

Sublime Text 2 Tutorial

Source: [Tuts+](#)

Basics and Setup

- You can open files by going to File > Open or Ctrl + O
- Click on **minimap** on the right to scroll to where you clicked
- Size changed by Ctrl++ and Ctrl+-
- Change themes by going to Preferences > Color Scheme
- Default Settings at Preferences > Settings - Default
 - Make changes in Preferences > Settings - User never to Settings - Default
 - Settings - Default is changed reset on every sublime update
- My user settings:

```
{  "color_scheme": "Packages/Color Scheme - Default/Solarized (Dark).tmTheme",  "default_line_ending": "unix",  "ensure_newline_at_eof_on_save": true,  "font_size": 13,  "ignored_packages":  [    "Vintage",    "SublimeCodeIntel"  ],  "rulers":  [    79  ],  "trim_trailing_white_space_on_save": true}
```
- OSX only:
 - To enable Open in Sublime Text 2 by right click, go to Automator > Service in Mac and
 - Type Shell and click on Run Shell Script
 - Type /Applications/Sublime\ Text\ 2.app/Contents/SharedSupport/bin/subl -n \$®, select Files and Folders and as arguments, Save and Done.
 - To enable subl in the command line, create an alias of subl in you bin folder by typing:

```
bash: sudo ln -s "/Applications/Sublime Text 2.app/Contents/SharedSupport/bin/subl" ~/bin/subl
cmd: doskey subl="C:\Program Files\Sublime Text 2\sublime_text.exe" %*
```

Multiple Cursors

- Replace is Ctrl+H
- Sublime way is to put your cursor on a word and hit Ctrl+D
 - If you keep hitting Ctrl+D, you should be able to select all the words one by one, to skip a work press Ctrl+K
 - Then type and see the magic!
 - Or you can highlight a certain group of words and then Ctrl+D to add a multiple cursor to the next occurrence
- To select all at once hit Alt+F3 - might want to change this to Ctrl+Shift+a
- If you hold Shift and drag your mouse down while holding the right mouse button, you will create a multiple cursor on every line
- If everything is already selected, hit Ctrl+Shift+I to add a multiple cursor on all those line

Finding stuff

- You can use Ctrl+F or Ctrl+I
- Ctrl+I is faster and prevents hitting the Esc key to hide the search box
 - It more of a way to jump to something you see on the screen without using the mouse

Command Palette

- Ctrl+Shift+P
- Fuzzy search - order doesn't matter
- So to go to Ctrl+Shift+P and then ssy and you can get Set Syntax: Python
- Can do a lot with this, all commands are here

Opening Files in Folder

- Ctrl+P and type the name of the file
- Fuzzy search, so you don't need to type the name of the file or folder exactly

Finding methods/variables

- In **current file**, Ctrl+R and type the name of the method
- In **current file**, Ctrl++ and type the name of the variable
- In **current file**, Ctrl+G and type the name of the variable
- From another file, Ctrl+P once the file is the first one, type @ or # and you get access to all the symbols

Your own keyboard shortcuts

- Go to Preferences > Key Bindings - Default to see the current shortcuts
- To make changes, go to Preferences > Key Bindings - Default

General

- Shortcut to comment line: Answer - Ctrl+/
 - To wrap words in something "wrappable" like () or " ", press Ctrl D and then (or "
- To move a line up or down hit Ctrl+Shift+Up-Arrow or Ctrl+Shift+Down-Arrow
- To toggle the sidebar press Ctrl+K then Ctrl+B
- Shortcut to duplicate line Ctrl+Shift+D

Packages

- To view the packages go to Preferences > Browse Packages
- To install packages that are not in package manager, you need to install them in this dir.
- To install packages that are not in package control, get the folder from github and paste it in the directory.

Package Control

- To download package control press Ctrl ~ and paste this:

```
import urllib2,os; pf='Package Control.sublime-package'; ipp=sublime.installed_packages_path(); os.makedirs(ipp) if not os.path.exists(ipp) else None; urllib2.install_opener(urllib2.build_opener(urllib2.ProxyHandler())); open(os.path.join(ipp,pf),'wb').write(urllib2.urlopen('http://sublime.wbond.net/'+pf.replace(' ','%20')).read()); print('Please restart Sublime Text to finish installation')
```

Link: http://wbond.net/sublime_packages/package_control/installation

- To access, use **Command Palette** or go to Preferences > Package Control
- To install, use Package Control:Install Package in the command palette

Snippets

- Type Snippet in the Command Palette to access snippets
- The shortcuts to the snippets will be visible on the right
- To get the snippet, either pick it from here, or type the shortcut
- If a drop down comes, press enter, else press tab
- When you insert a snippet, use tab to go from one fillable field to another

Create your own snippets

- Go to Tools > New Snippet...
- Type you snippet and save as snippetname.sublime-snippet in the User dir of the Packages folder, extension matters
- To create snippet do this:

```
<snippet>
  <content><![CDATA[
      Hello, ${1:this} is a ${2:snippet}.
  ]]>
</content>
<!-- Optional: Set a tabTrigger to define how to trigger the snippet -->
<tabTrigger>hello</tabTrigger>
<!-- Optional: Set a scope to limit where the snippet will trigger -->
<!-- <scope>source.python</scope> -->
</snippet>
```

