
ploogz Documentation

Release 0.0.9

Pat Daburu

Aug 18, 2017

Contents

1	API Documentation	3
1.1	plooz	3
1.2	plooz.ploogins	3
2	Indices and tables	5
3	Quickstart	7
	Python Module Index	9

Ploog it in and go.

To get the source, visit the [github repository](#).

Let's take a look at how ploogz works.

1.1 ploogz

Provide a brief description of the module.

1.2 ploogz.ploogins

CHAPTER 2

Indices and tables

- `genindex`
- `modindex`
- `search`

CHAPTER 3

Quickstart

ploogz is a pretty simple plugin framework, so there isn't a whole lot to know to get started. Step one is to extend the `ploogz.ploogins.Ploogin` class and override the important methods to create an object that can do some useful work. (In the code blocks below, we don't include docstrings to focus on the code itself... *but you should always document your code!*)

```
from ploogz.ploogins import Ploogin, upon_setup, upon_activate, upon_teardown

class MyUsefulThing(Ploogin):

    def __init__(self):
        super().__init__(name='My Useful Thing')

    @upon_setup
    def get_ready(self):
        print("Getting ready!")

    @upon_activate
    def get_busy(self):
        print("Here we go!!")

    @upon_teardown
    def wrap_it_up(self):
        print("Good night.")
```

You can place this class in a python file (with a `.py` extension) in a directory along with other files containing other ploogins.

When it comes time to load these useful classes into an application, you'll need a `ploogz.ploogins.PlooginHost`.

```
from ploogz.ploogins import PlooginHost

host = PlooginHost(search_path=['/path/to/ploogins/directory', '/another/path/to/more/
↳ploogins'])
```

```
host.load()
for ploogin in host.ploogins:
    ploogin.setup()
for ploogin in host.ploogins:
    ploogin.activate()
# When you're all done, you can call teardown() on the host which will tear down all
↳the ploogins.
host.teardown()
```

In the simplest terms, that's all there is to it. This library is in the early stages of its development and there will likely be some additions, but these examples demonstrate the simple approach we seek.

p

ploogz, 3

P

ploogz (module), 3