SDV602 Assessment 2

Milestone three

Josh Moss

Table of Contents

[Reflection 2](#_Toc182598325)

[History of python 2](#_Toc182598326)

[The characteristics strengths and weaknesses of python are: 3](#_Toc182598327)

# Reflection

## History of python

Python was designed by Guido van Rossum in 1991 developed by Python Software Foundation and publicly released in 1994. Python's primary focus was on code readability with its “English-like syntax”. ("Python," 2023). Python is an extremely popular language and due to its open-source nature, this helped it grow even more in popularity. “The inspiration for the name came from the BBC’s TV Show – ‘ Monty Python’s Flying Circus’” ("History of Python," 2022). Python has a huge community backing, this community helps create libraries and modules that aid in expanding Python's functionality and features. Python has had three versions throughout its lifespan. These are “Python 1.0, released in 1994; Python 2.0, in 2000; and Python 3.0, in 2008” ("Python," 2023).

Python is an interpreted language. This means that each line of the code is executed line by line by an interpreter. Because of pythons dynamically typed nature types are not specifically declared. Whereas statically typed languages such as C, types must be declared at compile time. Python is also an object-oriented language. This allows for objects to be created that reflect real-world things or concepts. Objects are instanced as “classes” These classes are like blueprints that create objects. These objects then have attributes and methods that help create a particular functionality for a chunk of code. These objects can then be inherited in a chain of hierarchy (Parent and child class). For example, say I have an object Animal. This animal class holds attributes, colour, eyes, fur, etc. I can then inherit these characteristics into a new class called Cat. This cat object will then contain all inherited properties from the animal class.

I love this approach to programming as it promotes code reuse and enhances structure in the codebase. However, I can see how OOP can lead to complexity and over-engineering as simple scripts could result in the same outcome with less code. The IDE we have used for the course was visual studio code. This IDE has extension support to run the python environment. Code libraries that we have used for the assessment were PySimpleGui, Pandas and Matplotlib, PySimpleGui **“**creates graphical user interfaces (GUIs) using Python”. (*PySimpleGUI*, n.d.) This module allows for our applications to have a view that users can interact with. The Pandas module provides you with simple to use tools for data analysis. For the assessment I use Pandas to read CSV data. “Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python” (*Matplotlib — Visualization with Python*, n.d.). This allows me to add charts and other data visualizations to my application.

Venv (virtual environment) is a module that we have been using for our assessments. This allows for the creation of an isolated workspace. The advantage of using a virtual environment is that you can install packages separately from your main system python packages. This can help resolve dependency issues and provides a cleaner environment.

## The characteristics strengths and weaknesses of python:

**Pros:**

* Simple syntax

Python is a very good language for beginners due to its easy-to-read syntax, it allows for higher productivity as code can be written with more efficiency compared to other languages.

* Large community support

Pythons’ community is active and contains plenty of resources and documentation. this makes it easy to find help for problems when they arise.

* Huge range of modules/libraries

Python has a large list of modules that can be used to enhance functionality. for example, all the modules I have used in my assessment made it possible to create my application.

* Flexible

Python has support for all major operating systems (Windows, mac, Linux). Making development with python on any environment as simple as possible

**Cons:**

* Slower than some languages:

Because of its interpreted nature. Python is slower than compiled languages such as C, this can create limitations for applications that relying on quick performance.

* Memory intensive:

“Python is not optimized to reduce memory” ("programmers’ guide to Python: Advantages & disadvantages," 2022). Python can use more memory than other languages making it unsuitable for memory strict software.

* Not great for large applications:

Large-scale projects may have issues with performance and maintenance due to python’s dynamic nature and large emphasis on third party-libraries.

References

*History of Python*. (2022, November 26). GeeksforGeeks. <https://www.geeksforgeeks.org/history-of-python/>

*Python*. (2023, October 16). Encyclopedia Britannica. <https://www.britannica.com/technology/Python-computer-language>

(n.d.). Matplotlib — Visualization with Python. <https://matplotlib.org/>

(n.d.). PySimpleGUI. <https://www.pysimplegui.com/>

*Venv — Creation of virtual environments*. (n.d.). Python documentation. <https://docs.python.org/3/library/venv.html>

*A programmers’ guide to Python: Advantages & disadvantages*. (2022, March 23). Linode Guides & Tutorials. https://www.linode.com/docs/guides/pros-and-cons-of-python/