



انجمن علمی
مهندسی کامپیوتر



معاونت فرهنگی و اجتماعی
دانشگاه ولی عصر عجل رفسنجان

خبررسانه

زبان های برنامه نویسی

شماره دو آذر ۱۴۰۰

فهرست

صاحب امتیاز



انجمن مهندسی کامپیوتر

مدیر مسئول



صبا منصور صفائیان

سر دبیر



صبا منصور صفائیان

طراحی جلد



علی زینلی

ویراستار



مائده محبی

صفحه آرا



صبا منصور صفائیان

یادداشت سردبیر

C++، زبان چند رگه

C#، جدید و کارآمد

Python، همه فن حریف

Kotlin، جوان و حریف جاوا



یادداشت سرذیر



دانشجویان رشته مهندسی کامپیوتر، از مهم‌ترین افرادی در جامعه هستند که به پیشرفت و شکوفایی علم و هموار کردن راه در جهان برای ساده‌تر زیستن کمک می‌کنند. این افراد نه تنها با استفاده از دانش خود می‌توانند به افراد بی‌شماری یاری برسانند، بلکه با استفاده از برنامه‌هایشان، می‌توانند به کشف بیشتر جهان کمک کنند و حتی در شرایطی، جهان جدیدی را خلق کنند.

نشریه تورینگ، وظیفه خود می‌داند که در این راه و در پیدا کردن هدف، به دانشجویان مهندسی کامپیوتر کمک کند و آن‌ها را بیش از پیش به سوی موفقیت هدایت کند. تیم نشریه تورینگ معتقد است که برنامه‌نویسان آینده، مهندسان کامپیوتر آتی و طراحان سیستم، در آینده می‌توانند جهان را بیشتر از قبل به سوی روشنایی و صلح هدایت کنند.

در همین راستا در این شماره از نشریه تورینگ، می‌خواهیم شما را با زبان‌های برنامه‌نویسی آشنا کنیم تا بتوانید برای آینده و راهی که می‌خواهید در آن قدم بردارید، اطلاعات بیشتری داشته باشید و بهترین و مناسب‌ترین راه را برای خود انتخاب کنید.

امید داریم که نشریه تورینگ می‌تواند به آینده، موفقیت شما و صلح و روشنایی در جهان، حتی در مقیاس‌های کوچک هم کمک کند.

از همه شما بابت انتخاب کردن نشریه تورینگ و حمایت‌هایتان تشکر می‌کنیم.

سردبیر نشریه تورینگ

صبا منصور صفائی‌ان

زبان چندرگه



نویسنده: رامین آقائی آغچه کهل

شناخت با بخش‌های معروف کد دستورات C++

با اینکه به احتمال خیلی زیاد با دستورات اصلی، آشنا هستید؛ ولی به جهت یادآوری مفاهیم پایه، برخی از آن‌ها مورد بررسی قرار می‌دهیم:

```
#include <iostream>
using namespace std;

int main()
{

    return 0;
}
```

#include <iostream>

با استفاده از کتابخانه‌ای از فایل هدر به نام «iostream» می‌توان کدهایی که نوشته شده را با کامپایلر با استفاده از رسانه‌ها و یا نمایشگر، نشان داد.

using namespace std;

این کد را تقریباً همه برنامه‌نویسان خبره، استفاده می‌کنند؛ زیرا به کمک آن از فضای استاندارد نامی به اسم std، در برنامه استفاده می‌کنیم که رجوع به آن را برای ما آسان می‌کند؛ یعنی لازم نیست در کدها «std:: cout» را تایپ کنیم.

زبان برنامه‌نویسی، یک راه ارتباطی بین انسان و ماشین است. یکی از زبان‌های محبوب برنامه‌نویسی، C++ است؛ که دروازه ورود به دنیای برنامه‌نویسی است. این زبان در آزمایشگاه بل اختراع شده و در اکثر دانشگاه‌ها برای دانشجویان تازه‌کار که در رشته مهندسی کامپیوتر تحصیل می‌کنند، آموخته می‌شود و دانشگاه ولی عصر (عج) هم از این قاعده، مستثنی نیست!

اهمیت زبان C++

زبان C++، به این دلیل دارای اهمیت است که ساختار و دستورات آن به نوعی معیار، تبدیل شده‌اند و در زبان‌هایی مانند جاوا و C#، مورد استفاده قرار گرفته‌اند. جالب است بدانید، نام C++ از روی یک ویژگی (عملگر افزایشی ++ که به متغیر یک واحد اضافه می‌گردد) در C که ریک ماشیتتی Rick Mascitti، پیشنهاد دهنده این نام بود، آورده شده است. در کل C++ به مفهوم کمی بیشتر از C است. با توجه به اینکه زبان C++، نسخه ارتقا یافته زبان C است و بخش C در آن یک زبان سطح پایین (زبانی که به زبان ماشین بیشتر شباهت دارد) است و اجرای آن سریع‌تر است.



```
int main() {
```

تابع اصلی برنامه است که در داخل آن بدنه دستورات و کدها نوشته میشوند و در آخر آنها ”؛“ تایپ میگردد؛ یا با استفاده از مفاهیم فراخوانی تابع های دیگر و یا از تری و... دستورات نهایی از این تابع، قابل اجرا هستند.

```
return 0;
```

در خط آخر برنامه، مورد استفاده قرار می گیرد و صفر هم نشانگر خروج از برنامه را بر می گرداند.

