Visual Studio | Marketplace

Sign in

Visual Studio Code > Programming Languages > AsciiDoc

New to Visual Studio Code? Get it now.

AsciiDoc

asciidoctor ✓ | ⤓ 222,230 installs | ★★★★⯪ (7) | Free

Provides rich language support for AsciiDoc.

Install   Trouble Installing? ⧉

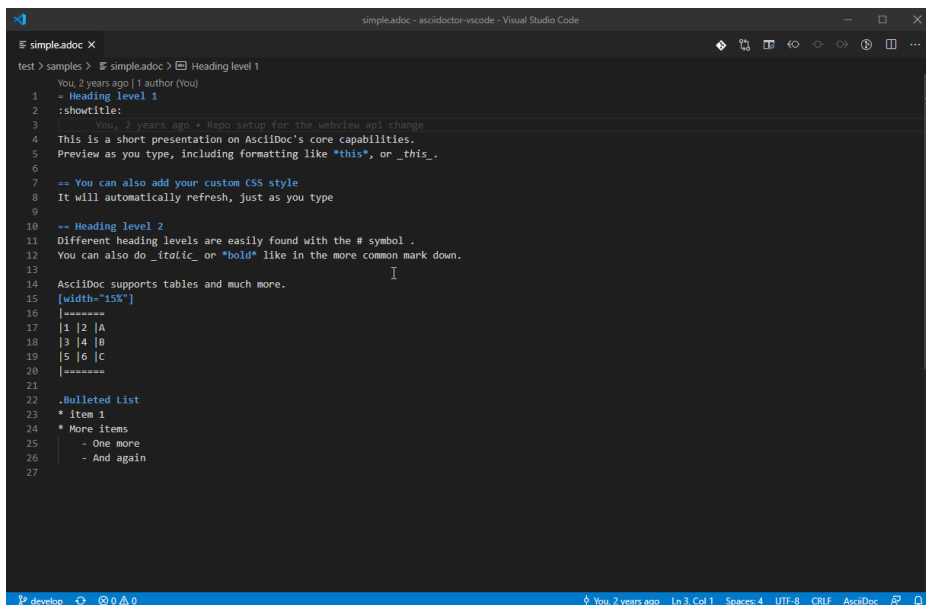Overview   Version History   Q & A   Rating & Review

AsciiDoc support for Visual Studio Code

`Visual Studio Marketplace` `v3.0.0` `installs` `222226` `rating` `average: 4.29/5 (7 ratings)` `zulip` `join chat`

An extension that provides live preview, syntax highlighting and snippets for the AsciiDoc format using Asciidoctor.

How to Install

Launch Visual Studio Code "Quick Open" (`Ctrl+P`), paste the following command, and press `Enter`:

```
ext install asciidoctor.asciidoctor-vscode
```

Alternatively, you can use the built-in extension browser to find the *AsciiDoc* by *asciidoctor* extension and install it.

This extension is also available as a pre-version (alpha) in Visual Studio Code for the Web and can be installed using the same procedure.

Feature	Desktop	Web
Document Outline and Symbols	✓	✓
Equations (via Mathjax)	✓	✓ (requires security to be disabled)
Export as PDF	✓	⊘
Kroki Integration for Diagrams	✓	✓
Paste Image	✓	⊘

Categories

Programming Languages

Tags

asciidoc   Asciidoctor   keybindings   snippet

Works with

Universal, Web

Resources

Issues

Repository

Homepage

License

Changelog

Download Extension

Project Details

⊙ asciidoctor/asciidoctor-vscode

🕐 Last Commit: 3 days ago

⑃ 3 Pull Requests

❗ 118 Open Issues

More Info

Version	3.0.0
Released on	6/27/2020, 2:54:06 AM
Last updated	7/4/2022, 1:06:15 PM
Publisher	asciidoctor
Unique Identifier	asciidoctor.asciidoctor-vscode
Report	Report Abuse

Feature	Desktop	Web
Save as HTML	✔	⛔
Save as DocBook	✔	⛔
Snippets	✔	✔
Syntax Highlighting	✔	✔ (requires security to be disabled)
Sync scrolling between the editor and the preview	✔	✔

How to Use

The extension activates automatically when opening an AsciiDoc file (.adoc, .ad, .asc, .asciidoc).

Preview

To show the preview you can use the same commands as the Markdown extension:

- Toggle Preview - `ctrl+shift+v` (Mac: `cmd+shift+v`)
- Open Preview to the Side - `ctrl+k v` (Mac: `cmd+k v`)

The extension updates automatically the preview but it can also be forced with the *AsciiDoc: Refresh Preview* command.

The preview supports setting AsciiDoc attributes through the `asciidoc.preview.asciidoctorAttributes` setting.

By default, the preview uses VS Code editor theme (`workbench.colorTheme`). To use Asciidoctor default style set the `asciidoc.preview.useEditorStyle` setting to `false`. It is also possible to set your own preview stylesheet with the `asciidoc.preview.style` setting.
(See more details under [Extension Settings](#))

Export as PDF

The extension provides a quick command to export your AsciiDoc file as PDF.

