Use and install extensions

In this unit, you'll learn how easy it is to locate and install Visual Studio Code extensions. As seen previously, Visual Studio Code installs with many built-in extensions, but it does not come bundled with support for every development language and environment. By using the Extensions Marketplace, you can locate the tools, languages, and debuggers you need for your particular workflow and personal preference. Additionally, there are many extensions that add to the powerful features of Visual Studio Code for even greater customization and control, efficiency, and maybe even some fun.

Extension Marketplace

In the previous unit, you saw the default list presented when you access the Extension Marketplace. Let's look more closely at that, and install a few things.

Searching for extensions

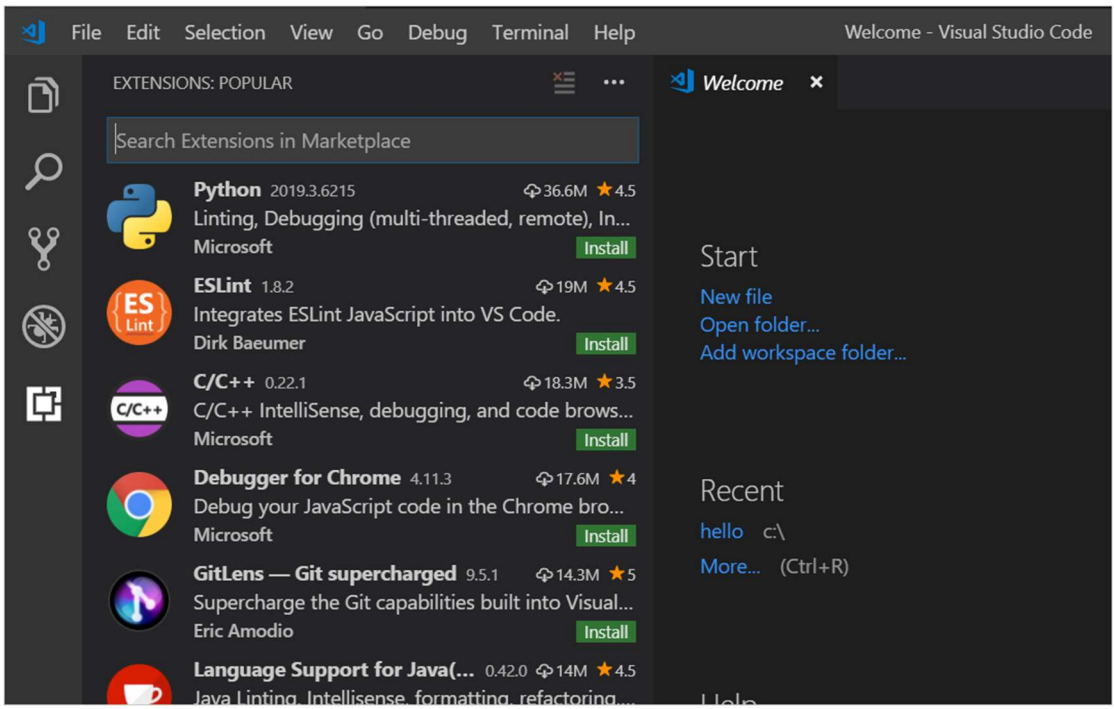
On the left of the Visual Studio Code UI is the Activity Bar. If you do not see an Activity Bar, toggle it on by navigating the pulldown menu sequence: **View > Appearance > Show Activity Bar**. The **Extensions** icon is the highlighted icon in the following graphic.

By default, all activity icons are showing. If you don't see this icon, right-click in the vacant area of the Activity Bar to see the list of available icons, and select **Extensions**. Open the Extensions list by selecting the icon shown below.

Note

There is a keyboard shortcut for extensions. Hover your mouse over the **Extensions** icon to see yours (varies by platform).

Since this is your first time installing any extensions, you'll see only a list of popular extensions on the marketplace.



This list is one way to browse, but with so many extensions available, it's usually quicker to use the Search box at the top.

To give you an idea of the variety of choices available for a given category of extensions, type **icons** in the search box. You should see many pages of extensions, sorted by default order (a combination of install count and rating). You can sort more specifically by choosing a sort from the ellipsis icon (...) in the upper-right corner of the Extensions area.

