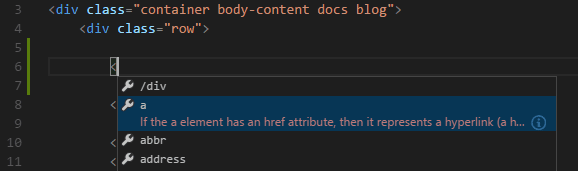
**HTML in Visual Studio Code**

Visual Studio Code provides basic support for HTML programming out of the box. There is syntax highlighting, smart completions with IntelliSense, and customizable formatting. VS Code also includes great Emmet support.

[**IntelliSense**](https://code.visualstudio.com/docs/languages/html#_intellisense)

As you type in HTML, we offer suggestions via HTML IntelliSense. In the image below, you can see a suggested HTML element closure </div> as well as a context specific list of suggested elements.



Document symbols are also available for HTML, allowing you to quickly navigate to DOM nodes by id and class name.

You can also work with embedded CSS and JavaScript. However, note that script and style includes from other files are not followed, the language support only looks at the content of the HTML file.

You can trigger suggestions at any time by pressing Ctrl+Space.

You can also control which built-in code completion providers are active. Override these in your user or workspace [settings](https://code.visualstudio.com/docs/configure/settings) if you prefer not to see the corresponding suggestions.

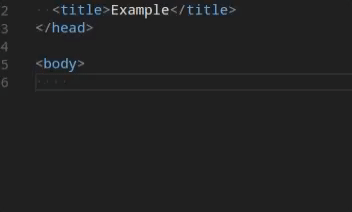
// Configures if the built-in HTML language suggests HTML5 tags, properties and values.

"html.suggest.html5": true

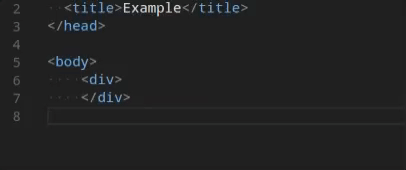
Copy

[**Close tags**](https://code.visualstudio.com/docs/languages/html#_close-tags)

Tag elements are automatically closed when > of the opening tag is typed.



The matching closing tag is inserted when / of the closing tag is entered.



You can turn off autoclosing tags with the following [setting](https://code.visualstudio.com/docs/configure/settings):

"html.autoClosingTags": false

Copy

[**Auto update tags**](https://code.visualstudio.com/docs/languages/html#_auto-update-tags)

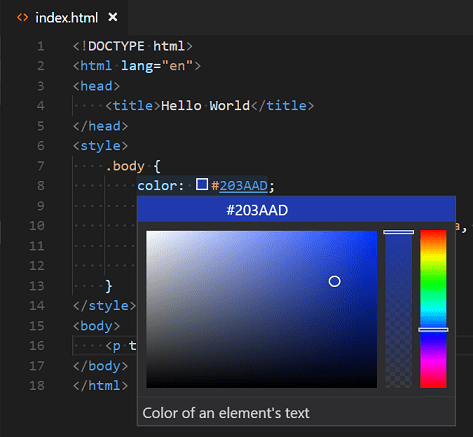
When modifying a tag, the linked editing feature automatically updates the matching closing tag. The feature is optional and can be enabled by setting:

"editor.linkedEditing": true

Copy

[**Color picker**](https://code.visualstudio.com/docs/languages/html#_color-picker)

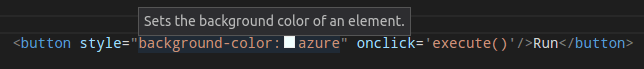
The VS Code color picker UI is now available in HTML style sections.



It supports configuration of hue, saturation and opacity for the color that is picked up from the editor. It also provides the ability to trigger between different color modes by clicking on the color string at the top of the picker. The picker appears on a hover when you are over a color definition.

[**Hover**](https://code.visualstudio.com/docs/languages/html#_hover)

Move the mouse over HTML tags or embedded styles and JavaScript to get more information on the symbol under the cursor.



[**Validation**](https://code.visualstudio.com/docs/languages/html#_validation)

The HTML language support performs validation on all embedded JavaScript and CSS.

You can turn that validation off with the following settings:

// Configures if the built-in HTML language support validates embedded scripts.

"html.validate.scripts": true,

// Configures if the built-in HTML language support validates embedded styles.

"html.validate.styles": true

Copy

[**Folding**](https://code.visualstudio.com/docs/languages/html#_folding)

You can fold regions of source code using the folding icons on the gutter between line numbers and line start. Folding regions are available for all HTML elements for multiline comments in the source code.