- Open the command palette - `ctrl+shift+p` or `F1` (Mac: `cmd+shift+p`)
- Select *AsciiDoc: Export document as PDF*
- Choose the folder and filename for the generated PDF

By default, the PDF export feature relies on `asciidoctor-pdf`. If you prefer to use `wkhtmltopdf`, set the `asciidoc.pdf.engine` setting to `wkhtmltopdf` and configure the `asciidoc.pdf.wkhtmltopdfCommandPath` if necessary.
(See more details under [Extension Settings](#))

Save as HTML

The extension provides a quick command to export your AsciiDoc file as HTML using the default Asciidoctor stylesheet.

- Open the command palette - `ctrl+shift+p` or `F1` (Mac: `cmd+shift+p`)
- Select *AsciiDoc: Save HTML document*
- The file is generated in the same folder as the source document

The shortcout key of `ctrl+alt+s` (Mac: `cmd+alt+s`) will also save the document.

Save to Docbook

The extension provides a quick command to export your AsciiDoc file as DocBook.

- Open the command palette - `ctrl+shift+p` or `F1` (Mac: `cmd+shift+p`)
- Select *AsciiDoc: Save to DocBook*
- The file is generated in the same folder as the source document

ℹ **Note:** Only DocBook 5 is supported.

Snippets

Several code snippets are provided including but not limited to: include statements, images, links, header, headings, lists, blocks, etc...

For a full list open the command palette and select *Insert Snippet*.

Identifying the VS Code Environment

The `env-vscode` attribute is set on all output documents. If you need to identify or handle the VS Code environment you can use an `ifdef` expression similar to the following:

```
ifdef::env-vscode[]
This is for vscode only
endif::[]
```

Diagram Integration

This extension supports a wide range of diagrams from BPMN to Graphviz to PlantUML and Vega graphs using Kroki and asciidoctor-kroki.

You can see the full range on the Kroki website.

Note that this extension will send graph information to https://kroki.io. If this is an issue it is also possible to use your own Kroki instance (see the instructions for further information).

To enable diagram support, set the `asciidoc.extensions.enableKroki` setting to `true`.

To cache and save diagrams locally set the `:kroki-fetch-diagram:` AsciiDoc attribute in your document header:

```
= My Amazing Document
:kroki-fetch-diagram:
```

This will store images by default in your document folder, however you may also set `imagesdir` to store them elsewhere:
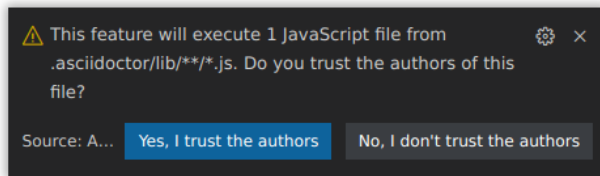
```
= My Amazing Document
:kroki-fetch-diagram:
:imagesdir: media
```

Use Asciidoctor.js extensions

When using the preview, the VS Code extension can load and register Asciidoctor.js extensions.

By convention, extensions must be located in `.asciidoctor/lib` (at the root of your workspace). The VS Code extension will recursively load all files with the extension `.js` as Asciidoctor.js extensions. For instance, the following files will be loaded: `.asciidoctor/lib/emoji.js`, `.asciidoctor/lib/emoji/index.js` and `.asciidoctor/lib/foo/bar/baz.js`.

To use an Asciidoctor.js extension, you should enable the feature by checking "Enable Asciidoctor.js extensions registration" in the extension settings. The first time, you will also need to confirm that you trust the authors of the Asciidoctor.js extensions located in *.asciidoctor/lib*.

❗ **Important:** This feature will execute JavaScript code and should not be enabled if you don't fully trust the authors of the Asciidoctor.js extensions.

💡 **Tip:** You can always update the trust mode using the command "Manage Asciidoctor.js Extensions Trust Mode".

You can create a new extension by creating a JavaScript file in the `.asciidoctor/lib` directory or use an existing one.
Here's an example of how to use the asciidoctor-emoji extension:

1. Install the npm package in the workspace directory:

```
npm i asciidoctor-emoji
```

2. Create a file a JavaScript file in *.asciidoctor/lib* with the following content:

```
module.exports = require('asciidoctor-emoji')
```

3. Enjoy :tada:

Extension Settings

This extension contributes the following settings:

Preview

Name	Description	Default Value
`asciidoc.preview.asciidoctorAttribute`	Asciidoctor attributes used in the preview (object of `{string: string}`).	`{}`
`asciidoc.preview.refreshInterval`	Interval in miliseconds between two consecutive updates of the preview. The value 0 means it will only update the preview on save.	`2000`
`asciidoc.preview.style`	An URL or a local path to CSS style sheets to use from the preview.	
`asciidoc.preview.useEditorStyle`	Use VS Code editor style instead of the default Asciidoctor style.	
`asciidoc.preview.fontFamily`	Control the font family used in the preview.	`"-apple-syst BlinkMacSyst 'Segoe WPC', UI', 'HelveticaNe Light', 'Ubu 'Droid Sans serif"`
`asciidoc.preview.fontSize`	Control the font size in pixels used in the preview.	`14`
`asciidoc.preview.lineHeight`	Control the line height used in the preview.	`1.6`

