

GATE WAY TO ML

PYTHON
A CRASH COURSE

DO YOU KNOW CODING????.....

WHY IS IT IMPORTANT TO US?





PYTHON

YOU DO NOT NEED TO BE A
PYTHON DEVELOPER TO
GET STARTED USING THE
PYTHON ECOSYSTEM FOR
MACHINE LEARNING.

Python is a general purpose interpreted programming language.

1. It is easy to learn and use primarily because the language focuses on readability.
2. It's a dynamic language and very suited to development and quick prototyping with the power to support the development of large applications..

PYTHON

YOU DO NOT NEED TO BE A
PYTHON DEVELOPER TO
GET STARTED USING THE
PYTHON ECOSYSTEM FOR
MACHINE LEARNING.

3. It is also widely used for machine learning and data science because of the excellent library support and because it is a general purpose programming language (unlike R or Matlab).
4. It is a popular language in general, consistently appearing in the top 10 programming languages in surveys on StackOverflow.

PYTHON

**YOU DO NOT NEED TO BE A
PYTHON DEVELOPER TO
GET STARTED USING THE
PYTHON ECOSYSTEM FOR
MACHINE LEARNING.**

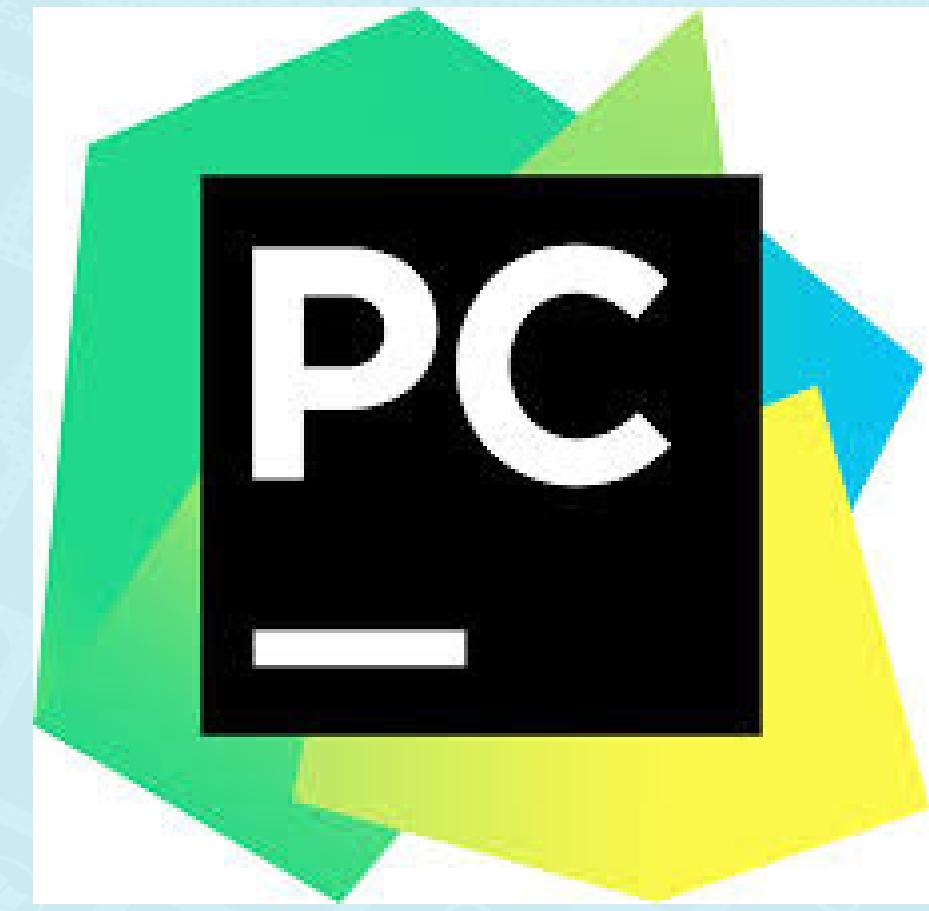
The philosophy of Python is captured in the Zen of Python which includes phrases like:

- Beautiful is better than ugly.
- Explicit is better than implicit. Simple is better than complex.
- Complex is better than complicated.
- Flat is better than nested
- Sparse is better than dense.
- Readability counts.

