

THE BEST VISUAL STUDIO CODE EXTENSIONS TO MAKE THE VS CODE EXPERIENCE EVEN BETTER

Top Visual Studio Code Extensions: 50 Powerful Tools

ALEXANDRA ALTVATER | APRIL 18, 2017 | [DEVELOPER TIPS, TRICKS & RESOURCES](#)

Microsoft's Visual Studio Code provides developers with a robust marketplace packed with useful tools and extensions that make VS Code even more functional and customizable to meet every developer's needs. New extensions are added regularly, and you can find tools, libraries of snippets, and just about anything else you could want to streamline the development process. In fact, Stackify Prefix has its own [code extension](#) in the Marketplace — the easiest and most powerful .NET profiler you'll find.

Extensions are a vital part of the Visual Studio Code experience. We decided to take a little tour of the other offerings in the Marketplace to come up with a list of 50 most helpful, must-try extensions (in addition to Stackify Prefix, which we, of course, consider a must-have) for any Visual Studio Code enthusiast. All of the following 50 extensions are free, although some require you to have other programs or services installed. We hope you'll find some great ideas in this list of tools.

Try Stackify's free code profiler, [Prefix](#), to write better code on your workstation. Prefix works with .NET, Java, PHP, Node.js, Ruby, and Python.

By the way, a few of these extensions are TypeScript related. If you aren't familiar with it, check out our article about [TypeScript vs JavaScript](#).

1. [TSLint](#)

Configuration options

- `tslint.enable` - enable/disable tslint.
- `tslint.jsEnable` - enable/disable tslint for js files. default is `false`.
- `tslint.run` - run the linter onSave or onType. default is `onType`.
- `tslint.rulesDirectory` - an additional rules directory, for user-created rules.
- `tslint.configFile` - the configuration file that tslint should use instead of the default `tslint.json`.
- `tslint.ignoreDefinitionFiles` - control if TypeScript definition files should be ignored.
- `tslint.exclude` - configure glob patterns of file paths to exclude from linting.
- `tslint.validateWithDefaultConfig` - validate a file for which there was no custom tslint configuration found. The default is `false`.
- `tslint.nodePath` - use this setting load tslint from a different location than the current workspace or the globally installed npm modules'.
- `tslint.autoFixOnSave` - fix auto fixable warnings when a file is saved. This option is ignored when `files.autoSave` is set to `afterDelay`.

If you're doubtful about your TypeScript quality, verify it with this Visual Studio code extension, which integrates the tslint linter for TypeScript language into VS Code.

Key Features:

Search Blog

Search

Topics/Keywords

ASP.NET	Product Updates
.NET Core	App Monitoring
Java	App Performance Tips
Azure	Error Handling
AWS	Logging Tips
Cloud	DevOps

Popular Posts

	ASP.NET Performance: 9 Types of Tools You Need to Know!
	How to Troubleshoot IIS Worker Process (w3wp) High...
	How to Monitor IIS Performance: From the Basic...
	SQL Performance Tuning: 7 Practical Tips for Developers
	Looking for New Relic Alternatives & Competitors?...

Recent Posts

	Netreo Launches Prefix Premium Real-Time...
	Node.js Performance Monitoring
	What to Do About Java Memory Leaks: Tools, Fixes,...
	The Top 5 Node.js Performance Measurement...
	How to Catch All Exceptions in C# & Find All Application Errors



Download eBooks

Find the best software development resources.

[Want to write for us?](#)

- Checks the quality of your static typescript code
- Analyzes the whole project or a particular section of your code
- Allows you to auto fix errors when scanning lines of code
- Supports automated fixing of warnings
- Requires tslint (installed globally or locally)

2. Auto Import



Auto Import

Automatically finds, parses and provides code actions and code completion for all available imports. Works with TypeScript and TSX.

A Visual Studio code extension that automatically finds parses and provides you with code actions and code completions for every import you have, Auto Import works with TypeScript and TSX.

Key Features:

- Easily scans code in large projects
- Import status bar show you the number of imports available
- Merge imports from the same location

3. Docker Support

Generating Dockerfile, docker-compose.yml, and docker-compose.debug.yml

Docker is one of the hottest trends in development today, and if you use Visual Studio Code, you don't want to miss out. This is the official Docker extension to help you create dockerfiles and add syntax highlights, commands, and snippets to them.

Key Features:

- Makes creating and launching containerized applications easy
- Automatically generates dockerfile and docker-compose.yml
- Dockerfile and docker-compose commands are built into the command palette
- Syntax highlighting and hover tips
- Snippets and linting for dockerfile files

4. Visual Studio Keymap



```

72 { "key": "View Focus into Left Hand Editor", "command": "viewFocusIntoLeftHandEditor", "when": "editorHasFocus", "args": "left" }
73 { "key": "View Focus into Next Editor on the Left", "command": "viewFocusIntoNextEditorOnTheLeft", "when": "editorHasFocus", "args": "left" }
74 { "key": "View Focus into Next Editor on the Right", "command": "viewFocusIntoNextEditorOnTheRight", "when": "editorHasFocus", "args": "right" }
75 { "key": "View Focus into Right Hand Editor", "command": "viewFocusIntoRightHandEditor", "when": "editorHasFocus", "args": "right" }
76 { "key": "View Minimize Other Editors", "command": "viewMinimizeOtherEditors", "when": "editorHasFocus" }
77 { "key": "ctrl+o", "command": "openFile", "when": "editorHasFocus", "args": "openFile" }
78 { "key": "ctrl+y", "command": "readFile", "when": "editorTextFocus", "args": "y" }
79 { "key": "tab", "command": "tab", "when": "editorTextFocus && editorTabMovesFocus" }
80 { "key": "ctrl+z", "command": "undo", "when": "editorTextFocus" }
81 { "key": "ctrl+shift+z", "command": "redo", "when": "editorTextFocus" }
82 { "key": "ctrl+u", "command": "acceptSelectedSuggestion", "when": "editorTextFocus & suggestWidgetVisible && edit" }
83 { "key": "ctrl+shift+u", "command": "acceptSelectedSuggestion", "when": "editorTextFocus & suggestWidgetVisible && edit" }
84 { "key": "-", "command": "acceptSelectedSuggestion", "when": "editorTextFocus & suggestWidgetVisible && edit" }
85 { "key": "ctrl+f1", "command": "extension.sayHello", "when": "editorTextFocus" }
86 { "key": "ctrl+l", "command": "removeSelectionCursors", "when": "editorTextFocus & editorHasMultipleSelections" }
87 { "key": "escape", "command": "cancelEdit", "when": "editorTextFocus" }
88 { "key": "ctrl+shift+r", "command": "o.execute", "when": "editorTextFocus" }
89 { "key": "ctrl+r", "command": "o.executeLastCommand", "when": "editorTextFocus" }
90 { "key": "shift+b", "command": "acceptSelectedSuggestion", "when": "editorTextFocus & suggestWidgetVisible && edit" }
91 { "key": "shift+f", "command": "acceptSelectedSuggestion", "when": "editorTextFocus & suggestWidgetVisible && edit" }
92 { "key": "shift+g", "command": "acceptSelectedSuggestion", "when": "editorTextFocus & suggestWidgetVisible && edit" }
93 { "key": "shift+9", "command": "acceptSelectedSuggestion", "when": "editorTextFocus & suggestWidgetVisible && edit" }
94 { "key": "ctrl+shift+9", "command": "acceptSelectedSuggestion", "when": "editorTextFocus & suggestWidgetVisible && edit" }
95 { "key": "ctrl+shift+0", "command": "acceptSelectedSuggestion", "when": "editorTextFocus & suggestWidgetVisible && edit" }
96 { "key": "ctrl+shift+1", "command": "acceptSelectedSuggestion", "when": "editorTextFocus & suggestWidgetVisible && edit" }

