Frederick Laroche

NMIT

SDV602

Assessment 3

Contents

[Python programming language 2](#_Toc182839949)

[Introduction 2](#_Toc182839950)

[Heritage and philosophy 2](#_Toc182839951)

[Development platforms and libraries 3](#_Toc182839952)

[Characteristics ,strength and weaknesses 3](#_Toc182839953)

[Conclusion 4](#_Toc182839954)

[References 5](#_Toc182839955)

# Python programming language

## Introduction

In this document, I will share my learning journey and my research on Python, the programming language. It will be from the point of view of a beginner who is currently learning it. I will go through its heritage, philosophy ,development platforms , libraries and its characteristics(strength and weaknesses).

## Heritage and philosophy

As I started learning programming with Python, I discovered that it is one of the most widely used programming languages. It was created by Guido van Rossum during his Christmas holiday in 1989. Python was named after “Monty Python's Flying Circus show” because he was a fan (van Rossum, 2018). Van Rossum was working with a language called ABC at CWI (a research institute), and while he liked many of its features, he was frustrated by its lack of extensibility. According to van Rossum himself, he wanted to make the language more extensible so users could add their own modules and features.(Python Institute, 2024).

Python was designed with a specific philosophy in mind, the “Zen of python” also known as [PEP 20](https://www.python.org/dev/peps/pep-0020). “It is a collection of 19 [guiding principles](https://en.wikipedia.org/wiki/Coding_conventions) for writing [computer programs](https://en.wikipedia.org/wiki/Computer_program) that influence the design of the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) [programming language](https://en.wikipedia.org/wiki/Programming_language)”(Kenneth, 2019) .

These principle has helped me understand the language more easily as it was made to be simple to write and read for beginners. When you type ‘import this’ in Python, it shows these principles in the console(see image below). This includes “Simple is better than complex” and “Readability counts” (Peters, 2004). The Python Software Foundation (PSF) is in charge of the Python's development and makes sure it remains free and open source (Python Institute, 2024).

A screenshot of a computer screen

Description automatically generated

## Development platforms and libraries

In my Python learning journey, I've been using Visual Studio Code (VS Code) as my main development environment. I chose it because it is what we normally uses in this program because of its extensive extension tools. It also helps me spot errors underlining syntax error which is really helpful when you're learning .I've learned that there are other development environments like PyCharm and Jupyter Notebooks, but they are more difficult to learn ,understand and use as a beginner (Canadian Agency, 2024).

Python’s extensive library collection is impressive as there is a library for each uses case. In my cases, I had to use Mathplotlib to create plot(graph) for my scientific data application. I also used Pandas that is great for data manipulation in tables and Request to get data from internet (JSNDrop).There is many more and I think van Rossum was successful by making Python more extensible by the large collection of library available.

## Characteristics ,strength and weaknesses

Python has several distinct characteristics that make it unique as a programming language. One of is its typing system. Python uses both strong and dynamic typing.

With strong typing, Python’s operations between incompatible types will result in errors. For example, trying to add a number to a text string will cause an error. This is to prevent mistakes in the code (Wikipedia contributors, 2024).

Then the dynamic typing means that Python determines the type of a variable when the program runs, based on what value is assigned to it. This differs from languages like C++ where you must declare the type before using a variable.

Python also has mutability in its data types. This means that some data types can be modified after creation (mutable) but not all data type can be modified, some are immutable. For example, lists and dictionaries are mutable but strings and tuples are immutable. This means that once they are defined, their values can or cannot be modified (Great Learning Team, 2024).

Python advantages:

* Code readability by its syntax and indentation style
* Extensive library collection
* Large community for support
* Cross-platform compatibility - Code can be run on different OS

Python limitations:

* Slower execution speed compared to compiled languages (Canadian Agency, 2024)
* Higher memory consumption
* Limited support for mobile application development
* The Global interpreter lock restrain python to have an optimized multithread

## Conclusion

As someone who is just starting to learn programming, I find Python to be a great first language. It is easier to read and understand compared to JavaScript code. The naming conventions make it easier to read and name variable. I still have some trouble with the indentation rules as I would have some code out of my class without me realising because the indentation was wrong. I am adapting but having brackets would , in my opinion, make it easier to ensure that codes block is includes in a class.

Yet , I think Python is great by its versatility as it can cater to data-oriented project , machine learning and also web development.

# References

Canadian Agency. (2024). Pros and cons of Python programming language.

<https://canadian.agency/pros-and-cons-of-python-programming-language/>

Hunter, J., Dale, D., Firing, E., & Droettboom, M. (2023). Matplotlib: Visualization with Python.

<https://matplotlib.org/>

McKinney, W. (2023). Pandas - Python Data Analysis Library.

<https://pandas.pydata.org/>

Peters, T. (2004). PEP 20 - The Zen of Python.

<https://www.python.org/dev/peps/pep-0020/>

Python Institute. (2024). About Python.

<https://pythoninstitute.org/about-python>

Python Software Foundation. (2024). What is Python? Python FAQs.

<https://docs.python.org/3/faq/general.html>

van Rossum, G. (2018). Python History.

<https://python-history.blogspot.com/>

Reitz, Kenneth (2011–2019). ["Code Style"](http://docs.python-guide.org/en/latest/writing/style/#zen-of-python). *The Hitchhiker’s Guide to Python*. § Zen of Python.

<https://docs.python-guide.org/writing/style/#zen-of-python>