Introduction to Python

Programming = Writing English

- Easy to get started.
- Well known for its simplicity.
- Well structured code.
- Main intension was to teach kids how to program.
- **Guido Van Rossum** a dutch programmer invented Python while working for Google.

Credits - Image from Internet

Zen / Poem of Python by Tim Peters

```
>>> import this
```

```
   txt
   The Zen of Python, by Tim Peters

   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.
   Special cases are not special enough to break the rules.
   Although practicality beats purity.
   Errors should never pass silently.
   Unless explicitly silenced.
   In the face of ambiguity, refuse the temptation to guess.
   There should be one and preferably only one obvious way to do it.
   Although that way may not be obvious at first unless you are Dutch.
   Now is better than never.
   Although never is often better than right now.
   If the implementation is hard to explain, it is a bad idea.
   If the implementation is easy to explain, it may be a good idea.
   Namespaces are one honking great idea let us do more of those!
```

Things that can be done using Python

- **Web Development**
  - Flask
  - Dash
  - Django
- **Game Development**
  - Pygame
  - OpenGL
- **Numeric & Scientific Computation**
  - Numpy
  - Scipy
  - OpenCV
  - Pandas
- **Data Analysis**
  - Pandas
  - Statsmodel
  - Statistics
- **Data Visualization**
  - Matplotlib
  - Plotly
  - Bokeh
- **Machine Learning and Artificial Intelligence (Research & Development)**
  - Scikit-learn
  - Tensorflow
  - Pytorch
  - Keras
  - Scikit-image
- **Mobile App Development**
  - Kivy
- **Desktop Application**
  - Tkinter
- **Web Scraping**
  - BeautifulSoup
  - Selenium (Automation)
  - Scrapy

The list goes on ...

Companies that use Python

Almost all companies use Python for their requirements. Some of the big companies that use python

- NASA
- Google
- IBM
- Yahoo
- Facebook
- Netflix
- YouTube
- etc

Note - Programming languages are created by other programming languages which are again created by other programming languages.

Unix → C → Python

Why Python?

- Python is very simple that anybody can get start.
- Personally I imporved my coding skills and done projects as it is easy and flexible.
- The community is so big that you can find anything that can be solved through programming.

`C` Program

```
#include<stdio.h>;
int main() {
    printf("Hello World!!!");
    return 0;
}
```

`Java` Program

```
import io.*;

class Simple {
    public static void main(String args[]){
        System.out.println("Hello World!!");
    }
}
```

`Python` Program

In [1]:
```
print("Hello World !!")
```

```
Hello World !!
```

Variables

- Mainly used to store the information or values
- Come in usage with Data Types

Initialization of a variable

- Initializing is simply assigning a value to a variable or placeholder.

In [2]:
```
var_name = 1231.9898
print(var_name)
```

```
1231.9898
```

Program Execution

- **Interpreter** → A program that is intended to execute the written program **line by line**.
- **Compiler** → A method of converting the written program into a **machine code** which is executed by a Computer. This method is done by a Compiler. In simple it is just a translator.

Interpreter

- Successful execution

In [3]:
```
a = 10
print("The value of a", a)
b = 20
print("The value of b", b)
```

```
The value of a 10
The value of b 20
```

- **Error** in the execution since the interpreter checks line by line. Here `d` is defined after it is been used and hence not supported in any language.

In [4]:
```
c = 10
print("the value of c is", c)
print("the value of d is", d)
d = 20
```

```
the value of c is 10
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-4-ef19e5fd0539> in <module>
      1 c = 10
      2 print("the value of c is", c)
----> 3 print("the value of d is", d)
      4 d = 20

NameError: name 'd' is not defined
```

Data Types

It is used to let the compiler know how the programmer (yourself) wants to use the data.

- `int` → numbers -

  `1, 2, 3`
- `float` → decimal values -

  `.5, .54, .54345`
- `bool` →

  `True, False`
- `str` →

  `"hi", 'hello', 'd'`
- `list` →

  `[int, float, str]` → `[1, 2.11, "hi"]`
- `tuple` →

  `(int, float, str)` → `(1, 2, 3, .465, .5, 6, "hi")`
- `dict` →

  ```
  {
    "key1" : "value1",
    "key2" : "value2",
  }
  ```

Questions

- Are `Interpreter` and `Compiler` same?
- In terms of process execution, are `interpreter` and `compiler` different?
- Which is faster `Compiler` or `Interpreter` ? Why?
- What makes `Python` slow when compared with `C` and `Java` ?
- Does `Python` have a `Compiler` ?

Helpful Links