



مقایسه کردن جاوا و پایتون به معنای روبه‌روی هم قرار دادن و رقابت بین آن‌ها نیست. این دو زبان برنامه‌نویسی کاربردهای متفاوت و طرفداران خاص خود را دارند که بهتر است به عنوان دو نمونه از معروف‌ترین زبان‌های برنامه‌نویسی موجود، نگاه دقیق‌تری به شباهت‌ها، تفاوت‌ها، مزایا، معایب و موارد کاربرد هر یک داشته باشید.

صرف‌نظر از اشتباه‌های رایج که این دو زبان را کاملا شبیه هم جلوه داده، مهم است بدانید که **جاوا** و **پایتون** به همان اندازه که به یکدیگر شبیه هستند، تفاوت‌های کلیدی نیز باهم دارند. هر دو زبان‌های برنامه‌نویسی قدرتمندی هستند که توسط طیف گسترده‌ای از متخصصان و توسعه‌دهندگان پشتیبانی می‌شوند و مجموعه بزرگی از کتابخانه‌های مختلف برای آن‌ها آماده‌سازی شده است. اگر قصد انجام کاری را داشته باشید که توسط خود این زبان‌ها قابل انجام نباشد، این شانس را خواهید داشت که به کمک یکی از این کتابخانه‌ها پروژه خود را پیش ببرید. در واقع، هیچ‌کس به‌طور کامل از تمام کتابخانه‌های پشتیبانی شده توسط هر کدام از این زبان‌های برنامه‌نویسی آگاه نیست، زیرا هیچ‌گاه کسی همه آن‌ها را باهم امتحان نمی‌کند. به یقین بعضی از کتابخانه‌ها از بقیه معروف‌تر هستند و برخی از آن‌ها تقریباً در هر اپلیکیشنی استفاده می‌شوند. اما **جاوا** و **پایتون** در بعضی از جنبه‌ها تفاوت‌های زیادی نیز با هم دارند. برخی از این تفاوت‌ها به حدی ملموس هستند که هیچ نیازی به بحث در مورد آن‌ها نیست، اما بعضی دیگر به شرایط و اولویت‌های استفاده و محیطی که برای آن برنامه‌نویسی می‌کنید، مربوط می‌شود.

