

اگرچه در رتبه‌بندی زبان‌های برنامه‌نویسی ماه ژوئن ۲۰۲۰ سایت RedMonk، زبان پایتون رتبه دوم را کسب کرده و بعد از جاوا اسکریپت قرار دارد، اما برنده این گزارش محسوب می‌شود زیرا این رتبه را در ماه ژانویه و با پشت سر گذاشتن جاوا به دست آورده است (رتبه بندی این سایت هر ۶ ماه منتشر می‌شود). پایتون، در آخرین رتبه‌بندی ۶ ماهه اول ۲۰۲۴ نیز کماکان این جایگاه را حفظ کرده است.

در رتبه‌بندی سال ۲۰۲۰ برای اولین بار است (از زمان شروع رتبه بندی در سال ۲۰۱۲) که زبانی به جز جاوا یا جاوا اسکریپت رتبه دوم را به دست آورده است. همچنین این اولین بار است که جاوا پایین‌تر از رتبه دوم قرار دارد. همچنین این سایت گزارش داده است که Python در هزاران پروژه کوچک استفاده شده و مبنای بسیاری از اسکریپت‌های شخصی است. از جمله قسمتی که داده‌های وبسایت RedMonk را بازیابی می‌کند. همچنین در زمینه‌هایی مانند Data Science کاربرد دارد.

وبسایت RedMonk رتبه‌بندی را براساس تجزیه و تحلیل داده‌های سایت‌های Github و Stack Overflow استخراج می‌کند. در گیت هاب ارزیابی توسط Pull request های زبان‌ها انجام می‌شود و در استک اورفلو، "بحث و گفتگو" در مورد زبان‌ها مورد ارزیابی قرار می‌گیرند.

همچنین، زبان جاوا هیچگاه با رقابت بیشتری با توجه به زمان توسعه و توجه توسعه دهندگان، مواجهه نشده بود. در حالی که جاوا یک زبان اینترپرایز است و در پروژه‌های بسیاری زیادی استفاده شده است، شرکت توسعه‌دهنده آن اعلام کرده است که دیگر نمی‌توان حفظ جایگاه برجسته جاوا را تضمین کرد.



the developer-focused industry analyst firm

---

## TECOSYSTEMS

# The RedMonk Programming Language Rankings: June 2024

By [Stephen O'Grady](#) | [@sograpy](#) | September 12, 2024

This iteration of the RedMonk programming Language Rankings is brought to you by Amazon Web Services. AWS manages a variety of [developer communities](#) where you can join and learn more about building modern applications in your preferred language.

As has become typical in recent years, our Q3 programming language rankings are arriving a few months late. Unlike the last run when that was primarily due to the anomalous results we observed, this quarter it's more attributable to summer vacation schedules. We're still waiting to see what the longer term implications of coding assistants will be on these rankings, but at least for the present we're continuing with the exercise as it continues to identify trends for us in the market. Trends which we'll get to shortly.

In the meantime, however, as a reminder, this work is a continuation of the work originally performed by Drew Conway and John Myles White late in 2010. While the specific means of collection has changed, the basic process remains the same: **we extract language rankings from GitHub and Stack Overflow, and combine them for a ranking that attempts to reflect both code (GitHub) and discussion (Stack Overflow) traction. The idea is not to offer a statistically valid representation of current usage, but rather to correlate language discussion and usage in an effort to extract insights into potential future adoption trends.**

این شاخص یک نمایش آماری معتبر از میزان استفاده فعلی از زبان‌ها را ارائه نمی‌دهد. بلکه، با ارتباط دادن حجم مباحث پیرامون زبان و میزان استفاده از آن، بینشی درخصوص روند پذیرش احتمالی زبان در آینده ارائه می‌دهد.

# Our Current Process

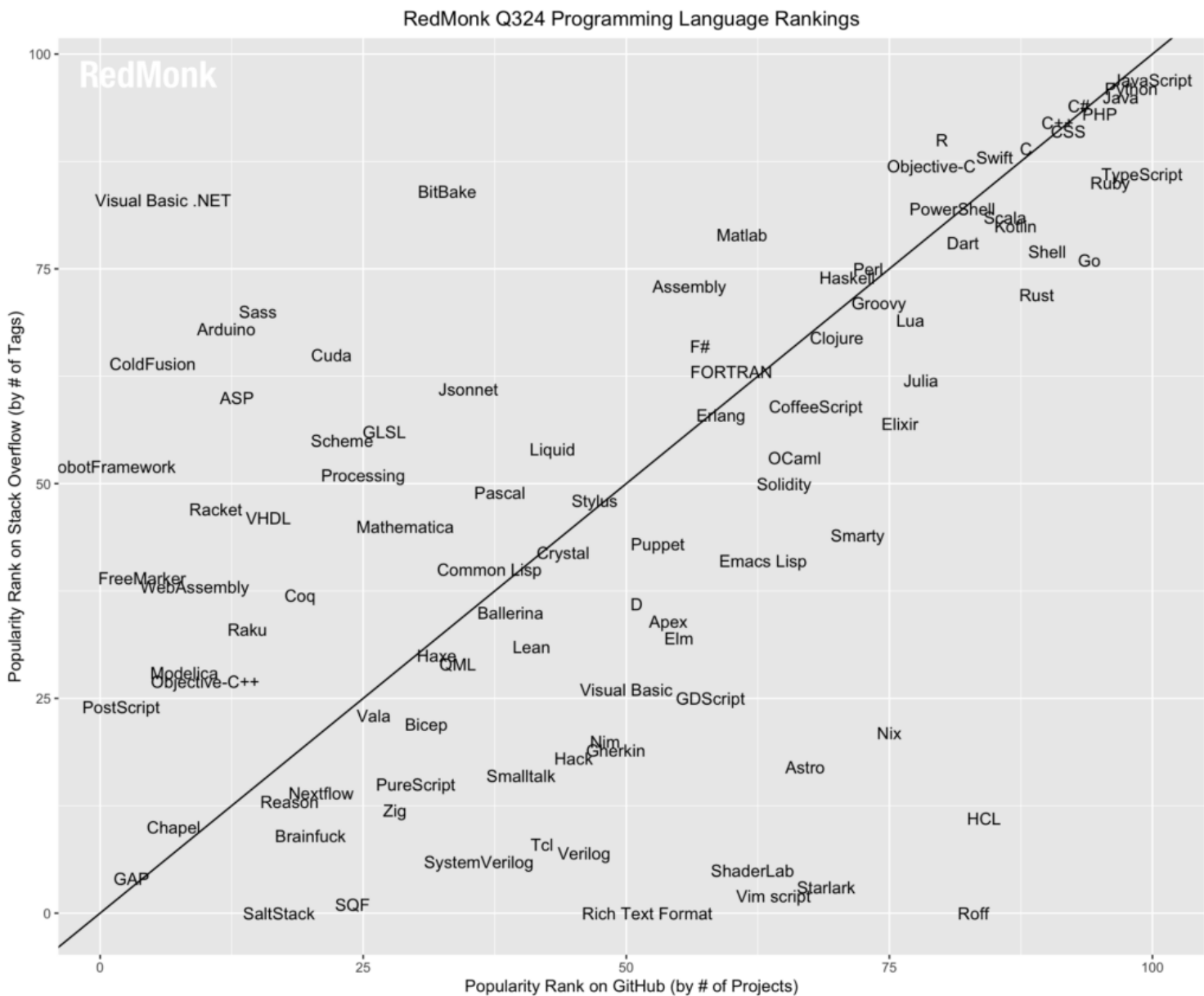
The data source used for the GitHub portion of the analysis is the GitHub Archive. We query languages by pull request in a manner similar to the one GitHub used to assemble the State of the Octoverse. Our query is designed to be as comparable as possible to the previous process.

