

OOP PYTHON coding style

for the BGCE Project 2015

Benjamin R  th

October 29, 2015

In this we are going to collect coding style guidelines for our project

1 General Style

- follow PyCharm style (PEP8)
- use meaningful names in any case! No abbreviations without a comment

2 Functions and Variables

- separate different words in variable names and function names with `_` (e.g. `my_variable` instead of `myVariable`)
- only use lowercase for variable and function names

3 Object oriented programming

- use camelcase for class names!
- start class names with uppercase (e.g. `MyObject` instead of `my_object`)
- there is no way for defining private variables in python! the following convention is applied:
 - variables starting with `_` or `__` (e.g. `_private` or `__private`) should not be touched from outside
 - for variable modification and access add appropriate getters and setters
 - from my personal experience and fails: `_` seems to me to mean something like protected and `__` derived. Also to the interpreter! This means, if you want to access variables from a base class via a derived class you have to be careful. See the following example:

```
class base(object):
    def __init__(self):
        self._foo = 1

class derived(base):
    def __init__(self):
        base.__init__(self)
        self._bar = 2
    def print_foo(self):
        print self._foo
```

```
a = derived()
a.print_foo()
```

This doesn't work, if `_foo` has two leading `_` instead of one (`__foo`)!

– see this post for additional information