

Pelican Web Site Generator Configuration

Pelican is a static web site generator. It is the tool we are using to maintain our static sites.

This document assumes we are using `venvwrapper` ¹.

Install Pelican

Do it within our python virtual environment. We named our environment `statsites`.

Activate it, and install Pelican

```
venv statsites
pip install pelican
```

Themes

In each of our `pelicanconf.py` configurations we select a theme for the given site. To get a collection of themes to choose from, we cloned as follows:

```
git clone --recursive https://github.com/getpelican/pelican-themes ${HOME}/pelican-themes
```

Site Theme Note

In our own sites we have made a few adjustments to the lightweight theme to suit our purpose. We maintain this derived theme and our web content under our git version control repository. Our `pelicanconf-en.py` configuration file has this reference.

```
THEME = u"/home/ubuntu/allrepos/mystatic_sites/derived-themes/lightweight-single-home-en"
```

Your site's theme will be for you to choose and/or modify.

Initialize Skeletal Site File System

Use 'pelican-quickstart' to generate initial static website content using pelican.

`pelican-quickstart` creates the a skeletal site generation file system beneath the current empty directory.

Afterwards you create `.rst` files in `content/anyname` directories as blog articles. Create `.rst` files in `content/pages` as blog pages.

Compilation

Generate the web site files once we have all our `.rst` source files. For this we only need to use the Makefile. The following will compile and emit all the web site content under `output/` using pelican.

```
make html
```

You can review your results by invoking the `output/index.html` file.

1

We are using the package `venvwrapper` to simplify virtual environment management. That fact is assumed in this document. If you are not using it, you need to replace lines here such as: `"venv statsites"`, with lines such as `"source ${HOME}/.venv/statsites/bin/activate"`.