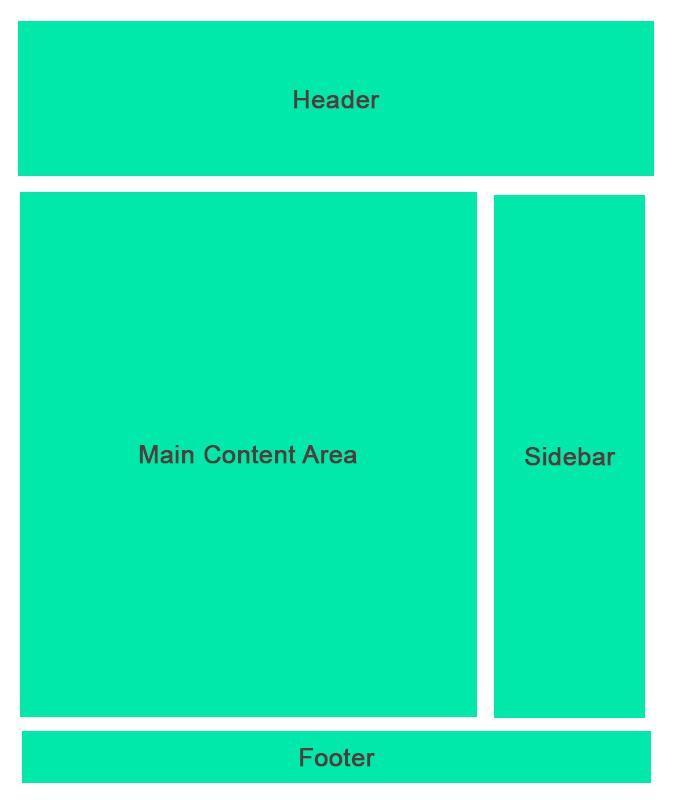
*BEM*

Todos queremos **hacer que nuestro código sea más fácil de leer**. Esto nos ayuda a trabajar más rápidamente y de manera eficiente, y cuando otros trabajen con nosotros podremos mantener claridad y coherencia. Pareciera que en estos días existen muchos estándares y términos en el mundo del CSS: [OOCSS](http://oocss.org/), [SMACSS](http://webdesign.tutsplus.com/articles/quick-tip-fix-your-messy-css-with-smacss--webdesign-18489), BEM y otros. Todos estos son **acercamientos para mejorar la estructura del CSS**, y hoy vamos a cubrir la metodología BEM.

BEM significa *Modificador de Bloques de Elementos (Block Element Modifier)* por sus siglas en inglés. Sugiere **una manera estructura de nombrar tus clases, basado en las propiedades del elemento en cuestión**. Si alguna vez has visto un nombre de una clase como header\_from-email eso es precisamente BEM en acción. Cuando utilices la metodología BEM, debes tomar en cuenta que solamente usarás nombres de clases (no IDs). Los nombres de clases te permiten repetir el nombre BEM si es necesario, y crear una estructura de código más consistente (en ambos archivos el HTML y CSS/Sass). Ahora vamos a descifrarlo.

Bloque

El *bloque* es un contenedor o contexto donde el elemento se encuentra presente. Piensa como si fueran partes estructurales de código más grande. Puede que tengas un encabezado, pie de página, una barra lateral y un área de contenido principal; cada uno de estos sería considerado como un bloque. Mira la siguiente imagen:

El bloque de elementos forma la raíz de la clase y siempre irá primero. Solo debes saber que una vez que has definido tu bloque, estarás listo para comenzar a nombrar tus elementos.

|  |
| --- |
| .block\_\_element {      background-color: #FFFFFF;  } |
|

El doble guión bajo te permite visualizar, navegar rápidamente y manipular tu código. Aquí hay algunos ejemplos de cómo funciona la metodología de elementos:

|  |
| --- |
| .header\_\_logo {}  .header\_\_tagline {}  .header\_\_searchbar {}  .header\_\_navigation {} |
|

Como puedes ver, hay espacio para la creatividad y apropiarte de esta metodología. En el ejemplo, "navigation" puede ser cambiado a "nav"; "tagline" puede cambiarse a "tag" o "tagLine". El punto es **mantener los nombres simples, claros, y precisos**. No lo pienses demasiado, y solo porque tus hojas de estilos y html sean estáticos, no quiere decir que tengas que volver a repetir el mismo código.

Actualizar el nombre de las clases no debería ser un problema cuando encuentras una mejor semántica que funcione (solo debes tratar de ser consistente y apegarte a ella). Los elementos se convertirán en el centro de los nombres de tus clases, y te ayudarán a entender cómo estructurar tus hojas de estilos y cómo manejar tu diseño.

Modificadores

Cuando nombras una clase, la intención es ayudar a que ese elemento pueda ser repetido para que no tengas que escribir nuevas clases en otras áreas del sitio si los elementos de estilo son los mismos. Cuando necesitas **modificar el estilo de un elemento específico**, puedes usar un modificador. Para lograr esto, añades un doble guión *--* luego del elemento (o bloque). Aquí tenemos un corto ejemplo:

|  |
| --- |
| .block--modifier {}  .block\_\_element--modifier {} |
|

Ten cuidado al usar los modificadores, recuerda que se quiere mantener todo más simple y no tener que repetir lo mismo o crear clases extra a menos que sea absolutamente necesario. Lo explicaremos con un código usando el encabezado del sitio como nuestro bloque:

|  |
| --- |
| .header\_\_navigation {}  .header\_\_navigation--secondary {} |
|

Si estás usando una segunda navegación, lo más probable es que el diseño y espaciado no cambien, pero puede que la navegación secundaria tenga un color distinto. Puedes duplicar los estilos originales, o mejor aún, usar un pre-procesador. Con Sass, podrías @extender el elemento principal (así el elemento secundario heredará todas las propiedades) y cambiar los estilos apropiados.

|  |
| --- |
| .header\_\_navigation {   background: #008cba;   padding: 1rem 0;   margin: 2rem 0;   text-transform: uppercase;  }  .header\_\_navigation--secondary {   @extend .header\_\_navigation;   background: #dfe0e0;  } |
|

Puede que pienses, "¡pero el nombre de la clase es muy largo!". Piénsalo así: Los nombres de las clases BEM son específicos, claros, fáciles de leer dentro del html y comunican claramente para qué existen.

Cuando usamos BEM solo utilizamos un nombre de clase por cada etiqueta html. Fíjate cómo funciona para la etiqueta "label". Selectores estándares:

|  |
| --- |
| .label .label-default {}  .label .label-alert {} |
|

vs. selectores BEM:

|  |
| --- |
| .label {}  .label--alert {} |
|

Mas información acerca de BEM:

<https://en.bem.info/>

<https://csswizardry.com/2013/01/mindbemding-getting-your-head-round-bem-syntax/>

<https://coding.smashingmagazine.com/2012/04/a-new-front-end-methodology-bem/>

<https://www.intelygenz.es/como-organizar-css-metodologias-oocss-bem-smacss/>

<https://www.espai.es/blog/2016/07/metodologias-css-oocss-bem-smacss/>