International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 3 | ISSN : 2456-3307

Programming Languages that are on the Verge of Death

Richa
Research Scholar, Department of Computer Science, Punjabi University, Patiala, Punjab, India

ABSTRACT

This paper is about the observation and analysis of the programming or computer languages which are not being used anymore and on the verge of death. This paper uses 2020 statistics for its discussion and explanation. The point is there is nothing wrong with these languages that made them disappear, it's just they get replaced by the better alternative that came in the market. Hence, it's up to us to analyze the demand of the tech market and be aware of all the latest trends in the programming world to survive in the IT world.

Keywords : Programming Languages, Objective-C, Haskell, Perl, Pascal, Erlang, CoffeeScript, VB.NET, ColdFusion, COBOL60

I. INTRODUCTION

Not all programming languages endure forever. In fact, even the once-most-popular languages crumble away at some point in time. It's inevitable! New generations of developers embrace other languages and frameworks they find easier to work with. We looked at what languages were popular with developers active on GitHub, Twitter, Stack Overflow, Freenode, and Reddit. Languages with more forks, repositories, and subscribers scored higher when it came to community engagement. We looked at Google Trends and Stack Overflow Trends to see which languages experienced an upward (or downward) growth trajectory between 2013 and 2018 to assess the which Programming Languages would be dying. But do you know there is one thing common with all these languages that with the emergence of the technologies and the advent of various trending programming languages, these veteran languages have witnessed an unpredicted decline in their popularity & usage.

And if you're assuming that these are the only languages that have experienced such situations then you must know that there have been a lot of programming languages that just came, rule, and fade away. Programming languages are a crucial medium of connecting humans to machines. The world is moving toward the most advanced technologies, and this is only possible because of programming languages. These languages help in harnessing the power of computing in all human endeavours.

However, in this article, we list down seven programming languages, in no particular order, which people think will die in a few years.

II. LIST OF PROGRAMMING LANGUAGES

The list of programming languages that are going to die soon is given below.

1. Objective-C

The first name that comes here in the list of top dying Programming Languages is Objective-C. The language that came around 36 years ago is a general-purpose, object-oriented language that follows the Smalltalk-derived syntax and is primarily used to develop applications for Apple platforms such as macOS, iOS, etc. The language single-handedly ruled the particular domain until the arrival of its alternative language Swift in 2014. Swift is comparatively being more preferred than Objective-C by the developers due to various remarkable features such as robustness, better memory management, etc. However, the language has still some occupancy in the market and probably not going to fade away anytime soon, but if you're looking forward to starting your journey particularly with the macOS or iOS domain, you're recommended to go with Swift instead of Objective-C. Apple's Objective-C is 35 years old, and it's clear that the company wants it dead. Five years ago, Apple executives took to the stage to unveil Swift, its new-and-improved programming language for its software ecosystem. No doubt they expected developers to quickly embrace Swift at Objective-C's expense. And to be fair, more developers have begun using Swift, especially as it's become more feature-rich, but Objective-C hasn't crashed as much in the popular-language rankings as some folks might have expected. Blame that on 35 years of legacy code, and many developers simply preferring to work with a language they've always used.

At some point, though, Objective-C will likely fade away entirely. Apple's too keen on its eventual demise, and Swift is becoming an incredibly effective language for building iOS, macOS, and cross-platform apps.

2. Haskell

Haskell is one of the oldest general-purpose, statically typed functional programming languages. The language was particularly designed for handling symbolic computations along with list processing applications and it is considered to be much relevant to research and industrial application domain. Moreover, due to several worthwhile features such as better reliability, shorter lead times, etc., the language is being preferred by various big tech companies such as Facebook, GitHub, IBM, etc. in the past.

However, if we take a look at the TIOBE popularity index, the language shows a consistent decline in its demand & usage in the last 8-10 years. There may be various reasons behind that such as difficult to learn, static-typing, etc. Hence, despite the major standard update of Haskell, you're recommended to not majorly rely on Haskell and consider learning other trending languages as well such as Python, Go, etc. for better career opportunities. Though, learning Haskell along with the other programming languages would also be the best option as having an understanding of functional programming language is always appreciated.

3. Perl

Perl, introduced in 1987, is a general-purpose programming language that incorporates Perl 5 and the latest sister version i.e., Perl 6. The language is used for a wide range of tasks such as web development, text manipulation, GUI development, network programming, and many more. The language has enjoyed a good run in the tech world around 2004 but after the advent of several other prominent programming languages especially Python, Perl, and even its latest sister version is finding it difficult to make a strong command over the market. However, the language is been doing quite well with the rankings in the current year but still, you're recommended to diversify your programming language choice considering the average downward trend of Perl in the last few years. Perl is a family of two high-level, general-purpose, interpreted, dynamic programming languages. While the language does have features that ease the task of the programmer, it comes at the expense of greater CPU and memory requirements. Nonetheless, widespread developer embrace of other languages for things like building websites means that Perl is going to just fall into increasing disuse. Even if Red Monk has Perl's popularity declining, it's still going to take a long time for the language to flatten out completely, given the sheer number of legacy websites that still feature its code. That being said, it's taking a while for Perl to finally give up the ghost. Its descent has been monitored for quite some time, even as coding boot camps and developers have given it up.

