Tema 2.10 - Generación de Interfaces I - CSS (BEM, funciones matemáticas)

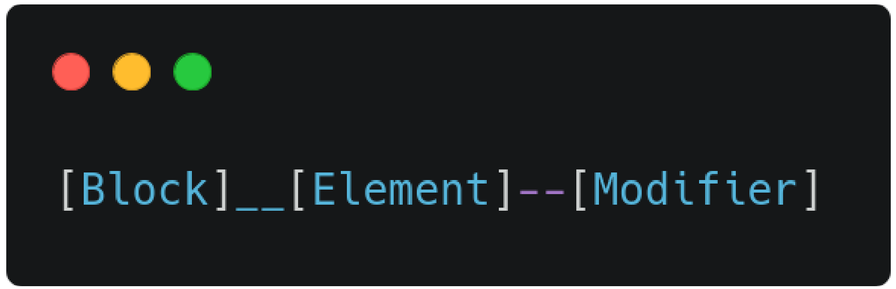
Módulo: Desarrollo de Interfaces

Curso 2º DAM

¿Qué es BEM (Block Element Modifier)?

BEM (Block-Element-Modifier) es una convención de nomenclatura ampliamente utilizada para nombres de clases CSS. Ayuda a escribir CSS más legible, comprensible y escalable.

Los nombres en BEM siguen esta estructura:



- **Block**: Representa el componente principal, como un botón o una tarjeta.
- **Element**: Subpartes o elementos dentro del bloque, como un icono o un título.
- **Modifier**: Variaciones en el estilo del bloque o de un elemento, como un botón secundario o deshabilitado.

Ejemplos simples BEM

1. Bloque sin elementos ni modificadores

```
<button class=”btn”></button>

.btn {...}
```

2. Bloque con modificador

```
<!-- CORRECTO -->
<button class="btn btn--secondary"></button>

.btn {...}
.btn--secondary {...}

<!-- INCORRECTO -->
<button class="btn--secondary"></button>

.btn--secondary {...}
```

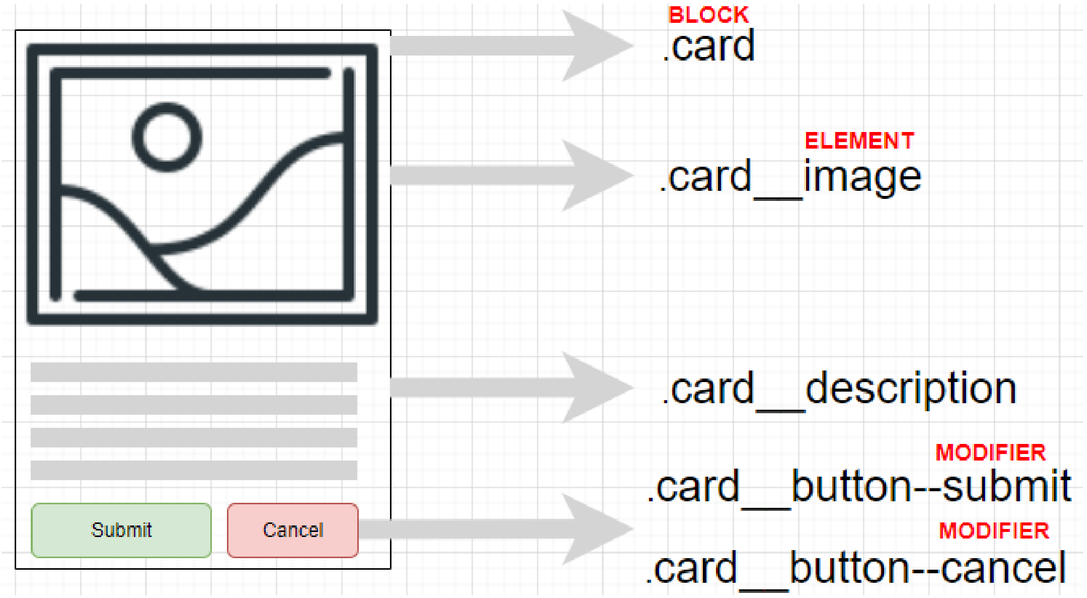
3. Bloque con elementos

```
<figure class="photo">
  <img class="photo__img" src="me.jpg">
  <figcaption class="photo__caption">Look at me!</figcaption>
</figure>

.photo {...}
.photo__img {...}
.photo__caption {...}
```

