Matthew Burton

Comparison of C# and Python programming languages

SDV602

Heritage and philosophy

C#:

Anders Hejlsberg, the creator of C# and the .NET framework was hired by Microsoft during the time that Microsoft was attempting to monopolize a coding language market particularly in comparison to Java (2000).

C# is designed with an OOP (Object Oriented Programming), Shlaer-Mellor originated style, is designed mostly for use with website application development, windows applications, or game development.

C# is a better version of C++, with more focus on OOP and a better overall design structure with the use of .NET to execute code.

C# is explicit, however has a sense of direction when using the software and will lead the designer down the path to a successful application, however it is very limited to the ability of the .NET framework and almost impossible to create own modules for.

Python:

Guido van Rossum, the creator of Python in 1991, did so with the intention of creating a coding language that had improved readability and usage, as it allowed for both functional oriented programming styles and object-oriented programming styles.

Python as a language follows the philosophies: Beautiful is better than ugly, Explicit is better than implicit, Simple is better than complex, Complex is better than complicated and Readability counts.

Python does not have a core foundation library such as C#, it relies upon modules and therefore is highly extensible and this philosophy and use of modules allows for Python to be used as a more low-level language in a more high-level way, giving the user more control over what is happening where with what code.

Platforms and libraries

C#:

C# has a wide variety of IDE (Interactive development environments) to choose from when developing software. Such as MonoDevelop, Rider, SlickEdit, Rextester, Jdoodle or its purpose made development environment of Visual Studio. Visual studio is the design environment I am familiar with and have experience using, this is a powerful environment with the ability to set breakpoints and watch dogs on variables along with complexity evaluation of functions and CPU usage of threads in a visual format.

C# has a strict use of libraries supported, maintained, and explained in detail by Microsoft, these libraries allow the best possible integration of any kind of code.

Python:

The IDE (Interactive development environments) available for Python are Sublime Text 3, Atom, Thonny, PyCharm, Visual Studio Code, Vim, Spyder, Eclipse, and IDLE. Python installs IDLE alongside the Python download. I have found that the PyCharm IDE is great for fixing any readability errors or showing the code standard Python requires users to follow with highlighting these errors. However I have found that Visual Studio code was better for designing Python applications, as PyCharm seemed to not refresh properly when I had made modules and was attempting to link them between one another.

The libraries available for Python are far and wide, with huge support over the whole of the language, a massive community and constant updates on software, Python libraries allow the user to do basically anything with directional design implementation of code.

Characteristics, strengths, weaknesses

C#:

C# is a great language for GUI design as it follows strict OOP styles during design, allowing for clean and concise interpretation of code, along with the visual “drag and drop” view of a GUI designer. During writing of C# code, as it is a very directional language, it is constantly prompting the user to make certain decisions based on past input or design. However, C# is inhibited by its ownership by Microsoft. Though C# will struggle to grow as a language as the public cannot use it to create their own modules effectively, C# is powerful and can be used by the public to make many different applications and interpret user interaction with the PC as it goes hand in hand with the operating system architecture.

Python:

Python is a great language for beginners as it is loosely typed and can be read much easier, with a huge community support, Python is constantly growing and changing with the requirements of coding languages. Python can be made to do virtually anything and is incredibly powerful for data handling and manipulation. Things have been made increasingly easy through community updates and building of software of libraries for Python to use. Though Python isn’t as fast as C# during runtime, it has the ability to be coded much faster, resulting in a better design, over a shorter period of time, with much more readability and repeatability in a straightforward, directionally focused open minded design format.

In conclusion, C# and Python are both high-level, object-oriented, and easy-to-learn languages as they both have brilliant communities and developer support documentation. Both languages have fast development and good performance. However, C# is a much clearer language, with better organization and with a faster runtime through its explicit declarations and explicit modularity. While Python is easier to learn and write than C# and has vast standard libraries.