

FROM JAVA TO PYTHON: NAVIGATING THE WORLD OF PROGRAMMING LANGUAGES

Dmitrii CRAVCENCO^{1*}, Dmitrii BUJLOV¹, Alexandr CIORNII¹

¹Department of Software Engineering and Automation, gr. FAF-223, Faculty of Computer, Informatics, and Microelectronics, Technical University of Moldova, Chisinau, Republic of Moldova

*Corresponding author: Cravenco Dmitrii, dmitrii.cravenco@isa.utm.md

Abstract. *In today's fast-paced digital world, understanding how modern IT is built has become crucial for businesses and individuals alike. The goal of this article is to provide readers with a basic understanding of the process behind modern IT development. We will first explore the personal preferences of modern IT specialists, such as their preferred tools and methods. Next, we will examine how changes in technology and user needs have shaped the development of modern IT. Finally, we will discuss the various programming languages used in modern IT and their specific purposes. By the end of this article, readers will have a better understanding of the complex nature of modern IT and the factors that influence its development.*

Keywords: *approaches, diversity, programming languages, tasks, technologies.*

Introduction

Nowadays there are lots of different programming languages and jobs that require different skills and knowledge. Young people, who want to become a programmer, but don't know where to start face the problem that they don't really understand which way to choose. This paper will help to acquire basic understanding of the domain and select the most appropriate way. The top programming languages of 2022: C/C++, JavaScript, R, Swift, PHP, Go/Golang, MATLAB, C#, Kotlin.

Personal preferences

Programming languages are highly subjective, with developers often having strong personal preferences based on factors such as readability, performance, and style. These preferences can vary widely, with some favoring functional languages like Haskell, others preferring dynamically-typed languages like JavaScript, and still others valuing the speed and control of low-level languages like C or Rust.

Main criteria in choosing favorite programming language are:

- Simplicity of learning, it's very important for new programmers;
- Compatibility with other languages, sometimes it's needed;
- Speed, everyone likes when code runs fast;
- Platform independence, it's important for some needs;
- Big community, it's important for studying and sharing ideas;
- Amount of libraries, for someone it's more convenient to use libraries;
- Object oriented, nowadays it's very popular paradigm;
- Nice syntax, it's obvious.

Also it may be based on the contrary, if other languages are slow, rarely used, hard to learn, etc. In the survey [1] results for Most Loved, everything is categorized so it's easier to compare like-to-like (i.e. languages vs. frameworks vs. libraries, etc.). We're going to take a cue from the survey and focus on programming languages for this question; drawing comparisons within types makes sense and avoids introducing another layer of complexity.

Rust, Elixir, Clojure, Typescript, and Julia top the list of the most popular programming languages. But if you look at the last three years, you'll see some movement.

Comparing Most Loved and Learning to Code Popularity in Table 1 to code popularity reveals an interesting pattern. People learning to code don't use the most popular languages. The difference

between these two measures of popularity will be important in distinguishing both as possible explanatory variables for trends in question posts. Less than 1% of learners reported using either Clojure or Elixir. 1.2% use Julia, 7.1% use Rust, and 15.1% use Typescript.

Table 1

Most Loved Programming Languages[2]				
	2020	2021	2022	Trend
Rust	1	1	1	✓
Elixir	NA	4	2	↑
Clojure	NA	2	3	↓
Typescript	2	3	4	↓
Julia	6	5	5	↑

For web searches, We're using the already established PYPL index, [3] which is an aggregated source for Google Trends data specifically for programming language tutorial search history. Fig.1 StackOverflow is a popular online platform where developers can ask and answer programming-related questions. The platform also provides insights into the most popular and highly ranked tags used by developers around the world, which can be used to gain insights into current trends in the industry.

