
COMPUTER PROGRAMMING LANGUAGES IN 2020: WHAT WE USE, WHO USES THEM, AND HOW DO THEY IMPACT JOB SATISFACTION

Alan Peslak, Penn State University, arp14@psu.edu
Mauri Conforti, Geisinger Health Systems, maconforti1@geisinger.edu

ABSTRACT

There has been an explosion of programming languages over the past decades. From Assembler to Java, a variety of different programming languages have been developed by software designers to meet a variety of needs and applications. But how many of these languages are being actively used by today's software professionals? And are there differences in language usage by gender or age? Our study analyzes the 2019 Stack Overflow annual survey and finds that the top languages used by all respondents of the survey were JavaScript, HTML/CSS, SQL, Python, and Java. For those who self-identified as software developers, Java leapfrogs Python and TypeScript comes above C++ but otherwise the top ten are unchanged. Many other insights have been discovered including number of languages used, significant differences in language use by age and gender, and the relationship of language usage and job satisfaction.

Keywords: Programming Languages, Job Satisfaction, JavaScript, Java, Python, Stack Overflow

INTRODUCTION

Programming languages, scripting languages, markup languages are all ways that we interact with and code computers to produce the output that we desire. There are more than 300 languages that IEEE has currently named that are in active use today. (IEEE, 2019). With all these options it is difficult to focus on the technologies that are most prevalent and in use in the field today. The IEEE whittled down the field to 47 languages that had a significant presence in their survey of a variety of media sources. They analyzed Google search data, Google Trends, Twitter, Github, Stack Overflow, Reddit, Hacker News, CareerBuilder, Dice, and their own IEEE Xplore Digital Library to find the most popular programming languages. Nearly all the source data came from searches or postings for the specific language. As a result, they suggest that their study shows the most popular programming languages. Their list of top ten languages is:

Table 1: IEEE 2019 Most Popular Programming Languages

1. Python
2. C++
3. Java
4. C
5. C#
6. R
7. JavaScript
8. Go
9. Assembly
10. Matlab
11. Scala
12. Ruby

But this popularity list only shows how often items were searched or mentioned. It cannot make any valid predictions about the frequency of use of these languages by students, software professionals, or even hobbyists. Fortunately, Stack Overflow has developed an annual survey that does specifically study what languages are actively used by its members. The 2019 survey collects a great deal of information from its users and includes specific questions on what

programming languages they are currently using. Included in the survey is a wealth of demographic information including age and gender and other information such as career satisfaction. (Stack Overflow, 2019).

It is important to understand what programming languages are essential for a variety of reasons. Educators need to understand what programming languages to study to prepare students for the workforce. Computer curriculum includes programming languages, which languages should the students have a basic knowledge of, which should be more rigorous, and how should they be taught. The question becomes, how much programming language can be taught to students with an already heavy credit load. Designing programs, algorithms, principles of programming languages, and systems are priority. What to include next is in question. Too little beyond this, students will not be prepared for a computing career or graduate majors. “the course must illustrate each covered linguistic mechanism with several examples from existing languages, ideally complemented by programming exercises (using the mechanisms or languages).” (Felleisen, 2008).

Practitioners need to understand the variety of influences that affect programming usage such as demographics or preferences for job satisfaction so that proper mapping of individual workers can be accomplished.

Researchers can explore the programming language landscape to better understand usage and preferences and consider how to possibly break demographic barriers as well as better analyze where and how programming languages are used. Our study attempts to develop a comprehensive analysis to address these varied user needs.

LITERATURE REVIEW

There is limited literature on the usage of programming languages used by students, practitioners or hobbyists. Siegfried, Siegfried, & Alexandro, (2016) analyzed the first programming language taught in introductory computer science courses from 410 regionally disbursed US Universities. They found that Java was the most popular and taught at 180 universities. This was followed by Python at 76 and C++ at 74. The only other programming language to crack double digits was C at 22. In their study they note that since the last time this study was done in 2011 (their new study was 5 years later), Java, C and C++ have declined a bit and Python has increased in popularity. The authors speculate the move to Python was influenced by the simplicity of Python versus Java, making it more accessible for beginning students. In order to determine the first course, the authors used the intro computer science programming course or an intro computer information systems programming course if the university did not have a computer science curriculum. Java was most popular in all regions of the US except the Northwest which favored C++. From 2011 to 2015 Python increased from 41 schools to 76 schools.