- Language is based on the base repository language. While this continues to have the caveats outlined below, it does have the benefit of cohesion with our previous methodology.
- We exclude forked repos.
- We use the aggregated history to determine ranking (though based on the table structure changes this can no longer be accomplished via a single query.)
- For Stack Overflow, we simply collect the required metrics using their useful data explorer tool.

With that description out of the way, please keep in mind the other usual caveats.

- To be included in this analysis, a language must be observable within both GitHub and Stack Overflow. If a given language is not present in this analysis, that's why.
- No claims are made here that these rankings are representative of general usage more broadly. They are nothing more or less than an examination of the correlation between two populations we believe to be predictive of future use, hence their value.
- There are many potential communities that could be surveyed for this analysis. GitHub and Stack Overflow are used here first because of their size and second because of their public exposure of the data necessary for the analysis. We encourage, however, interested parties to perform their own analyses using other sources.
- All numerical rankings should be taken with a grain of salt. We rank by numbers here strictly for the sake of interest. In general, the numerical ranking is substantially less relevant than the language's tier or grouping. In many cases, one spot on the list is not distinguishable from the next. The separation between language tiers on the plot, however, is generally representative of substantial differences in relative popularity.
- In addition, the further down the rankings one goes, the less data available to rank languages by. Beyond the top tiers of languages, depending on the snapshot, the amount of data to assess is minute, and the actual placement of languages becomes less reliable the further down the list one proceeds.
- Languages that have communities based outside of Stack Overflow such as Mathematica will be under-represented on that axis. It is not possible to scale a process that measures one hundred different community sites, both because many do not have public metrics available and because measuring different community sites against one another is not statistically valid.

With that, here is the first quarter plot for 2024.



- |              |                |
|--------------|----------------|
| 1 JavaScript | 11 Swift       |
| 2 Python     | 12 Go          |
| 3 Java       | 12 R           |
| 4 PHP        | 14 Shell       |
| 5 C#         | 14 Kotlin      |
| 6 TypeScript | 14 Scala       |
| 7 CSS        | 17 Objective-C |
| 7 C++        | 18 PowerShell  |
| 9 Ruby       | 19 Rust        |
| 10 C         | 19 Dart        |

The Top 20, in a fashion that has become typical in recent years, was not entirely devoid of movement, but nearly so. Outside of CSS moving down a spot and C++ moving up one, the Top 10 was unchanged. And even in the back half of the rankings, where languages tend to be less entrenched and movement is more common, only three languages moved at all.

While these rankings are, as explicitly acknowledged above, not intended to be an accurate representation of typical enterprise language usage – because the data is not available to measure that – what it is clearly evidence of is a landscape resistant to change. There are a few signs of languages following in TypeScript’s footsteps and working their way up the path, both in the Top 20 and at the back end of the Top 100 as we’ll discuss shortly, but they’re the exception that proves the rule.

It’s possible that we’ll see more fluid usage of languages, and increased usage of code assistants would theoretically make that much more likely, but at this point it’s a fairly static status quo.

With that, some results of note:

- **TypeScript (6)**: technically TypeScript didn’t move, as it was ranked sixth in our last run, but this is the first quarter in which it has been the sole occupant of that spot. CSS, in this case, dropped one place to seven leaving TypeScript just outside the Top 5. It will be interesting to see whether or not it has more momentum to expend or whether it’s topped out for the time being.
- **Kotlin (14) / Scala (14)**: both of these JVM-based languages jumped up a couple of spots – two spots in Scala’s case and three for Kotlin. Scala’s rise is notable because it had been on something of a downward trajectory from a one time high of 12th, and Kotlin’s placement is a mild surprise because it had spent three consecutive runs not budging from 17, only to make the jump now. The tie here, meanwhile, is interesting because Scala’s long history gives it an accretive advantage over Kotlin’s more recent development, but in any case the combination is evidence of the continued staying power of the JVM.
- **Objective C (17)**: speaking of downward trajectories and the 17th placement on this list, Objective C’s slide that began in mid-2018 continued and left the language with its lowest placement in these rankings to date at 17. That’s still an enormously impressive achievement, of course, and there are dozens of languages that would trade their usage for Objective C’s, but the direction of travel seems clear.
- **Dart (19) / Rust (19)**: while once grouped with Kotlin as up and coming languages driven by differing incentives and trends, Dart and Rust have not been able to match the ascent of their counterpart with five straight quarters of no movement. That’s not necessarily a negative; as with Objective C, these are still highly popular languages and communities, but it’s worth questioning whether new momentum will arrive and from where, particularly because the communities are experiencing [some friction](#) in growing their usage.
- **Ballerina (61) / Bicep (78) / Grain / Moonbit / Zig (87)**: as discussed during last quarter’s run, we’re keeping an eye on Bicep, Grain, Moonbit and Zig among others because of what they represent: an unusually visible cloud DSL, two languages optimized for WebAssembly and then a language that follows in the footsteps of C++ and Rust. Grain and Moonbit still haven’t made it into the Top 100, but Bicep jumped eight spots to 78 and Zig 10 to 87. That progress pales next to Ballerina, however, which jumped from 80 to 61 this quarter. The general purpose language from WS02, thus, is added to the list of potential up and comers we’re keeping an eye on.

