

# Προγραμματισμός σε C++ & Python & Εφαρμογές στη Ναυπηγική & Ναυτική Μηχανολογία

ΣΝΜΜ 2019

Μάθημα 4B: OOP-Classes

**Γεώργιος Παπαλάμπρου**

Επίκουρος Καθηγητής ΕΜΠ

`george.papalambrou@lme.ntua.gr`

Εργαστήριο Ναυτικής Μηχανολογίας (Κτίριο Λ)  
Σχολή Ναυπηγών Μηχανολόγων Μηχανικών  
Εθνικό Μετσόβιο Πολυτεχνείο

March 22, 2019

# Περιεχόμενα

## 1 Classes

# Περιεχόμενο Μαθήματος

- Εβδομάδα 1. Α. Εισαγωγή. Η γλώσσα. Το περιβάλλον Linux. Command line. Python interpreter. Ιστοσελίδα μαθήματος. Βιβλιογραφία. Editors: Sublime, Spyder. Β. Εισαγωγή στην γλώσσα Python. Hello World.
- Εβδομάδα 2. Α. Data types. Loops. Control. Β. Παραδείγματα
- Εβδομάδα 3. Functions. Modules
- Εβδομάδα 4. **OOP. Classes**
- Εβδομάδα 5. Α. Παραδείγματα: Μέτρηση και επεξεργασία δεδομένων  
Β. Βιβλιοθήκες Numpy, SciPy. Errors-Exceptions. Παραδείγματα: Γραμμική άλγεβρα, Γραφικά
- Εβδομάδα 6. Εφαρμογή: Hardware. Πλατφόρμες. Πρωτόκολλα. Βασικό I/O Εφαρμογή: Neural Networks. Machine Learning
- **ΑΝΑΠΛΗΡΩΣΗ: 1+1 ΩΡΑ, E5: File I/O, E6: Exceptions**

# Εισαγωγή

Για σήμερα η παράδοση προέρχεται από:

- **Core Python Programming, Wesley Chun**  
Chapter 13. OOP

# Classes: - [Βιβλίο: Wesley Chun]

Μελετήστε τα: 13.3 - 13.11.

▼ Chapter 13. Object-Oriented Programming	541
Section 13.1. Introduction	542
Section 13.2. Object-Oriented Programming	550
Section 13.3. Classes	553
Section 13.4. Class Attributes	555
Section 13.5. Instances	561
Section 13.6. Instance Attributes	566
Section 13.7. Binding and Method Invocation	575
Section 13.8. Static Methods and Class Methods	577
Section 13.9. Composition	579
Section 13.10. Subclassing and Derivation	580
Section 13.11. Inheritance	582
Section 13.12. Built-in Functions for Classes, Instances, and Other Ob...	593
Section 13.13. Customizing Classes with Special Methods	598
Section 13.14. Privacy	617

# Classes - PyCharm

- Εγκαταστήστε το IDE περιβάλλον PyCharm (<https://www.jetbrains.com/pycharm/>) - **Download "Community"**
- Ξεκινήστε φτιάχνοντας Project
- Δυνατότητα για εκτέλεση (Run), Debug, Execute selection in console
- Δυνατότητα για επιλογή Interpreter, Modules, ...
- Δείτε στην αρχή τα Quick Tours

# Classes - PyCharm - Αρχείο: class1.py

```
4 """Cal class definition"""
5 # pi is a class variable
6 pi = 3.142
7
8 def __init__(self, radius):
9     # self.radius is an instance variable
10    self.radius = radius
11
12 def area(self):
13    return self.pi * (self.radius ** 2)
14
15
16 class MyClass(object):
17    'MyClass class definition'
18    myVersion = '1.1' # static data
19    def showMyVersion(self): # method
20        print('MyClass.myVersion')
21
22
23 # 13.1 Wesley Chun
24 class P(object):
25    pass
26
27 class C(P):
28    pass
29
30
31 ..
```

Python Console

```
In[16]: C.__bases__
Out[16]: (_main__.P,)
```

```
In[17]:
```

Special Variables

- a = (Cal) < \_main\_\_.Cal object at 0x7fee85...
- c = (C) < \_main\_\_.C object at 0x7fee85408...
- mc = (MyClass) < \_main\_\_.MyClass object...

Python Debugger Extension Available: Cython extension speeds up Python debugging (... (today 5:26 μ)) 67:15 LF: UTF-8 : 4 spaces : 🔍 🔄

# Classes - PyCharm

The screenshot displays the PyCharm IDE interface. The top toolbar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The Project tool window on the left shows the project structure for 'course\_SNAME\_Python', with 'class1.py' selected. The main editor window shows the following Python code in 'class1.py':

```
31
32 """ Some echo to command line """
33 # select code + right-click -> "execute selection in console"
34 a = Cal(32)
35 a.area()
36 # Output: 3217.408
37
38 a.pi
39 # Output: 3.142
40
41 a.pi = 43
42 a.pi
43 # Output: 43
44
45 b = Cal(44)
46 b.area()
47 # Output: 6082.912
48
49 b.pi
50 # Output: 3.142
51
```

The Python Console at the bottom shows the execution output:

```
pass
In [15]: c=C()
Out[15]: c._class
In [16]: c._bases
Out[16]: (<_main__._C, )
In [17]: a.pi
Out[17]: 3.142
In [18]:
```

The Special Variables panel on the right lists the objects created during execution:

- a = (Cal) <\_main\_\_.\_Cal object at 0x7fee85
- c = (C) <\_main\_\_.\_C object at 0x7fee85408
- mc = {MyClass} <\_main\_\_.\_MyClass object

The bottom status bar indicates: Python Debugger Extension Available: Cython extension speeds up Python debugging // I... (today 5:26 np) 39:3 LF : UTF-8 : 4 spaces : [icon]