Chen (2018) discusses how every decade seems to have its own programming language. But Chen suggests that programming languages have a defined structure that inherently causes solutions to be built in a determined way based on the structure of the language itself. The author proposed Julia as a programming language that can allow for multiple approaches to a problem.

Kochlar, Wijedasa, and Lo (2016) published a comprehensive study of using multiple programming languages and their relationship to code quality. They studied 628 Github projects in 17 languages and used multiple regression analysis to uncover how multiple languages used in a project affected code quality. Overall, they found that using multiple languages does significantly reduce code quality as measured by defect proneness. They also found that commonly popular languages including C++ and Java are significantly lower in quality when used in conjunction with other languages in the same project. Kochlar also found that there are some languages that naturally go together in multiple language projects. Java pairs with other OO languages such as Python, C++ and C# as well as scripting languages such as Javascript, TypeScript, Ruby and CoffeeScript. Python was only paired with Javascript, C, C++, CoffeeScript and PHP.

An interesting study of Greek students found that there were significant differences in introductory Scratch programming lessons by gender. Females were actually found to have higher programming scores and skills but found the lessons harder and less fun than males. The authors suggest that building confidence levels may be an effective strategy when dealing with the programming gender gap (Schina, González, & Rodríguez, 2019). Noone and Mooney

(2019) found that there was no significant difference by gender for students who studied Java programming language in their introductory programming class.

The use of the Stack Overflow survey as a source for analyzing programming and computing activity is established in the literature. Silveira et al. (2019) used the Stack Overflow survey and found a statistically significant difference in confidence level of programming by gender. May (2019) used Stack Overflow data to analyze the gender gap in reputation for Stack Overflow users. They found that women generally had a much lower reputation score on Stack Overflow. Ahasanuzzaman, et al. (2016) used Stack Overflow questions and answers and found a preponderance of duplicate questions. They developed a methodology to classify and reduce these duplicate questions.

Johri and Bansal (2018) used Stack Overflows 11,097,716 user posts from 2014 and 2015 and performed LDA topic analysis on the posts. They found the rise in Mobile Apps and Data Visualization and a decline in OO programming and practice. The topics are meaningfully labeled based on top words allocated by LDA. Results show that Website Design/CSS is the most impactful topic. Data Analysis/Visualization and Mobile App Development are hot topics and their popularity is increasing, while impact of Object-Oriented Programming and Coding Style/Practice has decreased over time. On the other hand, topics like Authentication/Security and UI Development have shown steady trends over time. Furthermore, R and Python have dominated in Data Analysis/Visualization topics, Oracle and MySQL are the most popular database platform, Python is most impactful scripting language. The results of this analysis will help both developers and commercial vendors track latest trends in technology and programming languages. The work done in this project can be extended in a number of ways. Topic analysis of question and answer posts can be done separately, so that different topics can be found from the overall analysis. LDA can be applied to a smaller time interval of posts (e.g., 1 to 3 months), which may help reveal topics that emerge during specific months. Projects from GitHub can be analyzed in conjunction with Stack Overflow to validate the results. The proposed methodology can also be applied to other developer resources, blogs, and forums.

Sanatinia and Noubir (2016) analyzed Github for the most popular programming languages. They collected and studied Github, in its entirety investigating 10 million user profiles, over 16 million public repositories, 11 million user relationships, and over 3 billion contributions. Their top five languages were JavaScript, Ruby, Python, Objective-C, and Java.

