

Emmet Shortcuts for HTML - Complete Reference Guide

What is Emmet?

Emmet is a powerful toolkit that allows you to write HTML and CSS much faster using abbreviations that expand into full code blocks. It's built into VS Code by default.

How to use: Type the abbreviation and press `Tab` to expand it.

Basic HTML Structure

Document Structure

! + Tab

Expands to:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>

</body>
</html>
```

HTML5 Document with Custom Title

```
html:5 + Tab
```

Same as above, or you can use:

```
doc + Tab
```

Basic Elements

Simple Tags

Shortcut	Expands to
div + Tab	<div></div>
p + Tab	<p></p>
h1 + Tab	<h1></h1>
span + Tab	
a + Tab	
img + Tab	
br + Tab	
hr + Tab	<hr>

Input Elements

Shortcut	Expands to
input + Tab	<input type="text">
input:text + Tab	<input type="text" name="" id="">
input:email + Tab	<input type="email" name="" id="">

Shortcut	Expands to
<code>input:password</code> + Tab	<code><input type="password" name="" id=""></code>
<code>input:submit</code> + Tab	<code><input type="submit" value=""></code>
<code>input:button</code> + Tab	<code><input type="button" value=""></code>
<code>input:checkbox</code> + Tab	<code><input type="checkbox" name="" id=""></code>
<code>input:radio</code> + Tab	<code><input type="radio" name="" id=""></code>

Classes and IDs

Adding Classes

```
div.container + Tab
```

Expands to:

```
<div class="container"></div>
```

Multiple Classes

```
div.header.main.large + Tab
```

Expands to:

```
<div class="header main large"></div>
```

Adding IDs

```
div#header + Tab
```

Expands to:

```
<div id="header"></div>
```

Combining Classes and IDs

```
div#main.container.wrapper + Tab
```

Expands to:

```
<div id="main" class="container wrapper"></div>
```

Multiplication and Numbering

Multiple Elements

```
li*5 + Tab
```

Expands to:

```
<li></li>  
<li></li>  
<li></li>  
<li></li>  
<li></li>
```

Auto-numbering with \$

```
li.item$*3 + Tab
```

Expands to:

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

Custom Numbering Start

```
li.item$@3*3 + Tab
```

Expands to:

```
<li class="item3"></li>
<li class="item4"></li>
<li class="item5"></li>
```

Reverse Numbering

```
li.item$@-*3 + Tab
```

Expands to:

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

Zero-padded Numbers

```
img.pic$$$*3 + Tab
```

Expands to:

```
<img src="" alt="" class="pic001">
<img src="" alt="" class="pic002">
<img src="" alt="" class="pic003">
```

Nesting and Grouping

Child Elements (>)

```
nav>ul>li*3>a + Tab
```

Expands to:

```
<nav>
  <ul>
    <li><a href=""></a></li>
    <li><a href=""></a></li>
    <li><a href=""></a></li>
  </ul>
</nav>
```

Sibling Elements (+)

```
h1+p+div + Tab
```

Expands to:

```
<h1></h1>
<p></p>
<div></div>
```

Climb Up (^)

```
div>header>h1^main>p + Tab
```

Expands to:

```
<div>
  <header>
    <h1></h1>
  </header>
  <main>
    <p></p>
  </main>
</div>
```

Grouping with Parentheses

(header>h1+nav>ul>li*3>a)+(main>section*2>h2+p) + Tab

Expands to:

```
<header>
  <h1></h1>
  <nav>
    <ul>
      <li><a href=""></a></li>
      <li><a href=""></a></li>
      <li><a href=""></a></li>
    </ul>
  </nav>
</header>
<main>
  <section>
    <h2></h2>
    <p></p>
  </section>
  <section>
    <h2></h2>
    <p></p>
  </section>
</main>
```

Attributes

Custom Attributes

```
img[src="image.jpg" alt="My Image"] + Tab
```

Expands to:

```

```

Multiple Attributes

```
input[type="email" name="email" placeholder="Enter email" required] + Tab
```

Expands to:

```
<input type="email" name="email" placeholder="Enter email" required>
```

Dynamic Attributes with Numbering

```
input[name="field$"]*3 + Tab
```

Expands to:

```
<input type="text" name="field1">  
<input type="text" name="field2">  
<input type="text" name="field3">
```

Text Content

Adding Text with {}

