**a) The heritage and philosophy of the programming language is explored**

Python was developed by Dutch computer scientist Guido van Rossum and has a history of 33 years since its release in 1991. Python was first studied and developed in the late 1980s. There happened to be a programming language called ABC, which was well designed but not very usable and extensible. So Guido van Rossum started to develop a more flexible and easy-to-learn language, which is Python.

“Python is an open source, general purpose programming language. Guido van Rossum developed Python based on the defunct ABC programming language and named it after the Monty Python comedy troupe. Python was designed to be simple, readable, and highly extensible through the use of modules. The first version of Python was released in 1991, and the more fully featured Python 2.0 followed in 2000. Both releases have since been discontinued.

Python continues to increase in popularity and is now ranked as one of the top five languages. It is widely used in data science, machine learning, artificial intelligence, and server & web applications. Many web developers use Python alongside external frameworks, including [Django](https://www.djangoproject.com/) and [Flask](https://flask.palletsprojects.com/en/2.0.x/), or third-party libraries. These frameworks include ready-to-use components and are especially useful for web development.” (Novotny, 2022)

Python's design concept is to be a simple, easy-to-learn and easy-to-understand programming language. We can see the philosophy of the Python language from The Zen of Python.

“[**The Zen of Python**](https://peps.python.org/pep-0020/#the-zen-of-python)

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 aren't 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're Dutch.

Now is better than never.

Although never is often better than \*right\* now.

If the implementation is hard to explain, it's a bad idea.

If the implementation is easy to explain, it may be a good idea.

Namespaces are one honking great idea -- let's do more of those!” (Peters, 2004)

**b) The platform(s) for developing and running software applications for that language are analysed and described. (code libraries, IDEs)**

**Platform**

Python is a very flexible programming language. It is also a cross-platform programming language, which enables developers to develop on different operating systems and platforms. For this assessment, I choose vs code as my tool to develop my python project. In my opinion, vs code is a very popular and powerful lightweight code editor. I can install a variety of extensions in vs code to make it a complete IDE, which is very helpful for me to complete the assessment process.

“Visual Studio Code has built-in support for multiple languages and an extension model with a rich ecosystem of support for others.”(Fincher , n.d.)

**Code Libraries**

I believe that the flexibility and power of Python as a programming language is closely related to the code libraries provided by third parties. Python's third-party code libraries provide powerful functions to support various development needs and improve the development efficiency of developers.

“Third-party libraries are collections of code developed by external programmers. These libraries or packages encapsulate specific functionality that can be reused in different projects as needed. Instead of writing code over and over again whenever you need to perform a common or complex task, you can rely on third-party libraries that have already created and tested that code, saving you time and effort. To download a Python library, you can use the package management system called **pip** (Pip Install Packages).” (*What Are Third-Party Libraries Used for in Python?*, 2023)

In vs code, you can easily install third-party code libraries through commands in the terminal, such as **pip install pandas**

After that, you can import the code library in your code file using following command on terminal

**import pandas as pd**

Note: You can use the keyword as to give the code library an alias, but this does not mean it is necessary.

**c) The characteristics, strengths and weaknesses of the programming environment are described.**

**Characteristics**

1. **Flexible:** Unlike C#, which makes Python quite versatile, Python language lets you define variables straight without explicitly stating the variable type. It is permitted to give a variable a numerical value and subsequently a string.
2. **Cross-platform:** Python may be developed on several operating systems, including Linux, Windows, and macOS. Python thus is quite portable and cross-platform.

**Strengths**

1. **Easy to learn:** Python is definitely advised for newbies just beginning to pick programming languages. Because of its straightforward syntax and easy-to-use code, Python is quite fit for beginners learning since it makes sense. Python is so regarded as a programming language fit for novices.
2. **Flexible:** Python's great third-party code library reflects its own degree of adaptability. These code libraries offer really strong and versatile capabilities that developers could call straight rather of "reinventing the wheel". Above all, Python programs can readily include the features of these libraries.
3. **Strong community：**Python has been developed for decades, and along with it, there are also forums related to Python. When encountering difficult problems, seeking help through the forums is usually very effective.

**Weaknesses**

1. **Security:** Python is a dynamically typed programming language, and the data type is determined only when the code is running, which means it is more prone to security risks and hacker attacks.

“**2. Less secure**

In terms of security, Python is considered to be less secure than some other programming languages such as Java or C++. This is because Python is a dynamically typed language, which means that data types are determined at runtime rather than at compile time. This can lead to vulnerabilities, including buffer overflows or injection attacks.” (Gavrilova, 2023)

1. **Running speed:** Python is an interpreted programming language, so its running speed is Running speed of Python is not as quick as other compiled languages since Python code is interpreted as machine code line by line by the Python interpreter at runtime instead of being compiled into machine code prior running.

“**Not Very Fast**: Python is much slower than more efficient languages like C and Java. Python is interpreted and dynamically-typed, so the run-time compiler has a lot of work to do. It must constantly validate the type of each variable. This means Python is not the best choice for scenarios where speed is critical.”

# Reference

Fincher , J. (n.d.). *Python Development in Visual Studio Code – Real Python*. Realpython.com. <https://realpython.com/python-development-visual-studio-code/>

Gavrilova, Y. (2023, October 30). *Pros and Cons of Python*. Pros and Cons of Python. <https://serokell.io/blog/python-pros-and-cons>

Novotny, J. (2022, March 23). *A Programmers’ Guide to Python: Advantages & Disadvantages*. Linode Guides & Tutorials. <https://www.linode.com/docs/guides/pros-and-cons-of-python/>

Peters, T. (2004, August 19). *PEP 20 – The Zen of Python | peps.python.org*. Peps.python.org. <https://peps.python.org/pep-0020/>

*What Are Third-Party Libraries Used for in Python?* (2023, November 15). 4Geeks. <https://4geeks.com/lesson/what-are-third-party-libraries>