



ترکیب نحوی ساده و روان، کتابخانه‌ها و چهارچوب‌های متنوع و قدرتمند، انجمن‌های فعال و بزرگ برای پاسخ‌گویی به مشکلات، پشتیبانی بدون وقفه از چهارچوب‌ها، الگوهای طراحی راهگشا برای تیم‌های توسعه‌دهنده هر یک به‌تنهایی موضوعات مهم و قابل‌بحثی هستند. اما زمانی‌که یک محیط توسعه یکپارچه خوب نداشته باشید، این فاکتورها کمک چندانی به شما نخواهند کرد. هیچ توسعه‌دهنده‌ای حاضر نیست از یک نرم‌افزار واژه‌پرداز یا یک ویرایشگر ساده فایل‌های متنی برای نوشتن یک برنامه حرفه‌ای استفاده کند. برنامه‌نویسان به محیط‌های توسعه یکپارچه قدرتمند و هوشمندی نیاز دارند که فرمان‌های کلیدی یک‌زبان یا چهارچوب را متمایز از دستورات عادی نشان داده، قابلیت‌های حرفه‌ای را برای اشکال‌زدایی کدها در اختیارشان قرار داده و اجازه دهد کدهای یک برنامه را خط‌به‌خط اجرا کرده و نتیجه را مشاهده کنند. خوشبختانه برای زبان برنامه‌نویسی پایتون محیط‌های توسعه یکپارچه قدرتمندی ارائه شده که در این مقاله برترین محیط‌های توسعه سال 2018 به شما معرفی خواهد شد.

برنامه‌نویسان **پایتون** جمله جالبی با این مضمون دارند: «اولین فرزند ما و **پایتون** در اغلب موارد یک برنامه Hello World است که تنها از چند خط کد ساده تشکیل شده و ممکن است خطاهای ابتدایی نشان دهد. در نتیجه به ابزارهای تخصصی اشکال‌زدایی نیازی نداریم. ما می‌توانیم از یک ویرایشگر متنی برای بررسی کدها استفاده کرده، برنامه را به ترمینال برده و دوباره اجرا کنیم. اگر برنامه باز هم با مشکل روبه‌رو شد، دوبره به محیط ویرایشگر بازمی‌گردیم، خطاها را اصلاح می‌کنیم و دوباره برنامه را اجرا می‌کنیم.» رویکردی این‌چنینی برای برنامه‌های ساده و ابتدایی خوب جواب می‌دهد، اما زمانی‌که برنامه‌نویسی پیچیده‌تر می‌شود، کدها درون فایل‌های چندگانه‌ای قرار می‌گیرند و هر قطعه کد باید به شکل متفاوتی آزمایش شود، رفت‌وبرگشت مداوم میان نرم‌افزارها بهره‌وری را کاهش داده و فرآیند تولید یک نرم‌افزار را طولانی می‌کند. زندگی برای یک توسعه‌دهنده زمانی مفرح می‌شود که بتواند در همان مکانی که کدهای خود را می‌نویسد، کدها را آزمایش و خطاها را مشاهده کرده و خروجی برنامه را مشاهده کند. این درست همان نقطه‌ای است که یک ویرایشگر متنی ابتدایی کارایی خود را از دست می‌دهد و یک محیط توسعه یکپارچه جای آن را می‌گیرد. توسعه‌دهندگان برای ساخت یک برنامه کاربردی به نرم‌افزاری احتیاج دارند که ترکیب نحوی دستورات را مشخص کرده، تورفتگی کدها را اعمال کرده، با تایپ چند کاراکتر به‌طور خودکار دستورات را تکمیل کرده، فرآیند اجرای دستورات را تسهیل کرده و به‌درستی بداند هر یک از فایل‌های پروژه برای چه هدفی استفاده می‌شوند. در سطح پیشرفته‌تر، این نرم‌افزار کاربردی باید یک ابزار اشکال‌زدایی کارآمد و قابلیت‌هایی برای ادغام‌سازی کدها با مخازن آنلاین ارائه کند.

در ادامه با هشت مورد از بهترین محیط‌های توسعه یکپارچه آشنا خواهید شد که تلاش می‌کنند زندگی را برای توسعه‌دهندگان **پایتون** ساده کنند.