Additionally you can use the following region markers to define a folding region: <!-- #region --> and <!-- #endregion -->

If you prefer to switch to indentation based folding for HTML use:

"[html]": {

"editor.foldingStrategy": "indentation"

},

Copy

[**Formatting**](https://code.visualstudio.com/docs/languages/html#_formatting)

To improve the formatting of your HTML source code, you can use the **Format Document** command Shift+Alt+F to format the entire file or **Format Selection** Ctrl+K Ctrl+F to just format the selected text.

The HTML formatter is based on [js-beautify](https://www.npmjs.com/package/js-beautify" \t "_blank). The formatting options offered by that library are surfaced in the VS Code [settings](https://code.visualstudio.com/docs/configure/settings):

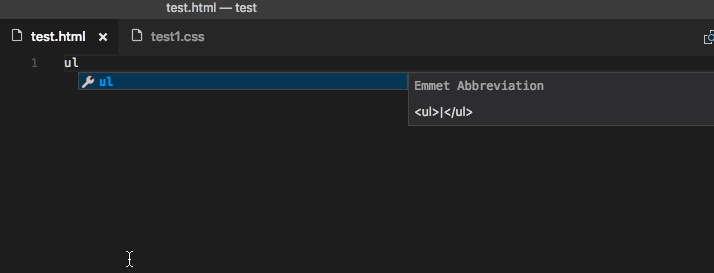
* [html.format.wrapLineLength](vscode://settings/html.format.wrapLineLength): Maximum amount of characters per line.
* [html.format.unformatted](vscode://settings/html.format.unformatted): List of tags that shouldn't be reformatted.
* [html.format.contentUnformatted](vscode://settings/html.format.contentUnformatted): List of tags, comma separated, where the content shouldn't be reformatted.
* [html.format.extraLiners](vscode://settings/html.format.extraLiners): List of tags that should have an extra newline before them.
* [html.format.preserveNewLines](vscode://settings/html.format.preserveNewLines): Whether existing line breaks before elements should be preserved.
* [html.format.maxPreserveNewLines](vscode://settings/html.format.maxPreserveNewLines): Maximum number of line breaks to be preserved in one chunk.
* [html.format.indentInnerHtml](vscode://settings/html.format.indentInnerHtml): Indent <head> and <body> sections.
* [html.format.wrapAttributes](vscode://settings/html.format.wrapAttributes): Wrapping strategy for attributes:
  + auto: Wrap when the line length is exceeded
  + force: Wrap all attributes, except first
  + force-aligned: Wrap all attributes, except first, and align attributes
  + force-expand-multiline: Wrap all attributes
  + aligned-multiple: Wrap when line length is exceeded, align attributes vertically
  + preserve: Preserve wrapping of attributes
  + preserve-aligned: Preserve wrapping of attributes but align
* [html.format.wrapAttributesIndentSize](vscode://settings/html.format.wrapAttributesIndentSize): Alignment size when using force aligned and aligned multiple in [html.format.wrapAttributes](vscode://settings/html.format.wrapAttributes) or null to use the default indent size.
* [html.format.templating](vscode://settings/html.format.templating): Honor django, erb, handlebars and php templating language tags.
* [html.format.unformattedContentDelimiter](vscode://settings/html.format.unformattedContentDelimiter): Keep text content together between this string.

**Tip:** The formatter doesn't format the tags listed in the [html.format.unformatted](vscode://settings/html.format.unformatted) and [html.format.contentUnformatted](vscode://settings/html.format.contentUnformatted) settings. Embedded JavaScript is formatted unless 'script' tags are excluded.

The Marketplace has several alternative formatters to choose from. If you want to use a different formatter, define "html.format.enable": false in your settings to turn off the built-in formatter.

[**Emmet snippets**](https://code.visualstudio.com/docs/languages/html#_emmet-snippets)

VS Code supports [Emmet snippet](https://emmet.io/) expansion. Emmet abbreviations are listed along with other suggestions and snippets in the editor auto-completion list.



**Tip:** See the HTML section of the [Emmet cheat sheet](https://docs.emmet.io/cheat-sheet) for valid abbreviations.

If you'd like to use HTML Emmet abbreviations with other languages, you can associate one of the Emmet modes (such as css, html) with other languages with the [emmet.includeLanguages](vscode://settings/emmet.includeLanguages) [setting](https://code.visualstudio.com/docs/configure/settings). The setting takes a [language identifier](https://code.visualstudio.com/docs/languages/overview#_language-identifier) and associates it with the language ID of an Emmet supported mode.

For example, to use Emmet HTML abbreviations inside JavaScript:

{

"emmet.includeLanguages": {

"javascript": "html"

}

}

Copy

We also support [User Defined Snippets](https://code.visualstudio.com/docs/editing/userdefinedsnippets).