**Disclosure:** WS02 is not currently a RedMonk client.

**Credit:** My colleague [Rachel Stephens](#) wrote the queries that are responsible for the GitHub axis in these rankings. She is also responsible for the query design for the Stack Overflow data.

**گزارش ۶ سال پیش:** در شاخص تیوبی (Tiobe) که به صورت ماهانه ارائه می‌شود پایتون به جایگاه جدیدی از محبوبیت دست پیدا کرده است. پایتون در این شاخص بعد از جاوا و C در مقام سوم قرار دارد اما به گفته تحلیلگران Tiobe اگر این روند رشد ادامه پیدا کند می‌تواند در چند سال بعدی پایتون به مقام اول جدول برسد. این یعنی پایتون می‌تواند جایگاه محبوب‌ترین زبان برنامه‌نویسی را بدست بیاورد.

براساس رتبه‌بندی جدید Tiobe در ماه ژوئن پایتون ۸,۵۳ درصد در این جدول رشد کرده؛ البته از نظر جایگاه تغییری حاصل نشده اما همانطور که گفته شد اگر این روند رو به رشد که ماه دسامبر سال قبل ۸,۳۷ بود ادامه پیدا کند می‌تواند از زبان‌های سی و جاوا پیشی بگیرد. براساس نظریات Tiobe پایتون از آنجایی که استفاده‌پذیری بسیار ساده‌ای داشته و کار با آن برای همه افراد مبتدی امکان‌پذیر است، می‌تواند محبوبیت بسیار بالایی را در گذر زمان کسب کند.

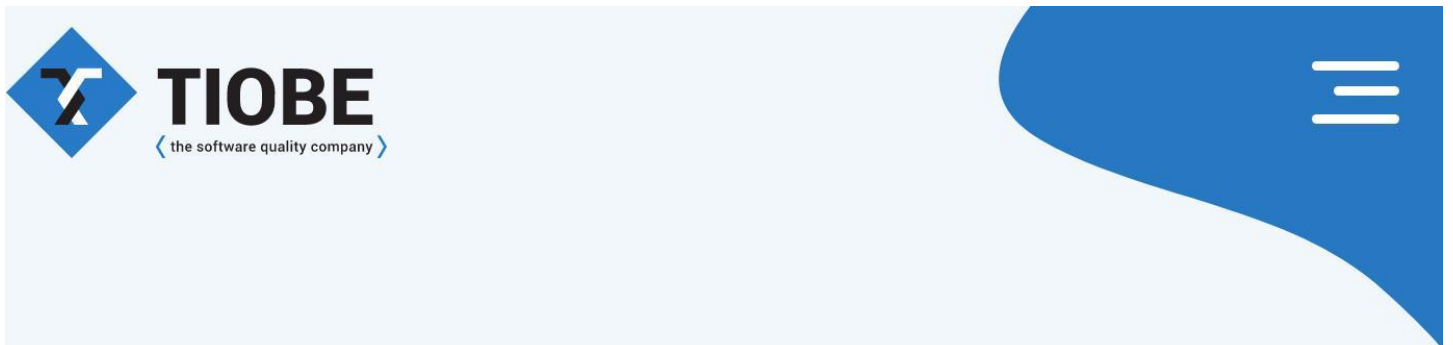
شاخص Tiobe اطلاعات خود را از طریق جستجوهای مختلف در موتورهای جستجوگر گوگل، یاهو! و بینگ بدست می‌آورد. فاکتورهای اصلی برای این شاخص محبوبیت، میزان آموزش‌ها، مهندسين توانا و کتابخانه‌های هر یک از زبان‌هاست.

شاخص تیوبی (Tiobe) از تعداد نتایج جستجو برای عبارت‌های حاوی نام زبان محاسبه می‌شود؛ موتورهای جستجوگر شامل Google, Google Blogs, MSN, Yahoo!, Baidu, Wikipedia و YouTube می‌شود.

در شاخص‌های دیگری مانند شاخص Pypl زبان پایتون به عنوان محبوب‌ترین زبان برنامه‌نویسی در جهان شناخته شده است.

**گزارش جدید در ژانویه ۲۰۲۵:**

اهمیت جدی بودن **The Importance of Being Earnest**:



# TIOBE Index for January 2025

## January Headline: Python is TIOBE's programming language of the year 2024!

Programming language Python has won the title "TIOBE's programming language of the year 2024". This award is given to the programming language with the highest increase in ratings in one year time. Python gained a whopping 9.3% in 2024. This is far ahead of its competition: Java +2.3%, JavaScript +1.4% and Go +1.2%. Python is everywhere nowadays, and it is the undisputed default language of choice in many fields. It might even become the language with the highest ranking ever in the TIOBE index. Python's only serious drawbacks are (and thus leaving room for competition) its lack of performance and that most errors occur run-time.






Apart from this, two interesting things happened in the TIOBE index top 10 in 2024. The C language lost a lot of popularity and was surpassed by C++ and Java. The main reason for this is that C is replaced by C++ in a lot of embedded software systems. Java and C++ are currently fighting for the top 2 position. The other interesting thing is that PHP said a final farewell to the top 10 and is replaced by Go, which is a top 10 keeper.