4. Pascal

Let's move to another programming language that is dying – Pascal. The language is a general-purpose, procedural programming language that was specifically developed for teaching programming practices in a structured manner. The journey of Pascal Language is in such a way that it was developed on the model of ALGOL 60 language and then led to the development of Object Pascal. In general, Object Pascal can be considered as an object-oriented derivative of Pascal language.

The language has indeed enjoyed its popularity and demand in the market due to various features such as easy to learn, extensive error checking, strongly-typed, etc. But when it comes to the growth and demand of Pascal in the current day scenario, you must know that old Pascal is almost dead and even the Object Pascal is on the verge of vanishing from the market as there is no buzz of the language among the developers.

5. Erlang

Erlang is a general-purpose, functional programming language that is primarily known for its support to concurrency, distribution, and fault tolerance. The language was initially used for huge telecommunication systems but then it expanded its implementation into various domains such as e-commerce, banking, and others. Though the language is still in utilization by some established organizations yet due to several factors such as steeper learning curve, availability of better alternatives like Elixir, Elm, etc. it is not being much preferred by the new developers and companies.

As per the reports, Erlang is experiencing a consistent decline in its popularity in the last few years. However, the jobs for Erlang Developers are still existed in the market and not going to disrupt anytime soon but if you're having a long-term career vision for yourself then you should opt for any other trending and relevant language rather than going with Erlang.

6. CoffeeScript

Though the language, CoffeeScript, came into the existence in 2009 and is not that much old that it should be here in the list of top dying programming languages – yet it is!! It is a programming language that compiles to JavaScript with several additional features like list comprehension, destructuring assignment, etc. and provides better syntax & readability. The language became very popular in a very short span however that popularity doesn't last long. As per Google Trends reports, CoffeeScript has witnessed a huge decline in the last 2-3 years in its popularity.

Several major reasons behind the decreasing demand for CoffeeScript are additional compilation process, prone to change, etc. Hence, if you're currently working with the CoffeeScript or looking forward to starting your career with the language, you're recommended to explore several other alternatives as well for better career opportunities in the long run.

7. VB.NET

Lastly, here comes one of the most renowned languages of its time – VB.NET!! The language is an object-oriented programming language developed by Microsoft and is implemented on the .NET Framework. Visual Basic was developed by Microsoft as a variant version of the BASIC and then Microsoft proposed VB.NET in 2002 as the successor of its initial Visual Basic language. Undoubtedly, VB.NET was the favourite child of the market for a longer period of time but it somehow gets on the weaker side against its competitor language C# which is also primarily built on the .NET framework.

As per the TIOBE Index reports, C# has taken over the VB.NET in terms of demand & popularity and VB.NET is losing its charm according to Google Trends results also. Meanwhile, due to its strong command over the programming world in the past, VB.NET still has some adequate occupancy in the market but if you're just about to start your career or in the beginner stage, you're recommended to prefer other options.

8. ColdFusion

As mentioned by several reports, a gradual decline is being observed in the popularity and usage of ColdFusion. Despite being updated by Adobe on a regular basis, ColdFusion is yet to gain momentum in the market and the programmer's community alike.

Adobe has launched ColdFusion 11 with a number of advanced features to help programmers to build and deploy both web applications and mobile apps rapidly. It further allows developers to use the enterprise, developer, standard, or express version of ColdFusion 11 according to their needs and budget.

However, there is a number of factors such as the poor quality of debugging, lack of package manager and lack of performance of the CFScript, that affect the popularity and market share of this commercial framework.

9. COBOL60

COBOL 60, created in 1960, is an acronym for Common Business-Oriented Language. As the name suggests it was designed primarily for business use. Programmers are more comfortable using static typing Java or dynamic typing Python. COBOL is difficult to use because it has strong typing rules and is more difficult to parse. As a result, big corporations are definitely showing symptoms of moving on!

It is still not clear whether COBOL is one of the dead programming languages or not. This is because it lives on in quite a few legacy systems that are expensive to update.

10. Elm

This purely functional, domain-specific language was developed for declaratively creating web browser-based graphical user interfaces, with emphasis on usability. While Elm's growth trajectory was rising from 2013 to 2018.

The problem is that it's been almost two years since the last update to Elm language. This makes Elm look sad and dead to the newcomers. This doesn't mean Elm is not usable or not mature enough for production work. However, it does make Elm looks stale by comparison. It means Elm will not get as much recurring exposure to broader and new adopters.

III. CONCLUSION

You must take into account that the above-mentioned languages like Objective-C, Perl, etc. are not going to die or disrupt immediately in the upcoming times, but they're not worth learning from scratch in this year as there are better and worthwhile alternatives available in the market. And, on top of that, this is not only about Programming languages; you're always required to stay updated with all the latest trends & technologies to do better in your career.

IV. REFERENCES

[1]. https://www.geeksforgeeks.org/top-dying-programming-languages

[2]. https://www.edureka.co/blog/top-10-dying-programming-languages
[3]. https://github.com
[4]. https://twitter.com/?lang=en
[5]. https://stackoverflow.com
[6]. https://freenode.net
[7]. https://www.reddit.com

Cite this article as :

Richa , "Programming Languages that are on the Verge of Death", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 3 Issue 3, pp. 2148-2152, March-April 2018.
Journal URL : http://ijsrcseit.com/CSEIT1952299