

Cómo escribir HTML / CSS más rápido con atajos de Emmet

[Luis Delgado](#) 31 de mayo de 2021

Emmet es una de mis funciones integradas favoritas de VS Code. También está disponible como una extensión en otros editores de código.

Piense en Emmet como una taquigrafía. Con él, puede escribir fácilmente una gran cantidad de código rápidamente. Acelera drásticamente su flujo de trabajo HTML y CSS.

Entender cómo usar Emmet es literalmente una superpotencia. ¿Algunos incluso lo han llamado un código de trucos para programar.

Y es solo una de las muchas características asombrosas integradas de VS Code.

Recientemente, lancé un curso completo y detallado llamado [Become A VS Code SuperHero](#). (Conviértete en un superheroe de VS Code). Cubre todos los aspectos de VS Code en gran profundidad.

Este artículo se basa en una de las 11 secciones del curso: **escritura y formato de código**.

HTML

Con Emmet puede crear rápidamente una plantilla HTML en un abrir y cerrar de ojos. En un archivo HTML, simplemente escriba `!` y verá que Emmet ha aparecido como sugerencia. Ahora presiona `Enter`. Ahí lo tiene, una página web HTML básica en blanco lista para usar.

Si nunca antes ha visto HTML y no tiene idea de qué es todo esto, no se preocupe. Solo estoy demostrando las capacidades de VS Code y Emmet. No es necesario que

sepas qué significa todo esto en este momento.

Etiquetas Básicas

Para crear etiquetas HTML básicas, simplemente escriba el nombre de la etiqueta y presione `Enter`. Observe cómo Emmet coloca su cursor dentro de la etiqueta. Ahora está configurado para continuar escribiendo dentro de la etiqueta.

- Intente escribir: `div` y luego `Enter`, o `h1` `Enter`, o `p` `Enter`.
- Estos también funcionan: `bq` para un `<blockquote>`, `hdr` para un `<header>`, `ftr` para un `<footer>`, `btn` para un `<button>` y `sect` para una sección.

Clases

Si está familiarizado con CSS, Emmet usa la misma forma para referirse a las clases usando un `..`. Para definir una clase junto con la etiqueta, simplemente agréguela así:

- `div.wrapper` \rightarrow `<div class="wrapper"></div>`
- `h1.header.center` \rightarrow `<h1 class="header center"></h1>`

ID's

Los Id funcionan de manera muy similar:

- `div#hero —> <div id="hero"></div>`

Combinatoria

Puede combinar estos juntos:

- `div#hero.wrapper —> <div id="hero" class="wrapper"></div>`

Atributos

También podemos especificar atributos para las etiquetas:

- `img[src="cat.jpg"][alt="foto de gato"] —> `

Contenido

Para incluir contenido dentro de la etiqueta, simplemente envolvemos el contenido entre llaves, { }.

- `p{Esto es un párrafo} —> <p>Esto es un párrafo</p>`

Etiquetas hermanos e hijos

Cada vez se pone mejor. También podemos especificar etiquetas hermanos e hijos usando los caracteres + y >.

- `section+section —> <section></section><section>`

</section>

- `ul>li —> `

Escalando

Tienes que intentar imaginar lo que estás construyendo en tu cabeza mientras escribes la taquigrafía de Emmet. En este ejemplo, "treparemos" por la estructura HTML usando ^.

```
div+div>p>span+em^bq
```

Resultado:

```
<div></div>
<div>
  <p><span></span><em></em></p>
  <blockquote></blockquote>
</div>
```

Aquí, queríamos que la etiqueta en `blockquote` estuviera al mismo nivel que el párrafo. Por eso, necesitábamos "trepar". De lo contrario, estaría dentro del párrafo.

Agrupar

Si su estructura es muy compleja, es posible que desee agrupar etiquetas en lugar de atravesar trepando.

En este ejemplo, crearemos un encabezado (header) y un

pie de página (footer) sin escalar usando paréntesis ().

```
div>(header>ul>li>a)+footer>p
```

Resultado:

```
<div>
  <header>
    <ul>
      <li><a href=""></a></li>
    </ul>
  </header>
  <footer>
    <p></p>
  </footer>
</div>
```

Multiplicación y \$

Podemos generar múltiples etiquetas multiplicando (*) y numerando elementos en secuencia usando un signo de dólar (\$).

- `ul>li*5 —> `
- `ul>li{Item $}*3 —> Item 1Item 2Item 3`

Incluso puede personalizar la secuencia de numeración rellenando con ceros, comenzando con un número específico e incluso en la dirección inversa.

Pad con ceros: `u>li.item$$$*5`

Resultado:

```
<ul>
  <li class="item001"></li>
  <li class="item002"></li>
  <li class="item003"></li>
  <li class="item004"></li>
  <li class="item005"></li>
</ul>
```

Comience con un número específico: `ul>li.item$@3*5`

Resultado:

```
<ul>
  <li class="item3"></li>
  <li class="item4"></li>
  <li class="item5"></li>
  <li class="item6"></li>
  <li class="item7"></li>
</ul>
```

Dirección inversa: `ul>li.item$@-*5`

Resultado:

```
<ul>
  <li class="item5"></li>
  <li class="item4"></li>
```

```
<li class="item3"></li>
<li class="item2"></li>
<li class="item1"></li>
</ul>
```

Dirección inversa desde un número específico:

```
ul>li.item$@-3*5
```

Resultado:

```
<ul>
  <li class="item7"></li>
  <li class="item6"></li>
  <li class="item5"></li>
  <li class="item4"></li>
  <li class="item3"></li>
</ul>
```

Nombres de etiqueta implícitos

Hay ciertas etiquetas que no es necesario escribir y pueden estar implícitas.

