The 5 Best Plugins for Python Development with Sublime Text

Sublime Text has an amazing plugin ecosystem. After more than three years Sublime is still my editor of choice for writing Python code. The list of plugins that I've used has changed but there are five that clearly stand out. This cheat sheet will help you get a great baseline Sublime Text setup for Python development. However, please feel free to add or remove plugins and settings as you see fit. Becoming productive with an editor is really about making the environment as comfortable for yourself as possible.

Alright, here we go—my list of five evergreen Sublime Text packages:

1. Package Control

A package manager for installing additional plug-ins directly from within Sublime. All other packages listed here can be installed through Package Control. It can also update installed packages automatically.
» packagecontrol.io

2. GitGutter

Highlights lines modified since the latest Git commit. GitGutter works by adding little icons to the editor gutter area indicating whether a line has been inserted, modified, or deleted according to Git. If you don't use Git I can recommend VCS Gutter as an alternative that supports Mercurial and Subversion.
» packagecontrol.io/packages/GitGutter

3. SublimeLinter + SublimeLinter-Flake8

Automatically lints .py files whenever they're modified using the Flake8 linter. Linter and syntax errors are displayed directly in the editor view. I use Sublime-Linter and the flake8 plugin for most Python projects that I work on.
» packagecontrol.io/packages/SublimeLinter
» packagecontrol.io/packages/SublimeLinter-flake8

4. Anaconda

Anaconda is full-featured solution for Python code completion inside Sublime Text. It's quite impressive how well it works and how well it's integrated with the editor. (Please note there's also a Python distribution for data science under the same name. I'm referring to the Sublime package here.)
» packagecontrol.io/packages/Anaconda

5. Zen Tabs

If you ever found yourself overwhelmed by a large number of open tabs in your Sublime Text window then this is *the* plugin for you. Zen Tabs lets you set a limit for open tabs and as soon as you open a new tab when you're at the limit the oldest one will be closed. Don't worry, Zen Tabs will never close any tab with unsaved work and in my two years of using this plugin I've never lost any work due to it. Again, a plugin I highly recommend!
» packagecontrol.io/packages/Zen%20Tabs

I hope you found this list of plugins useful! Like I said, treat this as a starting point to build your own setup from there. It takes some time trying things out and going back and forth on the different choices. But in the end you'll be rewarded with a custom editor setup that will make your life as a Python developer that much more enjoyable :)

– Dan (dbader.org)

P.S: Check out my *Sublime Python* course at SublimeTextPython.com