The most popular programming languages depend on what you are working on and what you need to accomplish. Since program languages are tools, in one domain a certain language may be necessary, and another it may not. The Cass article ranking system was taken from 10 data sources using 12 metrics. The 2015 top 5 are Java, C, C++, Python, and C#. These 5 have the same ranking as 2014. The number 6 rank, R, rose from 9th place in 2014. The second spot C is ranked 99.9, very close to Java, in 2014 it was ranked 99.3. C++ is ranked 99.4, in 2014 was 95.5. (Cass, 2015)

According to Cass, Python has taken the lead since the 2017 survey, followed by Java, C, C++, and R. Java is not running as close a second place with 96.3 as was C behind Java in 2015. Python rose from 4th place in 2015 with a rank of 96.5. It has gained popularity partly from the number of specialized libraries, especially artificial intelligence. One of the impacts with Python is with Deep-learning developers, it could not have been foreseen when it was released in 1991. Another impact is microcontrollers increased computing power meaning that Python's versions of CircuitPython and MicroPython is popular with makers. (Cass, 2019)

METHODOLOGY

In order to study the current usage of programming languages, we used the comprehensive 2019 Stack Overflow survey with over 88,000 respondents. Stack Overflow's annual Developer Survey is the largest and most comprehensive survey of people who code around the world. Each year, they field a survey covering everything from developers' favorite technologies to their job preferences. This year (2019) marked the ninth year they've published their annual Developer Survey results, and nearly 90,000 developers took the 20-minute survey earlier this year. (Stack Overflow, 2019). Despite our survey's broad reach and capacity in forming valuable conclusions, we acknowledge that our results don't represent everyone in the developer community evenly. The 2019 survey had nearly 90,000 respondents and nearly 70,000 of those respondents were employed as software professionals. Our results include selected questions from the survey as well as detailed demographics available. The results were analyzed using IBM SPSS 26.

RESULTS

A specific question in the survey asked what programming languages the participants had used in the past year. A checkbox list of 19 languages was used and respondents could select multiple boxes.

The top 10 programming languages from the Stack Overflow survey are shown in table 2.

Table 2: Stack Overflow Top 10 Languages

Language	Valid	Count	Percent
JavaScript	88884	59219	66.6%
HTML/CSS	88884	55466	62.4%
SQL	88884	47544	53.5%
Python	88884	36443	41.0%
Java	88884	35917	40.4%
Bash/Shell/PowerShell	88884	31991	36.0%
C#	88884	27097	30.5%
PHP	88884	23030	25.9%
C++	88884	20524	23.1%
TypeScript	88884	18523	20.8%

This table shows the results from all respondents including both professionals and non-professionals. It varies greatly from the IEEE survey. The most popular language is no longer Python but JavaScript. In fact, 66% of the respondents marked they have used JavaScript in the past year. Python was strong but only the fourth most used language and used by only 41% of respondents. The second most used language was the markup languages of HTML and CSS used by 62%. The database query language of SQL was third at 53%. So, the most popular languages used currently by over 50% of software professionals are not programming languages at all but a Scripting language, a markup language, and a data query language.

When we exclude non-professionals from the professional respondents, we still have over 65,000 respondents and the top three languages remain JavaScript, HTML/CSS, SQL and all are currently in use by more than 50% of the professionals. Amazingly, over 71% of the professionals are currently using JavaScript.

Table 3: Stack Overflow Top 10 Languages for Self-Identified Developers

Language	Valid	Count	Percent
JavaScript	65679	46945	71.5%
HTML/CSS	65679	41911	63.8%
SQL	65679	37235	56.7%
Java	65679	26348	40.1%
Python	65679	24285	37.0%
Bash/Shell/PowerShell	65679	24195	36.8%
C#	65679	21657	33.0%
PHP	65679	17042	25.9%
TypeScript	65679	16535	25.2%
C++	65679	13296	20.2%

Table 4: Stack Overflow List of Programming Languages for Professionals (All in Survey)

Language	Valid	Count	Percent
JavaScript	65679	46945	71.5%
HTML/CSS	65679	41911	63.8%
SQL	65679	37235	56.7%
Java	65679	26348	40.1%
Python	65679	24285	37.0%
Bash/Shell/PowerShell	65679	24195	36.8%
C#	65679	21657	33.0%
PHP	65679	17042	25.9%
TypeScript	65679	16535	25.2%
C++	65679	13296	20.2%
C	65679	11130	16.9%
Ruby	65679	5923	9.0%
Go	65679	5852	8.9%
Swift	65679	4652	7.1%
Kotlin	65679	4590	7.0%
Assembly	65679	3173	4.8%
VBA	65679	2987	4.5%
Scala	65679	2805	4.3%
R	65679	2731	4.2%
Rust	65679	1981	3.0%
Dart	65679	1244	1.9%
Elixir	65679	1064	1.6%
Clojure	65679	1029	1.6%
F#	65679	757	1.2%
WebAssembly	65679	745	1.1%
Erlang	65679	622	0.9%

When we examine the developer respondents by gender, we find that all languages have significant differences by gender at $p < .05$ except VBA (using Male, Female and Other). Between male and female, all languages showed higher usage by men than women. But some of these differences were not significant. The following languages showed statistically equal usage of the language for male and female: Clojure, JavaScript and Web Assembly. Surprisingly, Assembly was actually used by Other gender more than either men or women.

