



AL NAFI,
A company with a focus on education,
wellbeing and renewable energy.

Python Primer 104

Code and Functions Modules

Dua of the day

The righteous dua that Abu Bakr (r) made at the end of his lifetime.

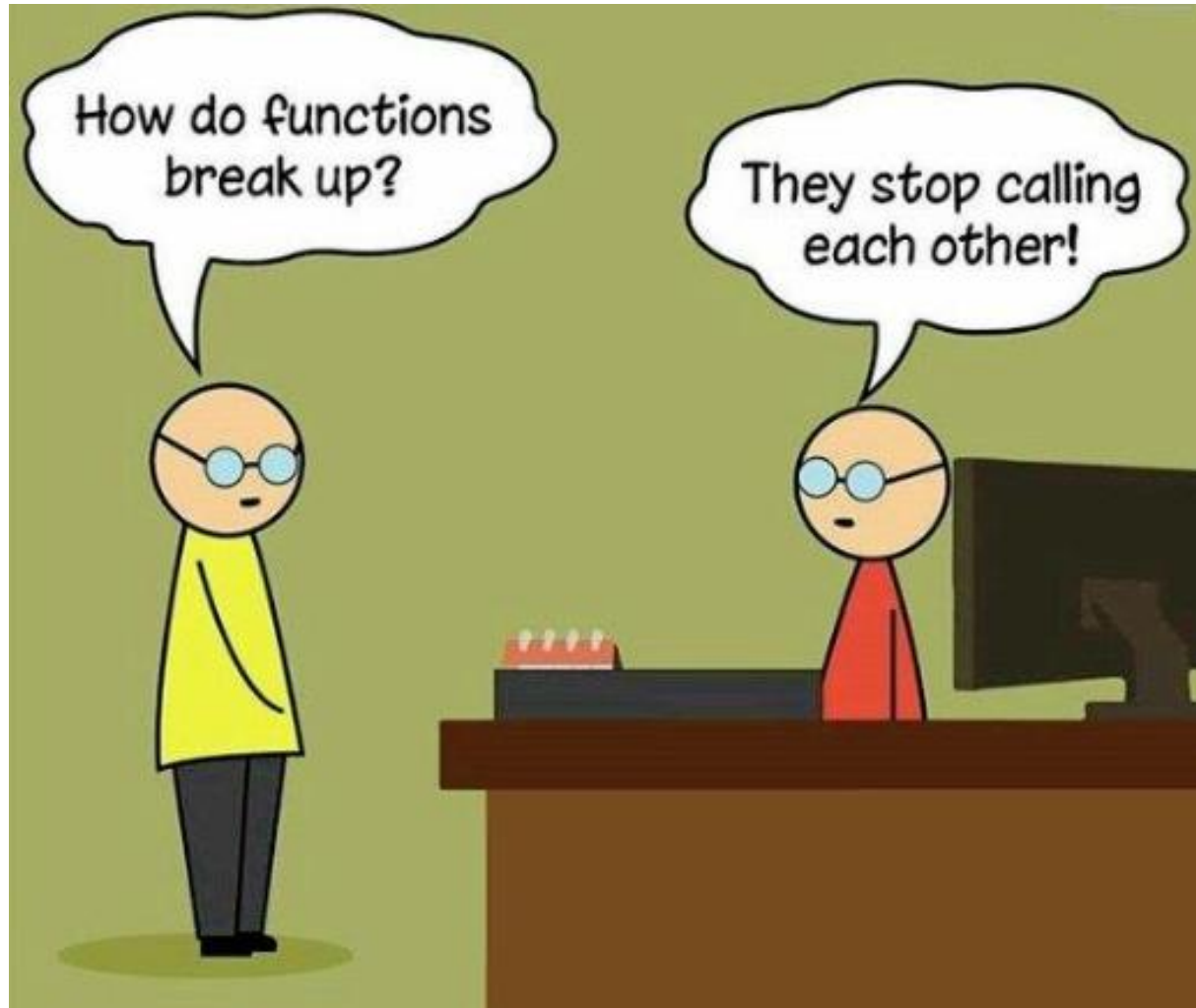
اللَّهُمَّ اجْعَلْ خَيْرَ زَمَانِي آخِرَهُ، وَخَيْرَ عَمَلِي خَوَاتِمَهُ، وَخَيْرَ أَيَّامِي يَوْمَ الْقَاكَ

“O Allah, let the best of my lifetime be its ending, and my best deed be that which I seal [my life with], and the best of my days the day I meet You.”

Reference: The Dua is taken from Sheikh Omer Sulaiman Dua Compilation.

Study, Rinse and Repeat

- Please subscribe to our [YouTube Channel](#) to be on top of your studies.
- Please logon to our website <https://alnafi.com/login/>
- Use your username and password to logon
- Please keep an eye on zone@alnafi.com emails
- Please review the videos of 30 minutes daily
- Please review the notes daily.
- Please take time for clearing up your mind and reflect on how things are proceeding in your studies.



Recycling code functions and modules

Talk about recycling

In the programming world, reuse is just as important. Obviously, your program won't disappear under a pile of garbage, but if you don't reuse some of what you're doing, you'll eventually wear your fingers down to painful stubs through overtyping. Reuse also makes your code shorter and easier to read.



Functions Come handy 😊

Using list and range

```
list(range(0, 5))
```

Or try another example

```
list(range(0, 1000))
```

Why functions?

When you're writing simple programs, functions are handy. Once you start writing long, more complicated programs, like machine learning, deep learning, AI, and games functions are essential (assuming you want to finish writing your program this century).

Parts of Functions

```
testfunc('mary')
```

```
def testfunc(myname):
```

```
    print('hello %s' % myname)
```


Multiple ways for writing functions

```
testfunc('Mohammad', 'Ali')
```

```
def testfunc(fname, lname):  
testfunc('Mohammad', 'Ali')
```

```
print('Hello %s %s' % (fname, lname))
```

Creating variables

```
firstname = 'Mohammad'
```

```
lastname = 'Omar'
```

```
testfunc(firstname, lastname)
```

```
print (firstname, lastname)
```

Using `return`

```
def variable_test ():  
    first_variable = 10  
    second_variable = 20  
    return first_variable * second_variable  
print(variable_test())
```

Variable Outside the functions

```
another_variable = 100
```

```
def variable_test2():
```

```
    first_variable = 10
```

```
    second_variable = 20
```

```
    return first_variable * second_variable * another_variable
```

```
print(variable_test2())
```

Python Modules

Python - Modules. ... A **module** is a **Python** object with arbitrarily named attributes that you can bind and reference. Simply, a **module** is a file consisting of **Python**code. A **module** can define functions, classes and variables. A **module** can also include runnable code.

OR

Modules are used to group functions, variables, and other things together into larger, more powerful programs. Some modules are built into Python, and you can download other modules separately.

Modules examples

```
import time  
print(time.asctime())
```

Here, the import command is used to tell Python that we want to use the module time.

The function asctime is a part of the time module that returns the current date and time, as a string.

Another module example

```
age=11
```

```
if age >= 10 and age <= 13:
```

```
    print('What is 13 + 49 + 84 + 155 + 97? A headache!')
```

```
else:
```

```
    print('Huh?')
```

جزاك الله

Please send us your questions at zone@alnafi.com
We will only answer our Nafi Members. So please
quote your membership number within the email.