● در روش برنامه نویسی شی گرای، مایه ای با رفتارها و ویژگی هایی داریم که توابع خارج از آن شی، امکان ایجاد تغییر در داده های داخل شی را نخواهند داشت.

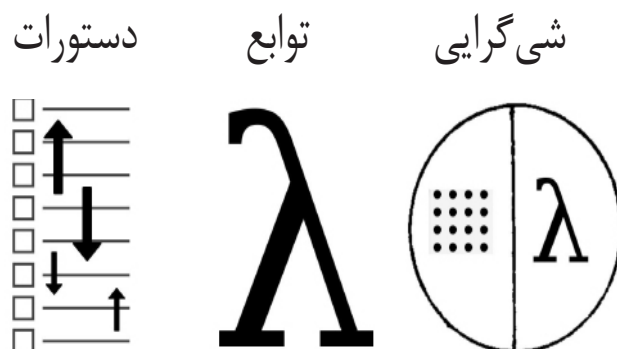
حال با اینکه ۲ پارادایم مثال زده شده، ممکن است به نوعی با هم در تضاد باشند؛ ولی برنامه نویسی C++ می تواند از هر کدام استفاده کند و یا از هر دو پارادایم بهره بگیرد؛ اما در جاوا و پایتون فقط باید از شی ها استفاده کنیم.

C++, زبان محبوب رباتیک



در مورد استفاده از زبان C++، میان افرادی که با زبان پایتون، برنامه نویسی می کنند؛ کمی اختلاف نظر وجود دارد. برنامه نویسان پایتون، همواره آسان بودن برنامه نویسی رباتیک را با زبان پایتون، مطرح می کنند، ولی کسانی که با C++ کار می کنند، نظر دیگری دارند. با اینکه برنامه نویسی با C++ به اندازه پایتون آسان نیست ولی چون رباتیک به عملکردی بدون درنگ، نیاز دارد و قبلا هم گفتیم، C++ بسیار به زبان سطح پایین شباهت دارد و اجرا و سرعت بهتری برای اجرا دارد؛ می توان نتیجه گرفت که این زبان به همراه کتابخانه های سخت افزاری، ^{Hardware Libraries} می تواند یک زبان استاندارد برای برنامه نویسی رباتیک باشد.

C++, یک زبان چند پارادایم



پارادایم به مفهوم راه های دسته بندی زبان های برنامه نویسی براساس ویژگی هایشان است که برای مثال شی گرایی (OOP)، روش رویه ای، ساخت یافته و... هر کدام یک پارادایم هستند

جالب است بدانید، C++ چند پارادایم برنامه نویسی را پشتیبانی می نماید، مثال هایی از هر پارادایم:

● در روش رویه ای یا همان پروسه ای، هر پروسه می تواند در هر نقطه ای در طول اجرای برنامه فراخوانده شود، شامل فراخوانی پروسه توسط خودش (برنامه نویسی بازگشتی)

برنامه‌هایی که با C++ نوشته شده‌اند





جدید و کارآمد



نویسنده: فوزیه السادات حسینی

همچنین فریمورک‌های ASP.NET MVC و ASP.NET Core، دو تکنولوژی برای طراحی و ایجاد برنامه‌های تحت وب مدرن هستند که امروزه در دنیای برنامه‌نویسی بسیار پرکاربرد و پرآوازه می‌باشند؛ بنابراین اگر تسلط خوبی به زبان برنامه‌نویسی سی‌شارپ داشته باشید، شما آمادگی کافی برای شروع یادگیری فریمورک قدرتمند ASP.NET Core را نیز خواهید داشت.

در این مقاله به معرفی زبان سی‌شارپ (C#) می‌پردازیم و شما را تا حد زیادی با این زبان برنامه‌نویسی و ویژگی‌هایش آشنا می‌کنیم. زبان برنامه‌نویسی سی‌شارپ، یک زبان برنامه‌نویسی Cross Platform است که به لطف NET Core، شما می‌توانید این زبان را روی همه سیستم عامل‌ها (Windows, Linux, Mac) اجرا کنید و به توسعه برنامه‌های خود بپردازید!

درباره C#

یکی از محبوبترین و پرکاربردترین زبان‌های برنامه‌نویسی، "سی‌شارپ" نام دارد و این زبان، جزو پنج زبان برنامه‌نویسی برتر در دنیا می‌باشد؛ که همچنین بازار کار بسیار خوبی در ایران دارد. از زبان برنامه‌نویسی سی‌شارپ می‌توان برای ساخت برنامه‌های تحت ویندوز (دسکتاپ)، برنامه‌های تحت وب، برنامه‌های موبایل و بازی‌ها استفاده کرد.

برای توسعه برنامه‌های ویندوز از طریق زبان سی‌شارپ، می‌توان از پلتفرم‌های WinForms و WPF، UWP استفاده کرد؛ حتی با استفاده از زبان سی‌شارپ و پلتفرم‌هایی مانند Xamarin و UWP می‌توان برای اندروید، iOS و ویندوزفون‌ها؛ برنامه ایجاد کرد.