Next, we examined usage of programming languages by age group. The following tables show and describe where there were significant differences by age group. The count is the actual number of programmers in the age group who used the language in the past year. The percent is the percentage of individuals within that group that used the language. As a result, though there are more 5-30 year olds who programmed in Assembly, as a percentage they are lower than the 50+ group. The analysis that follows is based on percentages since that a more relevant indicator of preference within the age group.

Table 5: Assembly by Age Group

	Age	Count	Percent
Assembly	5-30	1733	5.1%
	31-50	875	3.7%
	51+	158	7.9%

Many who are Assembly programmers are over 50 but there is a new crop coming up 1-30 exceeds the mid 31-50 age group suggesting there has been new hiring in this area and younger programmers.

Table 6: Bash/Shell/PowerShell

Bash/Shell/PowerShell	Age	Count	Percent
	5-30	11622	34.3%
	31-50	9903	41.7%
	51+	825	41.1%

Bash/Shell/PowerShell is one of the top languages in our survey, but users are generally older. There may be a need for education and training as well as hiring in this language.

Table 7: Java by Age Group

	Age	Count	Percent
Java	5-30	14512	42.8%
	31-50	8567	36.1%
	51+	668	33.3%

Clearly, Java is a programming language of the young with much higher usage in the 1-30 age group. Since this is the fourth most popular language, this bodes well for the supply and demand of Java in the workplace.

Table 8: JavaScript by Age Group

	Age	Count	Percent
JavaScript	5-30	24862	73.3%
	31-50	16946	71.4%
	51+	1187	59.1%

Likewise, JavaScript skews to a younger demographic with usage by a staggering 73% of the 1-30 age group.

Table 9: Python by Age Group

	Age	Count	Percent
Python	5-30	13296	39.2%
	31-50	8242	34.7%
	51+	682	34.0%

Python, the fifth most popular language also skews younger and is well positioned as a present and future language.

Table 10: SQL by Age Group

	Age	Count	Percent
SQL	5-30	33905	55.7%
	31-50	23752	59.0%
	51+	2009	60.3%

SQL data query language has been around for a long time and the age demographic of SQL exhibits this with an increasing use by age.

Table 11: C by Age Group

	Age	Count	Percent
C	5-30	18893	17.7%
	31-50	14004	13.9%
	51+	1211	26.3%

C is a language used much more by older programmers.

JOB SATISFACTION

Our study next explores the usage of a language and whether the language has an impact on job satisfaction. A regression analysis was performed on all 26 languages. The results were that there are many languages that have no significant impact on job satisfaction including Assembly, C, C#, C++, Dart, Elixir, Erlang, F#, JavaScript, Rust, Scala, SQL, and Web Assembly. But there are some languages that positively affect job satisfaction and others that negative affect job satisfaction. Table 11 shows the significant languages at $p < .05$ and their negative to positive impact based on regression coefficient. VBA is the least popular, but Java and PHP also have negative impact on job satisfaction. All the other languages correlate with positive job satisfaction led by Bash/Shell/PowerShell. Python is also viewed positively. This may suggest that as a general programming language Python is preferred to Java and could result in higher job satisfaction.

Table 12: Programming Language and Job Satisfaction Correlation Coefficient

Language	Corr. Coeff.	Sig.
VBA	-0.085	0.000
Java	-0.075	0.000
PHP	-0.073	0.000
Swift	0.042	0.031
HTML/CSS	0.045	0.001
TypeScript	0.055	0.000
Go	0.055	0.002
Python	0.057	0.000
Kotlin	0.064	0.002
R	0.071	0.005
Ruby	0.091	0.000
Clojure	0.096	0.017
Bash/Shell/PowerShell	0.102	0.000

Next our study examines the multi-lingual nature of our developer respondents. Number of languages decrease the older you get. Overall, most programmers have code in about 5 languages.