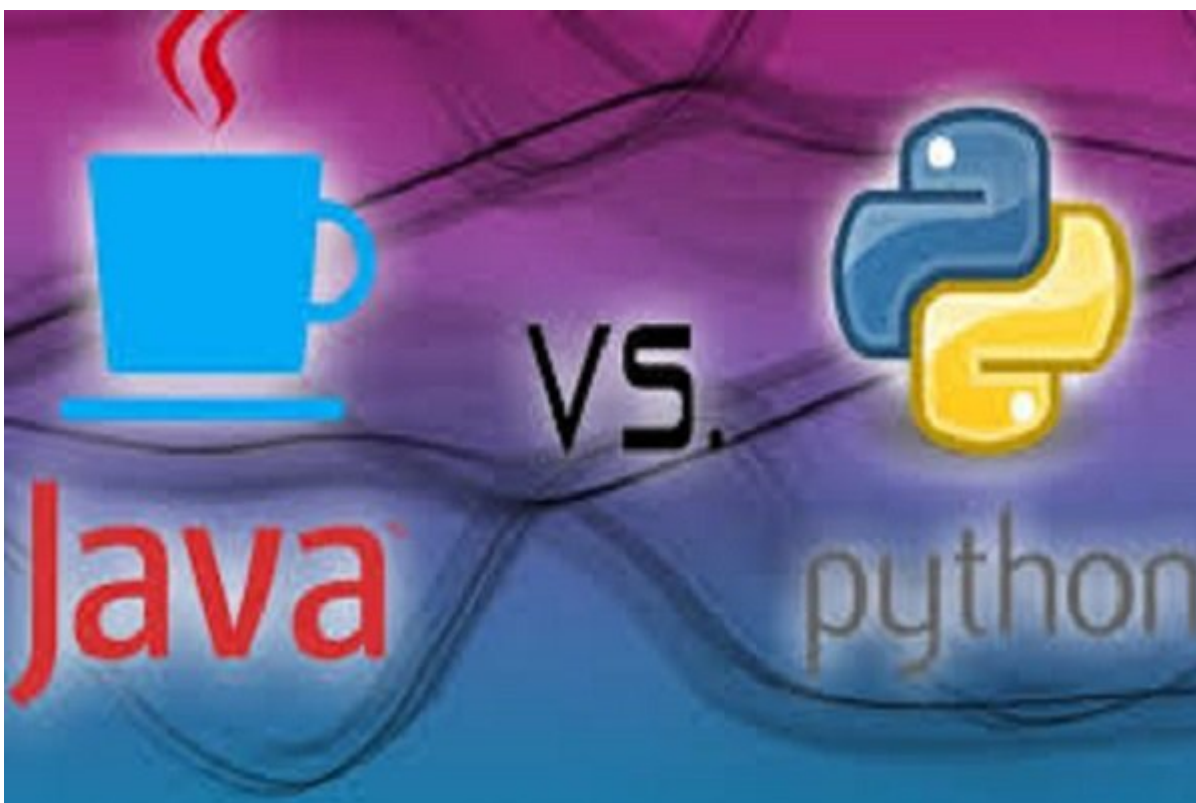
برای مثال، **جاوا** یک زبان ترجمه شده و **پایتون** یک زبان تفسیری است. این تفاوت باعث می‌شود تا هر کدام از این زبان‌ها مزایا و معایب خاص خود را داشته باشند. حتی در مورد این موضوع که کدهای ترجمه شده سریع‌تر اجرا می‌شوند یا کدهای تفسیری تاکنون به نظر واحدی نرسیده‌اند و موضوعی پیچیده است. سریع‌تر بودن یک زبان برنامه‌نویسی نسبت به زبان دیگر به عوامل دیگری در محیطی که در آن کار می‌کنید، بستگی دارد. برای مثال، **پایتون** با اجرا در حالت عمدتاً موازی با پردازنده‌های گرافیکی سازگارتر است. این دو زبان کاملاً متفاوت نوشته می‌شوند. اگر در **جاوا** یک ساختار ایجاد کنید و آن را در براکت قرار می‌دهید و **پایتون** برای همین مقصود از تورفتگی استفاده می‌کند. FreeCodeCamp کدهای **پایتون** را «مرتب، قابل خواندن و به خوبی ساختار یافته» توصیف می‌کند که تورفتگی‌های مناسب در آن نه فقط برای زیبایی بلکه مشخص‌کننده وضعیت اجرای کد در این زبان است. این تفاوت‌های ساختاری بر نوع نگاه برنامه‌نویسان به این زبان‌ها و سرعت تایپ آن‌ها تأثیر دارد و از لحاظ نظری این تفاوت‌ها بر سطح مورد نیاز برای یادگیری یک زبان تأثیرگذار است.

برنامه‌نویسان از جنبه‌های بسیاری در مورد برتری‌های این دو زبان بحث می‌کنند. برای مثال، بعضی از افراد مدعی‌اند در **پایتون** یک توسعه‌دهنده می‌تواند پنج تا ده برابر راندمان بالاتری داشته باشد، زیرا **پایتون** از تایپ پویا پشتیبانی می‌کند و عده‌ای دیگر نیز دیدگاه متفاوتی در مورد مسئله بهره‌وری دارند. بسیاری از این استدلال‌ها از رویکردهای متفاوتی برای اثبات عقیده خود استفاده می‌کنند که خیلی هم سودمند نخواهد بود. این‌که چه زبانی می‌تواند بهره‌وری بالاتری داشته باشد، معمولاً به میزان تجربه توسعه‌دهنده در کار با آن زبان، شیوه کدنویسی و احتیاج‌های اپلیکیشن مورد نظر بستگی دارد. توجه داشته باشید ابزارهای زبان در نهایت هر چیزی را که برنامه‌نویس می‌نویسد، به کد

قابل فهم توسط ماشین تبدیل می‌کند. بنابراین، انتخاب یک زبان به اندازه‌ای که برای ماشین قابل قبول باشد، برای ما نیست.