- This will create a snippet that will come out when you type hello tab
- Once the snippet is out you will be at the \${1:this} which has a default value of this and if you hit tab again you will go to \${2:snippet} which has default value of snippet
 - If you don't want a default value just type \${1}
- To make a snippet language specific, and not a global snippet, create a folder called, for example, Python in your User directory (this will pop up when you try to save the snippet) and save it in the Python folder.
 - this snippet will then only pop up when you are in a .py file
- Another way to do snippets for languages is to edit the scope in the xml above.
 - So, scope.python will make it a python snippet regardless of directory
- You can download snippet packs from package control
- If its not in package control, download from github and paste in the Packages/User directory

Useful packages

Emmet

- Cooler, more dynamic snippets
- use css like typing to generate html (Never add spaces)
 - ul tab results in
 - ul#someId tab results in <ul id="someId">
 - ul.someClass tab results in <ul class="someClass">
 - To create children use >
 - ul>li tab results in
 - To go up to the parent use ^
 - ^ .header>h1 .main tab results in: (if you dont put anything before the dot, it defaults to div)
- To create multiple, use * li*2 tab results in
- To create a sblings, use +
 - ul+ol tab results in
- To add an attribute use {}
 - So ul{color="red"} tab results in <ul color="red">
- To add data in the attribute use {}
 - So, li{my data} tab results in my data
- To add a div with a class, type .NameOfClass
 - Result: <div class="container"></div>
- In sum, ul>li*4>a[href=#]{Some Link} tab will result in:

```
<ul>
  <li><a href="#">Some Link</a></li>
  <li><a href="#">Some Link</a></li>
  <li><a href="#">Some Link</a></li>
  <li><a href="#">Some Link</a></li>
</ul>
```

- For css
 - p20 -> padding: 20px
 - w80p -> width: 80%
 - m10-auto -> margin: 10px auto
 - transition ->
- webkit-transition: 1s;
- moz-transition: 1s;
- ms-transition: 1s;
- o-transition: 1s;
- transition: 1s;
- lorem200 tab results in 200 words of lorem

Fetch

- Install package using the name Nettuts+ Fetch
- Add sources to Fetch: Manage using Command Palette
 - "jquery": "http://code.jquery.com/jquery.min.js"
 - The key is the name you wanna use, the value is the source to pull from
 - Use a link that will likely never change -> Github raw source of file
- To put it in a file
 - Create a file and type Fetch: File and pick the name - This will pull the latest file from that source link
- You can pull packages too that must be zip files
 - This is in the packages part of Fetch: Package
 - Pick a package and enter the folder where you want to extract it

Advanced New File

- Install AdvancedNewFile
- Fast way to create file instead of Ctrl N followed by Ctrl S and typing the name
 - Simply type Ctrl+Alt+N and type in the file name or relative file name
- To create a folder, you have to create a file in the folder and the folders will be automatically create
 - css/buttons.css will create the css folder and place the file in it if it doesn't already exist

SideBar Enhancements

- Install SideBarEnhancements
- You now have the Open in Browser functionality
 - Right click file and click open in browser of use Ctrl palette when the file is open
 - To open with a specific url, Project: Save As then add a url to the properties json file

HttpRequester

- Install using Http Requester
- Highlight any url and press Ctrl+Alt+R to open the page source in a new tab
 - Or you can highlight and right click and click Http Requester
- A good idea would be to put the url as a comment on the top of your view function and then use highlight it to test immediately
- Works best for json as the source is easy to read

LiveReload

- Install using LiveReload
- Need chrome extension for LiveReload
- Enable allow access to file urls in the extensions settings menu in chrome
- Now if you fun a html file using the file url, you will be able to see live changes without having to hit refresh

SublimeRope - Recommended for Django

- It will show errors, unused imports and definitions
- Smarter, project aware auto-completions (Ctrl+space to trigger manually)
 - Example: User.objects.filter()
- Use command palate to create a rope project
- Set settings in .ropeproject/config.py like so:
 - prefs.add('python_path', 'C:/Python27/Lib/site-packages')
- Go to definition: Right click on the word

Optional:

Sublime Linter

- Install SublimeLinter
- This will show all errors below at the bottom bar

Code Intel

- Smarter Autocompletion
- Jump to definition = Alt+Click
- Jump to definition = Control+Windows+Alt+Up
- Go back = Control+Windows+Alt+Left
- Manual CodeIntel = Control+Shift+space
- Might need to edit settings:

```
{
  "Python": {
    "python": 'C:\\Python27\\python',
    "pythonExtraPaths": ['C:\\Python27\\Lib\\site-packages'],
  },
}

Django:

{
  "folders": [
    {
      "name": "Run Evynty",
      "working_dir": "${project_path}",
      "cmd": ["python", "manage.py", "runserver"]
    }
  ],
  "build_systems": [
    {
      "name": "Run Evynty",
      "working_dir": "${project_path}",
      "cmd": ["python", "manage.py", "runserver"]
    }
  ]
}
```

- To edit settings, you can edit override any of the settings in Preferences > Settings - Default like so:

```
{
  "folders": [],
  "settings": {
    "tab_size": 2
  }
}
```

Split Windows

- To split windows, press Alt+Shift+1 or 2 or 3 or 4 or 5 depending on how many windows you want
- To move your cursor to another window press Ctrl+1 or 2 or so on...

Custom Builds

- Tools> Build System

```
{
  "Ctrl": [ "python", "$file" ],
  "selector": "source.python",
  // this will only apply to python files
}
```

Once you save this, select your build as the build system, and press Ctrl B it will run python filename.py and display it in the Sublime Command Prompt

- [Other variables](#)

Useful Links

- [Dropbox Sync](#)