Language combinations vary greatly. In other words, the multiple languages that developers know are not the same across the population. A correlation matrix (table 14) shows only a few combinations where correlation is over 25%. The differences between age groups was significant at $p < .001$. The language with the most correlations is JavaScript which correlates with HTML/CSS, SQL, PHP, and TypeScript, all web related languages. Assembly and C and C++ correlate as well as C and C++.

Table 13: Number of Languages by Age Group

	N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean		Minimum	Maximum
					Lower Bound	Upper Bound		
1-30	47544	5.05753	2.581990	.011842	5.03432	5.08074	.000	27.000
31-50	28722	4.80715	2.397386	.014146	4.77942	4.83488	.000	27.000
51+	2943	4.61298	2.641520	.048692	4.51751	4.70845	.000	27.000
Total	79209	4.95022	2.522559	.008963	4.93265	4.96779	.000	27.000

Table 14: ANOVA, Number of Languages by Age

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	1470.056	2	735.028	115.845	.000
Within Groups	502554.663	79206	6.345		
Total	504024.719	79208			

Table 15: Programming Language Correlation Matrix

	Assembly	C	Ehrlang	HTML/CSS	JavaScript
Assembly	1				
Bash/Shell/PowerShell	0.139597				
C	0.354841				
C#	0.048256				
C++	0.267989	0.531343			
Elixir			0.36459		
JavaScript				0.596513	
SQL				0.297141	0.261564
PHP					0.269576
TypeScript					0.282895

Table 16: Number of Languages by Gender

	N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean		Minimum	Maximum
					Lower Bound	Upper Bound		
Male	77919	4.96939	2.543853	.009113	4.95153	4.98725	.000	27.000
Female	6344	4.38651	2.474282	.031065	4.32561	4.44740	.000	27.000
Other	4439	4.56747	3.252605	.048819	4.47176	4.66318	.000	27.000
Total	88702	4.90759	2.584606	.008678	4.89058	4.92460	.000	27.000

Number of languages are significantly higher for men than women at $p < .001$. The results for all respondents is shown in table 16. For only developers the results are shown in table 17.

Table 17: Number of Languages by Gender – Developers Only

	N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean		Minimum	Maximum
					Lower Bound	Upper Bound		
Male	58371	5.01425	2.443706	.010115	4.99443	5.03408	.000	27.000
Female	4355	4.42801	2.281327	.034570	4.36024	4.49579	.000	27.000
Other	2827	4.84542	3.002443	.056469	4.73469	4.95614	.000	27.000
Total	65553	4.96803	2.464481	.009626	4.94916	4.98689	.000	27.000

IMPLICATIONS AND CONCLUSION

The results of our study compared to prior literature have areas of both agreement and extension. Our first exploration was on the use of specific programming languages. IEEE in 2019 published a study on the “most popular” programming languages and used a variety of sources. They found Python as the most popular followed by C++ and Java. Siegfried, Siegfried, & Alexandro, (2016) found that Java was the most popular language taught in university introductory courses. Sanatinia and Noubir (2016) found the scripting language Javascript to be the most used language in Github projects. Johri and Bansal (2018) found a decrease in object-oriented languages such as Java and C++. Our study used the Stack Overflow survey and found the actual languages used by programmers in the past year. The results support the top languages but in a somewhat different order. We found JavaScript, HTML/CSS, and SQL as the top languages but these are followed next by both Python and Java. This supports both the remaining but declining popularity of OO languages and the rising popularity of scripting, web, and database languages. We propose that this is an important trend that needs to be addressed in today’s university software development education.

We next explored the demographic differences for programming languages. We found that gender plays an impact in programming languages with web programming languages more utilized by females rather than males. This may relate to an inherent gender bias within the profession for “hard” coding languages versus “soft” scripting and presentation languages. This supports the Schina, González, & Rodríguez, 2019 finding that females found introductory programming (which is generally done in coding languages) to be less fun than males found it. Age also has an impact in programming. Languages such as Assembler and C are more used by older individuals whereas Java and JavaScript are used by younger programmers. Also, older individuals were found to use significantly less programming languages within the year.