Installing and managing extensions

Installing is as easy as clicking the **Install** button on an extension entry in the Marketplace. Let's install a few now. Use the Search box to locate the extensions listed below, and then use the **Install** button on each (we'll get into the function of each of these later):

- Material Theme
- Prettier Now
- Bracket Pair Colorizer
- Live Server

While installing, removing, disabling, and enabling extensions, the button may change into a **Reload** button. Be sure to select it if applicable.

Material Theme

Just for fun, let's play around with this one a little bit. Once it's installed and ready, bring up the Color Themes (remember, there are many ways to get there, from pull-down menu sequences to keyboard shortcuts and the Welcome page). We use the direct shortcut Ctrl-K, Ctrl-T (Cmd-K, Cmd-T on macOS). Now use the up and down arrow keys to change the theme. You'll see the change immediately as you select new themes. Try out some of the Material Themes that now appear in the list to find one you like. Even if you don't select one you prefer now, if you're like many of us, you'll eventually enjoy a quick change to the color theme once in a while. Adding lots of choices can help you with this, so feel free to add more themes.

This extension is here to illustrate customization features. You can safely remove or disable it if you don't like it.

Prettier Now

This extension is a *beautifier*. In case you aren't already aware, this term refers to an editor's ability to format your code automatically using certain rules the

designer has decided work well for the type of code you're' working with, so it can be a subtle but powerful tool to use. There are many beautifiers available in the Extensions Marketplace, but this one has a lot of options and is easy to use.

Again, if you don't care for this extension, feel free to disable it or choose a different one.

Bracket Pair Colorizer

As with the beautifier and theme we installed, this is an extension that can help you code more efficiently with visual feedback, but isn't required for web development. However, this extension can truly save you a lot of time troubleshooting, and provide your eyes some relief.

Live Server

This extension provides a local development server with a live reload feature for static and dynamic pages. This greatly eases web development by providing real-time updates to your content in a web browser. As with most extensions, there are other choices for this functionality in the Marketplace, but we'll use this one for now.

This is the only extension here that we'll be using in later units, and other modules might also direct you to install it or a similar extension. Next, we'll put together a simple web page and you'll see how this extension works.

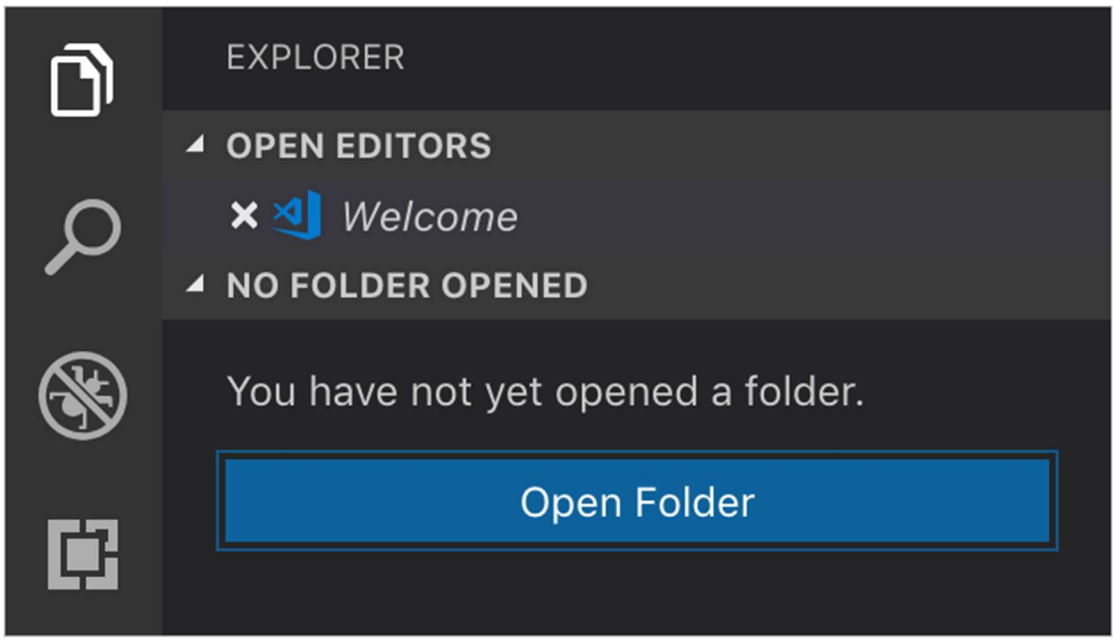
Create and auto-generate files in Visual Studio Code

