VaultofCodes

(Assignment 1 - Task 2)

By
Thatha Keerthi

Table Of Contents

Introduction

- Python is a high-level , interpreted and general-purpose programming language .
- Key Features:
  - Easy to read and write
  - Extensive standard library
  - Supports multiple programming paradigms
  - Open-source and community-driven
- Uses:
  web development ,data analysis , artificial intelligence , scientific computing , automation and scripting , software development etc.

History of Python

- History of python was developed by Guido Van Rossum in the **late eighties and early nineties** at National Research Institute for Mathematics and Computer science in the Netherlands .
- Python is derived from many other languages, including ABC, Modula-3, C ,C++, Algol-68, SmallTalk and Unix shell and other scripting languages.
- Python is copyrighted . Like Perl, Python source code is now available under the **GNU General Public Licence**(GPL).
- Python is now maintained by core development team at the institute , although Guido Van Rossum still holds a vital role in directing it's progress.

Functions and Modules

Functions

- Definition :
  - A function is a block of reusable code that performs a specific task.
- Syntax:

```
def function_name(parameters):
    """docstring"""
    # code
    return value
```

- Example;

```
def greet(name):
    """Greets the person with the given name."""
    return f"Hello, {name}!"

print(greet("Alice"))  # Output: Hello, Alice!
```

- Key points
  - Defining: Use ‘def’ keyword.
  - Calling: Use ‘function_name()’.
  - Parameters: Inputs to the function.
  - Return Value: Output from the function.
  - Docstring: Optional documentation string.

Modules

- Definition
  - A module is a file containing Python code, typically with related functions and variables.
- Creating a module:

```python
# my_module.py
def greet(name):
    return f"Hello, {name}!"

PI = 3.14159
```

- Using a module:

```
import my_module

print(my_module.greet("Alice"))  # Output: Hello, Alice!
print(my_module.PI)  # Output: 3.14159
```

- Key points:
  - Importing: Use ‘import module_name’.
  - From Import: Use ‘from module_name import item’.
  - Renaming: Use ‘import module_name as alias’.
  - Built-in Modules: Examples include ‘math’, ‘os’, ‘sys’ .

Conclusion

- Python supports both function-oriented and structure- oriented programming. It has features of dynamic memory management which can make use of computational resources efficiently.
- It is also compatible with all popular operating systems and platforms. Hence this language can be universally accepted by all programmers.
- Implementation to develop Python based GUI Application.
- Directing interacted with os using Python Programming.

THANK YOU!!!