

How developers pick languages for specific projects?

In general developers prioritize expressiveness and speed of production over linguistic correctness. Developers see more value in unit test, debugging rather than its types [4].

Uses of Programming Languages

Programming languages are used for various purposes such as:-

1. *Web Development:*

For building the websites web development is used. Websites are divided into two parts frontend development and backend development [5].

- **Frontend Development:** It is a part of web development. With the help of HTML, CSS and JavaScript frontend is made [6]. Frontend is basically a page which can be seen by user so that user can perform various operations such as reading, writing, copying, pasting etc. Web development that creates the application and runs on browser and adjust the style, colour and font etc.
- **Backend Development:** Another part of creating the websites is backend. It related to the server. Computer that runs the software, websites and servers to the world. It route the webpage according to the need of user. It also interacts with the database that stores the website information and sends the data to the user [7].

2. *Game Development:*

Game development is one of the most interesting fields of programming. Developers enjoy it and there are developers who create the games just for the fun. Developing games requires game engine. Game engine is software which is used as the infrastructure for building the game and defines what the game has and what it can do. Mostly language used in game development is C# and C++. It requires a lot of memory optimization and fast performance to create rich graphics.

3. *Mobile Development:*

Creating mobile application is little bite tricky. As there are more than one operating system are present for mobiles. An operating system is the device that is responsible for dealing with the hardware. It is the layer between the application and the hardware, whether it's a touchscreen or GPS. The most common operating systems are Android and IOS. For creating application in Android Java and Kotlin languages are used and for creating the IOS application Swift is used. Using JavaScript or Dart it is possible to create mobile applications for both Android and IOS [8].

How to learn a programming language?

Learning a programming language is requires dedication and practice. Without writing a code it is impossible to become a programmer. After understanding the basic concept of programming pick up the field and language [9].

According to the search the most popular programming language are:

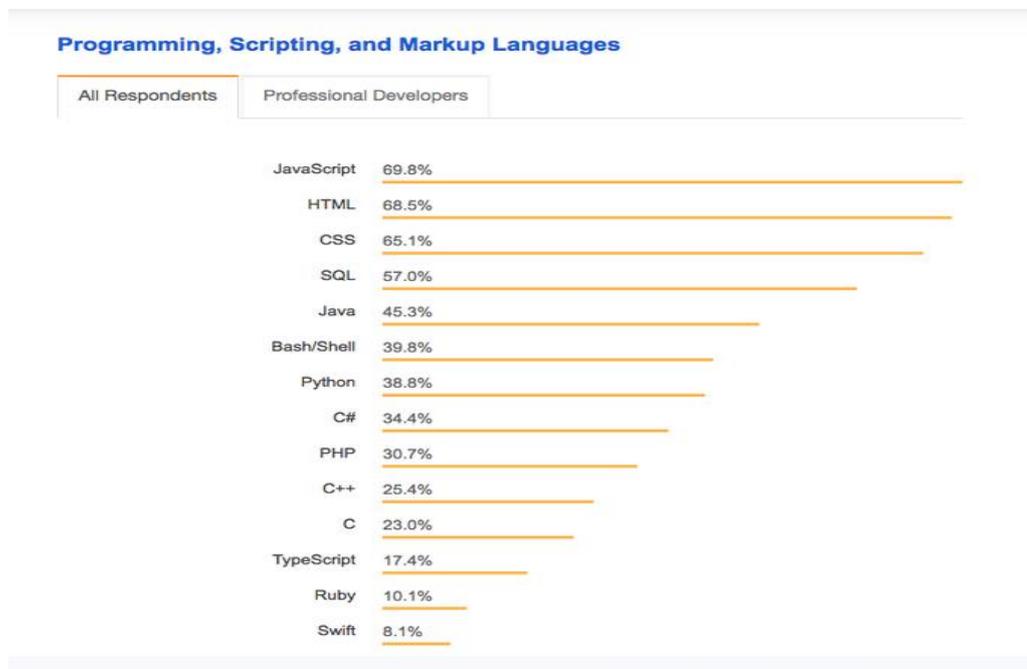


Fig. 2 Language usage

Type of Language

1. Machine and assembly language

It consists of the numeric codes for the operations that a particular computer can execute directly. The codes are in the form of 0s and 1s, or binary digits (“bits”), which are frequently converted both from and to hexadecimal (base 16) for human viewing and modification. Machine language instructions typically use some bits to represent operations, such as addition, and some to represent operands, or perhaps the location of the next instruction. Machine language is difficult to read and write, as it is not written in the human language. Its codes vary from computer to computer [10].