```

Keyboard shortcuts don't just save time but enable you to maintain your flow as you code. They're very popular with Visual Studio users, and this extension makes it possible to use your favorite VS keyboard shortcuts in VS Code.

Key Features:

- Provides the traditional Visual Studio keyboard shortcuts
- Easy to add your custom shortcuts
- Brings shortcuts of Windows, Linux, and Mac to Visual Studio code
- Find all keyboard shortcuts in the extension's contribution list

5. EditorConfig for VS Code

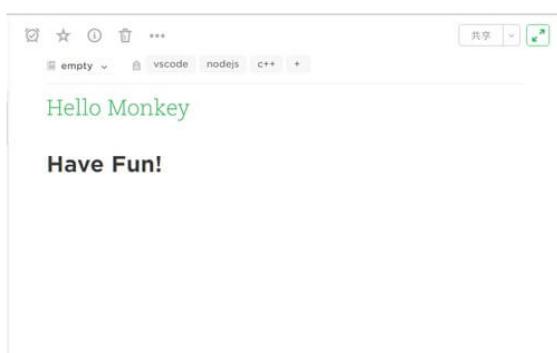


Maintain a consistent coding style with this Visual Studio code extension. Even if your teams use multiple IDEs and editors, consistency is achievable with EditorConfig for VS Code. Internally, EditorConfig uses the [editorconfig npm package](#), one of the few EditorConfig cores available. Check out the main EditorConfig site [here](#).

Key Features:

- Official release by EditorConfig for Visual Studio
- Define styles for indenting, tabs, and where lines begin and end
- Text editor plugins for editors to read and follow defined style rules
- Uses editorconfig npm package

6. Evermonkey



Evernote is one of the most popular note taking apps. It can be used to improve the quality and efficiency of your coding with the Evermonkey extension. This handy Visual Studio Code extension enables Markdown support for Evernote, following the [same principles as Sublime Text but with a faster experience](#).

Key Features:

- Provides support for Evernote in Visual Studio Code
- It caches data locally to avoid repetitive net requests for a faster experience
- UI design updates to be added to improve viewing

7. React Redux ES6 Snippets

React and Redux ES6 Snippets

This extension for Visual Studio Code adds snippets for React and Redux using ES6 for JavaScript.

Usage

Type part of a snippet, press enter, and the snippet unfolds.

Shortcut	Description
rpc	React Presentational(Dumb) Component Template
rcc	React Only Container(Smart) Component Template
rct	React Contructor
rcm	React Lifecycle Hook: Component Did Mount
rcp	React Lifecycle Hook: Component Will Receive Props
rct	React Lifecycle Hook: Component Did Unmount
rrcc	React-Redux Container(Smart) Component Template
imp	ES6 Import with Optional Alias
nimp	ES6 Named Import
rimp	ES6 React Import

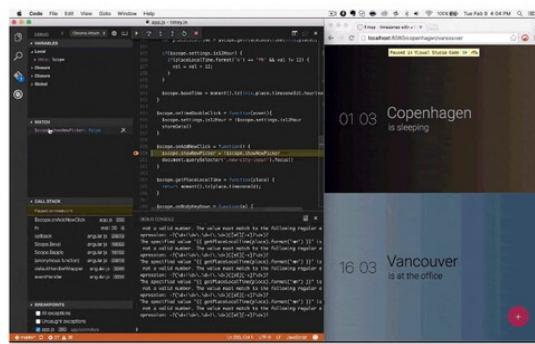
Alternatively, press Ctrl+Space to activate snippets from within the editor.

Get React and Redux snippets using ES6 for JavaScript to help you structure components with this simple, yet helpful VS Code extension.

Key Features:

- Provides snippets using JavaScript
- Quick access to snippets by pressing Ctrl+Space
- Assists with boilerplate as well

8. Debugger for Chrome



Chrome maintains the most market share among browsers, and it's widely used by developers thanks to its robust marketplace of tools and tricks for simplifying developers' lives. This VS Code extension assists you in debugging your JavaScript in Chrome, offering several helpful, time-saving features.

Key Features:

- Works with other targets that support Chrome debugging protocol
- Sets breakpoints even on source files when source maps are enabled
- Debugs script tags and eval scripts as they add added
- Locals pane, watches, and console

9. JSHint





JSHint is a great tool to quickly spot errors with your JavaScript code. If you're used to JSHint, you'll also want to use it in Visual Studio Code, and this extension makes it simple and easy. You can check out the JSHint website [here](#).

Key Features:

- Uses the default JSHint configuration as described on their site
 - All files are included in this extension by default
 - Allows you to exclude specific files and folders from the review
 - Option to disable JSHint for specified workspaces

10. C#

Working with C#

The C# support in VS Code is optimized for cross-platform .NET Core development (see [working with .NET Core](#) and [VS Code](#) for another relevant article). Our focus with VS Code is to be a great editor for cross-platform C# development. For instance, many [Unity](#) game developers enjoy using VS Code in place of the MonoDevelop IDE.

```
Program.cs ×

1  using System;
2
3  namespace DotnetBot {
4
5      public static class Program {
6
7          public static void Main(string[] args) {
8
9              string message = "";
10             if (args.Length < 1) {
11                 message = "Welcome to .NET Core!";
12             }
13             else {
14                 foreach (string item in args) {
15                     message += item;
16                 }
17             }
18         }
19     }
20 }
```

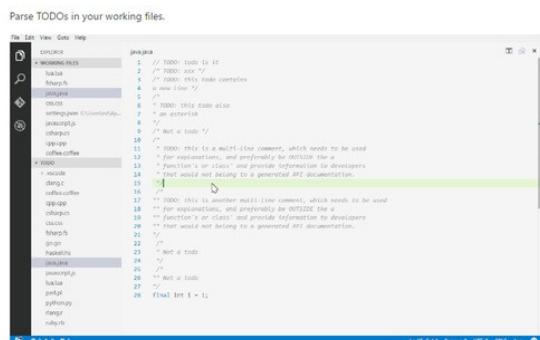
VS Code supports debugging of C# applications running on either .NET Core or Mono

Write great C# code for .NET Core with this extension, powered by OmniSharp. It even debugs your code and supports a variety of operating systems. VS Code aims to be a useful tool for cross-platform C# development, and in fact, many Unity game developers use VS Code rather than the MonoDevelop IDE, making this C# extension a particularly useful tool for many developers.