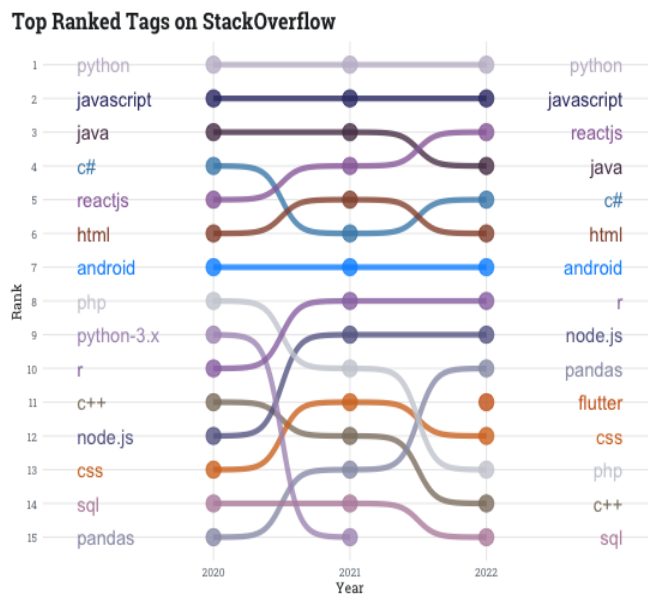


Figure 1. Top Ranked Tags on StackOverflow [2]

From this dataset (Figure 1), we will focus on annual trends in programming languages share of search. GitHub publishes statistics on public repositories for anyone to use as a handy public dataset within Google BigQuery, and although we lose the information from private repositories, we can assume the public accounts speak more directly to popularity as they are tied to learning initiatives, portfolios, and open-source collaboration, which are mostly self-directed rather than mandated by existing business rules. Posting questions most likely speaks to friction with coding, a friction that may lead to loving a programming language less.

Changes in technology and user needs

Programming languages are constantly evolving and change very often, because of the needs of the programmers. It happens, because languages should support new technologies and provide many possibilities for the programmer if their developers want them to be used by many programmers. “FORTRAN I [Backus et al. 1956] emerged as the first successful high-level language during the late 1950s. Predating nearly every major development in computer science (and, in fact, predating the term “computer science” [Janss 1999]), FORTRAN I now appears quaint, differing substantially from its modern incarnation, Fortran 2008, which reflects the benefits of half a century of accumulated experience. Since its inception, the Fortran language has been adapted to incorporate subprograms (FORTRAN 66), structured programming constructs (FORTRAN 77), modules and dynamic memory allocation (Fortran 90), object orientation and C language interoperability (Fortran 2003), and co-arrays (Fortran 2008). Even the much-younger Java language has evolved from its initial release in 1996. Although most of the changes have been in the library, the core language has been expanded as well, adding inner classes (JDK 1.1), strictfp (J2SE 1.2), assert statements (J2SE 1.4), annotations, autoboxing, enumerations, a new for loop, generics, import static, and varargs methods (J2SE 5.0).” [4] Obviously, a language is thought to be a good one or not, based on some criteria. There are many technical ones, but let’s consider the most general ones:

Readability and Writability: A good programming language should be easy to read and write. This means that the syntax and semantics of the language should be easy to understand, and the code should be easy to maintain and modify.

Expressiveness: A good programming language should allow programmers to express complex ideas in a concise and clear way. The language should have a rich set of built-in functions and data structures that make it easy to solve a wide range of problems.

Performance: A good programming language should be efficient and fast. It should be able to execute programs quickly and handle large amounts of data without slowing down.

Portability: A good programming language should be portable across different hardware and software platforms. This means that programs written in the language should be able to run on different operating systems and hardware architectures without requiring significant modifications.

Safety and Security: A good programming language should have built-in mechanisms to prevent common programming errors, such as buffer overflows and memory leaks. It should also provide features for secure programming, such as encryption and authentication.

Community and Support: A good programming language should have a strong community of developers and users who can provide support, share knowledge, and contribute to the development of the language and its ecosystem.

Documentation: A good programming language should have comprehensive and up-to-date documentation that makes it easy for developers to learn and use the language.

Flexibility: A good programming language should be flexible and adaptable to a wide range of programming paradigms, such as object-oriented, functional, and procedural programming. It should also be able to integrate with other languages and technologies.

Different tasks require different approaches

To begin with, to understand how much differ programming styles and approaches, it is likely to explore different Programming duties.