4. Bloque con elementos y modificadores

```
<figure class="photo">
  <img class="photo__img photo__img--framed" src="me.jpg">
  <figcaption class="photo__caption photo__caption--large">Look at me!</figcaption>
</figure>

.photo {...}
.photo__img {...}
.photo__caption {...}
.photo__img--framed {...}
.photo__caption--large {...}
```

Ejemplo estructura BEM




```html
<div class="card">
    <div class="card__image"></div>
    <p class="card__description"></p>
    <button class="card__button--submit"></button>
    <button class="card__button--cancel"></button>
</div>
```

```css
.card__image {
  display: block;
}
.card__description {
  display: block;
}
.card__button--submit {
  display: inline-block;
}
.card__button--cancel {
  display: inline-block;
}
```

Algunos errores comunes

https://dev.to/visuellverstehen/common-mistakes-when-writing-css-with-bem-4921

Funciones matemáticas útiles en CSS

- **calc()**: Realiza cálculos matemáticos.
- **min() / max()**: Devuelve el valor mínimo o máximo entre varias opciones (normalmente 2).
- **clamp():** Define un valor dinámico dentro de un rango mínimo y máximo. 3 valores (mínimo, preferido, maximo).

```css
width: calc(100% - 50px); /* Ancho calculado dinámicamente */
```

```css
width: min(50%, 400px); /* Usa el menor valor */
width: max(200px, 10%); /* Usa el mayor valor */
```

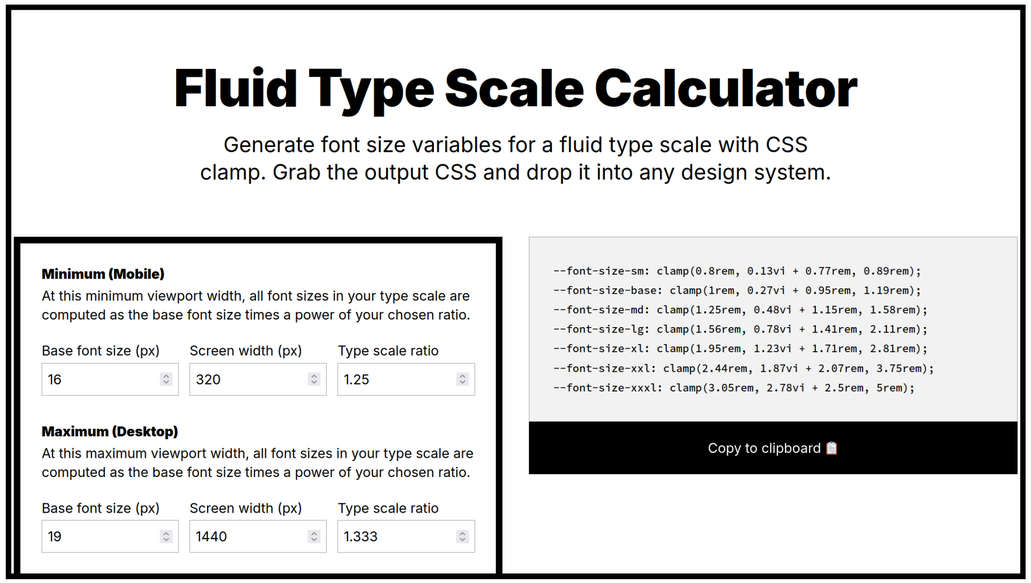
Comunmente utilizada para definir max-width, width o min-height, height.

```css
font-size: clamp(1rem, 2.5vw, 3rem); /* Escala entre 1rem y 3rem */
```

Comunmente utilizada realizar tamaños fluidos en el texto.

320px ⟷ 1440px

rango de px a tener en cuenta



https://www.fluid-type-scale.com/

Utilizaremos *clamp()* para definir de manera fluida el font-size de nuestra interfaz