Assembly language is one level above machine language. Instructions are written in the form of short mnemonic codes and allow the programmer to introduce names for blocks of memory that hold data. One might thus write “add pay, total” instead of “0110111000” for an instruction that adds two numbers. Assembly language is designed so that it can easily be converting into machine language [11].

2. Algorithmic languages

It is designed to express mathematical and symbolic computations. Algorithmic language is the first high-level language. They can express algebraic operation in the mathematical operation.

3. Fortran

The first language develop was Fortran. It is also called as formula translation design in 1957 by an IBM led by John Backus. Its control structures included conditional IF statements, loops and a GOTO statement that allowed no sequential execution of program code. FORTRAN made it convenient to have subprograms for common mathematical operations, and it had built libraries with them [12].

4. LISP

LIST means as list processing. It was developed by John McCarthy at the Massachusetts Institute of Technology (MIT) in 1960. It was founded on the mathematical theory of recursive functions (in which a function appears in its own definition). A LISP program is a function applied to data. Its uses a very simple notation in which operations and their operands are given in a parenthesized list example, (+ a

(* b c)) stands for $a + b*c$. List can be represented in the form of structure to represent the data. For artificial language LIST is commonly used language.

5. C-Language

C programming language was developed by Richie and Brian Kernighan at the AT&T Corporation for programming Computer operating system. It uses a compact notation and provides the programmer ability to operate with the addresses of data as well as with their values. C and C++ both have almost same syntax.

6. Business-oriented languages (COBOL)

COBOL means Common business oriented language. COBOL uses an English-like notation. Business computations organize and manipulate large quantities of data and COBOL introduced the record data structure for such tasks.

Object-oriented Programming Language (OOP)

The purpose of this programming language is to think about the fault by separating it into a set of objects offering services that can be used to solve a particular problem. One of the main objectives of OOP is encapsulation. Encapsulation is means everything an object will need must be inside of the object. It also contains other features like polymorphism, abstraction, Class and Inheritance.

CONCLUSION

In this research paper study of language type and using it relates to software quality. Github data is used by characterized by its complexity and the variance along multiple dimensions of language. During their careers, professional developers learn and forget languages, and that age has little to do with language choice. Some languages are easier to learn than other. Exposed to a language paradigm in school, developers are more likely to learn or remember similar languages later in their careers. This result will help broader to computer science community. Language selection is related to libraries, history and familiarity, historical effects are present and therefore potentially multiple stable equilibriums. It suggests that today popular programming language can be replaced or improved. The changes are durable. Before choosing the language the developer should know the field related to languages. Every language is connected with different field. But the logic in all the language is same.

REFERENCES

- [1] B. Carpenter *et al.*, "Stan: A probabilistic programming language," *J. Stat. Softw.*, 2017, doi: 10.18637/jss.v076.i01.
- [2] B. W. Kernighan and D. M. Ritchie, "The C Programming Language: The C Programming Language," *TI The effect of two different electronic health record user interfaces on intensive care provider task load.* 2015.
- [3] J. Meyerson, "The go programming language," *IEEE Softw.*, 2014, doi: 10.1109/MS.2014.127.
- [4] L. Objective-c, "The Objective-C Programming Language," *Management*, 2010.
- [5] T. D. Kulkarni, P. Kohli, J. B. Tenenbaum, and V. Mansinghka, "Picture: A probabilistic programming language for scene perception," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2015, doi: 10.1109/CVPR.2015.7299068.
- [6] B. Stroustrup, *the C++ Programming Language 4Th Edition.* 2013.
- [7] P. McKay, "Introduction to Programming in C," in *The Professional Programmers Guide to C*, 2010.
- [8] T. J. Stevens, "The Python Programming Language," in *Wilson and Walker's Principles and Techniques of Biochemistry and Molecular Biology*, 2018.
- [9] J. Nagel, "Getting started," in *Lecture Notes in Energy*, 2019.
- [10] J. J. Palestro, P. B. Sederberg, A. F. Osth, T. Van Zandt, and B. M. Turner, "A Tutorial," 2018.
- [11] R. Pickering, K. Eason, R. Pickering, and K. Eason, "Language-Oriented Programming," in *Beginning F# 4.0*, 2016.
- [12] R. E. Neapolitan and R. E. Neapolitan, "Neural Networks and Deep Learning," in *Artificial Intelligence*, 2018.