Lots of eyes were on Rust and Kotlin this year. Did they live up to their promises in 2024? Rust is still getting more popular. Despite the amazing speed of Rust programs, its steep learning curve will never make it become the lingua franca of the common programmer unfortunately. Kotlin, on the other hand, disappointed: it didn't break through and even lost (possibly permanently) its top 20 position in 2024.

Further down the list we see two new promising languages: Rust's competitor Zig climbed from #149 to #61 in 2024, whereas Mojo, the faster Python, jumped from position #194 to #68. Especially Mojo, which was first released only 2 years ago, addresses exactly what is needed in the programming field. I have high hopes that it will get close to a top 20 position in 2025.

The TIOBE Programming Community index is an indicator of the popularity of programming languages. The index is updated once a month. The ratings are based on the number of skilled engineers world-wide, courses and third party vendors. Popular web sites Google, Amazon, Wikipedia, Bing and more than 20 others are used to calculate the ratings. It is important to note that the TIOBE index is not about the *best* programming language or the language in which *most lines of code* have been written.

The index can be used to check whether your programming skills are still up to date or to make a strategic decision about what programming language should be adopted when starting to build a new software system. The definition of the TIOBE index can be found [here](#).

Jan 2025	Jan 2024	Change	Programming Language	Ratings	Change
1	1		 Python	23.28%	+9.32%
2	3	▲	 C++	10.29%	+0.33%
3	4	▲	 Java	10.15%	+2.28%
4	2	▼	 C	8.86%	-2.59%
5	5		 C#	4.45%	-2.71%

Jan 2025	Jan 2024	Change	Programming Language	Ratings	Change
6	6		 JavaScript	4.20%	+1.43%
7	11	⬆️	 Go	2.61%	+1.24%
8	9	⬆️	 SQL	2.41%	+0.95%
9	8	⬇️	 Visual Basic	2.37%	+0.77%
10	12	⬆️	 Fortran	2.04%	+0.94%
11	13	⬆️	 Delphi/Object Pascal	1.79%	+0.70%
12	10	⬇️	 Scratch	1.55%	+0.11%
13	7	⬇️	 PHP	1.38%	-0.41%
14	19	⬆️	 Rust	1.16%	+0.37%
15	14	⬇️	 MATLAB	1.07%	+0.09%
16	18	⬆️	 Ruby	1.06%	+0.25%
17	15	⬇️	 Assembly language	1.01%	+0.10%
18	23	⬆️	 R	1.00%	+0.27%
19	16	⬇️	 Swift	0.99%	+0.10%
20	20		 COBOL	0.95%	+0.17%

نظرسنجی و رتبه‌بندی‌هایی از این قبیل به کاربران و توسعه‌دهندگان کمک کرده تا متوجه محبوبیت زبان‌های برنامه‌نویسی و ترندهای آن شده و به دنبال یادگیری یا کار در حوزه‌های محبوب‌تر روند.





پایتون در آخرین رتبه‌بندی سالانه محبوب‌ترین زبان‌های برنامه‌نویسی از سوی IEEE (مؤسسه مهندسان برق و الکترونیک) مجدداً به رتبه اول دست پیدا کرده است.

"By popularity, we mean we are trying to rank languages that are in active use," IEEE Spectrum special projects editor Stephen Cass explains.

"We look at three different aspects of popularity:

languages in active use among typical IEEE members and working software engineers (the 'Spectrum' ranking),

languages that are in demand by employers (the 'Jobs' ranking), and

languages that are in the zeitgeist (the 'Trending' ranking)."

Zeitgeist (روح زمانه واژه مرکب از دو کلمه آلمانی «تسایت» (زمان) و «گایست» (روح) ترجمه واژه‌ای است فلسفی از زبان آلمانی که خود برگرفته از زبان لاتین «genius saeculi» است. این عبارت آلمانی از طریق زبان انگلیسی تقریباً به همه زبان‌های دنیا راه باز کرده و بدون تبدیل به زبان بومی، مورد استفاده قرار می‌گیرد. روح زمانه روش و نوع تفکر و احساسات یک مقطع زمانی یا به عبارت دیگر دوران خاصی را تشریح یا توجیه می‌کند.

طبق اعلام IEEE، پایتون به دلیل داشتن تعداد زیادی کتابخانه تخصصی مخصوصاً برای توسعه‌دهندگان فعال در حوزه هوش مصنوعی، توانسته به این جایگاه دست یابد. برای مثال می‌توان به کتابخانه Keras پایتون اشاره کرد که رابط‌هایی برای [TensorFlow](#) (توسعه‌داده‌شده از سوی گوگل)، [جعبه ابزار شناختی میکروسافت](#) (CNTK) و یادگیری عمیق [Theano](#) دارد.

حوزه دیگری که از زمان ظهور پایتون در سال ۱۹۹۱ به وجود آمده، میکروکنترلرهای رایانه‌های کوچک ارزان‌قیمت مانند [رزیبری‌پی](#) و آدافروت است.

طبق اعلام مجله IEEE Spectrum این رتبه‌بندی براساس یازده آمار از هشت منبع شامل CareerBuilder، [گوگل](#)، [گیت‌هاب](#)، هکر نیوز، IEEE، ردیت، Stack Overflow و [توییتر](#) است.

نکته مهم این رتبه‌بندی تغییر رتبه PHP و خارج شدن آن از لیست ده زبان برتر است که PHP از سال ۲۰۰۱ در آن حضور داشته‌است. تحلیلگران TioBE می‌گویند:

زبان php از ابتدا ویژوال بیسیک طراحی وب بود: یادگیری آسان، اجرای آسان. اما غالبا از سوی طراحان وب با پیش‌زمینه کمی از مهندسی نرم‌افزار مورد استفاده قرار می‌گرفت. افول php به دلیل حفره‌های امنیتی راحت‌الاستفاده آن است.