تاریخچه C#

در سال ۱۹۹۹، شرکت Sun اجازه استفاده از زبان برنامه‌نویسی جاوا را در اختیار شرکت مایکروسافت قرار داد تا در سیستم عامل خود از آن استفاده کند. جاوا در اصل به هیچ پلتفرم یا سیستم عاملی وابسته نبود؛ ولی مایکروسافت برخی از مفاد قرارداد را زیر پا گذاشت و قابلیت مستقل از سیستم عامل بودن جاوا را از آن برداشت. شرکت سان، پرونده‌هایی علیه مایکروسافت درست کرد و مایکروسافت مجبور شد تا زبان شی‌گرای جدیدی با کامپایل جدید که به C++ شبیه بود را ایجاد کند. در سال ۱۹۹۹، آندرس هلزبرگ، گروهی را برای طراحی زبان جدیدی تشکیل داد که در آن زمان نامش Cool و شبیه زبان C بود اما با خواص شی‌گرایی. مایکروسافت در نظر داشت اسم این زبان را برای همیشه Cool قرار دهد؛ ولی به دلیل مناسب نبودن برای اهداف تجاری، این کار را نکرد. در ارائه و معرفی رسمی چارچوب دات‌نت (.net)، در سال ۲۰۰۰، این زبان به C# تغییر نام یافت. مدیر و سرپرست طراحان سی‌شارپ در مایکروسافت، آندرس هلزبرگ بود که تجربه قبلی او در طراحی Framework و Borland C++، Delphi، Pascal، به آسانی در دستورات عمل‌های سی‌شارپ، قابل رویت است. یکی از نکات مثبتی که زبان سی‌شارپ دارد، شی‌گرایی آن است و برای درک بهتر موضوع شی‌گرایی بهتر است اول با مفهوم شی‌گرایی یا OOP آشنا شویم.

برنامه‌نویسی شی‌گرا

(Object Oriented Programming)

برنامه‌نویسی شی‌گرا که به اختصار OOP نیز نامیده می‌شود، در حقیقت منطقی برگرفته از زندگی روزمره‌ی ما، انسان‌ها است. اگر به اطراف خود بنگرید، اشیای مختلفی را می‌بینید و به خاطر همین موضوع، انسان عمدتاً براساس اشیا تفکر می‌کند. در برنامه‌نویسی شی‌گرا نیز ما تلاش می‌کنیم تا مفاهیم برنامه‌نویسی را همانند اشیای دنیای واقعی، مدل کنیم؛ زیرا ذهن انسان، توانایی بالایی در مدلسازی اشیا دارد.



مزایای برنامه‌نویسی شی‌گرا چیست؟

- ۱- افزایش امنیت برنامه
- ۲- کاهش هزینه نگهداری
- ۳- قابلیت استفاده مجدد
- ۴- تحلیل ساده‌تر برنامه
- ۵- قابلیت سازماندهی بهینه‌تر کدها
- ۶- عدم نیاز به نوشتن کدهای تکراری و قابلیت‌هایی که قبلاً پیاده‌سازی شده‌اند و صرف‌جویی در استفاده از منابع
- ۷- قابلیت تقسیم برنامه به برنامه‌های کوچک‌تر اما مستقل



چرا C#؟

دلایل بسیاری برای محبوبیت سی شارپ وجود دارد که از جمله آنها می توان به موارد زیر اشاره کرد:

- آسان بودن یادگیری برنامه نویسی سی شارپ
- سی شارپ، جزء زبان های سطح بالا است و به زبان های برنامه نویسی چون سی، سی پلاس پلاس و جاوا نزدیک است؛ بنابراین یادگیری این زبان، آسان است و شما با داشتن یک منبع مناسب، می توانید به خوبی این زبان برنامه نویسی را یاد بگیرید.

- استفاده گسترده از زبان سی شارپ برای توسعه برنامه های دسکتاپ (ویندوز) و تحت وب
زبان سی شارپ به صورت گسترده برای تولید برنامه های دسکتاپ، تحت وب و Web service مورد استفاده قرار می گیرد؛ مایکروسافت با خرید شرکت Xamarin، قدم بزرگی در مسیر رشد هر چه بیشتر استفاده از زبان سی شارپ برداشت.

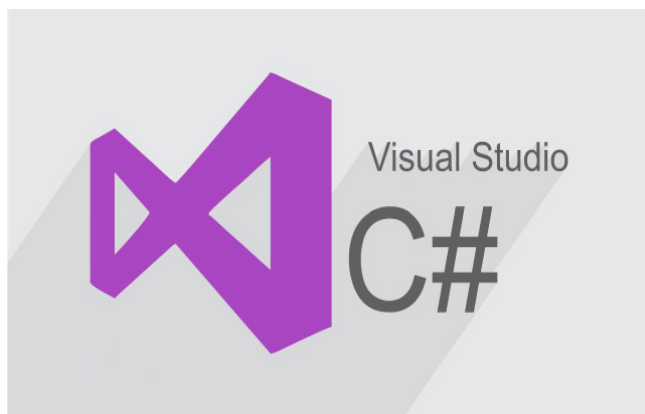
Xamarin، امکانی را فراهم کرده است که به سادگی می توان با استفاده از زبان سی شارپ، برای دستگاه های اندرویدی، آی او اس و اپلیکیشن موبایل، برنامه های مفیدی نوشت.