As with most of the Visual Studio Code functionality, there are many ways to create and manage files. In this unit, we'll create and edit a file. We suggest creating a folder called **intro-to-vscode** on your local computer and work from there. Although you can think of this as your project folder for this module, Visual Studio Code won't create any files other than the specific files needed for this lesson unless you create a Workspace, discussed at the end of this unit.

Using Explorer to manage files

The **Explorer** button, shown below, toggles the Explorer view in the sidebar on and off.

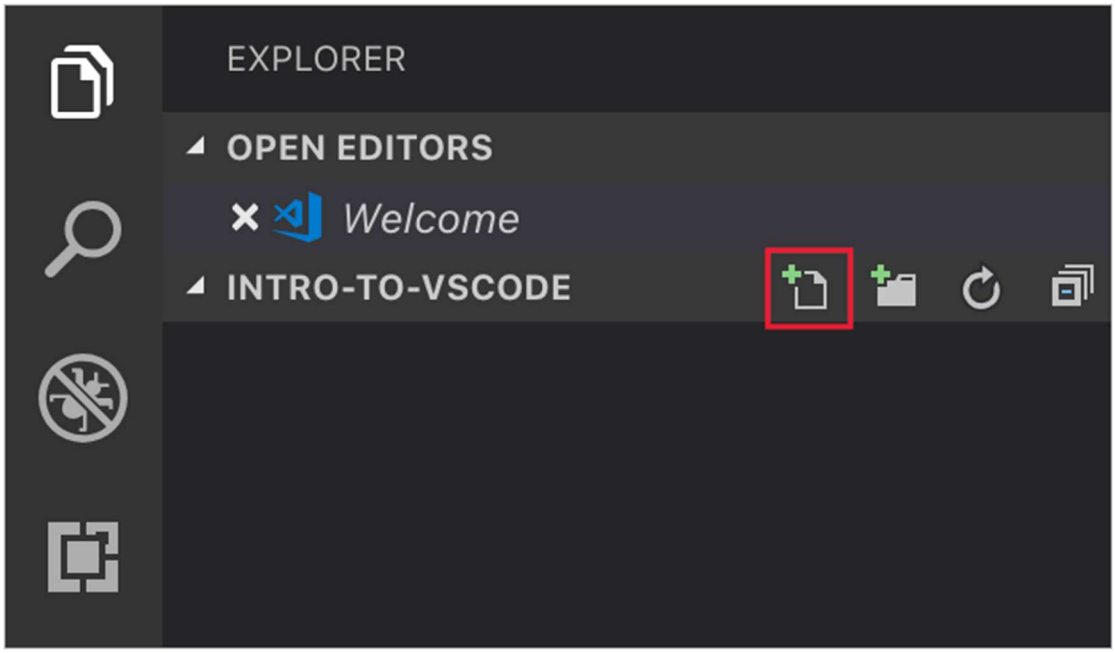
You should see the **Open Folder** button as shown here.



Select this button and navigate to the intro-to-vscode folder you created previously.

Create an HTML file

In this case you probably don't have any files yet, so the folder is empty. Let's solve that now. Use the **New File** icon, shown below, to create a new file in the current folder. Note that this icon doesn't show up unless you hover your mouse cursor over the portion of the Explorer where your folder name is listed.

