a la especificación CSS3 y cuando aplicamos a una propiedad el valor *initial* estamos dando el valor inicial y predefinido por el navegador en cuestión.

**Upset**. Este valor *unset* es una combinación entre *inherit* y *initial*, cuando utilizamos este valor en una propiedad esta tratara de heredar el valor de su elemento padre si este esta disponible, de no ser así este valor colocará el valor de la propiedad en su valor inicial, como si usáramos *inherit* e *initial* juntos.



**ESPECIFIDAD EN SELECTORES**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Tabla

Descripción generada automáticamente** **Diagrama

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

1. Los **!important** estarán por encima de los demás estilos. Sin embargo, son mala práctica y no se deberían usar.
2. Los estilos embebidos en el HTML, es decir **inline styles** están por encima de las **clases y IDs**. Sin embargo, también se deberían evitar.
3. Los **IDs** están por encima de las **clases**. Los **IDs** son específicos, si se usa uno en un archivo HTML ya no se podrá repetir más en ese mismo archivo. Mientras que las **clases** si se pueden repetir en cualquier elemento.
4. Un **estilo de etiqueta** es el último valor que el navegador tiene en cuenta antes de tomar los estilos por defecto de esa etiqueta. Los **estilos de etiqueta** son los que menos peso tienen.