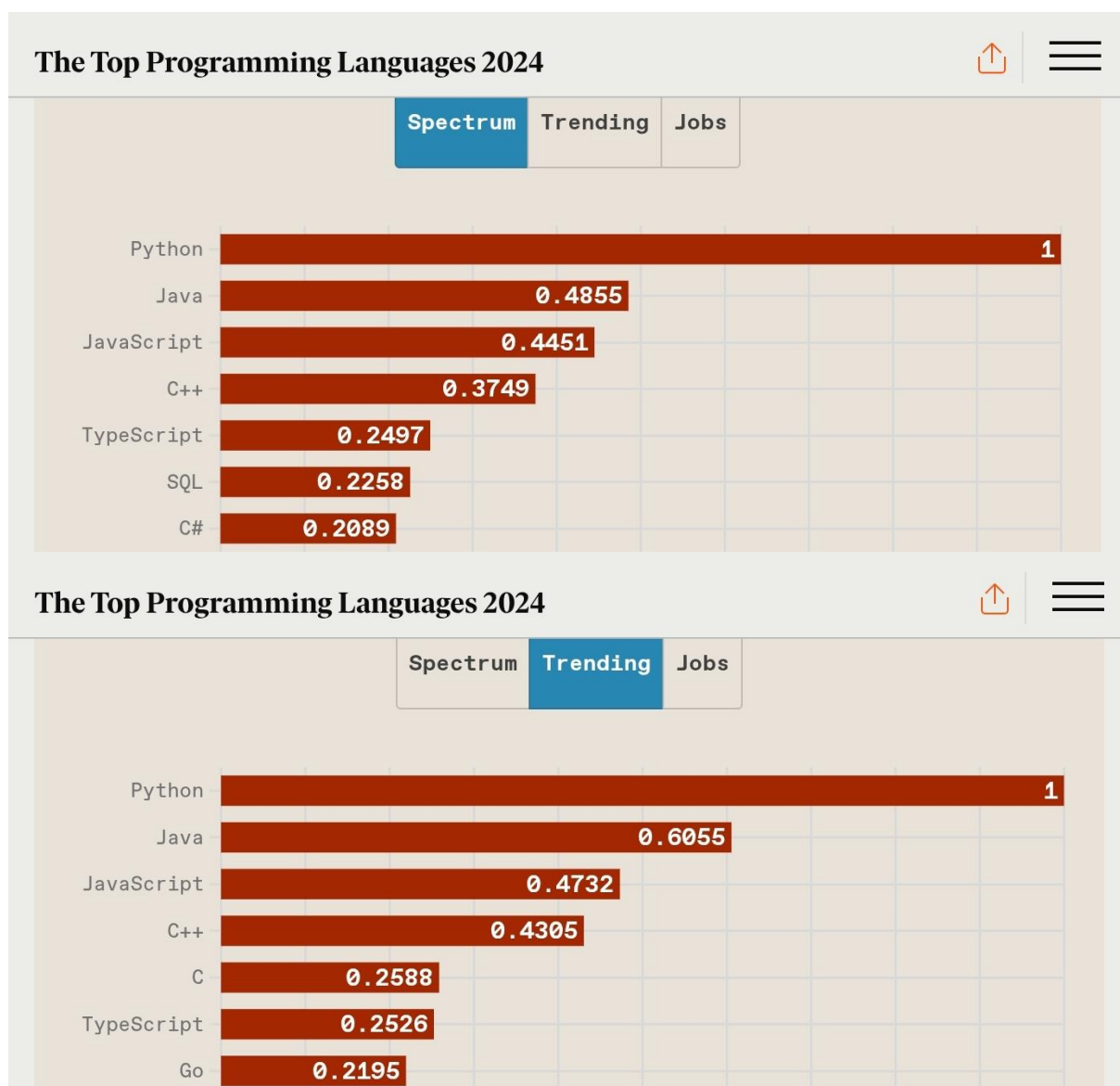
این تحلیل‌گران همچنین اشاره کردند که فیسبوک، که ابتدا با زبان PHP ساخته شده بود، جایگزین PHP یعنی Hack را در سال ۲۰۱۴ معرفی کرد و از آن پس استفاده از جاوااسکریپت، تایپ‌اسکریپت و پایتون رونق بسیار فراوانی یافت.

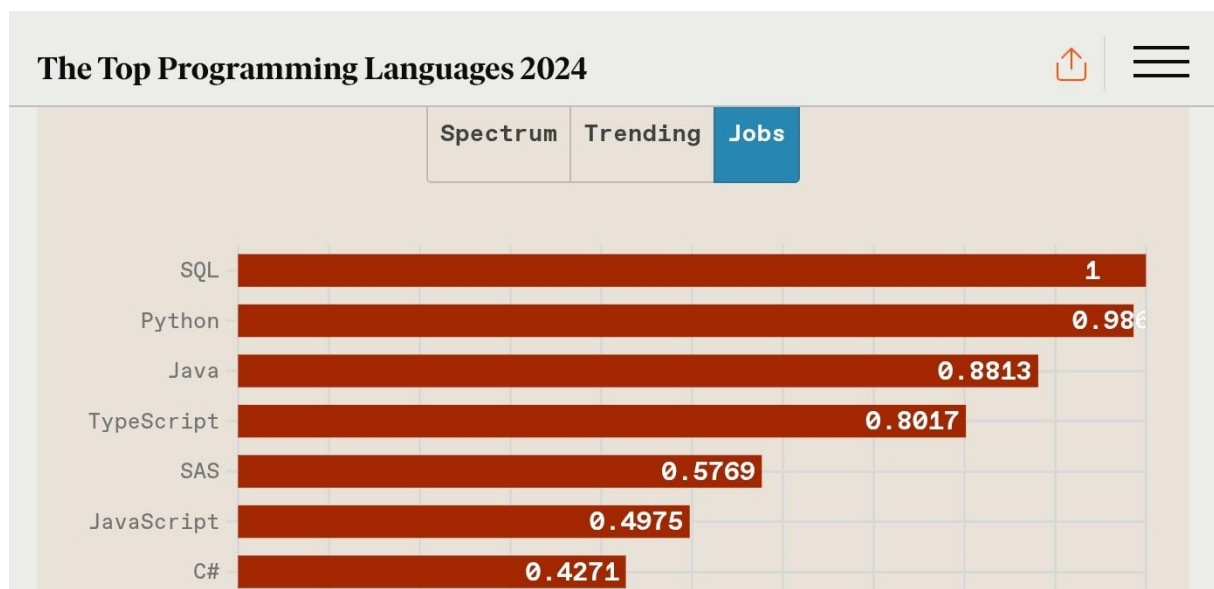
## The Top Programming Languages 2024

Stephen Cass

۲۲Aug 2024

Stephen Cass is the special projects editor at IEEE Spectrum.