[**HTML custom data**](https://code.visualstudio.com/docs/languages/html#_html-custom-data)

You can extend VS Code's HTML support through a declarative [custom data format](https://github.com/microsoft/vscode-html-languageservice/blob/main/docs/customData.md). By setting [html.customData](vscode://settings/html.customData) to a list of JSON files following the custom data format, you can enhance VS Code's understanding of new HTML tags, attributes and attribute values. VS Code will then offer language support such as completion & hover information for the provided tags, attributes and attribute values.

You can read more about using custom data in the [vscode-custom-data](https://github.com/microsoft/vscode-custom-data" \t "_blank) repository.

[**HTML extensions**](https://code.visualstudio.com/docs/languages/html#_html-extensions)

Install an extension to add more functionality. Go to the **Extensions** view (Ctrl+Shift+X) and type 'html' to see a list of relevant extensions to help with creating and editing HTML.

[[](https://marketplace.visualstudio.com/items?itemName=ecmel.vscode-html-css)](https://marketplace.visualstudio.com/items?itemName=ecmel.vscode-html-css" \t "_blank)

**[HTML CSS Support](https://marketplace.visualstudio.com/items?itemName=ecmel.vscode-html-css" \t "_blank)**

[25.3M](https://marketplace.visualstudio.com/items?itemName=ecmel.vscode-html-css" \t "_blank)

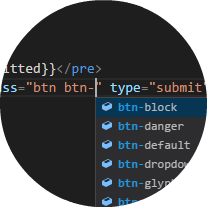
[ecmel](https://marketplace.visualstudio.com/items?itemName=ecmel.vscode-html-css" \t "_blank)

[[](https://marketplace.visualstudio.com/items?itemName=abusaidm.html-snippets)](https://marketplace.visualstudio.com/items?itemName=abusaidm.html-snippets" \t "_blank)

**[HTML Snippets](https://marketplace.visualstudio.com/items?itemName=abusaidm.html-snippets" \t "_blank)**

[11.2M](https://marketplace.visualstudio.com/items?itemName=abusaidm.html-snippets" \t "_blank)

[abusaidm](https://marketplace.visualstudio.com/items?itemName=abusaidm.html-snippets" \t "_blank)

[[](https://marketplace.visualstudio.com/items?itemName=Zignd.html-css-class-completion)](https://marketplace.visualstudio.com/items?itemName=Zignd.html-css-class-completion" \t "_blank)

**[IntelliSense for CSS class names in HTML](https://marketplace.visualstudio.com/items?itemName=Zignd.html-css-class-completion" \t "_blank)**

[8.9M](https://marketplace.visualstudio.com/items?itemName=Zignd.html-css-class-completion" \t "_blank)

[Zignd](https://marketplace.visualstudio.com/items?itemName=Zignd.html-css-class-completion" \t "_blank)

[[](https://marketplace.visualstudio.com/items?itemName=mkaufman.HTMLHint)](https://marketplace.visualstudio.com/items?itemName=mkaufman.HTMLHint" \t "_blank)

**[HTMLHint](https://marketplace.visualstudio.com/items?itemName=mkaufman.HTMLHint" \t "_blank)**

[1.2M](https://marketplace.visualstudio.com/items?itemName=mkaufman.HTMLHint" \t "_blank)

[mkaufman](https://marketplace.visualstudio.com/items?itemName=mkaufman.HTMLHint" \t "_blank)

Tip: Click on an extension tile above to read the description and reviews to decide which extension is best for you. See more in the [Marketplace](https://marketplace.visualstudio.com/).

[**Next steps**](https://code.visualstudio.com/docs/languages/html#_next-steps)

Read on to find out about:

* [CSS, SCSS, and Less](https://code.visualstudio.com/docs/languages/css) - VS Code has first class support for CSS including Less and SCSS.
* [Emmet](https://code.visualstudio.com/docs/languages/emmet) - Learn about VS Code's powerful built-in Emmet support.
* [Emmet official documentation](https://docs.emmet.io/) - Emmet, the essential toolkit for web-developers.

[**Common questions**](https://code.visualstudio.com/docs/languages/html#_common-questions)

[**Does VS Code have HTML preview?**](https://code.visualstudio.com/docs/languages/html#_does-vs-code-have-html-preview)

No, VS Code doesn't have built-in support for HTML preview but there are extensions available in the VS Code [Marketplace](https://marketplace.visualstudio.com/vscode). Open the **Extensions** view (Ctrl+Shift+X) and search on 'live preview' or 'html preview' to see a list of available HTML preview extensions.

Top of Form