

Setup

- Clone [microsoft/vscode](#)
- Run `yarn` at `/`, this will install
 - Dependencies for `/extension/html-language-features/`
 - Dependencies for `/extension/html-language-features/server/`
 - devDependencies such as `gulp`
- Open `/extensions/html-language-features/` as the workspace in VS Code
- In `/extensions/html-language-features/` run `yarn compile` (or `yarn watch`) to build the client and server
- Run the [Launch Extension](#) debug target in the Debug View. This will:
 - Launch a new VS Code instance with the `html-language-features` extension loaded
- Open a `.html` file to activate the extension. The extension will start the HTML language server process.
- Add `"html.trace.server": "verbose"` to the settings to observe the communication between client and server in the `HTML Language Server` output.
- Debug the extension and the language server client by setting breakpoints in `html-language-features/client/`
- Debug the language server process by using `Attach to Node Process` command in the VS Code window opened on `html-language-features`.
 - Pick the process that contains `htmlServerMain` in the command line. Hover over `code-insiders` resp `code` processes to see the full process command line.
 - Set breakpoints in `html-language-features/server/`
- Run `Reload Window` command in the launched instance to reload the extension

Contribute to `vscode-html-languageservice`

[microsoft/vscode-html-languageservice](#) contains the language smarts for html. This extension wraps the html language service into a Language Server for VS Code. If you want to fix html issues or make improvements, you should make changes at [microsoft/vscode-html-languageservice](#).

However, within this extension, you can run a development version of `vscode-html-languageservice` to debug code or test language features interactively:

Linking `vscode-html-languageservice` in `html-language-features/server/`

- Clone [microsoft/vscode-html-languageservice](#)
- Run `yarn` in `vscode-html-languageservice`
- Run `yarn link` in `vscode-html-languageservice`. This will compile and link `vscode-html-languageservice`
- In `html-language-features/server/`, run `yarn link vscode-html-languageservice`

Testing the development version of `vscode-html-languageservice`

- Open both `vscode-html-languageservice` and this extension in two windows or with a single window with the [multi-root workspace](#) feature
- Run `yarn watch` at `html-language-features/server/` to recompile this extension with the linked version of `vscode-html-languageservice`
- Make some changes in `vscode-html-languageservice`
- Now when you run `Launch Extension` debug target, the launched instance will use your development version of `vscode-html-languageservice`. You can interactively test the language features.