Key Features:

- Supports debugging and lightweight development tools for .NET Core
 - Editing support for C# including syntax highlighting, IntelliSense, Go to Definition, Find All References, and more
 - C# debugger available on Windows, Mac, and many Linux operating systems

11. TODO Parser



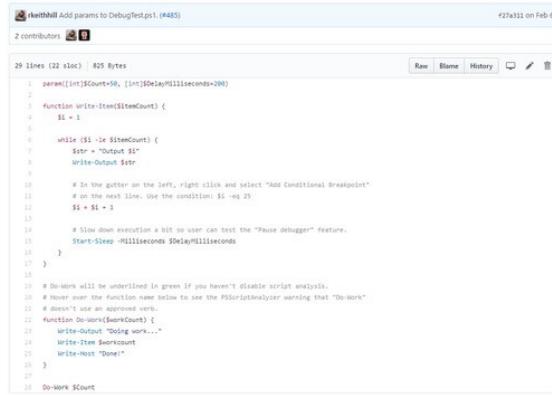
If you have trouble parsing TODOs in the working files of your project, this Visual Studio extension can help:

Studio code extension can help. It lets you parse multi-line TODOs and makes your workflow a lot more actionable.

Key Features:

- TODO counter is available in the status bar of your current document
- The counter provides list of all available TODOs
- Option to parse single and multiple TODOs at once
- Supports any programming language

12. PowerShell



A screenshot of a PowerShell script in Visual Studio Code. The status bar at the top shows '29 lines (22 sloc) | 425 Bytes'. The code itself is a script named 'DebugTest.ps1' with the following content:

```
param([int]$ItemCount, [int]$DelayMilliseconds=200)

function Write-Item($ItemCount)
{
    $i = 1

    while ($i -le $ItemCount) {
        $str = "Output $i"
        Write-Output $str
        Start-Sleep $DelayMilliseconds
        $i += 1
    }
}

# Do Work will be underlined in green if you haven't disable script analysis.
# Never ever use the function name below to see the PowerShellAnalyzer warning that "Do-Work"
# doesn't use an approved verb.
function Do-Work($ItemCount) {
    Write-Output "Doing work..."
    Write-Item $ItemCount
    Write-Host "Done."
}

Do-Work $Count
```

PowerShell is a great [object-oriented](#) programming language. If you use this programming language, this is a must-use extension offering PowerShell language support for Visual Studio Code.

Key Features:

- Write and debug PowerShell scripts in VS Code's IDE-like interface
- Works on platforms like Windows, Linux, and iOS
- Provides tools to find variables and cmdlets references

13. Angular 4 and TypeScript/HTML VS Code Snippets

How To Use the Snippets

Type one of the snippet names below (or part of it) and then press return.

TypeScript Snippets

```
ag-Bootstrap      - Bootstrap snippet
ag-AppModule     - Create the root app module (@NgModule)
ag-AppFeatureModule - Angular app feature module (@NgModule) snippet
ag-AppFeatureRoutingModule - Angular app feature routing module (@NgModule) snippet
ag-CanDeactivateRoutingGuard - Create a CanDeactivate routing guard
ag-Component     - Component snippet
ag-HttpImport    - Http module import snippet
ag-HttpMap       - Http map() snippet
ag-HttpService   - Service with http snippet
ag-InputProperty - Input property snippet
ag-DotEvent      - @Output event snippet
ag-Pipe          - Pipe snippet
ag-Routes        - Angular routes snippet
ag-Route         - Route definition snippet
ag-Service       - Service snippet
ag-Subscribe     - Observable subscribe snippet
```

Use this Visual Studio code extension to add code snippets of Angular (v2 or higher), TypeScript or HTML to your editor. It's an incredibly easy-to-use extension that streamlines coding productivity.

Key Features:

- Adds Angular version 2 and up, TypeScript or HTML code snippets
- Multiple snippets provided in the description for both TypeScript and HTML
- Instantly improve productivity in Angular 2 and up with these snippets

14. Sublime Text Keymap

This extension ports the most popular Sublime Text keyboard shortcuts to Visual Studio Code. After installing the extension and restarting VS Code your favorite keyboard shortcuts from Sublime Text are now available.

What keyboard shortcuts are included?

You can see all the keyboard shortcuts in the extension's contribution list.

Details	Contributions
Commands (34)	
Name	
workbench.files.action.closefile	⌘W
editor.action.sortLinesAscending	F5
editor.action.sortLinesDescending	⌃F5
workbench.action.previousEditor	Shift↑
workbench.action.nextEditor	Shift↓
workbench.action.files.saveAll	⌃S
workbench.action.toggleSidebarVisibility	⌘K ⌘B

Sublime Text is an excellent text editor for coding. If you are used to Sublime Text keyboard shortcuts and want to use them in VS, this extension brings the most popular Sublime Text keyboard shortcuts to Visual Studio code.

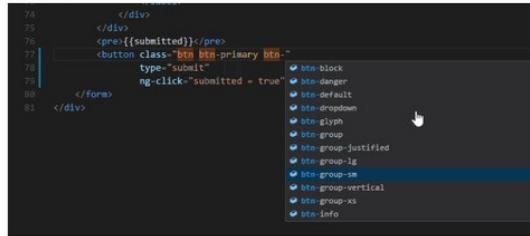
Key Features:

- Extension officially released by Microsoft to import keyboard shortcuts
- Imports the popular Sublime Text keyboards shortcuts used in programming
- Brings keyboard shortcuts used on Windows, Linux and Mac

15. HTML CSS Class Completion

HTML CSS Class Completion

A Visual Studio Code extension that provides CSS class name completion for the HTML class attribute based on the CSS files on your workspace.



As awesome as CSS is, it's hard to remember all CSS classes. This extension auto-completes CSS class names as you type, based on the CSS files in your workspace.

Key Features:

- Customize the directories where the extension will search for files
- Provides manual re-caching for class definition auto completion
- Provides support for PHP, Razor, and Jade

16. View in Browser

View In Browser

Extension for vscode to view a html file in a browser.

Usage

Use command or context menu in the explorer.

Command

- Open the command list (Press F1 or Ctrl + Shift + P)
- Select "View in Browser"

Keybindings

- Press Ctrl + F1

Context Menu

![[contextmenu](https://github.com/hellopao/view-in-browser/raw/master/.images/contextmenu.png)]

If you like viewing your HTML in a browser frequently for feedback as you code, this Visual Studio code extension will help you do it. It renders HTML files in your system's default browser.

