

## مقدمه

زمانی که در بهار سال ۲۰۱۰ برای اولین بار از JetBrains دیدار کردم، تقریباً مطمئن بودم که دنیا به یک زبان برنامه‌نویسی دیگری نیاز ندارد. فکر می‌کردم زبان‌های موجود JVM<sup>۱</sup> کافی هستند، و چه کسی با عقل سلیم یک زبان جدید می‌سازد؟ پس از حدود یک ساعت بحث در مورد مشکلات تولید در پایگاه‌های کد<sup>۲</sup> در مقیاس بزرگ، من قانع شدم که اینطور نیست و اولین ایده‌هایی که بعداً بخشی از کاتالین شدند، ابتدا روی تخته سفید طراحی شدند. من کمی بعد به JetBrains پیوستم تا طراحی زبان را رهبری کنم و روی کامپایلر کار کنم. امروز، بیش از شش سال بعد، ما به انتشار دومین نسخه‌مان نزدیک می‌شویم. بیش از ۳۰ نفر در تیم هستند و هزاران کاربر فعال داریم، و هنوز ایده‌های طراحی هیجان‌انگیزی داریم که من به راحتی نمی‌توانم با آن‌ها کنار بیایم. اما نگران نباشید، این ایده‌ها باید از یک بررسی دقیق عبور کنند قبل از اینکه وارد زبان شوند. ما می‌خواهیم کاتالین در آینده همچنان در یک کتاب با اندازه منطقی جای گیرد. یادگیری یک زبان برنامه‌نویسی کار هیجان‌انگیز و اغلب بسیار پاداش بخشی است. اگر اولین زبان شما باشد، شما از طریق آن دنیای جدید برنامه‌نویسی را یاد می‌گیرید. اگر اینطور نباشد، باعث می‌شود در مورد چیزهای آشنا با دیدگاه جدید فکر کنید و بنابراین آن‌ها را عمیق‌تر و در سطح بالاتری درک کنید. این کتاب عمدتاً برای خوانندگانی است که با جاوا آشنا هستند. طراحی یک زبان از ابتدا ممکن است به خودی خود یک کار چالش‌برانگیز باشد، اما باعث شدن اینکه آن به خوبی با زبان دیگری کار کند،

---

۱. منظور از زبان JVM، زبان برنامه‌نویسی است که برای اجرا روی ماشین مجازی جاوا (JVM) طراحی شده است. یک موتور است که محیط اجرایی را برای اجرای برنامه‌های جاوا فراهم می‌کند، اما همچنین از دیگر زبان‌های برنامه‌نویسی که به بایت‌کد جاوا ترجمه می‌شوند نیز پشتیبانی می‌کند.

۲. پایگاه کد به مجموعه‌ای از فایل‌های سورس کد اشاره دارد که برای ساخت یک سیستم نرم‌افزاری خاص، برنامه، یا جزء نرم‌افزاری استفاده می‌شوند. معمولاً شامل تمام کد نوشته شده توسط انسان که برای توسعه نرم‌افزار لازم است و همچنین ممکن است شامل فایل‌های پیکربندی، اسکریپت‌ها، و سایر منابع مورد نیاز برای عملکرد صحیح نرم‌افزار باشد.

داستان دیگریست. تعامل‌پذیری جاوا (یعنی اینکه جاوا و کاتلین چگونه می‌توانند با هم ترکیب شوند و یکدیگر را فراخوانی کنند) یکی از اصول اساسی کاتلین بود، و این کتاب توجه زیادی به آن دارد. تعامل‌پذیری برای معرفی تدریجی کاتلین به یک پایگاه کد جاوا موجود، بسیار مهم است. حتی هنگام نوشتن یک پروژه جدید از ابتدا، باید زبان را در تصویر بزرگتر پلتفرم با تمام کتابخانه‌های نوشته شده به جاوا جای دهید. در حالی که این متن را می‌نویسم، دو پلتفرم جدید در حال توسعه هستند: کاتلین اکنون روی ماشین‌های مجازی جاوا اسکریپت اجرا می‌شود که توسعه وب فول‌استک را ممکن می‌سازد، و به زودی قادر خواهد بود به طور مستقیم به کد بومی<sup>۳</sup> کامپایل شود و در صورت نیاز بدون هیچ ماشین مجازی اجرا شود. بنابراین، در حالی که این کتاب محوریت JVM دارد، بسیاری از چیزهایی که از آن یاد می‌گیرید می‌تواند به محیط‌های اجرایی دیگر نیز تطبیق داده شود. نویسندگان این کتاب از روزهای اولیه تشکیل تیم کاتلین، عضو بوده‌اند، بنابراین با زبان و جزئیات آن به طور نزدیک آشنا هستند. تجربه آن‌ها در ارائه‌های کنفرانس، کارگاه‌ها، و دوره‌های آموزشی درباره کاتلین به آن‌ها امکان داده است تا توضیحات خوبی ارائه دهند و پرسش‌های رایج و اشتباهات احتمالی را پیش‌بینی کنند. این کتاب مفاهیم سطح بالایی که پشت ویژگی‌های زبان اند را توضیح می‌دهد و همچنین تمام جزئیات لازم را نیز فراهم می‌کند. امیدوارم از وقت خود با زبان ما و این کتاب لذت ببرید. همانطور که اغلب در پست‌های کامیونیتی خودمان می‌گوییم: کاتلین خوبی داشته باشید!

- آندری برسلو، طراح اصلی کاتلین در JetBrains

---

۳. کدبومی (Native code) به کدی اطلاق می‌شود که به زبان ماشینی سیستم عامل یا پردازنده ترجمه شده است. این کد می‌تواند به صورت مستقیم و بدون نیاز به ماشین مجازی یا تفسیرگر اجرا شود. این کد به زبانی نوشته شده است که سیستم عامل یا پردازنده می‌تواند بفهمد و اجرا کند.