

# *Python Programming*

## Enumerate and Override

**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)*



# Enumerate Func

- Built-in
- Helps in iterating on sequences

```
2  # enumerate() function returns an enumerated object.
3  for item in enumerate(range(5, 8)):
4      #print(type(item))    # <class 'tuple'>
5      idx, value = item
6      print(idx, value)
7      """
8      0 5
9      1 6
10     2 7 """
11  for idx, value in enumerate(range(5, 8)):
12      print(idx, value)
13      """
14     0 5
15     1 6
16     2 7 """
17  for idx, value in enumerate('ali'):
18      print(idx, value)
19      """
20     0 a
21     1 l
22     2 i """
```

# Override!

- Using a previously used name (of a function or a variable), override the old one
  - It is gone!
- Be careful!

```
1
2 def sum(a, b):
3     return a + b
4
5 print(sum(2, 3)) # 5
6
7
8 # now we override previous one. You can't call
9 def sum(a, b, c):
10     return a + b + c
11
12 # TypeError: sum() missing 1 required positional argument: 'c'
13 # print(sum(2, 3)) # 5
14
15 print(sum(2, 3, 5)) # 10
16
17 # be careful with functions with the same name
18 # With decorator we can do some other nice stuff
19
20 print(len('mostafa')) # 7
21
22 len = 1 # now len binds to something else
23 # TypeError: 'int' object is not callable
24 # print(len('mostafa')) # 7
```

*“Acquire knowledge and impart it to the people.”*

*“Seek knowledge from the Cradle to the Grave.”*