

Compare 2 Visual Studio Code and Atom editors

Atom has been around for a while, but it's only been around recently. Visual Studio Code, from an 'anonymous' tool, has now become the hottest 'editor'.

If you're looking for the perfect code editor, chances are you already know about both Atom and Visual Studio Code. Sure, there are many other editors, but these two options are among the most talked about software.

Atom has been around for a while, but it's only been around recently. Visual Studio Code, from an 'anonymous' tool, has now become the hottest 'editor'. However, not everyone wants to switch from Atom to Visual Studio Code.

Is Visual Studio Code or Atom the right code editor for you?

1. What are the similarities between Visual Studio Code and Atom?
2. Difference between Visual Studio Code and Atom
 1. Efficiency
 2. The core features
 3. Plugin
 4. Community and user base
3. Download Visual Studio Code and Atom

What are the similarities between Visual Studio Code and Atom?

For starters, Visual Studio Code and Atom are referred to as having the same 'DNA'. Both editors use Electron, allowing developers to create complete desktop applications with web technologies like JavaScript, HTML and CSS. Some people vehemently oppose this idea, but it's a different story.

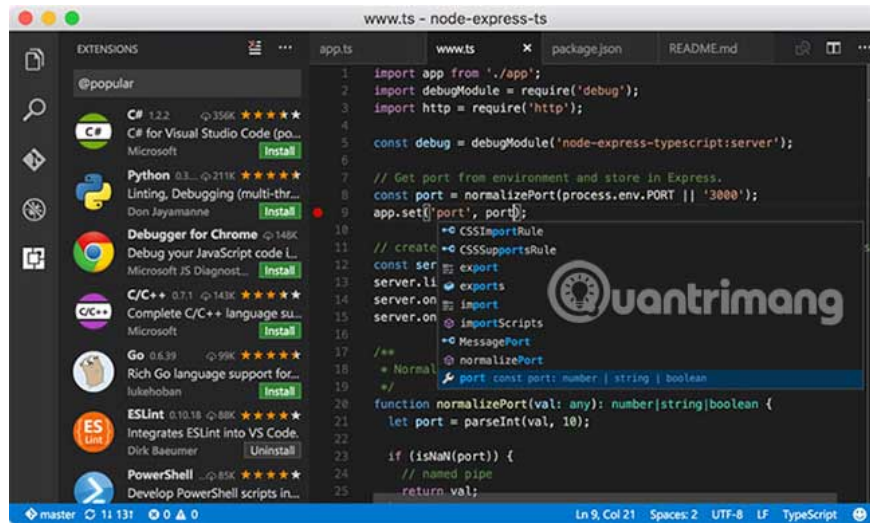
These two editors are more similar than you think in a different way. Atom is created at GitHub, while Visual Studio Code is created by Microsoft. In 2018, Microsoft announced it would buy GitHub. This has raised concerns about Atom's death, but Microsoft has made it clear that both editors will continue to exist.

Difference between Visual Studio Code and Atom

Efficiency

Visual Studio Code fans often compare its performance against Atom and other Electron-based applications. Electron applications are pretty much notorious for slow performance and long boot times, but Visual Studio Code has avoided this.

The difference in performance between Visual Studio Code and Atom shows on several factors, but the main aspect is the approach that each application is developed. Visual Studio Code has a strictly controlled core set of functions, with plugins that help add features.

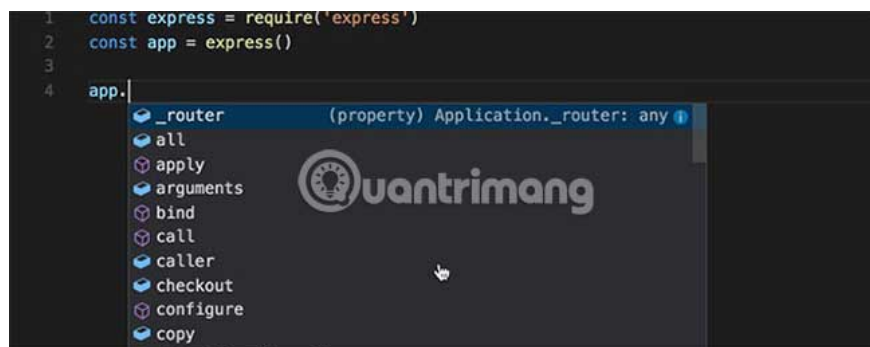


On the other hand, Atom uses a plugin-based approach for almost everything. This approach is beneficial, but also has drawbacks. Atom is a bit slower and this only gets worse when adding certain plugins.

Visual Studio Code has a clear advantage when it comes to performance, but no editor is slow on modern computers. This changes when you edit large files. Visual Studio Code is better than Atom, but significantly slower when compared to an editor like Vim or even Sublime Text.

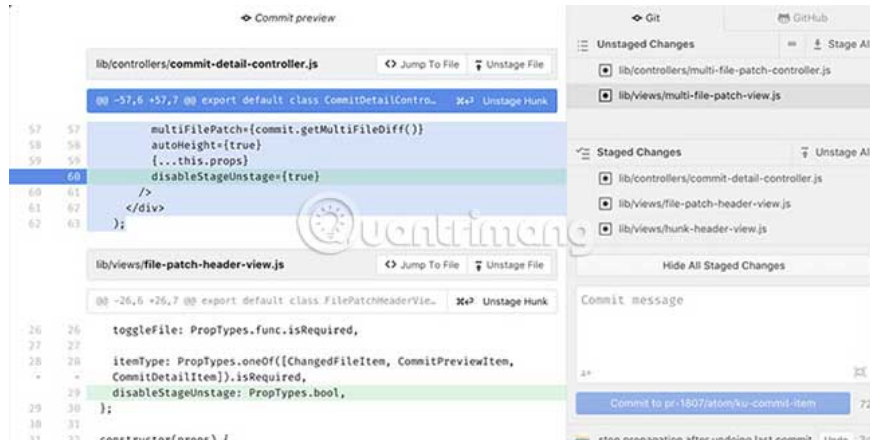
The core features

Visual Studio Code possesses more functions than Atom or even many other editors. It does not have the features of the Integrated Development Environment (IDE), but is nearly equivalent. In addition to the standard drafting features, Visual Studio Code also has support for building and debugging applications.