- زبان سی شارپ به صورت گسترده برای تولید بازی های مختلف، مورد استفاده قرار می گیرد.
- زبان سی شارپ دارای برنامه های تحت وب، یکپارچه ای است.

- کدنویسی آسان در نرم افزار ویژوال استودیو که قدرتمندترین IDE موجود برای برنامه نویسی است.

- زبان برنامه نویسی سی شارپ، دارای کتابخانه بزرگی است که می تواند عملکرد سطح بالاتری نسبت به دیگر زبان ها همچون جاوا و C++ ارائه دهد.

- دارای بازار کار عالی در ایران و خارج از ایران، با پشتیبانی و بروزرسانی مداوم توسط شرکت مایکروسافت است و زبان برنامه نویسی سی شارپ، کاملاً شی گرا است.



آینده و بازار کار C#

با توجه به پیشرفت روزافزون تکنولوژی و فضای مجازی و حرکت بیشتر دنیا به سوی اینترنت، قطعاً نیاز به برنامه نویس سی شارپ، روز به روز افزایش خواهد یافت و ایران نیز از این قضیه مستثنی نخواهد بود، زیرا شرکت های ایرانی زیادی هستند که در زمینه توسعه نرم افزارهای بزرگ و سازمانی، وبسایت ها، بازی های کامپیوتری و اپلیکیشن های موبایل فعالیت می کنند و در طراحی و توسعه برنامه ها و نرم افزارهای خود از زبان سی شارپ استفاده می کنند، به همین دلیل به جرات می توان گفت زبان برنامه نویسی سی شارپ، آینده خوب و مطمئنی خواهد داشت.

آر همه فن حریف

نویسنده: آتنا خالق

خیدو فان روسوم خالق پایتون



خیدو در نظر داشت توسعه‌ی اپلیکیشن، به سادگی و بدون نگرانی بابت سخت‌افزار، مدیریت حافظه و پیچیدگی‌هایی از این دست، امکان‌پذیر باشد؛ پس به این فکر افتاد زبان برنامه‌نویسی خودش را ابداع کند، ایده‌هایش را از ABC الهام بگیرد؛ اما زمان انجام پروژه را از سه سال به پروژه‌ای سه‌ماهه کاهش داد و اینگونه بود که پایتون متولد شد.

در سال‌های اولیه پس از منتشر شدن این زبان، پایتون به دلیل جهانی نبودن اینترنت، محبوبیت چندانی نداشت و قاعدتاً متقاعد کردن افراد برای استفاده از این زبان برنامه‌نویسی جدید، کار بسیار دشواری بود.

اگر شما هم به آموختن زبان‌های برنامه‌نویسی مختلف و کد زدن با آنها علاقه‌مندید، حتماً در ذهن خودتان با این سوال که “کدام زبان علاقه‌مندان بیشتری دارد و اصطلاحاً محبوب‌ترین زبان برنامه‌نویسی دنیاست؟” مواجه شده‌اید.

برای پاسخ به این سوال، ما بارها و بارها آخرین نتایج را چک کرده‌ایم، سایت‌های مختلف و معتبر ایرانی و خارجی را مورد بررسی قرار داده‌ایم و می‌توانیم به شما بگوییم که در چند سال اخیر، زبان “پایتون”، جزء زبان‌های شماره یک برنامه‌نویسی در جهان بوده است. زبانی که به دلیل داشتن کتابخانه‌های گسترده؛ فرآیندهای هوش مصنوعی، علم داده و یادگیری ماشین را تسهیل کرده و لقب پادشاه زبان‌های برنامه‌نویسی را از آن خود کرده است.

نام گذاری پایتون

شاید شما هم وقتی برای اولین بار، نام زبان پایتون را شنیده‌اید، حدس زده باشید که این زبان، نامش را از نوعی مار گرفته است؛ اما در واقع این زبان نامش خلاصه شده‌ای از نام برنامه تلویزیونی کمدی به اسم “سیرک پرنده مونتی پایتون” است که در سال ۱۹۷۰، در شبکه بی‌بی‌سی کشور بریتانیا، ساخته و پخش می‌شد. این برنامه یکی از برنامه‌های موردعلاقه خیدو فان روسوم (Guido Van Rossum) طراح و نویسنده زبان پایتون بوده است.



در زبان پایتون، انواع زبان‌های C و Java وجود دارد. با وجود اینکه پایتون یک زبان برنامه‌نویسی مستقل است؛ اما برای زبان‌های برنامه‌نویسی C و java نیز تغییراتی را ارائه می‌دهد. CPython، مشتقی از زبان C است تا به پایتون مزایای C را بدهد؛ این تنوع می‌تواند به عنوان مفسر و همچنین کامپایلر استفاده شود. JPython نیز پیاده‌سازی جاوا پایتون است که به سیستم این اجازه را می‌دهد تا برخی از مهم‌ترین ویژگی‌های جاوا مانند بهره‌وری، روی یک ماشین مجازی اجرا شود. برخلاف چیزی که در اوایل سال‌های انتشارش در مورد این زبان وجود داشت، همان‌طور که در اول مقاله گفتیم اکنون یکی از محبوب‌ترین زبان‌های برنامه‌نویسی حال حاضر دنیا است؛ به حدی که توجه والدین نسبت به اهمیت یادگیری این زبان برای فرزندان خود، جلب شده و در سال‌های گذشته درخواست آموزش این زبان در دبستان‌ها به جای زبان فرانسوی، به طور چشم‌گیری افزایش داشته است. با افزایش محبوبیت و بالا رفتن میزان استفاده از این زبان در اینترنت اشیاء، هوش مصنوعی، داده‌های علمی و... باید به اطلاع شما برسانیم که زبان پایتون در حال حاضر یکی از زبان‌های اصلی گوگل است. این زبان برنامه‌نویسی، یکی از معدود زبان‌هایی است که با این عنوان شناخته شده است. این زبان به دلیل کارایی بالای خود برای استفاده در شرکتی به بزرگی گوگل انتخاب شده و برای این شرکت بسیار سودمند بوده است؛ زیرا استفاده از آن حتی در پروژه‌های بزرگ نیز ساده است و حتی یوتوب نیز از برنامه‌نویسی پایتون پشتیبانی می‌کند.