Atom، یک محیط توسعه یکپارچه است که روند کدنویسی را تسهیل می‌کند. از ویژگی‌های شاخص اتم می‌توان به رابط هوشمند و به دور از پیچیدگی و مدیریت جامع پکیج‌ها اشاره کرد. البته سرعت این ویرایشگر در مقایسه با نمونه‌های مشابه کمی پایین‌تر است و اگر فایل‌های سنگینی را باز کنید، این مسئله را احساس خواهید کرد. تیم سازنده، اتم را این‌گونه توصیف کرده‌اند: «اتم یک ویرایشگر متنی قابل‌هک قرن بیستمی است که اجازه می‌دهد بنا بر احتیاج خود آن را سفارشی کنید.» این محیط توسعه یکپارچه (اگر دقیق‌تر بگوییم ویرایشگر حرفه‌ای) از سوی انجمن کدنویسان Megalith گیت‌هاب نگهداری و پشتیبانی می‌شود. پشتیبانی از سوی یک تیم توسعه‌دهنده حرفه‌ای نشان از این حقیقت دارد که اتم به نیازهای حرفه‌ای توسعه‌دهندگان به‌خوبی پاسخ می‌دهد. اگر به قابلیت نیاز دارید که درون اتم وجود ندارد، در بیشتر موارد افزونه‌ای برای پاسخ‌گویی به این نیاز ساخته شده است. برای مثال، افزونه Git Time Machine اجازه می‌دهد نسخه‌های مختلف فایل‌های پروژه و تغییرات اعمال‌شده روی فایل‌ها در گیت‌هاب را به شکل گرافیکی مشاهده کنید. یک توسعه‌دهنده وب ممکن است به دنبال پیاده‌سازی عنصر خاصی در یک پروژه باشد که کدنویسی آن عنصر زمان‌بر خواهد بود، اما مطمئن نیست همه مرورگرها از آن پشتیبانی می‌کنند یا خیر، افزونه Can I Use، اجازه می‌دهد جست‌وجویی انجام داده و این مسئله را بررسی کند. پس از دانلود و نصب این افزونه در پنجره Command Palette ویرایشگر اتم با تایپ عبارت Can I Use کادری ظاهر شده که اجازه می‌دهد جست‌وجویی برای عناصر مختلف انجام دهید. مهم‌ترین ویژگی یک ویرایشگر حرفه‌ای به پشتیبانی نرم‌افزار از پکیج‌ها بازمی‌گردد. پکیج‌هایی که عملکرد نرم‌افزار را بهبود می‌بخشند، اتم به شکل پیش‌فرض از ویژگی مدیریت پکیج‌ها پشتیبانی کرده و با گیت و گیت‌هاب ادغام شده است، با این حال به توسعه‌دهندگان اجازه می‌دهد از پکیج Teletype استفاده کرده و به شکل بلادرنگ روی پروژه‌های کدنویسی کار کنند. کدنویسان پایتون همواره به دنبال پکیج‌های کارآمدی هستند که قابلیت‌های کاربردی اسکریپتی در اختیارشان قرار دهد. برای این منظور پکیج‌های متنوعی روی گیت‌هاب وجود دارد که توسعه‌دهندگان به راحتی می‌توانند آن‌ها را دانلود کرده و استفاده کنند. اتم بر پایه چهارچوب الکترون ساخته شده، در نتیجه یک ویرایشگر چند سکویی است. از جمله نکات مثبت اتم می‌توان به ویژگی‌های قدرتمند، رابط کاربری ساده، نمایش پروژه‌های کاربردی به بهترین شکل، شکستن پانل‌های ویرایشگر به مولفه‌هایی مجزا از رابط کاربری اشاره کرد.

لینک دسترسی:

<https://ide.atom.io>

مطلب پیشنهادی



چرخه تولید نرم‌افزارها

به‌عنوان یک توسعه‌دهنده نرم‌افزار تا چه اندازه با چرخه ساخت برنامه‌های کاربردی آشنا هستید؟

2. IDLE

IDLE، یکی از کوچک‌ترین محیط‌های توسعه یکپارچه ارائه‌شده برای پایتون است. از ویژگی‌های شاخص این ابزار می‌توان به روان بودن و مکانیسم اشکال‌زدایی قدرتمند آن اشاره کرد، اما متأسفانه فاقد قابلیت برای مدیریت پروژه است. این محیط توسعه یکپارچه سفارشی برای پایتون در مقایسه با نمونه‌هایی که به آن‌ها اشاره خواهیم کرد، کوچک‌تر است، اما هر آن چیزی که برای کدنویسی حرفه‌ای به آن نیاز دارید در اختیارتان قرار می‌دهد. خالقان زبان پایتون تصمیم گرفتند، ساده‌ترین و کاربردی‌ترین محیط توسعه را پیاده‌سازی کرده و درون بسته نصبی پایتون قرار دهند. محیطی که به نام IDLE از آن نام می‌بریم و به اعتقاد برخی از کارشناسان به نام خانوادگی Eric Idle و کاراکترهای کمدی Monty Python است و به راحتی قابل استفاده است. زمانی که بسته پایتون را نصب کردید، در کادر جست‌وجوی ویندوز عبارت IDLE را تایپ کرده و کلید اینتر را فشار دهید تا نرم‌افزار اجرا شود. برای کار با IDLE دو پنجره Edit Windows و Shell Window در اختیارتان قرار داد. پنجره Shell Window پنجره اصلی نرم‌افزار در زمان اجرا است. پنجره‌ای که بالای آن نسخه پایتونی را که استفاده می‌کنید، نشان می‌دهد. با کلیک روی منوی File و انتخاب گزینه New File پنجره Edit Window ظاهر می‌شود که پنجره ویرایش کدها است. توسعه‌دهندگان می‌توانند در حالت اسکریپتی یا حالت تعاملی از IDLE استفاده کنند. حالت تعاملی زمانی استفاده می‌شود که قطعه کدهای کوچکی باید آزمایش و اجرا شوند و خروجی به سرعت نشان داده شود. دقت کنید زمانی که از