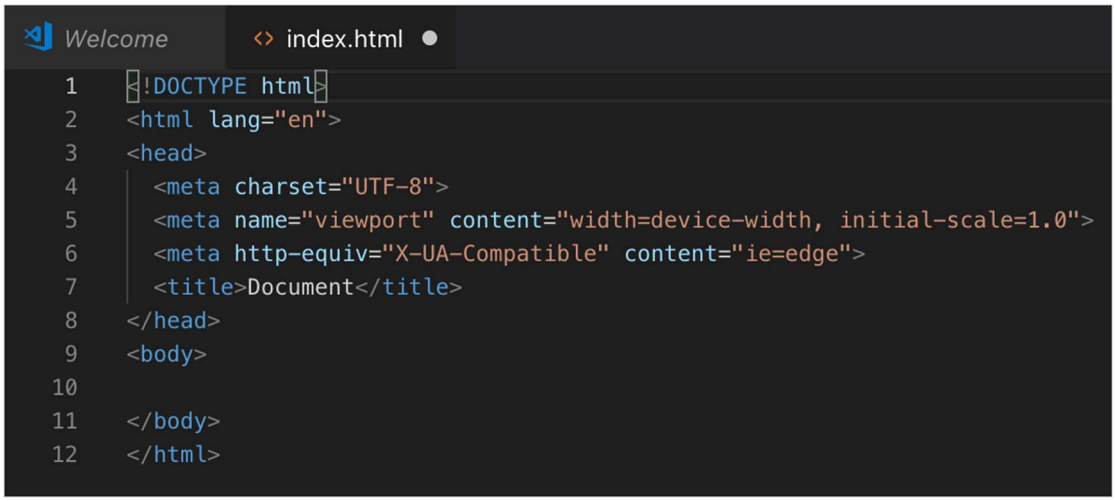


Name the new file `index.html` and press Enter.

Using Emmet to create code

You should now have an empty file named `index.html` open in the editor and listed in the Explorer. If you start typing valid HTML, you'll see color coding come in as you work. Instead of typing everything manually, we can use Visual Studio Code's built-in Emmet support to do a lot of the tedious bits.

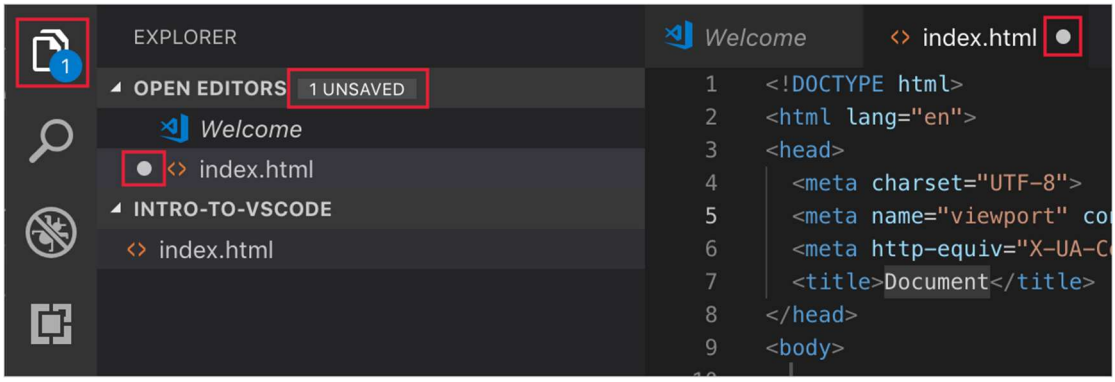
In your empty editor window `index.html`, type `!` (exclamation point) then select the Tab key. This will tell Emmet to fill in the minimum HTML needed to make a web page using default values, as shown below.



We've taken a direct line to creating the HTML here, skipping over optional features of Emmet. For example, when you type the exclamation point but before you type the Tab key, you'll see something like the following.

Note the little `i` in the blue circle off to the right, which you can click for more info. Also, be sure to take note of the location of the full Emmet reference at Emmet In Visual Studio Code. You don't need to know all the details at this moment, but it's useful to have a feel for the scope of Emmet and what it can do to make you more efficient.

Let's head back to our new file we're editing. Notice the changes to the UI after adding content.



There's a number 1 on the **Explorer** icon in the Activity Bar, which means there's one file open that's been changed but not saved. After that is a count of unsaved changes in the Explorer, next to the Open Editors header. Lastly, the file name in the list of open editors and the editor tab for the file itself each have a large dot.

All these various indicators are provided to notify you that you have unsaved changes, so you'll know regardless of which parts of the UI you decide to hide or show.

Saving files manually is no different than usual, from the File menu or using keyboard shortcuts. Visual Studio Code also has a handy automatic file save feature you can enable, which is in the File menu. (Alternately, hit F1, start typing "auto save," and select File: Toggle Auto Save).