A feature that appears more frequently in modern editors is Git integration. Again, Visual Studio Code also has this feature, which allows you to easily handle version control without having to open a terminal window. Markdown support is also built-in, complete with preview, so you can ensure your README.md file will fit on GitHub.

Although Atom doesn't have nearly the same number of features at first launch, it does have Git integration. Atom is even one step further, providing complete GitHub integration. Based on the project's origins, this is almost always expected because it's handy, especially if you use GitHub for everything.



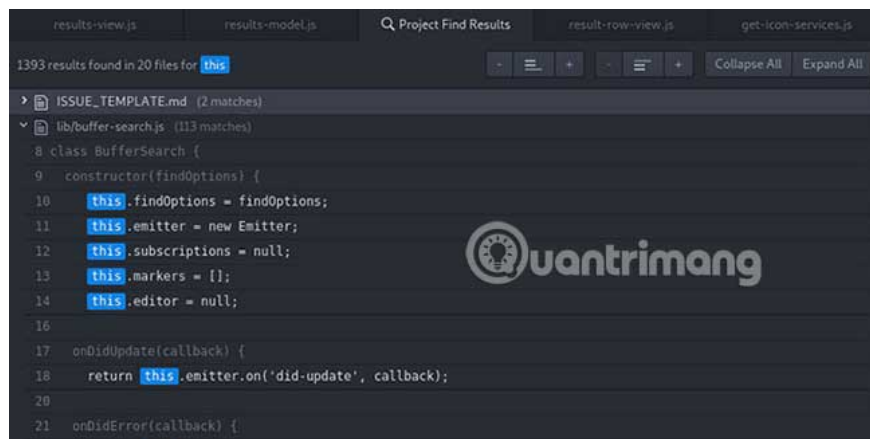
However, for the most part, Atom depends on plugins. They are easy to install from the app itself, allowing you to customize the editor in almost any way you can think of.

Plugin

Scalability is the biggest difference between these two editors. For Visual Studio Code, the plugin helps to add features. You can install themes, support for new languages ??and build tools, which help you when you talk about programming in Rust or Go, for example.

On the other hand, since most of the Atom editor's functionality comes from integrated plugins, the right plugin can create an almost entirely new application. This makes Atom a highly scalable application. Atom even has a part in the user manual called Hacking Atom. Reference at:

<https://flight-manual.atom.io/hacking-atom/>

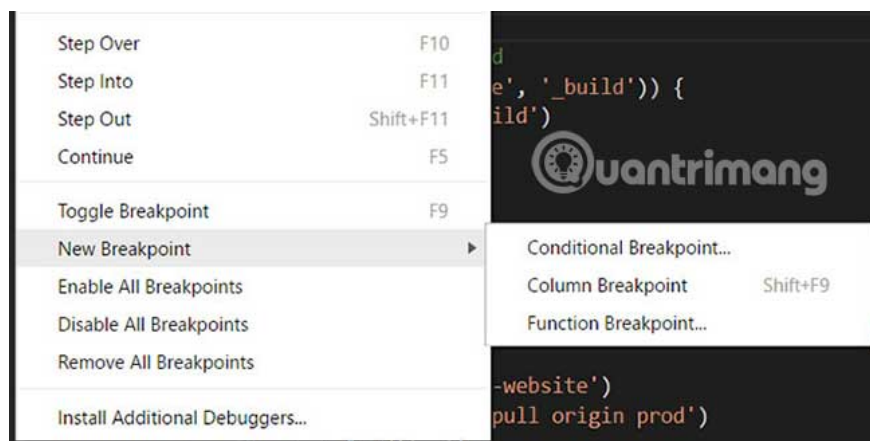


This is somewhat similar to the battle between Vim and Emacs. The sheer amount of built-in functionality later led to some people considering Vim as the editor and Emacs as the operating system. Atom is not as customizable as Emacs (no one has written an Atom email client), but it is more similar to Visual Studio Code.

However, it cannot be said that the Visual Studio Code plugins do not provide much functionality. TipsMake.com's list of 9 extension Visual Studio Code for easier programming proves it.

Community and user base

Both Visual Studio Code and Atom currently have large communities and user bases. While Visual Studio currently seems more popular, Atom still has a community for users and developers. If Microsoft decides to drop support for either editor, this may change, but it is unlikely to happen in the near future.



Atom seems to be more popular with the open source community than Visual Studio Code due to Microsoft's later affiliation. There was an effort from the community to create a free version of Visual Studio Code from Microsoft. Both editors are open source, but some users don't like the data collection used by either of these apps.

Download Visual Studio Code and Atom

Both Visual Studio Code and Atom are available for free. The Atom editor is available under the MIT license, while Visual Studio Code is under a proprietary license (although this editor is open source). Both editors are available for macOS, Linux and Windows.

1. Download Atom (Free).
2. Download Visual Studio Code (Free).

In short, Atom is a highly customizable editor. With the right plugins, it is possible to get the functions of the IDE. The same goes for Visual Studio Code, but its feature set feels closer to the IDE in the first place. However, both do not have the features of a full IDE.

Wish you find yourself a suitable choice!

You finished reading the article "**Compare 2 Visual Studio Code and Atom editors**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.