در دهه ۱۳۹۰ میلادی، فان روسوم پس از سال‌ها تلاش برای معرفی و تبلیغ ABC مجبور به سفر و توزیع نوارهای مغناطیسی بین افراد بود. خیدو انتظار خاصی از پایتون نداشت چون تا آن زمان نتوانسته بود در بین برنامه‌نویسان جا باز کند؛ اما در سال ۱۹۹۵، شرکتی به نام Zope تأسیس شد که متخصص در زمینه‌ی تولید موتور آگهی‌نامه برای اینترنت بود. شرکت Zope صفحات وب داینامیکی ایجاد می‌کرد که با پایتون نوشته شده بودند، در واقع می‌توان این شرکت را یکی از دلایل عمده برای محبوبیت پایتون در روزهای اولیه نامید. Zope را تیمی از توسعه‌دهندگان پایتون هدایت می‌کنند و در سال ۲۰۰۰، فان روسوم هم به آن‌ها ملحق شد. حول و حوش همین وقت‌ها بود که فان روسوم لقب benevolent dictator for life "دیکتاتور خیرخواه جاویدان" را گرفت؛ چون او مبدا این زبان بود و مراحل توسعه آن را کنترل می‌کرد.

پایتون، زبانی برای همه چیز

پایتون در واقع، یک نوع زبان برنامه‌نویسی همه منظوره است. زبانی تفسیر شده، شی‌گرا و سطح بالا است که کارایی بسیار بالایی دارد و برای اولین بار در سال ۱۹۹۱، طراحی و منتشر شده است. پایتون به‌گونه‌ای طراحی شده است که برنامه‌نویس می‌تواند به صورت کاملاً واضح و بدون ابهام، پروژه‌های برنامه‌نویسی بزرگ و کوچک را ایجاد کند؛ همچنین، می‌توان از آن در طیف گسترده‌ای از سناریوهای کدنویسی مانند توسعه وب، هوش مصنوعی، یادگیری ماشین، تجزیه و تحلیل داده‌ها، و اینترنت اشیاء و بسیاری موارد دیگر استفاده کرد. پایتون زبان مفسری است؛ به این معنی که کد متنی، در قالب فایلی نوشته می‌شود و سپس مفسری، فایل را خط به خط می‌خواند و کد را به دستورعمل‌های قابل فهم برای کامپیوتر تبدیل می‌کند. درواقع، زبان پایتون بر اصل قابل فهم کردن برنامه‌نویسی برای همه پایه‌گذاری شد.



گوگل را می‌توان بی‌اغراق، برترین و قوی‌ترین موتور جستجوی جهان دانست و در عین حال باید بدانید که گوگل در قسمت‌های اصلی و پایه‌ای سیستم موتور جستجوی خود و همچنین بسیاری از برنامه‌هایش از زبان برنامه‌نویسی پایتون استفاده کرده است.



یکی از پلتفرم‌های محبوب در سراسر دنیا، اینستاگرام است. برنامه‌ای که هم به شما فرصت خوبی برای تبلیغ و ارائه کارتان را می‌دهد و اگر هم عاشق عکاسی از مکان‌ها و لحظات زندگی خودتان هستید و دوست دارید آن‌ها را به صورت آنلاین با دوستان خود به اشتراک بگذارید، باید از پایتون به خاطر کمک در توسعه بخش‌هایی از اینستاگرام، متشکر باشید.



اگر علاقه‌مند هستید که به صورت آنلاین به موسیقی گوش فرا دهید، حتماً اسپاتیفای را می‌شناسید؛ یک وبسایت قدرتمند در زمینه پخش موسیقی آنلاین که با استفاده از زبان برنامه‌نویسی پایتون، طراحی و پیاده‌سازی شده است.



یکی از دلایلی که می‌تواند شما را برای ساده بودن این زبان قانع کند، این است که در پایتون بلوک‌های کد در آکولاد نوشته نمی‌شوند، بلکه از فاصله یا space برای نوشتن کدها استفاده می‌شود. پایتون بیشتر از دیگر زبان‌های ماشینی، شبیه به زبان انسانی و زبان انگلیسی است و به شما این امکان را می‌دهد تا در زمان کمتری، کدنویسی کنید.