Welcome to IEEE Spectrum’s 11th annual rankings of the most popular programming languages. As always, we combine multiple metrics from different sources to create three meta rankings. The “Spectrum” ranking is weighted towards the profile of the typical IEEE member, the “Trending” ranking seeks to spot languages that are in the zeitgeist, and the “Jobs” ranking measures what employers are looking for.

You can find a full breakdown of our methodology here, but let’s jump into our results. At the top, Python continues to cement its overall dominance, buoyed by things like popular libraries for hot fields such as AI as well as its pedagogical prominence. (For most students today, if they learn one programming language in school, it’s Python.) Python’s pretty popular with employers too, although there its lead over other general purpose languages is not as large and, like last year, it plays second fiddle to the database query language SQL, which employers like to see paired with another language. SQL popularity with employers is a natural extension of today’s emphasis on networked and cloud-based system architectures, where databases become the natural repository for all the bytes a program’s logic is chewing on.

Stalwarts like Java, Javascript, and C++ also retain high rankings, but it’s what’s going on a little further down that’s particularly interesting. Typescript—a superset of Javascript—moves up several places on all the rankings, especially for Jobs, where it climbs to fourth place, versus 11th last year. Typescript’s primary differentiator over Javascript is that it enforces static typing of variables, where the type of a variable—integer, floating point, text, and so forth—must be declared before it can be used. This allows for more error checking when Typescript programs are compiled to Javascript, and the increase in reliability has proven appealing.

Another climber is Rust, a language aimed at creating system software, like C or C++. But unlike those two languages, Rust is “memory safe”, meaning it uses a variety of techniques to ensure programs can’t write to locations in memory that they are not supposed to. Such errors are a major source of security vulnerabilities. Rust’s profile has been rising sharply, boosted by things like a February cybersecurity report from the White House calling for memory safe languages to replace C and C++. Indeed, C’s popularity appears to be on the wane, falling from fourth to ninth place on the Spectrum ranking and from 7th to 13th on the Jobs ranking.

Two languages have entered the rankings for the first time: Apex and Solidity. Apex is designed for building business applications that use a Salesforce server as a back end, and Solidity is designed for creating smart contracts on the Ethereum blockchain.

This year also saw several languages drop out of the rankings. This doesn't mean a language is completely dead, it just means that these languages' signal is too weak to allow them to be meaningfully ranked. Languages that dropped out included Forth, a personal favorite of mine that's still popular with folks building 8-bit retro systems because of its tiny footprint. A weak signal is also why we haven't included some buzzy languages such as Zig, although those proficient in it can apparently command some high salaries.

As these other languages come and go from the rankings, I have to give the shout out to the immortals, Fortran and Cobol. Although they are around 65 years old, you can still find employers looking for programmers in both. For Fortran, this tends to be for a select group of people who are also comfortable with high-energy physics, especially the kind of high-energy physics that goes boom (and with the security clearances to match). Cobol is more broadly in demand, as many government and financial systems still rely on decades-old infrastructure—and the recent paralyzing impact of the CloudStrike/Microsoft Windows outage incident probably hasn't done much to encourage their replacement!

## معرفی زبان‌های برنامه‌نویسی برتر وب در سال ۲۰۲۴

زبان نشانه گذاری HTML یا [HyperText](#)، پایه و اساس توسعه وب است که هر توسعه دهنده‌ای باید با آن آشنا باشد. در حقیقت HTML و CSS واقعاً زبان‌های برنامه‌نویسی نیستند اما آنها ستون فقرات توسعه وب هستند و هر توسعه دهنده وب باید آنها را یاد بگیرد HTML. ساختار را ارائه می‌دهد در حالی که CSS سبک را ارائه می‌دهد و به آنها کمک می‌کند بهتر و از نظر بصری جذاب تر به نظر برسند. خوبی این دو زبان این است که می‌توانید به راحتی در عرض چند هفته به آنها مسلط شوید و همچنین یادگیری آنها نسبتاً راحت تر است.

[جاوا اسکریپت](#) یک زبان برنامه نویسی سطح بالا است که یکی از فناوری های اصلی شبکه جهانی وب است. ۹۷٫۸ درصد از تمام وب سایت‌ها از این زبان برنامه نویسی سمت کلاینت استفاده می‌کنند. جاوا اسکریپت در ابتدا فقط برای توسعه مرورگرهای وب استفاده می‌شد، اما اکنون از آن برای استقرار وب سایت سمت سرور و برنامه های غیر مرورگر وب نیز استفاده می‌شود.

جاوا اسکریپت در سال ۱۹۹۵ ایجاد شد و در ابتدا با نام LiveScript شناخته می‌شد. با این حال، جاوا در آن زمان یک زبان بسیار محبوب بود، بنابراین به عنوان برادر کوچکتر جاوا تبلیغ می‌شد. همانطور که در طول زمان تکامل یافت، جاوا اسکریپت به یک زبان کاملاً مستقل تبدیل شد. بدون شک جاوا اسکریپت یکی از محبوب‌ترین زبان‌ها در بین توسعه‌دهندگان وب است. همچنین جاوا اسکریپت تنها زبانی است که به شما امکان می‌دهد برنامه‌های تحت وب، فرانت‌اند، بک‌اند و اپلیکیشن‌های موبایل (React Native) ایجاد کنید. در واقع یک زبان برنامه نویسی همه کاره است.