One of the main goals that every programmer wants to achieve is to make our phones and computers work smoothly and reliably. In addition, they design, develop and test software that meets customer needs. Computer programmers utilize their knowledge and skills to guarantee that software development adheres to industry standards and best practices pertaining to reliability, performance, and security. In order to ensure accuracy, computer programmers frequently iterate through designing, developing, integrating software components, testing, and gathering feedback from customers. Once an application or program operates efficiently and is implemented, computer

programmers modify and enhance the features of the pre-existing software. So, there are a lot of different tasks that a programmer should complete and in order to increase the effectiveness of creating a solution, a specialist uses a variety of different languages, including Python(48.07%), C++(22.55%) , Java(33.27%) and JavaScript(65.36%) [1].

There are different domains that require special knowledge such as Web Development, Mobile App Development and Game Development, let's take a closer look at them.

A web developer is responsible for the general appearance and functionality of the website you visit, and oversees both the design and technical elements of the website, including its speed and resilience. A front-end developer brings her website to life using HTML, CSS, or JavaScript. The backend developer is less concerned with user experience design, more concerned with making sure that all moving parts work as her one unit. Usually they use PHP, Ruby or Python to build applications that connect the website backend to the frontend.

To understand the importance of Mobile App Development, we can check a few statistics from the domain: there were more than 2.8 million apps in the Google Play store as of March 2018, according to market and consumer data firm Statista. And each should be supported and looked for, what Mobile App Developers do. The most common languages are Kotlin, Java, C++ and Swift. According to Angelo Firen [5] main skills required for mobile app development are Product Management, User Interface and User Experience, Design, Business Analytics, Quality Assurance and Security.

Mostly, what game programmers do is designing video and creating online and mobile games. Video game developers help transform the perception of games into a playable reality. Create features that make video games easier to use. Special skills are required, such as animation programming, trigonometry, calculus, and linear algebra knowledge. A knowledge of physics also helps you understand concepts like mass, thermodynamics, and inertia [6].

To better understand the popularity of different domains, th In Stack Overflow's 2022 survey, about 43.38% of developers identified as backend developers, 25.96% as front-end developers, and 46.82% as full-stack developers(can do both front and back-end), 12.45 are mobile developers. Game development is one of the rarest professions in the industry, only 3% are involved in it [7].

To sum up, modern technology is so complex that one person is not capable of maintaining all the processes of a project. That is why there are a lot of different specialists who are responsible for their own domain. Each of them has his/her specific approach to the problem solving process. Depending on the issue there are different programming languages that he/she can rely on.

Conclusions

In conclusion, programming languages are essential tools that enable humans to communicate with computers and develop software applications. The diversity of programming languages reflects the variety of human perspectives, goals, and technical requirements. Each programming language has its strengths and weaknesses, and different programming paradigms offer different approaches to problem-solving. The choice of programming language depends on several factors, such as the application domain, the development team's skills, and the available resources. As technology continues to evolve, new programming languages will emerge, and existing languages will evolve to meet new challenges. Therefore, developers need to stay up-to-date with the latest trends and innovations in programming languages to design efficient and effective software solutions.

References

1. Stack Overflow, *Survey Most Loved Programming Languages*. [online]. [accessed 10.03.2023]. Available: <https://survey.stackoverflow.co/2022/#most-loved-dreaded-and-wanted-database-love-dread>

2. YEPIS, E., Comparing tag trends with our Most Loved programming languages
3. [online]. [accessed 10.03.2023]. Available: <https://stackoverflow.blog/2023/01/26/comparing-tag-trends-with-our-most-loved-programming-languages>
4. PYPL, *PopularitY of Programming Language*, <https://pypl.github.io/PYPL.html>
5. OVERBEY, J. L., JOHNSON, R. E., *Regrowing a Language*, Department of Computer Science, University of Illinois at Urbana-Champaign, 201 N. Goodwin Ave. MC 258, Urbana, IL 61801).
6. FIRENZE, A., *14 Essential App Development Team Skills You Must Have* <https://ascendle.com/ideas/14-skills-your-app-development-team-must-have/>
7. Simplilearn, *How to become a game developer*, [online]. [accessed 09.03.2023] Available: <https://www.simplilearn.com/how-to-become-a-game-developer-article#:~:text=Know%20Your%20Computer%20Languages&text=To%20achieve%20this%2C%20it%20is,%2C%20MS%20SQL%2C%20or%20Oracle>
8. Stack Overflow, *Survey Most Loved Programming Languages*, [online]. [accessed 07.03.2023] Available: <https://survey.stackoverflow.co/2022/#education-ed-level-prof>