گرایش‌ها به سمت جاوا و پایتون

جاوا همچنان یکی از محبوب‌ترین زبان‌های برنامه‌نویسی از تمام جهات است. از سویی رشد و پیشرفت **پایتون** نیز به ویژه در کشورهای توسعه یافته چشمگیر بوده و درآمد بالایی را نصیب توسعه‌دهندگان این زبان برنامه‌نویسی کرده است. طبق اعلام برخی از منابع، در نهایت معروفیت و محبوبیت پایتون از **جاوا** پیشی خواهد گرفت. از دلایل این رشد فوق‌العاده می‌توان به بهره‌وری توسعه‌دهندگان، انعطاف‌پذیری زبان، پشتیبانی کتابخانه، پشتیبانی جامعه و سهولت یادگیری اشاره کرد. همچنین **پایتون** در حوزه‌های مطرح و داغ فناوری مثل علم داده و هوش مصنوعی و اپلیکیشن‌های وب و دسکتاپ و سرورهای شبکه و ابزار رسانه به میزان زیادی مورد استفاده قرار گرفته است. از سوی دیگر، محبوبیت جاوا به دلیل کم‌رنگ شدن وجهه امنیتی آن در حال افول است. به طوری که بعضی از سایت‌های خبری حوزه فناوری مانند The Register مدعی‌اند: 88 درصد اپلیکیشن‌های **جاوا** با مشکلات امنیتی دست به گریبان هستند. البته نباید فراموش کنیم که در بدنام شدن **جاوا** مرورگر پلاگین بی‌تأثیر نبوده است، وگرنه مشکلات امنیتی خود این زبان خیلی بدتر از سایر زبان‌های برنامه‌نویسی نیست. سوابق امنیتی **پایتون** از همه نظر فراتر از سطح قابل قبول بوده است. پشتیبانی منظم و سهولت استفاده از پایتون باعث می‌شود تا توسعه‌دهندگان کم تجربه‌تر بتوانند امنیت کدهای خود را راحت‌تر فراهم کنند. اما غیرعقلانی است که به **جاوا** به‌عنوان یک زبان تاریخ مصرف گذشته نگاه کنیم. توسعه‌دهندگان جاوا با اضافه کردن قابلیت‌های جدید به آن باعث می‌شوند استفاده از آن به ویژه در مقیاس‌های بزرگ توسعه سبک‌تر، سریع‌تر و انعطاف‌پذیرتر شود. ماشین مجازی قدرتمند جاوا (JVM) امکان ساخت اپلیکیشن‌های **جاوا** سازگار با پلتفرم‌های مختلف را ساده‌تر می‌کند و ساخت اپلیکیشن‌های بزرگ سنتی با جاوا بهتر انجام می‌شود. **جاوا** و **پایتون** از مدل‌های ریسمان متفاوتی استفاده می‌کنند. Python GIL یا Global Interpreter Lock به این معنا است که برخلاف **جاوا**، **پایتون** به طور موثرتری می‌تواند روی یک هسته از پردازنده مرکزی اجرا شود. به عبارت دیگر، استفاده از یک پردازنده گرافیکی با **پایتون** در مقایسه با انجام همان کار با جاوا ساده‌تر انجام می‌شود. بنابراین اجرای یک اپلیکیشن **پایتون** در حالت موازی روی یک پردازنده گرافیکی 5120 هسته بسیار بهینه‌تر از اجرای یک اپلیکیشن جاوا روی یک پردازنده مرکزی 8 هسته‌ای است. در گزارش اخیر منتشر شده توسط freeCodeCamp اعلام شده: «برنامه‌نویسان **پایتون** نسبت به توسعه‌دهندگان **جاوا** درآمد بیشتری دارند: 116 هزار دلار در مقابل 102 هزار دلار. البته شاید این آمار درآمدی به دلیل اشباع شدن بازار توسط تعداد بیشتری توسعه دهنده و برنامه‌نویس جاوا باشد.»



معصل سواد برنامه‌نویسی

منحنی یادگیری یک زبان برنامه‌نویسی به میزان زیادی به سطح دانش فعلی شما، زبانی که قصد استفاده از آن را دارید و محیط یادگیری‌تان بستگی دارد. برای مثال، اگر از قبل برنامه‌نویسی با ++C, C یا جاوا اسکریپت را بلد باشید، یادگیری جاوا به نسبت ساده‌تر خواهد بود، به ویژه، اگر قصد داشته باشید از جاوا برای نوشتن کد اپلیکیشن استفاده کنید. اما اگر قبلاً هرگز برنامه‌نویسی نکرده‌اید و قصد دارید برای یک جامعه علمی برنامه بسازید، پایتون می‌تواند بهترین انتخاب برای شما باشد. در حال حاضر، پایتون به عنوان زبان منتخب برای آموزش در مدارس در نظر گرفته شده، زیرا به افراد کمک می‌کند تا فنون کدنویسی درست را به سرعت یاد بگیرند. معصلی که بسیاری از برنامه‌نویسان به آن توجه نمی‌کنند.

