Make Your Life Easier with Virtualenvwrapper

Reindert-Jan Ekker

@rjekker http://nl.linkedin.com/in/rjekker

Overview

Virtualenvwrapper

- A user-friendly wrapper around virtualenv
- Easy creation and activation
- Bind projects to virtualenvs
- Great with large numbers of projects

Demo

Installing virtualenvwrapper

- On Linux and Mac OS
- Windows: virtualenvwrapper-win

Demo

Using Virtualenvwrapper

- Activating a project
- Switching projects
- Creating projects
- Removing projects
- Linking projects with virtualenvs

Installing virtualenvwrapper on Mac/Linux

```
# Global install: use sudo if necessary
pip install virtualenvwrapper


# Edit .profile
# Use 'which' to check location of virtualenvwrapper.sh
source /usr/local/bin/virtualenvwrapper.sh
# After: restart shell
# Default location for envs: ~/.virtualenvs
```

Installing virtualenvwrapper on Windows

```
pip install virtualenvwrapper-win

# No extra setup needed

# Default location for envs:

# %USERPROFILE%\Envs
```

Virtualenvwrapper Commands

```
# List environments
workon

# Activate environment, switch to project
workon projectname

# Deactivate environment
deactivate
```

Virtualenvwrapper Commands: Projects

```
# Create a project and a virtualenv and bind them
mkproject new_proj
mkproject -p python3 new_proj

# Bind an existing project to a virtualenv
# Binds active venv to current working dir
setvirtualenvproject
```

Virtualenvwrapper Commands: venvs

```
# Create a virtualenv

mkvirtualenv new_env


# Remove a virtualenv

rmvirtualenv some_env


# Create a temporary virtual environment
mktmpenv
```

```
export WORKON_HOME="/home/rj/envs"    # Optional
export PROJECT_HOME="/home/rj/dev"    # Needed for mkproject
```

Configuration

Configure location of venvs and projects

By setting environment variables in `.profile`

Windows: set up variables in Advanced System Settings

Summary

Virtualenvwrapper

- Project creation
- Project switching
- Configuration