An area that has not been studied is the need for multi-lingual software language education. In 2020, professionals and non-professionals alike have a great many options in computer coding languages. In general, no one language

suffices for all applications as noted by Cass (2019). We found the average professional has worked with between four and five languages in the past year. They have actively used four to five languages in the year that was surveyed. As noted, older individuals were found to use significantly less programming languages within the year. Today's coder must be multi-lingual. This also needs to be addressed in today's university technology programs.

Another area examined was the relationship between job satisfaction and programming in a specific language. Some languages are less satisfying than others. VBA, Java, and PHP correlated with lower job satisfaction whereas TypeScript, Python, R, Ruby, and Bash/Shell/PowerShell contributed to higher job satisfaction. This may suggest that alternate languages can be examined to improve programmer job satisfaction.

Overall, though we see a diverse panoply of skills sets that employers are utilizing and in practice by their employees. This presents interesting challenges for educating and developing young programmers. It also presents an opportunity to make sure that the most popular languages including JavaScript, SQL, and HTML/CSS are properly emphasized in a curriculum since more than 50% of professionals are currently using these languages. Employers would be eager for students to possess these skills to hit the ground running in their beginning careers.

REFERENCES

Ahasanuzzaman, M., Asaduzzaman, M., Roy, C. K., & Schneider, K. A. (2016, May). Mining duplicate questions of stack overflow. In *2016 IEEE/ACM 13th Working Conference on Mining Software Repositories (MSR)* (pp. 402-412). IEEE.

Bissyand, T., et al: Popularity, Interoperability, and Impact of Programming Languages in 100,000 *Open Source Projects: Computer Software & Apps. Conference (COMPSAC)*, pp. 303-312, 2013.

Cass, S. (2015). The 2015 Top Ten Programming Languages. *IEEE Spectrum*, July, 20.

Cass, S. (2019) The Top Programming Languages. *IEEE Spectrum*, Sept, 6. <https://spectrum.ieee.org/computing/software/the-top-programming-languages-2019>

Chen, J. (2018). Linguistic Relativity and Programming Languages. *arXiv preprint arXiv:1808.03916*.

Felleisen, M. (2008). Why teach programming languages in this day and age and how to go about it. *ACM Sigplan Notices*, 43(11), 59-61.

Hafkin, N. J., & Taggart, N. (2001). *Gender, information technology, and developing countries: An analytic study*. Office of Women in Development, Bureau for Global Programs, Field Support and Research, United States Agency for International Development.

Johri, V., & Bansal, S. (2018, January). Identifying trends in technologies and programming languages using Topic Modeling. In *2018 IEEE 12th International Conference on Semantic Computing (ICSC)* (pp. 391-396). IEEE.

Kochhar, P. S., Wijedasa, D., & Lo, D. (2016, March). A large-scale study of multiple programming languages and code quality. In *2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)* (Vol. 1, pp. 563-573). IEEE.

May, A., Wachs, J., & Hannák, A. (2019). Gender differences in participation and reward on Stack Overflow. *Empirical Software Engineering*, 24(4), 1997-2019.

Noone, M., & Mooney, A. (2019). First Programming Language-Java or Snap? A Short Course Perspective.

Sanatinia, A., & Noubir, G. (2016). On GitHub's Programming Languages. *arXiv preprint arXiv:1603.00431*.

Schina, D., González, V. E., & Rodríguez, M. U. (2019, March). Gender differences in students' feedback and performance in Scratch programming. In *Conference Proceedings EDUNOVATIC 2018: 3rd Virtual International Conference on Education, Innovation and ICT* (p. 36). Adaya Press.

Siegfried, R. M., Siegfried, J., & Alexandro, G. (2016). A longitudinal analysis of the Reid list of first programming languages. *Information Systems Education Journal*, 14(6), 47.

Silveira, K. K., Musse, S., Manssour, I. H., Vieira, R., & Prikladnicki, R. (2019, May). Confidence in programming skills: gender insights from StackOverflow developers survey. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)* (pp. 234-235). IEEE.

Stack Overflow <https://insights.stackoverflow.com/survey/2019>