Vault of Codes

Python Programming Internship

Introduction of Python

- Python is developed by **Guido Von Rossum** in the year 1991.
- It is a interpreted language.
- It has numerous libraries and modules to perform various actions and operations.
- Python is known as user-friendly language, because of it’s simple and easy syntax.
- It is one of the recommended programming language for the beginners.
- Python supports Graphical User Interface.
- Python is a object oriented high level programming language.
- It is a free and open source language, which having number of developers world wide.
- It is cross platform language(platform independent) which equally works on different operating systems like windows, Linux, macros etc.
- Python Libraries such as Pandas, Numpy, Matplotlib and scikit Learn is widely used in emerging technologies includes Machine Learning, Data Science etc., in order to find meaningful insights of data and develop a machine learning model for solving the real world problems

Applications of Python

- Web development
- Game Development
- Scientific and numeric applications
- AI and ML
- Desktop GUI
- Software Development

History of Python

- Python was invented by Guido van Rossum in 1991 at CWI in Netherland.
- The idea of Python programming language has taken from the ABC programming language or we can say that ABC is a predecessor of Python language.
- There is also a fact behind the choosing name Python.
- Guido van Rossum was a fan of the popular BBC comedy show of that time, "Monty Python's Flying Circus". So he decided to pick the name Python for his newly created programming language.
- Python has the vast community across the world and releases its version within the short period.

Functions in Python

- A function is defined as a block of code or a set of instructions that performs the specific task.
- Functions also called as modules.
- **Syntax to write a function in python is:**

```
def function_name(parameters):
        body of the function(required task that to be performed)
        return (return value/variable should be mentioned)
```

- **Note**: we need to call the function to perform the required task, if function is not called the body of the function will not be executed.
- A function can return data as a result.

Example: def sum(a, b): # definition of a function

```
        c=a + b
        return c
            a=2
            b=3
            c=sum( a, b) # calling of function
            print("the sum is:", c)
```

Mentioning of parameters are optional and make sure that some functions does not returns any value.
Note that the number of parameters should match with the number of arguements passed in the function.

Advantages of functions:

1)Functions support modular programming through which we can divide the larger programs into smaller one.
2)With the help of functions we can debug the code easily if the program is large.
3)Functions supports reusability of the code. Since we can call the function any number of times to perform the specific task. No need to write the same code once again for completion of the task.

Modules in Python

Modules are simply Python files with the `'.py'` extension, which implement a set of functions.
Modules are imported from other modules using the `import` command.
Modules consist of a series of functions which can be functioned by importing the particular module.
Syntax to import the module in python: import module name as alias_name
Example: import pandas as pd # it is used for data analysis
It is not compulsory to add alias name for the module imported. It is completely optional
Different modules in python: 1)random module for generating random numbers.
2)math module for mathematical operations
3)datetime module for manipulating dates and times
4)numpy module for scientific computing for numerical operations.
5)pandas module for data analysis
6)matplotlib module for data visualization
Note: we can also create our own modules by saving the fiile with .py extension that should consists of desired functions and classes.