ENVIRONMENT SETUP

PY CHARM IDEA
PYTHON 3.7

Py Charm is an integrated development environment used in computer programming, specifically for the Python language.



HELLO WORLD

print "hello world"

The screenshot shows the PyCharm 2.7.1 IDE interface. The title bar reads "TestProject - [C:\Users\Shellver\PycharmProjects\TestProject] - ...\\test.py - PyCharm 2.7.1". The menu bar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The toolbar has various icons for file operations like Open, Save, and Run. The Project tool window on the left shows a "TestProject" folder containing a "test.py" file. The code editor on the right contains the single line of Python code: "print 'hello world'". Below the code editor is the Debug tool window, which displays the output of the run command: "C:\Python27\python.exe "C:\Program Files (x86)\JetBrains\PyCharm 2.7.1\helpers\pydev\pydevd.py" --multiproc --client 127.0.0.1 --port 50412 --file C:/Users/Shellver/PycharmProjects/TestProject/test.py". It also shows the connection status: "pydev debugger: process 1352 is connecting", the connection message: "Connected to pydev debugger (build 125.92)", the print statement: "hello world", and the process exit message: "Process finished with exit code -1". The bottom status bar shows the error message "Error running test: Cannot run program "C:\Python27\python.exe" (in directory "C:\Users\Shellver\PycharmProjects\TestProject"): CreateProcess error=123, The filename, directory name, or volume label ... (3 minutes ago)" and the system status: 8:1, UTF-8, Insert.

No need to explicitly define the data type of a variable

only space indentation is required no {}, ; etc

python don't have pointer

python have lot of libraries

ENOUGH TALK

LET'S CODE



ASSIGNMENTS

STRING

```
# STRINGS  
DATA ='HELLO WORLD'  
PRINT(DATA[0])  
PRINT(LEN (DATA))PRINT(DATA)
```

ASSIGNMENTS

NUMBERS

```
# NUMBERS
```

```
VALUE = 123.1
```

```
PRINT(VALUE)
```

```
VALUE = 10
```

```
PRINT(VALUE)
```

ASSIGNMENTS

BOOLEAN

```
# BOOLEAN  
A = TRUE  
B = FALSE  
PRINT(A, B)
```

FLOW CONTROL



There are three main types of flow control that you need to learn:

If-Then-Else conditions,
For-Loops
While-Loops.

FLOW CONTROL

IF - ELSE

```
VALUE = 99  
IF VALUE == 99  
:PRINT('THAT IS FAST')  
ELIF VALUE > 200:  
PRINT('THAT IS TOO FAST')  
ELSE:  
PRINT('THAT IS SAFE')
```

FLOW CONTROL

FOR LOOP

```
# FOR-LOOP  
FOR I IN RANGE(10):  
    PRINT(I)
```

FLOW CONTROL

WHILE LOOP

```
# WHILE-LOOP  
I = 0  
WHILE I < 10:  
    PRINT(I)  
    I += 1
```

DATA STRUCTURES



There are three data structures in Python that you will find the most used and useful. They are:

- tuples
- lists
- dictionaries.

DATA STRUCTURE

TUPLE

Tuples are read-only collections of items.

```
A = (1, 2, 3)
```

```
PRINT(A)
```

DATA STRUCTURE

LIST

Lists use the square bracket notation and can be index using array notation.

```
MYLIST = [1, 2, 3]
PRINT("ZEROOTH VALUE: %D") %MYLIST[0]
MYLIST.APPEND(4)
PRINT("LIST LENGTH: %D") % LEN(MYLIST)
FOR VALUE IN MYLIST:
    PRINT(VALUE)
```

DATA STRUCTURE

DICTIONARIES

Dictionaries are mappings of names to values, like key-value pairs. Note the use of the curlybracket and colon notations when defining the dictionary

DATA STRUCTURE

DICTIONARIES

```
mydict = {'a': 1,'b': 2,'c': 3}  
print("A value: %d") % mydict['a']  
    mydict['a'] = 11  
print("A value: %d") % mydict['a']  
print("Keys: %s") % mydict.keys()  
print("Values: %s") % mydict.values()  
for key in mydict.keys():  
    print(mydict[key])
```

FUNCTIONS



The biggest gotcha with Python is the whitespace. Ensure that you have an empty new line after indented code. The example below defines a new function to calculate the sum of two values and calls the function with two arguments.

```
# Sum function  
def mysum(x, y):  
    return x + y  
  
# Test sum function  
result = mysum(1, 3)  
print(result)
```