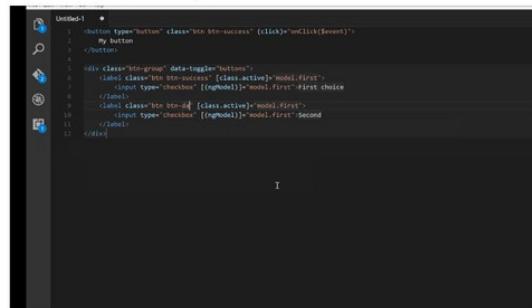
Key Features:

- Provides two options to edit, one from context menu and another from command
- Set your default browser according to your preference
- Supports Mozilla, Chrome, Internet Explorer and Safari

17. NgBootstrap Snippets

NgBootstrap Snippets

A set of NgBootstrap (Angular + Bootstrap 4) Snippets for Visual Studio Code.

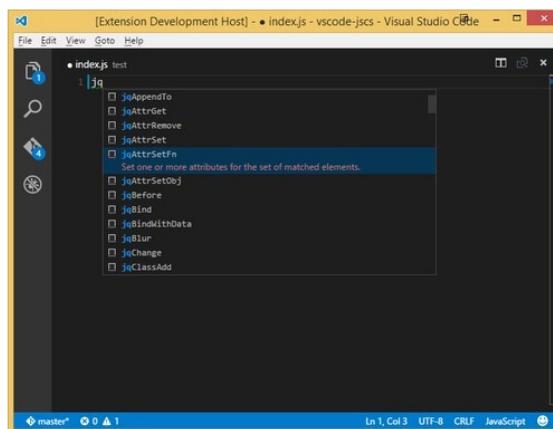


Get NgBootstrap snippets, which consist of Angular and Bootstrap 4 snippets, with this helpful extension.

Key Features:

- Typing nbg gives a quick list of the snippets available
- Unfolds snippets as soon as you press enter
- Provides alerts, dropdown, progressbar and many other snippets

18. jQuery Code Snippets



jQuery is a powerful JavaScript library. You won't remember every single function, and with this extension, you don't need to. jQuery Code Snippets brings a vast library of over 130 jQuery Code snippets to Visual Studio code.

Key Features:

- Provides a quick list of all the snippets after typing 'jq'
- Sets single or multiple attributes from the matched elements
- Snippets have been added from the 2015 Visual Studio jQuery Snippets

19. Bower



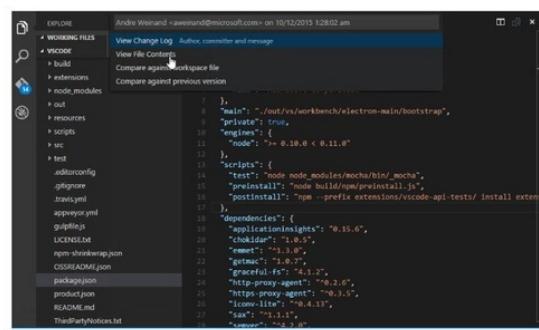


Bower makes building apps easy with its package management capabilities, providing access to most Bower commands within VS Code. With the Bower Visual Studio Code extension, you can install, uninstall, search for, and update packages effortlessly.

Key Features:

- Interactive Bower command prompts have been built in the IDE
- Write, install and uninstall bower.json and other Bower packs
- Find files according to the files you are editing

20. Git History

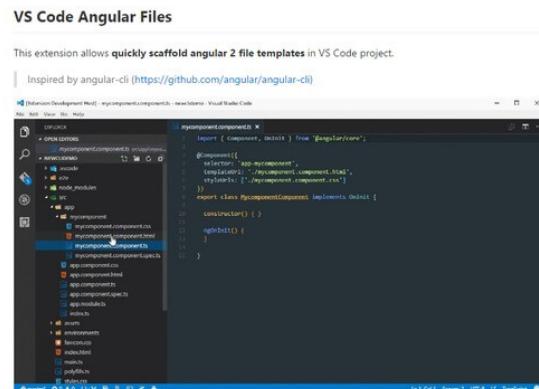


Git is the most popular version control system today. It enables distributed version control for modern web apps. Sometimes, you want to be able to re-trace your code to its previous versions and restore that version. The Git History extension gives you the complete Git history, including a history of inline files, all within Visual Studio Code.

Key Features:

- Shows history in graphs and charts according to your preference
- Get multiple history graphs to create a comparison
- Keyboard shortcuts are provided for quick and easy history viewing

21. Angular Files



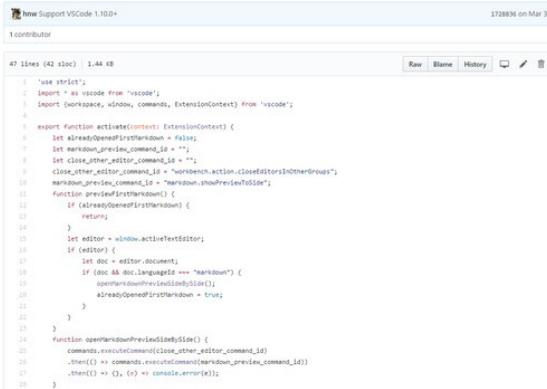
Scaffolding can be an important first step when building an app, but it can be complex. This Visual Studio extension lets you quickly scaffold Angular file templates.

Key Features:

- Supports non Angular-cli file structure

- Also supports angular-cli.json with different style extensions
- Provides good support for app structure

22. Auto-Open Markdown Preview



The screenshot shows the settings page for the 'Auto-Open Markdown Preview' extension. It contains 47 lines of code in a JSON file named 'package.json'. The code defines a function to handle markdown preview requests and another to execute commands related to closing other editors.

```

{
  "version": "0.1.0",
  "windows": {
    "command": ".\\node_modules\\.bin\\eslint"
  },
  "linux": {
    "command": "./node_modules/.bin/eslint"
  },
  "osx": {
    "command": "./node_modules/.bin/eslint"
  }
}

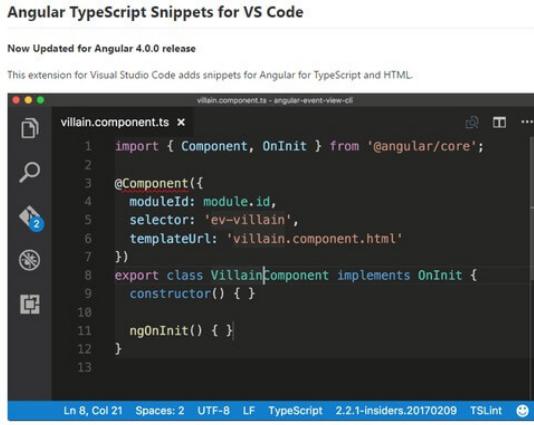
```

Previewing markdown files in a parallel window can be a pain due to the need to repeatedly type 'Ctrl + K V' or 'Cmd + KV'. With the help of this Visual Studio Code extension, you can get automatic previews of the Markdown files you open.