- Una clase definida inicialmente sin etiqueta estará implícita como `<div>`.

```
.wrapper —> <div class="wrapper"></div>
```

- Una clase definida dentro de una etiqueta de énfasis estará implícita como ``.

```
em>.énfasis —> <em><span class="énfasis">
</span></em>
```


- Una clase definida dentro de una lista estará implícita como un elemento de lista.

```
ul>.item —> <ul><li class="item"></li></ul>
```

- Una clase definida dentro de una tabla estará implícita como <tr> y dentro de una fila será un <td>.

```
table>.fila>.col —> <table><tr class="fila"><td class="col"></td></tr></table>
```

Envolver con etiquetas

Habrán ocasiones en las que tenga un código existente que desee envolver con una etiqueta. Podemos hacerlo fácilmente con Emmet.

Simplemente resalte el código que desea envolver y abra la paleta de comandos (F1). Luego busque Emmet: Ajustar con abreviatura Emmet: Wrap with Abbreviation. A continuación, se le presentará un cuadro de diálogo donde puede escribir el elemento.

```
test —> <div>test</div>
```

También puede utilizar la sintaxis estándar de Emmet en este cuadro de diálogo. Intente ajustar un poco de texto con `span.wrapper`.

De forma predeterminada, esta funcionalidad no está asignada a un método abreviado de teclado. Por lo tanto, si lo usa con frecuencia, es posible que desee agregar un

acceso directo personalizado para él.

Lorem Ipsum

"Lorem Ipsum" es un texto ficticio utilizado por los desarrolladores para representar datos en una página. Simplemente escriba `lorem` y presione `Enter`. Emmet generará 30 palabras de texto falso que puede usar como relleno en su proyecto.

También se puede definir la cantidad de palabras generadas.

- `lorem10` te dará 10 palabras de texto aleatorio.

Poniéndolo todo junto

Usemos varias cosas que hemos aprendido hasta ahora. Prueba esto:

```
ul.mi-lista>lorem10.item-$*5
```

Resultado:

```
<ul class="mi-lista">
  <li class="item-1">Lorem ipsum dolor sit amet.</li>
  <li class="item-2">Numquam repudiandae fuga porro consequ
  <li class="item-3">Culpa, est. Tenetur, deleniti nihil?</
  <li class="item-4">Numquam architecto corrupti quam repud
</ul>
```

CSS

En CSS, Emmet tiene una abreviatura para cada propiedad. No voy a enumerarlos todos, pero señalaré los más utilizados. Para ver la lista completa, consulte la [hoja de referencia](#) de Emmet.

Position

- `pos` —> `position:relative;` (Por defecto es relative)
- `pos:s` —> `position:static;`
- `pos:a` —> `position:absolute;`
- `pos:r` —> `position:relative;`
- `pos:f` —> `position:fixed;`

Display

- `d` —> `display:block;` (Lo predetermina a block)
- `d:n` —> `display:none;`
- `d:b` —> `display:block;`
- `d:f` —> `display:flex;`
- `d:if` —> `display:inline-flex;`
- `d:i` —> `display:inline;`
- `d:ib` —> `display:inline-block;`

Cursor

- `cur` —> `cursor:pointer;`

Color

- `c` —> `color:#000;`
- `c:r` —> `color:rgb(0, 0, 0);`
- `c:ra` —> `color:rgba(0, 0, 0, .5);`
- `op` —> `opacity: ;`

Margin & Padding

- `m` —> `margin: ;`
- `m:a` —> `margin:auto;`
- `mt` —> `margin-top: ;`
- `mr` —> `margin-right: ;`
- `mb` —> `margin-bottom: ;`
- `ml` —> `margin-left: ;`
- `p` —> `padding: ;`
- `pt` —> `padding-top: ;`
- `pr` —> `padding-right: ;`
- `pb` —> `padding-bottom: ;`
- `pl` —> `padding-left: ;`

Box Sizing

- `bxz` —> `box-sizing:border-box;`

Width

- `w` —> `width: ;`
- `h` —> `height: ;`

- `maw` → `max-width: ;`
- `mah` → `max-height: ;`
- `miw` → `min-width: ;`
- `mih` → `min-height: ;`

Border

- `bd` → `border: ;`
- `bd+` → `border:1px solid #000;`
- `bd:n` → `border:none;`

¡Emmet es increíble!

Con Emmet, puede crear una estructura HTML realmente compleja con una línea. Es realmente asombroso. Y también funciona con CSS.

Puede ver cómo Emmet puede aumentar drásticamente su eficiencia y velocidad al escribir HTML y CSS.

Si desea aumentar aún más su eficiencia y velocidad mientras usa VS Code, consulte el curso Conviértase en un Superhéroe de VS Code ([Become A VS Code Super Hero](#)).

El curso profundiza mucho más en estos conceptos y te ayuda a convertirte en un desarrollador de superhéroes rápido y eficiente :)

Traducido del artículo de [Jesse Hall](#) - [How to Write](#)

HTML/CSS Faster with Emmet Cheat-Codes