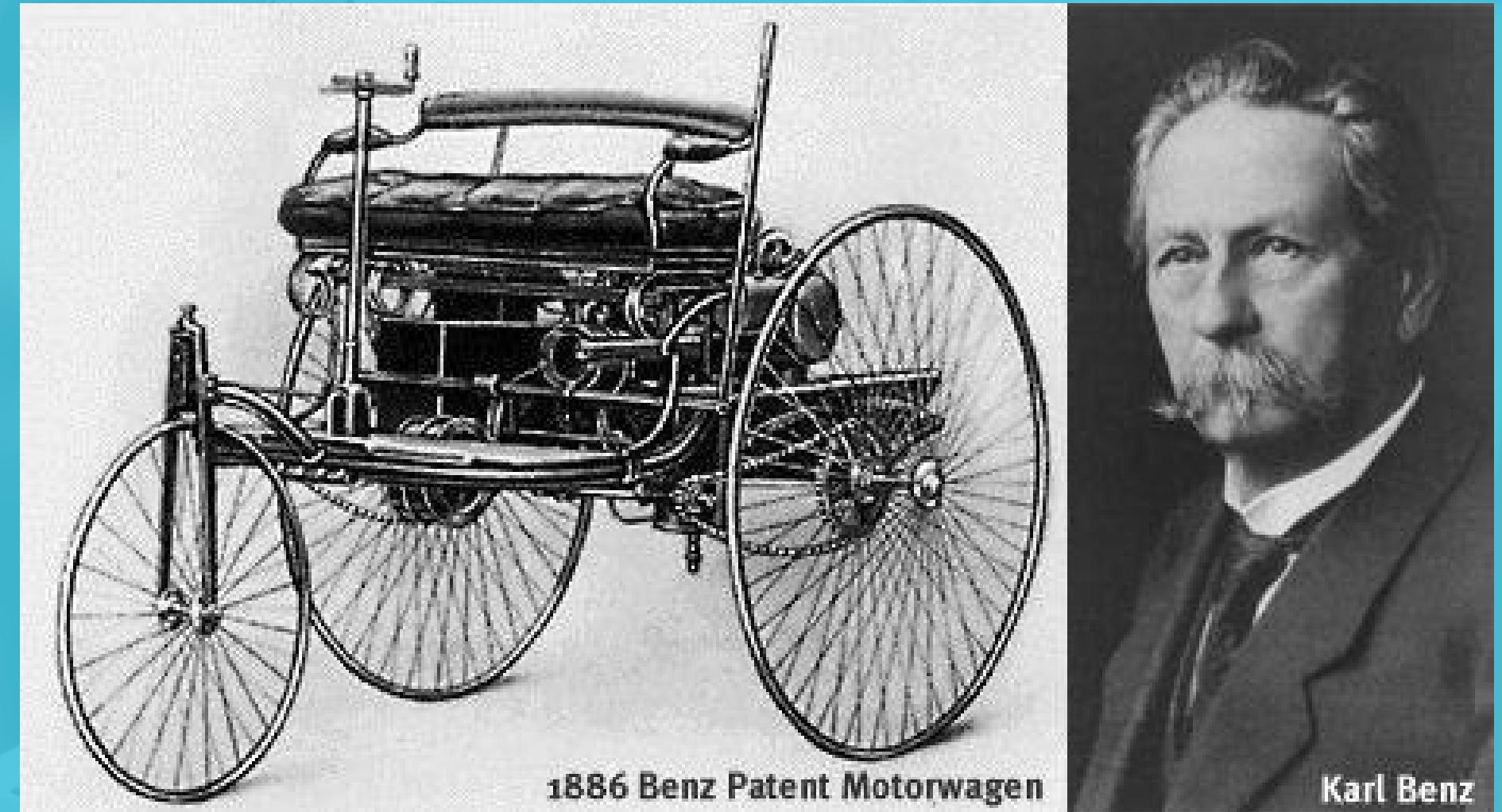
LIBRARIES IN PYTHON

Python library is a collection of functions and methods that allows you to perform lots of actions without writing your own code.

