

Python Programming

Modules

Mostafa S. Ibrahim

Teaching, Training and Coaching since more than a decade!

Artificial Intelligence & Computer Vision Researcher

PhD from Simon Fraser University - Canada

Bachelor / Msc from Cairo University - Egypt

Ex-(Software Engineer / ICPC World Finalist)



Modules & Packages

- In practice, real projects are so big!
 - We can't keep all code in a single file
 - We keep **breaking** the project to smaller tasks.
 - This is called Modular programming
 - Why: Simplicity, Maintainability, Reusability, etc
- Python way
 - The smallest task in python is called a **module**
 - **Module**: A single .py file focusing on a specific task(s)
 - **Package**: Group of modules (py files), so a bigger sub-problem scope
 - **Scoping**: Each module has a different namespace ⇒ No name collision
 - E.g. If same variable name in 2 modules, no problem at all

Hospital System \Rightarrow Sub-Tasks

- We wrote the code as all in a single file. Here is a split to **4 modules**:
- `utilities.py`
 - `def input_valid_int(msg, start = 0, end = None)`
- `patient.py`
 - `class Patient:`
- `hospitalmgr.py`
 - `class HospitalManger:`
- `frontendmgr.py`
 - `class FrontendManager:`
- Now: Assign to different team members different tasks
 - This is how we work as a team. Take a subtask and develop a module / package

Modules: Develop, use or install

- **In practice we may:**
- 1) develop our **own** modules (like the hospital system)
- 2) make use of the **built-in modules**
 - We already used built-in functions (builtins.py) such as max, len, dir
 - Python has more built-in modules that are focused on specific tasks
- 3) install **external packages** to use its modules
 - Python has a great community!

Using built-in modules

```
2  # please we wanna use math module
3  import math
4
5  print(math.sqrt(16))      # 4
6  print(math.factorial(5))  # 120
7  print(math.pi)           # 3.141592653589793
8  print(math.cos(math.pi/2)) # 0 - don't cos(90)
9
10 # More: Google python math module Or control over math
11
12 import math as XXX
13 print(XXX.pi)
14
15 from math import pi, factorial
16 print(pi)
17 print(factorial(5))
18
19 from math import * # all is now visible: avoid
```

Modules in functions

```
3 def f():
4     from math import pi, factorial
5     print(pi)
6     print(factorial(5))
7
8     #SyntaxError: import * only allowed at module level
9     #from math import *
10
11 if __name__ == '__main__':
12     f()
```

“Acquire knowledge and impart it to the people.”

“Seek knowledge from the Cradle to the Grave.”