بزرگترین اپلیکیشن‌های نوشته شده با پایتون



دراپ‌باکس، ابتدا به عنوان یک برنامه پر قدرت مطرح شد و در حال حاضر توسط انواع مختلف کسب‌وکارها، شرکت‌ها، سازمان‌ها، افراد عادی و تیم‌های کوچک و بزرگ مورد استفاده قرار گرفته است. دراپ‌باکس به شما این امکان را می‌دهد تا فایل‌های خود را بر روی سیستم‌های کلود ذخیره‌سازی کنید و در هر زمان و مکان به آن‌ها دسترسی داشته باشید.



یکی از معروف‌ترین سایت‌های اینترنتی و یکی از پر قدرت‌ترین پروژه‌هایی است که با پایتون، نوشته شده است.



جوان و حریف جاوا



نویسنده: مریم حسنی

می‌دهد. از لحاظ سینتکس (قواعد و گرامر یک زبان برنامه‌نویسی)، زبان‌های برنامه‌نویسی کاتلین و جاوا، باهم سازگار نیستند؛ ولی برای ایجاد همکاری بین جاوا و همچنین قابلیت تبدیل به زبان جاوا، باعث شده که این زبان به وجود بیاید. کدی که در کاتلین نوشته شده است با جاوا به طور طبیعی کار می‌کند؛ درواقع کاتلین، خلاصه‌شده زبان جاوا است.



تاریخچه کاتلین

در جولای ۲۰۱۱، جت برینز، پس از گذشت یک سال که کاتلین در حال توسعه بود، از آن به عنوان زبان جدیدی که برای ماشین مجازی جاوا بود، پرده برداشت. سرپرست جت برینز، دیمیتری جمروف ^{Dmitry Jemerov}، اظهار داشت که یکی از اهداف اصلی کاتلین این است که با همان سرعت زیاد جاوا کامپایل شود؛ زیرا آن‌ها به دنبال ویژگی‌هایی بودند که زبان‌های آن دوره نداشتند. جالب است بدانید مدیر توسعه کاتلین، اندری برسلاو ^{Andrey Breslav}، بر این عقیده بود همان‌طور که نام جاوا، نام یک جزیره در اندونزی است.

اگر که به برنامه‌نویسی اندروید، علاقه‌مند هستید و اخبار این حوزه برنامه‌نویسی را دنبال می‌کنید، حتماً بارها و بارها در حوزه برنامه‌نویسی اندروید، به زبان برنامه‌نویسی کاتلین برخورد کرده‌اید و ممکن است برایتان سوال پیش آمده باشد، چه اتفاقی افتاد که جاوای پیر، جای خود را به کاتلین جوان داد؟

برای شروع مطلب، بهتر است به سوال **”کاتلین چیست؟“** پاسخ دهیم. همان‌طور که می‌دانید، برای ساخت هر نرم‌افزار و برنامه‌ای، باید از زبان برنامه‌نویسی استفاده کنیم؛ یکی از پرکاربردترین و محبوب‌ترین زبان‌های برنامه‌نویسی، «زبان برنامه‌نویسی کاتلین» است که در ابتدا برای ماشین مجازی جاوا (JVM) و اندروید طراحی شد که ویژگی‌های برنامه‌نویسی شی‌گرا و کاربردی را با هم ترکیب می‌کند. برای اولین بار کاتلین، توسط تیم برنامه‌نویسان جت برینز ^{Jet Brains}، در روسیه توسعه یافت.

کاتلین چند پلتفرمی است؛ یعنی این زبان برنامه‌نویسی در وب‌سرور، وب سمت مشتری (client) و اندروید کاربرد دارد. از زمانی که گوگل، کاتلین را به رسمیت شناخت، کاتلین به یکی از محبوب‌ترین زبان‌ها تبدیل شد و طبق نظرسنجی توسعه‌دهندگان

Stack Overflow، در سال ۲۰۲۰، کاتلین در رده چهارم از محبوب‌ترین زبان‌ها قرار گرفت؛ به همین دلیل گوگل تصمیم گرفت زبان ترجیحی خود برای برنامه‌نویسی اندروید را به کاتلین تغییر

لینوکس پشتیبانی می‌کند و همچنین توسط شرکت‌های بزرگ و قابل اعتماد مثل گوگل، آپاچی و... حمایت می‌شود.

جاوا یک زبان قدیمی است، با این حال شی‌گرا، متن‌باز و قدرتمند است. در قدرت و سرعت جاوا هیچ شکی نیست و اگر ما در حین مقایسه کاتلین و جاوا از قدرت کاتلین حرف می‌زنیم هیچگاه قدرت جاوا را نهی نمی‌کنیم. اگر بخواهیم در اندروید و تاریخچه‌اش کمی دقیق‌تر شویم، اندروید در سال ۲۰۰۷ پدید آمده است؛ اما جاوا از سال ۱۹۹۷ ساخته شده است، پس جاوا برای اندروید ساخته نشده است؛ اما کاتلین به کمک برنامه‌نویسان اندروید آمد تا عملکرد برنامه‌هایشان بهبود یابد. جاوا نسخه‌های متنوعی دارد که هر کدام کاربردهای خاص خود را دارند؛ مثلاً با یک نسخه از جاوا می‌توان برنامه‌های مناسبی برای ویندوز نوشت و یا با کمک نسخه دیگری از این زبان، می‌توان سرورهای قدرتمندی پیاده‌سازی کرد؛ پس در قدرت جاوا، هیچ شک و شبه‌ای نیست.