```
h1{Welcome to My Site} + Tab
```


Expands to:

```
<h1>Welcome to My Site</h1>
```

Text with Numbering

```
h2{Chapter $}*3 + Tab
```

Expands to:

```
<h2>Chapter 1</h2>
<h2>Chapter 2</h2>
<h2>Chapter 3</h2>
```

Complex Text Structure

```
ul>li.item${Item $}*5 + Tab
```

Expands to:

```
<ul>
  <li class="item1">Item 1</li>
  <li class="item2">Item 2</li>
  <li class="item3">Item 3</li>
  <li class="item4">Item 4</li>
  <li class="item5">Item 5</li>
</ul>
```

Common HTML Patterns

Navigation Menu

```
nav>ul>li*5>a[href="#"]{Menu Item $} + Tab
```

Expands to:

```
<nav>
  <ul>
    <li><a href="#">Menu Item 1</a></li>
    <li><a href="#">Menu Item 2</a></li>
    <li><a href="#">Menu Item 3</a></li>
    <li><a href="#">Menu Item 4</a></li>
    <li><a href="#">Menu Item 5</a></li>
  </ul>
</nav>
```

Card Layout

```
div.card*3>div.card-header>h3{Card $}^div.card-body>p{Card content for card $} + Tab
```

Expands to:

```
<div class="card">
  <div class="card-header">
    <h3>Card 1</h3>
  </div>
  <div class="card-body">
    <p>Card content for card 1</p>
  </div>
</div>
<div class="card">
  <div class="card-header">
    <h3>Card 2</h3>
  </div>
  <div class="card-body">
    <p>Card content for card 2</p>
  </div>
</div>
<div class="card">
  <div class="card-header">
    <h3>Card 3</h3>
  </div>
  <div class="card-body">
    <p>Card content for card 3</p>
  </div>
</div>
```

Table Structure

```
table>thead>tr>th*4{Header $}^^tbody>tr*3>td*4{Cell $} + Tab
```

Expands to:

```
<table>
  <thead>
    <tr>
      <th>Header 1</th>
      <th>Header 2</th>
      <th>Header 3</th>
      <th>Header 4</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Cell 1</td>
      <td>Cell 2</td>
      <td>Cell 3</td>
      <td>Cell 4</td>
    </tr>
    <tr>
      <td>Cell 1</td>
      <td>Cell 2</td>
      <td>Cell 3</td>
      <td>Cell 4</td>
    </tr>
    <tr>
      <td>Cell 1</td>
      <td>Cell 2</td>
      <td>Cell 3</td>
      <td>Cell 4</td>
    </tr>
  </tbody>
</table>
```

Contact Form

```
form>div.form-group*4>label[for="field$"]{Label $}+input[type="text" id="field$" name="field$"]'
```

Expands to:

```
<form>
  <div class="form-group">
    <label for="field1">Label 1</label>
    <input type="text" id="field1" name="field1">
  </div>
  <div class="form-group">
    <label for="field2">Label 2</label>
    <input type="text" id="field2" name="field2">
  </div>
  <div class="form-group">
    <label for="field3">Label 3</label>
    <input type="text" id="field3" name="field3">
  </div>
  <div class="form-group">
    <label for="field4">Label 4</label>
    <input type="text" id="field4" name="field4">
  </div>
  <div class="form-group">
    <button type="submit">Submit</button>
  </div>
</form>
```

Lorem Ipsum Text Generation

Basic Lorem

```
p>lorem + Tab
```

Expands to:

```
<p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Animi, explicabo quod molestiae do.
```

Specific Word Count

```
p>lorem10 + Tab
```

Expands to:

```
<p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Animi, explicabo.</p>
```

Multiple Paragraphs with Lorem

```
p*3>lorem5 + Tab
```

Expands to:

```
<p>Lorem ipsum dolor sit amet.</p>
<p>Lorem ipsum dolor sit amet.</p>
<p>Lorem ipsum dolor sit amet.</p>
```

Advanced Emmet Features

Implicit Tags

Emmet can guess tag names based on parent element:

```
.header + Tab (inside body)
```

Expands to:

```
<div class="header"></div>
```

```
.item + Tab (inside ul)
```

Expands to:

```
<li class="item"></li>
```

```
.row + Tab (inside table)
```

Expands to:

```
<tr class="row"></tr>
```

CSS-like Selectors

```
div#page.container + Tab
```

Expands to:

```
<div id="page" class="container"></div>
```

Boolean Attributes

```
input[disabled] + Tab
```

Expands to:

```
<input type="text" disabled>
```

HTML5 Semantic Elements

Semantic Structure

```
header>nav>ul>li*5>a[href="#"]{Nav Item $}^^main>section*3>h2{Section $}+p>lorem^^footer>p{Copyr
```

Expands to:

```
<header>
  <nav>
    <ul>
      <li><a href="#">Nav Item 1</a></li>
      <li><a href="#">Nav Item 2</a></li>
      <li><a href="#">Nav Item 3</a></li>
      <li><a href="#">Nav Item 4</a></li>
      <li><a href="#">Nav Item 5</a></li>
    </ul>
  </nav>
</header>
<main>
  <section>
    <h2>Section 1</h2>
    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Animi, explicabo quod molestiae.
  </section>
  <section>
    <h2>Section 2</h2>
    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Animi, explicabo quod molestiae.
  </section>
  <section>
    <h2>Section 3</h2>
    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Animi, explicabo quod molestiae.
  </section>
</main>
<footer>
  <p>Copyright 2025</p>
</footer>
```

Quick Reference Cheat Sheet

Syntax Elements

Symbol	Meaning	Example
>	Child	div>p
+	Sibling	h1+p

Symbol	Meaning	Example
<code>^</code>	Climb up	<code>div>p^div</code>
<code>*</code>	Multiply	<code>li*3</code>
<code>()</code>	Grouping	<code>(div>p)*2</code>
<code>[]</code>	Attributes	<code>a[href="#"]</code>
<code>{}</code>	Text	<code>p{Hello}</code>
<code>\$</code>	Numbering	<code>li.item\$*3</code>
<code>.</code>	Class	<code>div.container</code>
<code>#</code>	ID	<code>div#header</code>

Common Shortcuts

Purpose	Shortcut	Result
HTML5 Document	<code>!</code>	Full HTML5 template
Div with class	<code>.container</code>	<code><div class="container"></div></code>
Div with ID	<code>#header</code>	<code><div id="header"></div></code>
Navigation menu	<code>nav>ul>li*5>a</code>	Complete nav structure
Form group	<code>div.form-group>label+input</code>	Label and input pair
Lorem text	<code>lorem10</code>	10 words of lorem ipsum

VS Code Emmet Settings

Enable Emmet in More File Types

Add to VS Code settings.json:


```
{
  "emmet.includeLanguages": {
    "javascript": "javascriptreact",
    "vue-html": "html",
    "razor": "html",
    "plaintext": "pug"
  }
}
```

Trigger Expansion Key

Change from Tab to another key:

```
{
  "emmet.triggerExpansionOnTab": false
}
```

Then use `Ctrl+E` (or `Cmd+E` on Mac) to expand.

Practice Exercises

Beginner

1. Create a basic HTML5 document structure
2. Make a navigation menu with 5 items
3. Create 3 paragraphs with lorem text
4. Build a simple contact form

Intermediate

1. Create a blog post layout with header, main content, and sidebar
2. Build a product grid with 6 cards
3. Make a complex table with headers and data
4. Create a multi-level navigation menu

Advanced

1. Build a complete landing page structure
 2. Create a dashboard layout with multiple sections
 3. Generate a complex form with various input types
 4. Build a responsive grid system using semantic HTML
-

Tips for Learning Emmet

1. **Start Simple:** Begin with basic abbreviations like `div`, `p`, `h1`
2. **Practice Daily:** Use Emmet for all your HTML writing
3. **Learn Incrementally:** Master one concept before moving to the next
4. **Use in Real Projects:** Apply Emmet shortcuts in actual development
5. **Customize:** Create your own snippets for frequently used patterns
6. **Experiment:** Try combining different operators to see what you can create

Remember: The key to mastering Emmet is practice. Start with simple shortcuts and gradually work your way up to more complex abbreviations. Soon you'll be writing HTML at lightning speed!

This guide covers the most useful Emmet shortcuts for HTML development. For CSS Emmet shortcuts and advanced features, check the official Emmet documentation at emmet.io.