example numpy,pandas,tensorflow etc



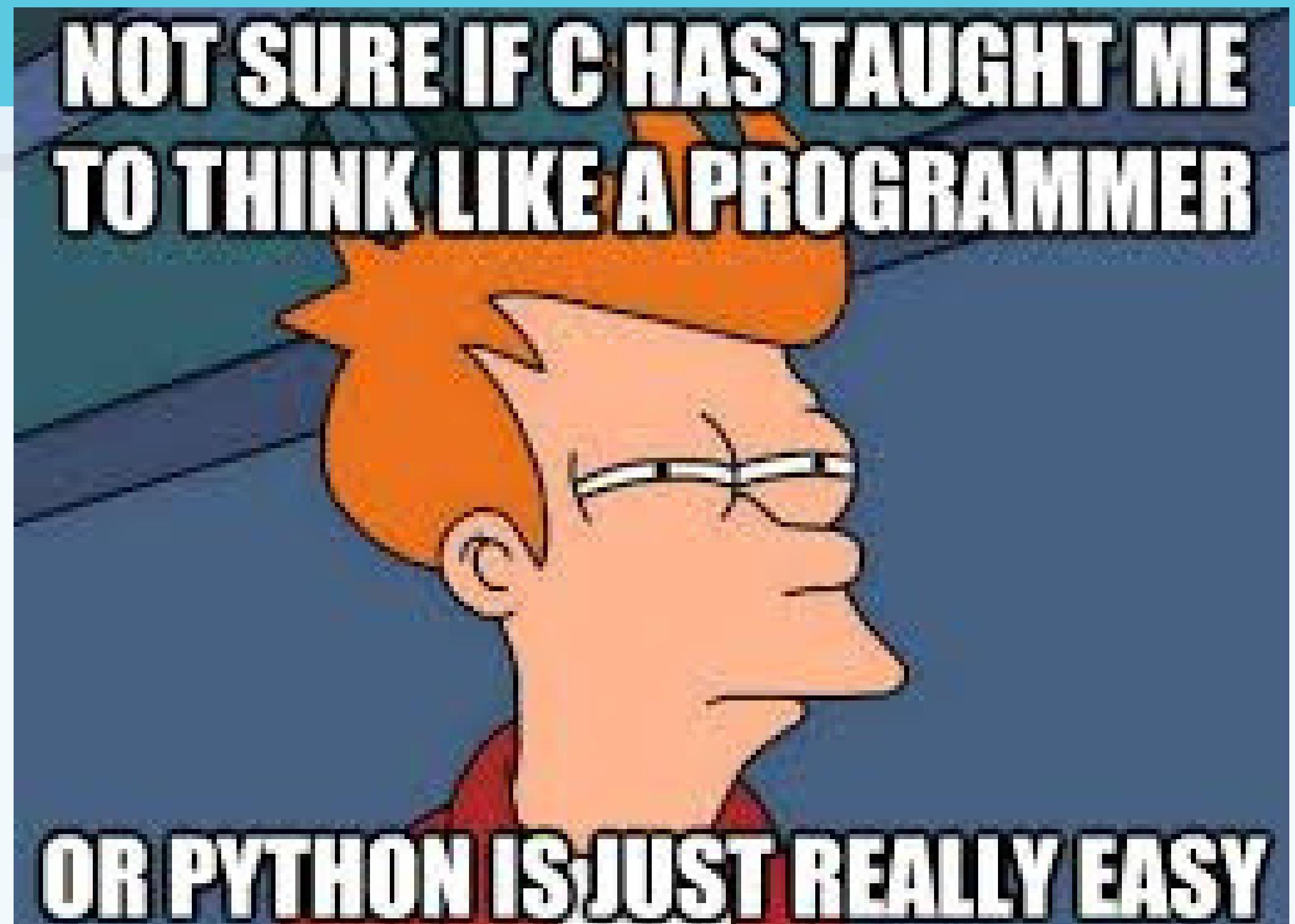
CAR - WHEEL ANALOGY TO PYTHON LIBRARY



Wheel has been already invented, So the person who invented car didn't waste his time in re-inventing the wheel again, Here car is an invention which has imported wheel. so, wheel is a library (can be used in other inventions as it is).

WHAT YOU FEEL ABOUT PYTHON NOW??

Lets find out
how much
you know?



FACTORIAL

Python Program to Find the Factorial of a Number



MULTIPLICATION TABLE

Python Program to Display the multiplication Table of a given number

NEXT HANDS ON SECTION ON
scipy and scikit-learn