Documenting Python Code

Sebastian Raschka

Jan 22, 2014

PEP 8

- Limit all lines to a maximum of 79 characters.

- For [...] docstrings or comments, the line length should be limited to 72 characters.

http://www.python.org/dev/peps/pep-0008/

PEP 8
(Naming Conventions)

- **Modules** should have short, all-lowercase names

  e.g., `screenlamp`

- **Class** names should normally use the CapWords convention.

  e.g., `PdbObj`

- **Methods**: lowercase with words separated by underscores as necessary to improve readability.

  e.g., `def get_chain(self, id):`

- **Function** names should be lowercase, with words separated by underscores

  e.g., `def calc_distance(coords1, coords2):`

- **Constants**: All Caps

  e.g., `MAXVAL = 10`

PEP 8
(Public VS Private)

- **_single_leading_underscore**: weak "internal use" indicator. E.g. from M import * does not import objects whose name starts with an underscore.

- **single_trailing_underscore_**: used by convention to avoid conflicts with Python keyword, e.g.

- **__double_leading_underscore**: when naming a class attribute, invokes name mangling (inside class FooBar, __boo becomes _FooBar__boo; see below).

PEP 8
(Comments)

Inline comments are unnecessary and in fact distracting if they state the obvious.
Don't do this:

```
x = x + 1                 # Increment x
```

But sometimes, this is useful:

```
x = x + 1                 # Compensate for border
```

Docstrings
(One liner)

```
def calc_eucl_dist(coords1, coords2):
"""Returns Euclidean distance from two lists of XYZ coordinates"""
    return sum([(j-i)**2 for i,j in zip(coords0,coords1)])**0.5
```

Docstrings
(Multi liner)

```
 9 def _filter_atoms(mol2_cont, atom_dict):
10     '''Searches for atom types in a mol2 file.
11
12     Keyword Arguments:
13     mol2_cont (list): mol2 file content as where each list item represents
14         a line
15     atom_dict (dict): Dictionary that consists of atom types as keys
16         and their allowed charge range as 2 integers in a list
17         (inclusive lower and inclusive upper boundary). If
18         charge list is empty, charge comparison is ignored.
19         e.g., {'O.3':[-0.42, -0.12], 'O.2':[]}
20
21     Returns:
22     List of mol2 lines that contain matches as items in a list.
23
24     '''
25     matched_lines = []
26     for line in mol2_cont:
```

For more info: http://www.python.org/dev/peps/pep-0257/

Why Docstrings?

```
[bash]~ >python3
Python 3.3.3 (default, Nov 26 2013, 10:37:54)
[GCC 4.4.7 20120313 (Red Hat 4.4.7-1)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import screenlamp
>>> help(screenlamp.mol2utils.filter_mol2)
```