Name	Description	Default Value
`asciidoc.preview.scrollPreviewWithEditor`	When the preview is scrolled, update the view of the editor.	`true`
`asciidoc.preview.scrollEditorWithPreview`	When the editor is scrolled, update the view of the preview.	`true`
`asciidoc.preview.markEditorSelection`	Mark the current editor selection in the preview.	`true`
`asciidoc.preview.doubleClickToSwitchToEditor`	Double click in the preview to switch to the editor.	`true`
`asciidoc.preview.openLinksToAsciidocFiles`	Control how links to other AsciiDoc files in the preview should be opened. Possible values: `"inPreview"`, `"inEditor"`.	`"inPreview"`

PDF

Name	Description	Def
`asciidoc.pdf.engine`	Control the PDF engine used to export as PDF. Possible values: `"asciidoctor-pdf"`, `"wkhtmltopdf"`.	`"a` `pd`
`asciidoc.pdf.asciidoctorPdfCommandPath`	External `asciidoctor-pdf` command to execute. It accepts a full path to the binary, for instance: `/path/to/asciidoctor-pdf`.	`"b` `as` `pd`
`asciidoc.pdf.asciidoctorPdfCommandArgs`	List of arguments, for instance: `-a`, `pdf-themesdir=resources/themes`, `-a`, `pdf-theme=basic`. Please note that the argument key and value should be added separately (i.e., two items). By default, it passes the following arguments: `--quiet` and `--base-dir` with the full directory path to the AsciiDoc document.	`[]`
`asciidoc.pdf.wkhtmltopdfCommandPath`	External `wkhtmltopdf` command to execute. It accepts a full path to the binary, for instance: `/path/to/wkhtmltopdf`. If the value is empty, use either `wkhtmltopdf` on Linux/macOS or `wkhtmltopdf.exe` on Windows.	`""`

Name	Description	Def...
`asciidoc.pdf.wkhtmltopdfCommandArgs`	List of arguments, for instance: `--orientation`, `Landscape`. Please note that the argument key and value should be added separately (i.e., two items). By default, it passes the following arguments: `--enable-local-file-access`, `--encoding`, `utf-8`, `--javascript-delay`, `1000`, `--footer-center` (if enabled) and `cover` (if it has a cover page).	`[]`

Extensions

Name	Description	Defa... Valu...
`asciidoc.extensions.enableKroki`	Enable Kroki extension to generate diagrams.	`fals...`
`asciidoc.extensions.registerWorkspaceExtensions`	Enables Asciidoctor.js extensions registration from the workspace directory `.asciidoctor/lib`.	`fals...`

General

Name	Description	Default Value
`asciidoc.useWorkspaceRootAsBaseDirectory`	When in a workspace, use the workspace root path as the base directory.	`false`

Debug

Name	Description	Default Value
`asciidoc.debug.trace`	Enable debug logging for this extension. Possible values: `"off"`, `"verbose"`.	`"off"`
`asciidoc.debug.enableErrorDiagnostics`	Provide error diagnostics.	`true`

Build and Install from Source

Manual

```
git clone https://github.com/asciidoctor/asciidoctor-vscode
cd asciidoctor-vscode
npm install
npm run package
code --install-extension *.vsix
```

Issues

If you encounter any problems with the extension and cannot find the solution yourself, please open an issue in the dedicated GitHub page: asciidoctor-vscode/issues.

Before opening an issue, please make sure that it is not a duplicate. Your problem may have already been brought up by another user and been solved: asciidoctor-vscode/issues all.

When you do open an issue, remember to include the following information:

1. Description of the issue
2. VSCode version, OS (*Help -> About*) and extension version
3. Steps to reproduce the issue
   IMPORTANT: We cannot solve the issue if you do not explain how you encountered it
4. If the problem occurs only with a specific file, attach it, together with any screenshot that might better show what the issue is.

If your issue only appeared after updating to a new version of the extension, you can roll back to a previous one via the extensions browser. Click on the small gear icon beside the AsciiDoc extension, then select *Install Another Version...*. A selection menu will appear allowing you to select which version you want to install.

Contributing

To contribute simply clone the repository and then commit your changes. When you do a pull request please clearly highlight what you changed in the pull comment.

Do not update the extension version or changelog, it will be done by the maintainers when a new version is released.

If you want to update the readme, you are free to fix typos, errors, and add or improve descriptions; but, if you have a style change in mind please use an issue (or specific pull request) so that it can be discussed.

Credits

- AsciiDoc by Stuart Rackham
- Asciidoctor organization for the language flavor
- Asciidoctor.js for the preview
- Asciidoctor PDF for the *Export to PDF* function
- wkhtmltopdf for the *Export to PDF* function
- Each and every contributor to this extension.