Key Features:

- Latest version supports Visual Studio code 1.10.0 and higher
- Removes the need to press 'Ctrl + K V' for a preview of the Markdown file
- Shows previews next to the Markdown file you open

23. Angular v4 TypeScript Snippets



The screenshot shows the code editor in VS Code with a file named 'villain.component.ts'. It contains a basic Angular component definition with imports and a constructor. The status bar at the bottom indicates the file is 'villain.component.ts - angular-event-view-cs' and shows the current line is 'Ln 8, Col 21'.

```

import { Component, OnInit } from '@angular/core';
@Component({
  moduleId: module.id,
  selector: 'ev-villain',
  templateUrl: 'villain.component.html'
})
export class VillainComponent implements OnInit {
  constructor() { }
  ngOnInit() { }
}

```

Save time when coding using the snippets you get with this Visual Studio code extension, which adds snippets for Angular for TypeScript and HTML.

Key Features:

- Works with Visual Studio 0.10.1 and higher
- Has been updated for Angular v4.0.0 release
- Keyboard shortcuts to activate snippets from within the editor

24. ESLint

Using the extension with VS Code's task running

The extension lints an individual file only. If you want to lint your entire workspace or project and want to see the warnings in the Problems panel, then you can define a VS Code task which extracts VS Code problems from the ESLint output. To do so use the following 'tasks.json' and tweak the args ('*.js') to your setup:

```
{
  "version": "0.1.0",
  "windows": {
    "command": ".\\node_modules\\.bin\\eslint"
  },
  "linux": {
    "command": "./node_modules/.bin/eslint"
  },
  "osx": {
    "command": "./node_modules/.bin/eslint"
  }
}
```

```

        "isShellCommand": true,
        "args": ["$*.jsc"],
        "showOutput": "silent",
        "problemMatcher": "$eslint-stylish"
    }
}

```

ESLint enables better code quality by bringing the ESLint library to Visual Studio Code. It also allows you to identify and act on patterns in JavaScript.

Key Features:

- Uses ESLint library from the opened workspace folder
- [Documentation](#) for ESLint is provided for new ESLint users
- Searches global version of the file if the library doesn't have one

25. Bracket Pair Colorizer

Bracket Pair Colorizer

This extension allows matching brackets to be identified with colours. The user can define which characters to match, and which colours to use.

Screenshot:

```

/**
 * Returns a text line denoted by the line number. Note
 * that the returned object is *not* live and changes to the
 * document are not reflected.
 *
 * @param line A line number in [0, lineCount).
 * @return A [line] (#TextLine).
 */
lineAt(line: number): TextLine;

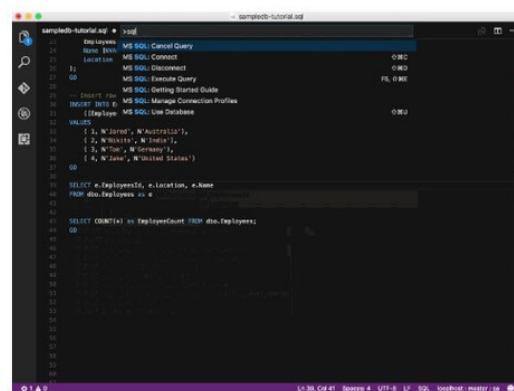
```

If different brackets confuse you, this Visual Studio code extension will help you match brackets with color-coding options. You can define which characters to match as well as preferred colors.

Key Features:

- Allows custom characters in custom brackets to be configured
- List of various colors available even for orphaned brackets
- New brackets can be added or removed from the list

26. MsSQL

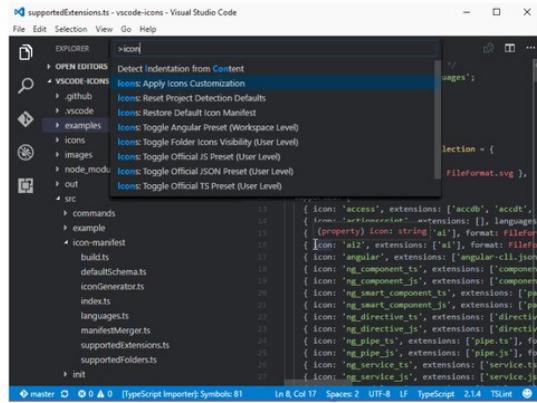


MySQL is a very helpful database tool. This extension helps you develop and connect various systems to the database you use, which helps your team access data more efficiently.

Key Features:

- Customizable command shortcuts and extension options
- Allows you to create and manage various connection profiles
- Connects to SQL Data warehouse, Azure SQL Database, and Microsoft SQL Server
- Evaluate scripts and view results in a grid
- Save results to JSON or .csv file format to see in the editor

27. VScode-Icons

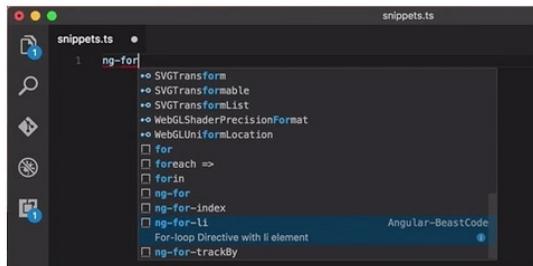


If the simplistic Visual Studio does not appeal to you, this extension adds icons to VS Code, making it more visually appealing and fun to work in. This extension was originally created to enable icons before VS Code began supporting them for files and folders. Now that VS Code does support icons, VScode-Icons aims to provide the most comprehensive set of icons available.

Key Features:

- Provides a complete set of icons to choose from in VS Code
- User can make icons look different and change the extension association at will
- Allows suggestions to add to the growing icon list

28. Angular 2+ Snippets



For programmers using the new Angular 2+ and need a supporting snippet extension, this Visual Studio extension is the one you need.

Key Features:

- Provides about 89 Angular 2+ snippets (and growing)
- Snippets are updated for versions 2.4.11 and 4.0.0
- Snippets of TypeScript, HTML, ngRx and code examples are available
- Suggests Angular 2 solutions when Angular 1 keywords are entered

29. TypeScript Hero

Commands		
Command	Extension part	Description
showCmdGui	general	Shows a small gui with all available internal commands
resolve.addImport	import resolver	Shows a pick list with all recognized, importable symbols
resolve.addImportUnderCursor	import resolver	Imports the symbol under the cursor
resolve.addMissingImports	import resolver	Imports all missing symbols for the actual document
resolve.organizeImports	import resolver	Removes unused imports and orders all imports
resolve.rebuildCache	import resolver	Rebuilds the whole symbol cache (or index)
restartDebugger.toggle	debug restarter	Toggles the active state of the debug restarter

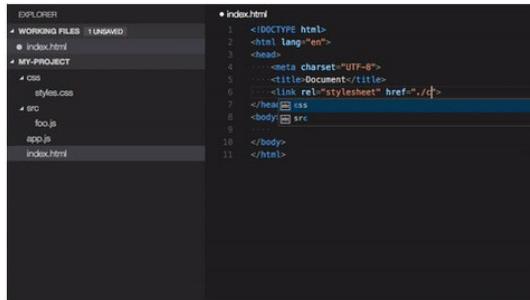
TypeScript lets you check and refactor JavaScript code as you type. TypeScript Hero brings this powerful tool to Visual Studio Code, automatically importing your project

and libraries to your current file.