(اگرچه زبان برنامه‌نویسی جاوا احتمالاً از قهوه نامگذاری شده است)، نام زبان برنامه‌نویسی کاتلین هم از جزیره کاتلین در نزدیکی سن پترزبورگ گرفته شده است. پس از گذشت حدوداً شش ماه بعد از رونمایی، در ژانویه سال ۲۰۱۲، کاتلین به عنوان زبان ماه انتخاب شد. اولین ورژن پایدار کاتلین در سال ۲۰۱۶، به نام ورژن ۱.۰ ارائه شد. در ۷ می ۲۰۱۹، گوگل رسماً کاتلین را به عنوان زبان دوم برنامه‌نویسی اندروید اعلام کرد. با پشتیبانی روزافزون اصلی تلفن‌همراه، می‌توان امیدوار بود که در آینده کاتلین، تنها به اندروید محدود نمی‌شود. برنامه رسمی KotlinConf ۲۰۱۷ کاملاً در کاتلین ساخته شد و از آی‌اواس و اندروید و همچنین بیشتر مرورگرها از طریق برنامه وب خود پشتیبانی می‌کرد.

مقایسه جاوا و کاتلین

با اینکه کاتلین در طول این چند سال، رشد چشمگیری داشته و برنامه‌نویسان زیادی از توسعه آن خوشحالند و توسط آن برنامه‌های زیادی را می‌نویسند، اما با این حال نمی‌توان منکر محبوبیت جاوا شد. با اینکه کاتلین ثابت کرد می‌تواند جایگزین خوبی برای جاوا و قدرتمند باشد، اما جاوا هنوز هم رقیب سرسختی برای کاتلین به حساب می‌آید. بهتر است برای مقایسه دقیق‌تر، توضیح کوتاهی هم درباره جاوا داشته باشیم.

جاوا را بهتر بشناسیم:

این زبان برنامه‌نویسی برای اولین بار توسط جیمز گاسلینگ، در شرکت سان مایکروسیستمز معرفی شد و الان مالک جاوا، شرکت Oracle است. جاوا زبان قدرتمندی است که هنوز هم می‌توان با او در حوزه برنامه‌نویسی اندروید برنامه نوشت. جاوا از همه سیستم‌عامل‌ها مثل اندروید، ویندوز یا





مزایا کاتلین نسبت به جاوا

در ادامه مزایای زبان برنامه‌نویسی کاتلین را مورد بررسی قرار می‌دهیم.

مختصر بودن کاتلین: کاتلین، زبان مختصری است و همین باعث کم شدن تعداد خطوط یک برنامه می‌شود و حدوداً ۴۰ درصد، کدنویسی کمتری نسبت به جاوا دارد.

```
fun main() {
    val name = "stranger"
    println("Hi, $name!")
    print("Current count:")
    for (i in 0..10) {
        print(" $i")
    }
}
```

تعامل کاتلین با جاوا: زبان برنامه‌نویسی کاتلین، همکاری زیادی با زبان جاوا دارد؛ یعنی اگر ما در یک پروژه‌ی جاوا از کاتلین استفاده کنیم، با مشکل مواجه نمی‌شویم.

یادگیری خیلی ساده کاتلین: اگر شما قبلاً با زبان جاوا آشنایی داشته باشید و کدنویسی کرده باشید، یادگیری کاتلین برای شما خیلی ساده است.

خطای کم در برنامه‌نویسی با کاتلین: زبان کاتلین، استاتیک می‌باشد؛ یعنی زمانی که برنامه کامپایل می‌شود، هر خطایی که وجود داشته باشد را پیدا می‌کند و به ما نشان می‌دهد. این ویژگی کاتلین از خاصیت **Statically typed** آن می‌باشد.

محبوبیت کاتلین بین برنامه‌نویسان: با وجود اینکه کاتلین جدید و نوپا است، اما برنامه‌نویسان اندروید، شور و اشتیاق زیادی به کدنویسی با آن دارند.

کاتلین، زبان اصلی سیستم عامل فیوشا: چرا سیستم عامل فیوشا باید جز مزایای زبان کاتلین باشد؟ خبرها درباره سیستم عامل فیوشا از آنجا شروع شد که به تازگی گوگل اعلام کرد، قصد دارد سیستم عامل اندروید را بردارد و سیستم عامل فیوشا را جایگزین آن کند و این کار تا چند سال آینده انجام می‌شود. آیا سیستم عامل فیوشا دیگر کاتلین و جاوا را پشتیبانی نمی‌کند؟ شنیده‌ها حاکی از این می‌باشد که کاتلین زبان اصلی سیستم عامل فیوشا خواهد شد، که این نیز یکی دیگر از مزایای زبان کاتلین محسوب می‌شود.

ایمنی بالا در کدهای نوشته شده با کاتلین: همان‌طور که قبلاً به این موضوع اشاره شد، کد کاتلین نسبت به جاوا مختصرتر است. این امر موجب ایمنی بالا در کدهای نوشته‌شده زبان کاتلین می‌شود. این کوتاه بودن کدها، باعث کم شدن خطاها نیز می‌گردد.