نقطه قوت جاوا اسکریپت فقط این نیست که می‌تواند روی مرورگر و سرور با استفاده از Nodejs اجرا شود، بلکه فریمورک‌ها و کتابخانه‌های فوق‌العاده‌ای برای توسعه وب و توسعه برنامه دارد. به عنوان مثال، می‌توانید از

React.js و Angular برای فرانت‌اند استفاده کنید Nodejs. را برای بک‌اند استفاده کنید و React Native را برای ایجاد برنامه‌های موبایلی چند پلتفرمی (هم اندروید و هم iOS) استفاده کنید. با [یادگیری JavaScript](#) می‌توانید هم در بخش بک‌اند و هم در بخش فرانت‌اند یک وب سایت کار کنید.

**Python** توسط برنامه نویس هلندی Guido van Rossum ایجاد شد و اولین بار در سال ۱۹۹۱ منتشر شد. زبان های برنامه نویسی اغلب برای انجام یک کار خاص طراحی می شوند و استفاده از آنها را محدود می کنند. با این حال، پایتون یک زبان چند پارادایم است. به عنوان یک زبان برنامه نویسی شی گرا عالی است، اما اگر ترجیح می دهید برنامه ها را به سبک برنامه نویسی تابعی بنویسید، می‌توانید این کار را نیز انجام دهید. این انعطاف‌پذیری باعث شده است که پایتون به زبان اصلی برای بسیاری از زمینه‌ها و صنایع تبدیل شود. ممکن است پایتون یک زبان برنامه نویسی جدید نباشد اما رشد مداوم آن نشان داده است که آزمون زمان را پس داده است. محبوبیت مداوم پایتون بر اساس سه عامل مهم است:

### در دسترس بودن عمومی

پایتون تقریباً بر روی هر دستگاه و سیستم عاملی که امروزه مورد استفاده قرار می‌گیرد می‌تواند اجرا شود. این بدان معناست که می‌توانید برنامه‌ای را روی دستگاه ویندوز خود بنویسید، آن را با شخصی که از مک استفاده می‌کند به اشتراک بگذارید و همچنان انتظار داشته باشید که به درستی برای او اجرا شود.

### جامعه موجود

Python.org مکانی عالی برای شروع برای منابع، اسناد و عضویت در بنیاد نرم افزار پایتون است. ماموریت PSF پشتیبانی و توانمندسازی کاربران پایتون از طریق کمک‌های مالی است که کنفرانس‌ها، جلسات، گروه‌های کاربری و موارد دیگر را تامین می‌کند.

### در دسترس بودن کتابخانه‌ها برای کمک به شما در دستیابی به اهدافتان

طبق نظرسنجی توسعه‌دهندگان Stack Overflow در سال ۲۰۲۱، پایتون همواره مورد علاقه توسعه‌دهندگانی است که در حال حاضر از آن استفاده می‌کنند و رتبه اول را در میان زبان‌هایی دارد که توسعه‌دهندگان دیگر آرزو دارند از آن استفاده کنند.

پایتون یک زبان برنامه نویسی قدرتمند و منعطف است که برنامه نویس با انتخاب آن می‌تواند در حوزه‌های مختلف تکنولوژی فعالیت داشته باشد. اگر شما به عنوان یک برنامه نویس دنبال زبانی هستید که کارهای متنوعی انجام دهد، پایتون بهترین گزینه است. از زبان برنامه نویسی پایتون برای ساخت وب سایت، اپلیکیشن‌های موبایل، نرم افزارهای دسکتاپ و بازی‌های سبک استفاده می‌شود و در زمینه تست نفوذ و هک کاربرد زیادی دارد. با یادگیری پایتون می‌توانید در علوم مختلف هوش مصنوعی مثل یادگیری ماشین یا داده کاوی نیز فعالیت کنید و یک برنامه نویس همه فن حریف شوید. این ویژگی پایتون را تبدیل به یکی از برترین زبان‌های برنامه نویسی کرده است.

**جاوا** یک زبان برنامه نویسی شی گرا و مبتنی بر کلاس است که توسط جیمز گاسلینگ در سال ۱۹۹۵ ایجاد شد. جاوا یکی از پرکاربردترین زبان‌های بک‌اند برای توسعه وب است و دارای کتابخانه‌های غنی است و جاوا می‌تواند حجم عظیمی از داده‌ها را مدیریت کند و از امنیت بالایی برخوردار است.

در کنار ++C، جاوا یکی از بهترین انتخاب‌ها برای توسعه نرم افزار است. از آنجایی که از نظر طراحی به وابستگی‌های زیادی نیاز ندارد، در بین تمام زبان‌های برنامه نویسی، تا حد زیادی سازگارترین بین پلتفرم‌ها است. به این ترتیب، یادگیری نحوه کدنویسی با جاوا به این معنی است که زمان زیادی را صرف نگرانی در مورد اینکه برنامه شما با چه سیستم‌هایی کار می‌کند، نخواهید داشت. به عنوان یکی از اولین زبان‌های برنامه نویسی شی گرا نیز، تمام نکات مثبت این سیستم مدل‌سازی داده را در بر می‌گیرد، به این معنی که با یادگیری آن، به طور کلی، اگر بخواهید با برنامه نویسی کامپیوتری OOP آشنا شوید، درک خوبی خواهید داشت.

با این حال، علاوه بر نرم‌افزار، اگر به دنبال توسعه برای موبایل هستید، یادگیری کدنویسی با جاوا قطعاً کمک بزرگی است به خصوص اگر هدف‌تان اندروید باشد. در واقع، رابط کاربری سیستم عامل اندروید با جاوا کدگذاری شده است، بنابراین اگر آن مسیر را دنبال کنید، می‌توانید انتظار سطوح بالایی از سازگاری را برای توسعه برنامه‌های موبایل داشته باشید.

