

Python Basicks Trainig

Contents

1	PyPI - the Python Package Index	2
2	Python Virtual Environment	3
	What it is	3
	Use it	4
	Reuse	5
3	Python Coding Conventions	6
	Why	6
	How	7
	The table	8
	Automate the process	9
4	PyLint - Code analyse	10
	Why	10
	How	11
	Automate the process	13

1 PyPI - the Python Package Index

The Python Package Index is a repository of software for the Python programming language.

```
$ python3.4 -m pip list
$ python3.4 -m pip install requests
$ python3.4 -m pip search xml
$ python3.4 -m pip show numpy
$ python3.4 -m pip uninstall requests
```

2 Python Virtual Environment

What it is

- A Virtual Environment is a tool to keep the dependencies required by different projects in separate places.
- When Project "A" depends on module version "1.x" but, Project "B" needs "4.x"
- It keeps your global site-packages directory clean and manageable.

```
# instal virtual env
$ sudo apt-get install python3-venv
$ pip install virtualenv

# create new environment
$ python3.5 -m venv training
```


3 Python Coding Conventions

Why

- To make long term code maintenance easier.
 - You will write code once
 - but other developers will read this all the time
- Better teamwork.
- To make our lives easier, not harder.



How

PEP8 - code style guide for Python. A high quality, easy-to-read version of PEP8 is also available at <http://pep8.org>.

```
# try to limit all lines to a maximum of 79 characters.
# 4 space indents
# every import item in new line, asterisk imports is not allowed
from os import path
from os import open

class ClassName:
    """
    Write doc string in rest format
    """
    def long_function_name(var_one,
                           var_two):
        """
        Print summarize of report

        :param var_one: description
        :param var_two: description
        :type var_one: SomeClass
        :type var_two: int
        :return: return description
        :rtype: the return type
        """
        print(var_one)
```

The table

Type	Public	Internal
Packages	lower_with_under	
Modules	lower_with_under	_lower_with_under
Classes	CapWords	_CapWords
Exceptions	CapWords	
Functions	lower_with_under()	_lower_with_under()
Constants	CAPS_WITH_UNDER	_CAPS_WITH_UNDER
Variables	lower_with_under	_lower_with_under
Attributes	lower_with_under	_lower_with_under (protected) or __lower_with_under (private)
Method	lower_with_under()	_lower_with_under() (protected) or __lower_with_under() (private)
Parameters	lower_with_under	
Modules	lower_with_under	_lower_with_under
Local Variables	lower_with_under	

Automate the process

Install pep8 module, then run it on a file or series of files to get a report of any violations.

```
$ pip install pep8
$ pep8 optparse.py
optparse.py:69:11: E401 multiple imports on one line
optparse.py:77:1: E302 expected 2 blank lines, found 1
optparse.py:88:5: E301 expected 1 blank line, found 0
optparse.py:222:34: W602 deprecated form of raising exception
optparse.py:347:31: E211 whitespace before '('
optparse.py:357:17: E201 whitespace after '{'
optparse.py:472:29: E221 multiple spaces before operator
optparse.py:544:21: W601 .has_key() is deprecated, use 'in'
```

The program autopep8 can be used to automatically reformat code in the PEP 8 style. Install and use it to format a file in-place with:

```
$ pip install autopep8
$ autopep8 --in-place optparse.py
```

4 PyLint - Code analyse

Why

- To make long term code maintenance easier.
 - You will write code once
 - but other developers will read this all the time
- Better teamwork.
- To make our lives easier, not harder.



How

Pylint - a source code, bug and quality checker for the Python programming language. <https://www.pylint.org/>

Install pylint module, then run it on a file or series of files to get a report of any violations.

```
$ pip install pylint
$ pylint lab/gui/tkinter/main.py
No config file found, using default configuration
***** Module main
C:  9, 0: Trailing whitespace (trailing-whitespace)
C: 19, 0: Trailing whitespace (trailing-whitespace)
C: 23, 0: Trailing whitespace (trailing-whitespace)
C: 32, 0: Trailing whitespace (trailing-whitespace)
C: 92, 0: Final newline missing (missing-final-newline)
C:  1, 0: Missing module docstring (missing-docstring)
C:  4, 0: Missing class docstring (missing-docstring)
C: 14, 4: Missing method docstring (missing-docstring)
C: 17, 4: Missing method docstring (missing-docstring)
R: 17, 4: Method could be a function (no-self-use)
C: 20, 4: Missing method docstring (missing-docstring)
C: 45, 4: Missing method docstring (missing-docstring)
C: 49, 4: Missing method docstring (missing-docstring)
C: 56, 4: Missing method docstring (missing-docstring)
C: 62, 4: Missing method docstring (missing-docstring)
C: 84, 4: Missing method docstring (missing-docstring)
C: 87, 0: Missing function docstring (missing-docstring)
```

Report

=====

62 statements analysed.

Global evaluation

Your code has been rated at 7.26/10

Automate the process

Create .pylintrc file in your project root dir.

```
$ pylint --generate-rcfile > .pylintrc
$ cat .pylintrc | grep line-length
max-line-length=120

$pylint --rcfile=.pylint lab/gui/tkinter/main.py
No config file found, using default configuration
***** Module main
C:  9, 0: Trailing whitespace (trailing-whitespace)
C: 19, 0: Trailing whitespace (trailing-whitespace)
C: 23, 0: Trailing whitespace (trailing-whitespace)
C: 32, 0: Trailing whitespace (trailing-whitespace)
```