Key Features:

- Adds every missing import in one command
- Fixes coding errors with a feature called 'Light bulb'
- Sort and organize your imports and remove unused imports

30. Path Intellisense

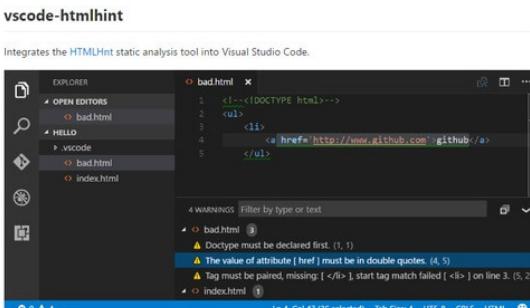


If you have a hard time remembering the path of a file, Path Intellisense can help. It auto-completes filenames as you work, saving you both time and energy.

Key Features:

- Autocomplete feature fills in the file path as you type
- Hidden files can be made visible with a simple setting
- Names with dashes are also shown in autocomplete

31. HTMLHint

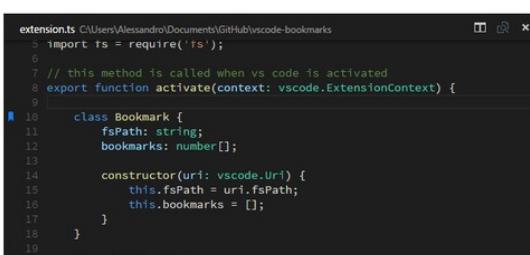


HTMLHint is a static code analysis tool that can be used with an IDE to improve the quality of code. If you're looking to use HTMLHint in Visual Studio Code, there's an extension for that.

Key Features:

- Highlights errors in the HTML file with squiggles
- Reports numbers of errors on status bar for easy viewing
- Provides options to set and define rules for checking

32. Bookmarks



```
21     bookmarks: Bookmark[];
```

Tired of endlessly scrolling through your code, or using a 'Ctrl + F' to find that exact line of code you need to fix? This Visual Studio code Bookmarks extension makes creating and jumping through code easy and smooth.

Key Features:

- Set bookmarks anywhere in your code for easy access later
- Autocomplete feature makes navigating to your bookmarks easy
- Bookmarks are saved on your current project in sessions
- Bookmarks are restored on reopening the file

33. Office UI Fabric Snippets

```
11   <div class="ms-peoplePicker">
12     <div class="ms-PeoplePicker-searchBox">
13       <input class="ms-PeoplePicker-searchField" type="text">
14     </div>
15     <div class="ms-PeoplePicker-results">
16       <div class="ms-PeoplePicker-resultGroups">
17         <!-- Insert code snippet "fabric:peoplepicker:result-group" -->
18         <div class="ms-PeoplePicker-resultGroup">
19           <div class="ms-PeoplePicker-resultGroupTitle">Top results</div>
20           <ul class="ms-PeoplePicker-resultList">
21             <!-- Insert code snippet "fabric:peoplepicker:result" -->
22             <li class="ms-PeoplePicker-result">
23               <div tabindex="0" role="button" class="ms-PeoplePicker-resultBtn">
24                 <!-- Insert code snippet "fabric:person:(tiny|large)" -->
25                 <div class="ms-Person">
26                   <div class="ms-Person-imageArea">
27                     <div class="ms-Person-initials ms-Ctrl + click to follow link">
28                       
29                     </div>
30                   </div>
31                 </div>
32               </li>
33             </ul>
34           </div>
35         </div>
36       </div>
37     </div>
```

Get the Office UI Fabric Snippets support to create HTML document with this Visual Studio code extension. It has many UI elements that can be easily added to your code.

Key Features:

- Supports UI Fabric 3.0.0 Beta 2
- Lists available after typing Fabric while coding
- Provides overall change to the HTML document
- Breadcrumbs, buttons, callouts, date picker and label are some of many snippets available

34. Haskell

```
28
29
30 insertUser :: T.Text -> DB.Action IO DB.Value
31 insertUser name = DB.insert "users" ["name" =: name]
32
33 getAllUsers :: DB.Action IO [T.Text]
34 getAllUsers = do
35   res <- DB.rest <=< DB.f[show :: forall a. Show a => a -> String]
36   return $ map (T.pack . show) res
37
```

Program using Haskell causally or as an expert using this Visual Studio extension that enables support for Haskell development.

Key Features:

- Supports Haskell and Cabal (.hs and .cabal)
- Code snippets for structures and popular functions
- Provides complete all round support for Haskell developers
- Customizable according to the preference of the user

35. Angular UI Bootstrap Snippets





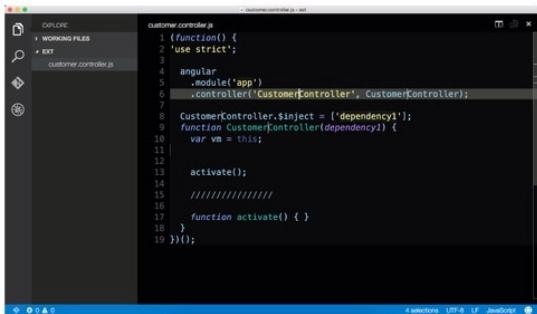
```
3   <span dir="rtl" uib-modal="true" portman>
4   </button>
5   <h4 class="modal-title">My Modal</h4>
6   </div>
7   <div class="modal-body">
8     Modal Body
9   </div>
10  <div class="modal-footer">
11    <button class="btn btn-default" type="button" ng-click="$dismiss()">
12      Cancel
13    </button>
14    <button class="btn btn-primary" type="button" ng-click="successCallback()">
15      OK
16    </button>
```

Use snippets of UI Bootstrap to code your Angular project using this Visual Studio Code extension. In HTML or JavaScript files, simply start typing uib- and Angular UI Bootstrap Snippets provides autocompletions.

Key Features:

- Autocomplete feature to make finding snippets easy
- Quickly toggle through different highlighted placeholder by pressing tab
- Can be added to any JavaScript or HTML file

36. Angular 1 JavaScript and TypeScript Snippets



You may be happy with things that just work, and don't feel the urge to keep updating to the latest version of a framework. If that sounds like you, and you're still using Angular 1, you'll love this extension. It provides snippets for Angular 1 for both JavaScript and TypeScript.