جاوا به عنوان یکی از زبان‌های برنامه نویسی قدیمی در لیست، کاملاً پایدار در صنعت توسعه است. در کنار پایتون، توسعه دهندگان جاوا بالاترین حقوق را دریافت می‌کنند، اگرچه تخصص قطعاً می‌تواند پتانسیل حقوق شما را افزایش دهد. با دانش جاوا، می‌توانید نرم‌افزار توسعه دهید، توسعه‌دهنده اندروید شوید یا بیشتر، بنابراین یادگیری این زبان برنامه نویسی فرصت‌های خوبی را به شما ارائه می‌دهد.

**سی شارپ** یک زبان برنامه نویسی شی گرا است که توسط مایکروسافت در سال ۲۰۰۰ برای کار با چارچوب دات نت ایجاد شد. این برنامه نه تنها به گونه ای طراحی شده است که بهترین ویژگی‌های مدرن را از زبان‌های برنامه نویسی دیگر مانند ++C و جاوا در خود جای دهد، بلکه با توجه به کسب و کارها ساخته شده است تا برنامه نویسان بتوانند با استفاده از آن نرم‌افزارهای مختلفی را ایجاد کنند. با توجه به این ابتکار، این زبان یک زبان همه منظوره در نظر گرفته می‌شود و در حالی که یک زبان سطح بالا است، نسبت به سایر زبان‌های برنامه نویسی مانند پایتون به زبان ماشین نزدیک‌تر است.

سی شارپ به دو صورت عمده استفاده می‌شود. با توجه به سازگاری آن با چارچوب دات نت، بسیاری از توسعه دهندگان آن را برای توسعه برنامه های وب و دسکتاپ خود انتخاب می‌کنند. در واقع، برای توسعه دهندگان ویندوز، این زبان اغلب اولین انتخاب برای ایجاد برنامه‌های ویندوز است. با این حال، سی شارپ نیز به عنوان زبان پیش فرض موتور Unity نقش مهمی در صنعت بازی سازی بازی می‌کند. بسیاری از گزارش‌ها نشان می‌دهند که حداقل ۳۳ درصد از بازی‌ها با یونیتی توسعه داده می‌شوند که آن را به یک بازیکن بزرگ تبدیل می‌کند.



از آنجایی که این زبان برنامه نویسی به طور خاص توسط مایکروسافت توسعه داده شده است، شکی نیست که برای مدت طولانی یک زبان پایدار باقی خواهد ماند. همچنین می‌توانید انتظار توانایی‌های بین پلتفرمی شگفت‌انگیز را با آن داشته باشید، به این معنی که توسعه آن از نظر سازگاری به دلیل چارچوب دانت بسیار آسان است. این زبان در حالی که برخی از بهترین ویژگی‌های ++C مانند جمع‌آوری زباله را دارد، یادگیری آن برای مبتدیان بسیار آسان‌تر باشد. با این حال، این زبان همچنین بسیار مقیاس پذیر است، بنابراین شما به راحتی می‌توانید آن را با دامنه اکثر پروژه‌های #C تطبیق دهید. بنابراین، این زبانی است که یادگیری نحوه کدنویسی آن توانایی‌های طولانی مدت و قوی زیادی را به شما می‌دهد.

اگر به طور خاص به توسعه بازی با یونیتی علاقه دارید، یادگیری کدنویسی #C قطعاً ضروری است. با این حال، با کتابخانه‌های یونیتی که پیرامون زبان سی‌شارپ ایجاد کرده است، بسیاری از وظایف پیچیده به‌طور خودکار انجام می‌شوند و آن را به یک زبان برنامه‌نویسی ساده تبدیل می‌کنند که حتی برای پروژه‌های بازی‌های ویدیویی پیچیده‌تر استفاده می‌شود.

از نظر شغلی، یادگیری کدنویسی سی‌شارپ به شما این امکان را می‌دهد تا زمینه‌های متنوعی از فرصت‌ها را دنبال کنید. به عنوان مثال برای مشاغلی که به دنبال توسعه هر نوع نرم‌افزاری هستند، تقاضای بالایی دارد.

## C++

C++ یک زبان همه منظوره است که در ابتدا برای گسترش زبان برنامه نویسی C ایجاد شد. این زبان که توسط Bjarne Stroustrup توسعه داده شد، برای ترکیب کارایی و ویژگی‌های پیشرفته C (C یک زبان نسبتاً سطح پایین است) با برخی از جنبه‌های برنامه نویسی شی‌گرا مانند کلاس‌ها در نظر گرفته شد. به این ترتیب، این زبان اغلب جایی بین زبان‌های برنامه نویسی سطح پایین و سطح بالا در نظر گرفته می‌شود، زیرا از ویژگی‌های هر دو برای انجام وظایف خود استفاده می‌کند.

C++ یکی از پرکاربردترین زبان‌های برنامه نویسی در کل صنعت است. توسعه دهندگان می‌توانند به راحتی آن را برای ایجاد برنامه‌های دسکتاپ پیاده‌سازی کنند و آن را بر روی سرورهای وب و غیره تطبیق دهند. می‌توان گفت در بین زبان‌های محبوب این زبان برنامه نویسی بسیار پرکاربرد است به همین دلیل منابع یادگیری آن نیز زیاد است و پشتیبانی جامعه بالایی دارد. با یادگیری این زبان برنامه نویسی می‌توانید پروژه‌های مختلفی بسازید و یک توسعه دهنده حرفه‌ای شوید حال این توسعه می‌تواند توسعه یک نرم افزار، توسعه وب یا بازی باشد. به هر حال بازار کار بسیار خوبی در انتظارتان خواهد بود.

## سخن پایانی

وقتی صحبت از یادگیری زبان‌های برنامه نویسی محبوب می‌شود، انتخاب درست یا غلط واقعی وجود ندارد. در عوض، زبان برنامه نویسی “درست” برای یادگیری نحوه کدنویسی، بهترین زبان برای پروژه‌ها، اهداف و سلیقه شخصی شماست. خیلی مهم نیست که چه زبانی را برای یادگیری کدنویسی انتخاب می‌کنید، اما دنیای

فوق العاده ای از فرصت ها در انتظار شماست. با دانش هر زبان برنامه نویسی، می توانید نرم افزار، بازی و برنامه های موبایلی بسازید که دنیا را به سهم خود تغییر دهید.