pyhoe Documentation

Release 0.0.1a5

modocache

February 18, 2012

CONTENTS

Authors modocache <modocache@gmail.com>

Version 0.0.1a5

OVERVIEW

Like Ruby's Hoe, pyhoe aims to help developers by providing a basic project skeleton. Projects created with pyhoe's default settings include the following:

- A basic directory structure with templates for README, LICENSE, setup.py, .gitignore, and other ubiquitous project files.

- A virtualenv for the new project.

- Automated tests using Nosy, which is built upon nose. Mac OSX users will find a nose plugin called Sneazr to display Growl notifications on tests results.

- Tox, for testing across multiple Python environments.

QUICKSTART

To create a new project directory with the default skeleton, install pyhoe and run the `sow` command.

```
$ pip install pyhoe
$ pyhoe sow my_new_project
```

See *commands* for details on all commands available).

REPORT A BUG

Feel free to open an issue on Github if you notice something wrong with the program or this documentation, or if you think of a feature for pyhoe you think might be worth adding. I'll try to get back to you as quickly as possible.

Contributions to the project are also very welcome! Fork me!

DOCUMENTATION CONTENTS

4.1 Installation

Installation is easy using `pip`.

```
$ pip install pyhoe
```

Or you can grab it from Github.

```
$ pip install git+https://modocache@github.com/modocache/pyhoe.git
```

You can also download the source from Github.

```
$ git clone git@github.com:modocache/pyhoe.git
$ cd pyhoe
$ python setup.py install
```

4.1.1 Requirements

pyhoe is compatible and tested on Python 2.7 and Python 3.2. Other versions are untested.

pyhoe has no requirements for plain usage, although several packages are installed to new virtual environments when starting a new project using the `sow` command.

4.2 Commands

4.2.1 sow

`sow` initializes a new project with the specified name. It can also take any number of extra arguments. Type `pyhoe sow --help` at the command line for a full list of all the arguments `sow` accepts.

```
$ pyhoe sow my_new_project --python python3.2 --template package
```

Some details on particular arguments follow.

project_name
>   The name of the project to create. A new directory and virtualenv will be created using this name.

-t, -template

>   **The template to use for your project. Current options are:**

> • package - A Python package including tests, a README, and a basic directory structure.

-p, -python
> The Python version for the virtualenv pyhoe creates.

INDICES AND TABLES

- *genindex*
- *search*