Key Features:

- Provides snippets for JavaScript and TypeScript
- Provides snippets for users of Angular 1
- Quick and easy search for snippets
- Keyboard shortcuts to streamline your workflow

37. Language Support for Java(TM) by RedHat

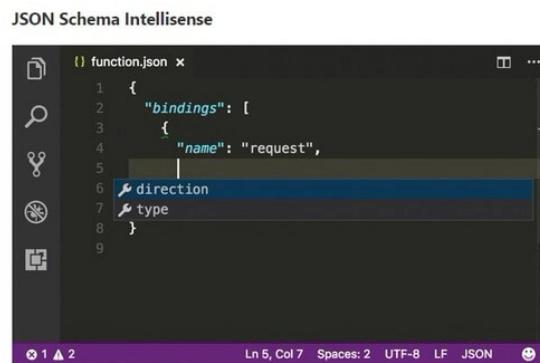


Many enterprise applications rely on Java. If you're having trouble with developing or editing in Java in Visual Studio Code, this VS code extension will make your life easier by providing language support via Eclipse JDT Language Server, which utilizes Eclipse JDT, M2Eclipse and Buildship.

Key Features:

- Provides Java support through Eclipse JDI server language
- Shows status of language tools on lower right corner
- Provides details about failures in language coding
- Supports Maven, Gradle and Eclipse
- Code completion, code navigation, highlights, formatting, snippets and more

38. Azure Functions Tools

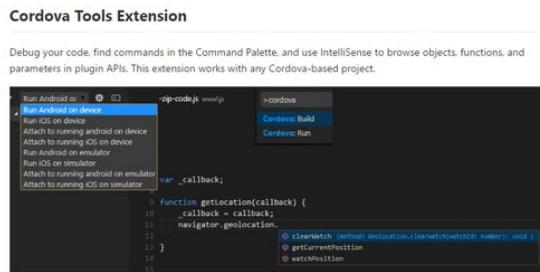


Microsoft Azure is one of the most popular cloud services used by developers and Ops teams to host and manage applications. Azure Functions is a powerful tool that lets you implement code based on event triggers, and the Azure Functions extension makes this possible in Visual Studio Code.

Key Features:

- Includes JSON IntelliSense for host and function
- Provides shortcut to quickly activate Intellisense
- Works on Visual Studio code 1.11.0 and higher

39. Cordova Tools



Apache Cordova is a popular mobile development framework. This Cordova extension assists in integrating commands, code-hinting and debugging for Apache Cordova within VS Code.

Key Features:

- Command palette finds errors in code and debugs them
- Uses Intellisense to work around plugin APIs
- Works smoothly with any project based on Cordova
- Added support for the Ionic framework

40. Vim



Bind : to show the command palette:

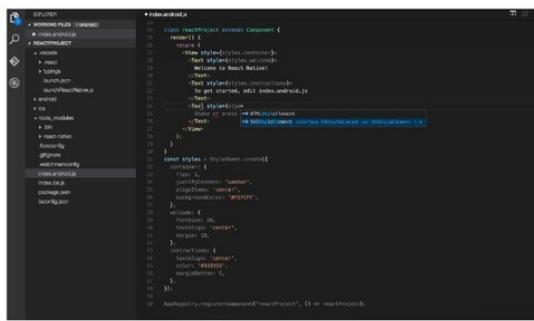
```
"vim.otherModesKeyBindingsNonRecursive": [
  {
    "before": ["`"],
    "after": [],
    "commands": [
      {
        "command": "workbench.action.showCommands",
        "args": []
      }
    ]
  }
]
```

A favorite text editor of many programmers, Vim is now emulated in Visual Studio Code. This Vim extension makes writing and editing code easier.

Key Features:

- Multiple code commands combinations
- Possess a versatile command remapping capability
- Multiple Vim commands can be followed by multiple cursors

41. React Native Tools

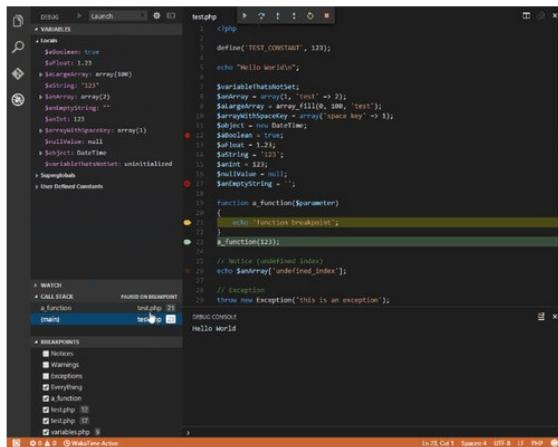


React Native is the popular mobile development framework open sourced by Facebook. This Reach Native extension lets you edit, develop, debug and integrate commands for React Native, providing a development environment for React Native projects.

Key Features:

- Creates an environment to support mobile project development
- Use command palette to debug and quickly run commands
- Use IntelliSense to browse parameters for React Native APIs

42. PHP Debug

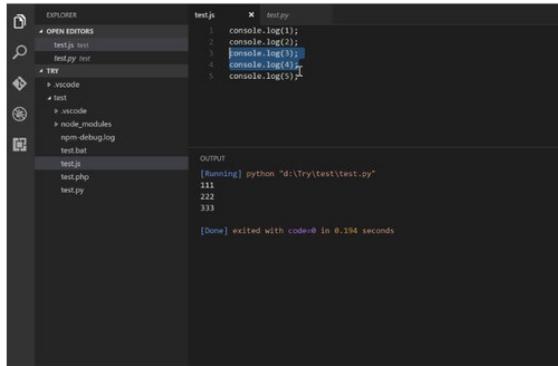


If you're programming with PHP and are looking for an extension to help you debug your PHP code, PHP Debug is the VS Code extension you need, offering a variety of features to streamline development and ensure bug-free code.

Key Features:

- Acts as an adapter between VS code and XDebug
- Connects to remote debuggers as well
- Allows user to set breakpoints according to their preference (line, conditional, and function breakpoints)
- Break on entry or on uncaught exceptions and errors / warnings / notices
- Supports multiple, parallel requests
- Runs as CLI or without debugging

43. [Code Runner](#)

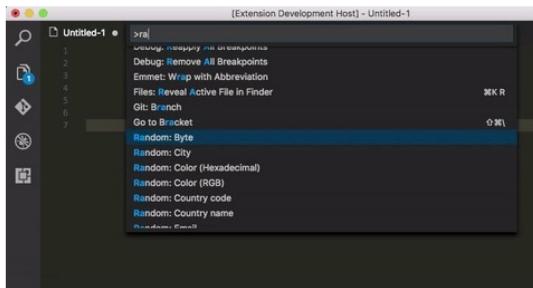


Code Runner is an extension that enables you to run any language's code snippets in Visual Studio Code, with support for every popular programming language including both legacy languages and those that have gained popularity in recent years such as Clojure, Objective-C, Rust, Racket, AutoHotkey, AutoIt, and many others.