یکی دیگر از تفاوت‌های مهم بین این دو زبان برنامه‌نویسی ارائه کد است. در گذشته توسعه دهندگان به میزان زیادی از کدهای تکراری برای ساخت اپلیکیشن‌ها استفاده می‌کردند. این کدها توسط سایر توسعه‌دهندگان و ماشین‌ها خوانده می‌شد و برای مدت زمان طولانی کدهای نوشتاری موجز و مختصر جاوا رایج بود. اما امروزه، مردم با دیدگاه‌ها و مهارت‌های متفاوت، از زبان‌های برنامه‌نویسی برای اهداف گوناگونی (نه لزوماً توسعه اپلیکیشن) استفاده می‌کنند و شما باید نیاز و مهارت‌های آن‌ها را در نظر بگیرید. شاید پایتون در اینجا برتری داشته باشد، زیرا به طور کامل از اصول برنامه‌نویسی درست پشتیبانی می‌کند.

مقایسه وضعیت عملکرد

بعضی از توسعه‌دهندگان معتقدند: «تفسیر کردن» همیشه به معنای «کند شدن» است. از طرفی کاربران جاوا هم در جواب می‌گویند: «کد خود را به زبان پایتون یا هر چیز دیگری که می‌خواهید بنویسید، اما هر زمان که اپلیکیشن شما به گسترش نیاز داشت، مجبورید دوباره آن را در جاوا بنویسید.»

مشکل مقایسه سرعت یک زبان با زبان دیگر اغلب به محیط آن محدود می‌شود. هر چند شما می‌توانید در یک محیط آزمایشگاهی شرایط را برای مقایسه دو زبان فراهم کنید، اما نتیجه کار زمانی ارزش خواهد داشت که در عمل به دست می‌آید. علاوه بر این باید مسائل دیگری را مثل کتابخانه‌های استفاده شده و سبک کدنویسی به کار رفته در ساخت اپلیکیشن (لااقل برای پایتون) در نظر داشته باشید.

تاثیر وضعیت عملکرد نسخه‌های مختلف یک زبان موضوع مهمی است که باید در نظر داشته باشید. در دنیای واقعی، همیشه آخرین نسخه‌ها مورد مقایسه قرار نمی‌گیرند. در جاوا اغلب اپلیکیشن‌ها 2-3 نسخه عقب‌تر از نسخه آخر هستند. در پایتون معمولاً Python 3.x سریع‌تر از x.2 اجرا می‌شود، اما تنها با کتابخانه‌هایی که برای Python 3.x بهینه سازی شده‌اند. با وجود این، بعضی از برنامه‌نویسان علم داده (و سایرین) هنوز ترجیح می‌دهند از Python 2.x استفاده کنند تا بتوانند از مزایای برخی از کتابخانه‌های خاص بهره‌مند شوند.

عملکرد، یک معیار بسیار پیچیده است که نمی‌توان آن را با دقت اندازه‌گیری کرد. این که یک زبان تا چه اندازه می‌تواند سریع اجرا شود، به عواملی مانند محیط اجرا، نحوه نوشتن کد و کتابخانه‌ها و افزونه‌های استفاده شده در آن بستگی دارد.

در پایان، باید در نظر داشته باشید که تنها می‌توان کلیاتی را در مورد مقایسه بین زبان‌های برنامه‌نویسی ارائه کرد. به طور خلاصه می‌توان گفت معمولاً زبان پایتون بیشتر برای علوم داده، هوش مصنوعی و وظایف یادگیری ماشین استفاده می‌شود و زبان جاوا بیشتر برای نوشتن اپلیکیشن‌های کاربر نهایی، به ویژه اپلیکیشن‌های الصاقی و سازگار با تمام پلتفرم‌ها استفاده می‌شود. جاوا معمولاً در صدر فهرست زبان‌های مورد استفاده توسعه‌دهندگان حرفه‌ای قرار دارد و یک انتخاب شایسته برای ساخت کدهای سمت سرور است.

تاریخ انتشار:

نشانی منبع:

<https://www.shabakeh-mag.com/workshop/15092/%D8%AC%D9%80%D9%80%D8%A7%D9%88%D8%A7-%D8%AF%D8%B1-%D9%85%D9%82%D8%A7%D8%A8%D9%84-%D9%BE%D8%A7%DB%8C%D8%AA%D9%88%D9%86-%DA%A9%D8%AF%D8%A7%D9%85%E2%80%8C%DB%8C%DA%A9-%D8%B2%D8%A8%D8%A7%D9%86-%D8%A8%D8%B1%D9%86%D8%A7%D9%85%D9%87%E2%80%8C%D9%86%D9%88%DB%8C%D8%B3%DB%8C-%D8%A8%D8%B1%D8%AA%D8%B1%DB%8C-%D8%A7%D8%B3%D8%AA%D8%9F>