Programming Languages are essential to computing and technology fields. Programming languages are the sets of commands, instructions, and syntax that allows for computer and human interaction. However, as time advances forward many of the languages that built the foundation of computing have lost popularity and are no longer used and new languages have been developed and are now growing in popularity. Therefore, the question of what makes a language increase or decrease in popularity comes into light.

Many programming languages are no longer popular in the computing industry today. One reason why said languages are losing popularity is due to their use-case or functionality. For example, the R programming language is not a popular language by today's standards. The R language is a language that focuses on statistics and data visualization. It is not a general-purpose language and therefore, its use cases are specific (Cogswell, 2015). Another reason for languages to lose popularity is the syntax of a language. Syntax refers to the rules that specify the correct combined sequence of symbols that can be used to format correctly structured program within a given language (Techopedia, 2018). If the syntax of a language is hard for users to understand or use, then that language will typically lose popularity. Some syntax is highly specific and unique which is good when performing those desired tasks in specific use cases. However, that will not necessarily boost the popularity of a language (maybe until it becomes obsolete). Another aspect to consider with the loss of use with a programming language is the matter of open source and other resources (Gewirtz, 2017). With open source materials, programmers can have access to a community of other minds and resources who may have already solved the problem they are facing. When this source of knowledge is taken away then programmers tend to move away due to the lack of resources and assistance.

Programming languages can also gain popularity due to differing factors. Aspects like use-case, syntax, and resources available all contribute to the success of a language. The most popular languages today include Java, C, Python, and C++ (Gewirtz, 2017). These languages are not tied to a specific programming platform or focus, therefore making them more universal languages and has increased their appeal (Gewirtz, 2017). Platform-specific languages can also become appealing to programmers. For example, web development languages like JavaScript, C#, PHP, and Swift are quite popular in today's computing realm (Gewirtz, 2017). Syntax also makes languages more successful, the easier a language is to understand by programmers (sometimes computers too) the more use the language will have. The most popular are typically high-level languages (closer to human language), but even some lower-level languages can still be quite popular (“High- and Low-Level Languages.”, Computer Science GCSE GURU). High-level languages are typically easier and faster to write, as well as easier to debug in development. Another aspect of high-level programming language is that they are designed to run on multiple types of machines, making use cases increase (“High- and Low-Level Languages.”, Computer Science GCSE GURU). Another reason that some programming languages are more appealing than others is that there are open resources available to help solve problems. Not all programmers like to have to work from scratch and therefore, there is a draw to open resources (libraries, open source code, etc.). The Internet is a highly valuable resource and the ranking of popularity for programming languages is even based off of the resources available online in those languages and the number of Internet searches (including job searches) concerning certain languages (Gewirtz, 2017).

Each successive generation of programming languages aims to provide a higher level of abstraction and to make languages more productive and programmer-friendly.

There is not one language that fits every situation and therefore even unpopular programming languages are still needed. However, all programming languages are a part of the waning and increasing state of popularity. Whether it be use-case, syntax, resources, or ease of understanding there will always be programming languages that bring greater appeal than others. As the realm of technology and computing continue to advance, programming languages popular or not will be essential.

Work Cited

Brush, Ryan. “97 Things Every Programmer Should Know: Collective Wisdom from the Experts.” *97 Things Every Programmer Should Know: Collective Wisdom from the Experts*, by Kevlin Henney, O'Reilly, 2010, pp. 24–25.

Cogswell, Jeff. “Unpopular Programming Languages That Are Still Lucrative.” *Dice Insights*, Dice, 24 Apr. 2015, insights.dice.com/2014/09/08/unpopular-programming-languages-prove-lucrative/.

Ferguson, Andrew. “A History of Computer Programming Languages.” *History of Computer Programming Languages*, 2004, cs.brown.edu/~adf/programming\_languages.html.

Freeman, Steve. “Code Layout Matters.” *97 Things Every Programmer Should Know: Collective Wisdom from the Experts*, by Kevlin Henney, O'Reilly, 2010, pp. 26–27.

Gewirtz, David. “Which Programming Languages Are Most Popular (and What Does That Even Mean)?” *ZDNet*, ZDNet, 4 Oct. 2017, www.zdnet.com/article/which-programming-languages-are-most-popular-and-what-does-that-even-mean/.

Gordeev, Vladimir. *What Makes Any One Programming Language Better than Another?*, Quora, Dec. 2015, www.quora.com/What-makes-any-one-programming-language-better-than-another.

“High- and Low-Level Languages.” *Computer Science GCSE GURU*, www.computerscience.gcse.guru/theory/high-low-level-languages.

Mwasi, Flavian. “Which Programming Language Will Gain Popularity in the next Years?” *Which Programming Language Will Gain Popularity in the next Years?*, Quora, June 2017, www.quora.com/Which-programming-language-will-gain-popularity-in-the-next-years.

“Rosetta Code.” *Knuth Shuffle - Rosetta Code*, Jan. 2016, www.rosettacode.org/wiki/Rosetta\_Code.

Techopedia. “What Is Syntax? - Definition from Techopedia.” *Techopedia.com*, 2018, www.techopedia.com/definition/3959/syntax.

TechTerms. “Programming Language.” *Programming Language Definition*, techterms.com/definition/programming\_language.

TIOBE. “TIOBE Index for January 2019.” *TIOBE Index | TIOBE - The Software Quality Company*, Jan. 2019, www.tiobe.com/tiobe-index/.