Key Features:

- Provides an Output Window to check the output of a snippet
- Supports a wide range of programming languages like C++, PHP, Clojure, Racket, Python and much more
- Custom commands

44. [vscode-random](#)



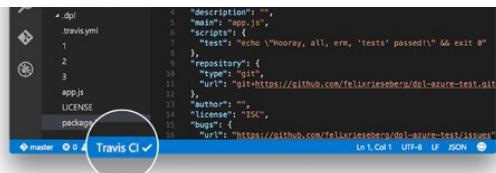
Generating and adding random data directly into your project, this Visual Studio extension does exactly what its name suggests. It uses the ChangeJS library and was inspired by the Random extension for Atom.

Key Features:

- Configures names, years, numbers, etc.
- Multiple commands can be accessed and worked
- Quick install and launch option

45. [Travis CI Status](#)





Apart from Jenkins, Travis is a widely known and used CI platform. Checking the build status of your code during the build stage is key. This Visual Studio code extension enables you to check your project's Travis CI status without leaving VS Code.

Key Features:

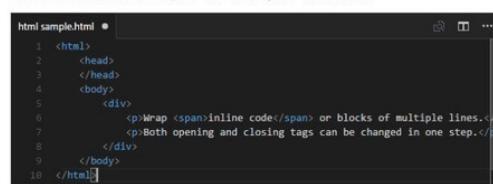
- Small status indicator present in the status bar for convenience
- Update status and open in Travis are two default commands
- Uses a combination of repository/username to find test

46. [htmltagwrap](#)

What is it

Wraps your selection in HTML tags. Can wrap an inline selection or a selection of multiple lines.

To use, select a chunk of code and press "Alt + W" ("Option + W" for Mac).



This simple, useful Visual Studio code extension provides you with the option of wrapping the codes you select in HTML brackets.

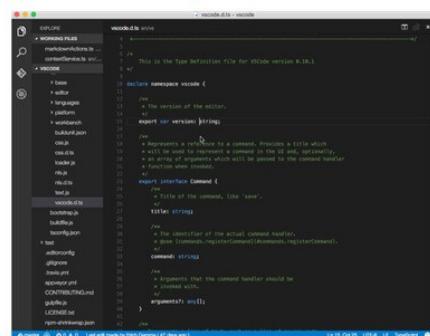
Key Features:

- Wraps single or multiple inline selections
- Quick wrap shortcuts available
- Supports spaces and tabs for indentation

47. [Git Blame](#)

Git Blame

See Git Blame information in the status bar for the currently selected line.



If you're looking for a tool to help you see the Git Blame to view the revision history for your code, this Visual Studio code extension does just that. It appears in your status bar for easy access.

Key Features:

- Shows git blame information for selected lines
- Git blame information pings in the status bar

- Reduced text size for visibility

48. Reactjs Code Snippets

```

• panel.jsx
  1 import React, {Component} from 'react';
  2
  3 class MyPanel extends Component {
  4   onChange(newState) {
  5     this.setState(newState);
  6   }
  7   render() {
  8     return (
  9       <div>
10         this is my panel
11       </div>
12     );
13   }
14 }
15
16 export default MyPanel;

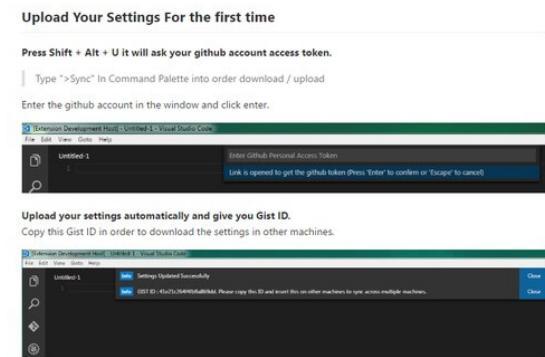
```

Reactjs is a popular open source UI framework released by Facebook. For the developers using Reactjs, this Visual Studio code extension provides snippets in ES6 syntax.

Key Features:

- Supports languages like JavaScript, TypeScript and React
- Based on the bable-sublime-snippet package
- View already-installed snippets or add new snippets

49. Settings Sync

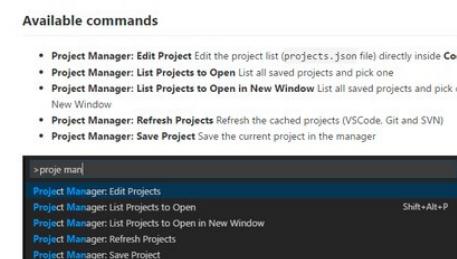


Github Gist is a nifty way to share code and other pieces of text. With the Settings Sync extension, you can sync snippets, themes and other extensions across different machines using Gist.

Key Features:

- Single click upload and download
- Supports anonymous, Gist and Github account token to log in
- Synchronizes settings, snippets, extensions and themes

50. Project Manager



Switching between projects can become tiresome, slowing down your progress. This

visual Studio code extension makes the navigation process seamless, allowing you to multitask like a pro.

Key Features:

- Makes projects within Visual Studio easily accessible, and easy to switch between
- Scans projects and differentiates them according to code used
- Automatically caches the projects after scanning

These are a few of the many more extensions available in the [Visual Studio](#)

[Marketplace](#). Each extension serves a particular purpose in helping you program. They perform a variety of tasks like enabling you to add snippets, navigate through projects, use different programming language and add other features. We hope this list helps you and you find the extension you have been searching for and opens up your mind to the ways you can extend Visual Studio Code to simplify and supercharge your development.

Improve Your Code with Retrace APM

Stackify's APM tools are used by thousands of .NET, Java, PHP, Node.js, Python, & Ruby developers all over the world.

Explore Retrace's product features to learn more.


App Performance Management


Code Profiling


Error Tracking


Centralized Logging


App & Server Metrics

[Learn More](#)

 About the Author  Latest Posts

 About Alexandra Altvater

Get the latest news, tips, and guides on software development.

Join the 50,000 developers that subscribe to our newsletter.

Email

Sign Up Today

Get In Touch	Products	Solutions	Resources	Company
Contact Us	Retrace	Application Performance Management	What is APM?	About Us
Request a Demo	Prefix	Centralized Logging	Pricing	News
Start Free Trial	.NET Monitoring	Code Profiling	Case Studies	Careers
	Java Monitoring	Error Tracking	Blog	GDPR
	PHP Monitoring	Application & Server Monitoring	Documentation	Security Information
7171 Warner Ave Suite B787 Huntington Beach, CA 92647 866-638-7361	Node.js Monitoring	Real User Monitoring	Free eBooks	Terms & Conditions
	Ruby Monitoring	For Developers	Free Webinars	Privacy Policy
	Python Monitoring	For DevOps	Videos	
	Retrace vs New Relic		Ideas Portal	
	Retrace vs Application Insights		ROI Calculator	
			Support	
			News	