پشتیبانی برنامه‌نویسی تابعی، به بهترین نحو در کاتلین: یک الگوی برنامه‌نویسی مناسب برنامه‌نویسان را قادر می‌سازد که تا مسائل خیلی آسان حل شود. این موضوع در جاوا هم معرفی شده بود؛ ولی نه این‌چنین که در کاتلین، پشتیبانی می‌شود. اگر به طور خلاصه بخواهیم بیان کنیم، مفاهیم تابعی، بسیار مختصرتر با نوع صحیح تابع، مورد استفاده قرار می‌گیرد. صرفه‌جویی در زمان و هزینه با کاتلین: به دلیل مختصر بودن زبان کاتلین، سرعت کدنویسی و کار با کاتلین، خیلی بیشتر از جاوا می‌باشد و همین امر باعث می‌شود که توسعه‌دهندگان، هزینه و همچنین زمان کمتری صرف کنند.

معایب زبان کاتلین

هیچ زبان برنامه‌نویسی، بدون عیب و اشکال نیست و کاتلین هم بی‌عیب و نقص نیست: اگر هم‌اکنون با استفاده از جاوا، پروژه‌های اندروید خود را کدنویسی می‌کنید و می‌خواهید به کاتلین کوچ کنید، باید در نظر داشته باشید که هزینه و همچنین زمانی را برای یادگیری کاتلین صرف کنید.

کاتلین، هنوز به جاوا تبدیل نشده است؛ اگرچه شباهت بسیاری به جاوا دارد؛ اما هنوز نمی‌توان منکر تفاوت‌های اساسی در هسته‌ی این دو زبان شد؛ بنابراین برنامه‌نویسان، بدون یادگیری پیش‌نیازهای زبان کاتلین، قادر به کار کردن با این زبان نخواهند بود.

در کاتلین، در سرعت فرآیند کامپایل، نوسان وجود دارد. در برخی موارد مانند سازه‌های افزایشی ^{Incremental Builds}، سرعت کاتلین به مراتب بهتر از جاوا است؛ اما در زمینه سازه‌های تمیز و شفاف، جاوا حرف اول را می‌زند.

یکی از معایب کاتلین کمبود نیروی متخصص است. پس از کنفرانس I/O که در آن گوگل، حمایت خود از کاتلین را به صورت رسمی اعلام کرد، کمپانی‌ها با مشکل کمبود برنامه‌نویس کاتلین مواجه شدند. در طرف دیگر برنامه‌نویسان جاوا، به وفور و به راحتی در بازار کار، پیدا می‌شوند.

بازار کار برنامه‌نویسی کاتلین چگونه است؟

برای جواب دادن به این سوال که "بازار کار برای یک توسعه‌دهنده کاتلین، چگونه است؟" باید به این نکته توجه کرد که چون زبان برنامه‌نویسی کاتلین، یک زبان برنامه‌نویسی جدید است و گوگل هم از آن پشتیبانی می‌کند

و هنوز متخصص‌ها در این زمینه اشباع نشده و زیاد نیستند، به همین علت اگر آن را بلد باشید، در آینده می‌توانید درآمدزایی خوبی از آن داشته باشید.

آینده کاتلین

در زمینه توسعه اپلیکیشن‌های اندروید، اینط به نظر می‌رسد که گوگل تمرکز خود را بیش بر روی زبان کاتلین گذاشته است. در حقیقت اسم کمپانی گوگل در پشت زبان کاتلین، باع محبوبیت و جایگزینی آن با جاوا خواهد شد اما فراموش نکنید که زبان جاوا، فقط مخصوص توسعه نرم‌افزارهای موبایل نیست و ابعاد خیل گسترده‌تری نسبت به کاتلین دارد؛ بنابراین استفاده همزمان از این دو زبان برای توسعه اپلیکیشن‌های درجه یک موبایل، منطقی کم‌ریسک‌تر به حساب می‌آید.

نتیجه‌گیری

برای استفاده از زبان کاتلین در توسعه اپلیکیشن‌های موبایل، اصلاً عجله نکنید! عوض سعی کنید گام‌به‌گام و آهسته آن را به جاوا جایگزین کنید؛ برای مثال، اگر اپلیکیشن بر پایه زبان جاوا منتشر کرده‌اید، سعی کنید در بروزرسانی بعدی خود از کاتلین نیز استفاده کنید تا بدون نیاز به بازنویسی کدها، با آن آشنا شوید. زبان کاتلین بسیار جوان است؛ اما در حال حاضر برای توسعه اپلیکیشن‌ها؛ اندروید یا iOS، از آن استفاده می‌شود و به نظر می‌رسد که در آینده‌ای نه چندان دور کاملاً جایگزین زبان جاوا در زمینه طراحی اپلیکیشن‌های موبایل می‌شود.



آلن ماتیسون تورینگ (۲۳ ژوئن ۱۹۱۲ - ۷ ژوئن ۱۹۵۴)
ریاضی‌دان، دانشمند رایانه، منطق‌دان، فیلسوف،
زیست-ریاضیدان و رمزنگار انگلیسی بود.

تورینگ به‌عنوان پدر علوم کامپیوتر و هوش مصنوعی
شناخته شده و مهم‌ترین جایزه علمی رایانه به افتخار
وی جایزه تورینگ نام گرفته است.