Shell Window برای کدنویسی استفاده می‌کنید با بستن آن پنجره شل همه کدهای درون آن پاک‌شده و غیرقابل‌بازرسی خواهد بود. مفسر تعاملی **پایتون** به این شکل عمل می‌کند که با تایپ کدها و فشار کلید اینتر، کدها یا اسکریپت‌ها تحلیل و اجرا شده و خروجی در پایین کدها نشان داده می‌شود. در پنجره تعاملی کدها متمایز از یکدیگر نشان داده‌شده و به شکل خودکار تورفتگی‌ها اعمال می‌شود، البته IDLE اجازه می‌دهد، برخی از تنظیمات و پیکربندی‌ها را سفارشی کنید. درست است که IDLE فاقد امکاناتی برای مدیریت پروژه است، اما اگر پروژه شما تنها از چند فایل تشکیل‌شده باشد، مشکل خاصی نخواهید داشت. قابلیت اشکال‌زدایی قدرتمند که اجازه می‌دهد کدها را تک‌به‌تک بررسی و ارزیابی کنید، در بیشتر موارد راهگشا است. ابزار اشکال‌زدایی می‌تواند وضعیت Call Stack را برای متغیرهای عمومی و متغیرهای سراسری به بهترین شکل نشان دهد. لینک دسترسی:

<https://docs.python.org/3/library/idle.html>

3. Thonny

Thonny، یک محیط توسعه یکپارچه برای **پایتون** است که دانشگاه Tatu واقع در کشور استونی آن را ساخته است. نکته جالب توجه آن‌که این محیط توسعه یکپارچه با زبان **پایتون** نوشته‌شده و رویکرد آموزش در عمل، ابزارها و قابلیت‌های مفید را ارائه می‌کند. اما رویکرد آموزش در عمل چیست؟ یعنی شما برای اشکال‌زدایی کدها از دیباگر استفاده می‌کنید، اما در عین حال نکاتی آموزشی را نیز یاد می‌گیرید. از ویژگی‌های شاخص این محیط توسعه یکپارچه می‌توان به دیباگر قدرتمند برای اشکال‌زدایی سریع کدها و مناسب برای افراد تازه‌کار اشاره کرد. با این حال، فرآیند نصب Thonny خیلی ساده نیست. Thonny، در میان محیط‌های توسعه یکپارچه‌ای که به آن‌ها اشاره شد حداقل ابزارهای توسعه‌محور را ارائه می‌کند. با این حال، هنوز هم محیط توسعه قدرتمندی به شمار می‌رود. قابلیت اشکال‌زدایی این ابزار به دو دلیل مورد توجه است: اول آن‌که نکات زیادی در مدت‌زمان اشکال‌زدایی یاد خواهید گرفت و همانند یک ابزار آموزشی عمل می‌کند؛ دوم آن‌که اجازه می‌دهد بدون نگرانی از بابت این‌که نقاط شکست (Breakpoints) بیرونی و درونی چگونه کار می‌کنند، فرآیند اشکال‌زدایی را انجام دهید. دیباگر این محیط توسعه یکپارچه به بهترین شکل وضعیت متغیرهای در حال اجرا در برنامه را نشان می‌دهد. دیباگر Thonny اجازه می‌دهد یک بلوک از کدها را به یک‌باره اشکال‌زدایی کرده یا کدها را گام‌به‌گام اشکال‌زدایی کنید. این رویکرد زمانی کاربردی است که خطاهایی ظریف در یک برنامه رخ داده و پیدا کردن خطاها کار ساده‌ای نیست. Thonny به‌طور خودکار برای کدها و دستوراتی که تایپ می‌کنید، تورفتگی‌ها را اعمال می‌کند. این قابلیت برای افرادی که تازه به دنیای **پایتون** وارد شده‌اند مناسب است، دنیایی که در آن **پایتون** از تورفتگی‌ها برای تعریف توابع، حلقه‌ها، کلاس‌ها، دستورات شرطی و... استفاده می‌کند. Thonny می‌تواند به‌طور خودکار کدها را کامل کرده و هر کجا توسعه‌دهندگان فراموش کردند پرانتزها و براکت‌های باز را ببندند، به‌طور خودکار این کار را انجام می‌دهد. یکی دیگر از ویژگی‌های شاخص دیباگر به برجسته کردن هر نوع خطای مربوط به ترکیب نحوی بازمی‌گردد. محیط توسعه یکپارچه Thonny یک ابزار مناسب برای افراد تازه‌کار است، اما همان‌گونه که اشاره کردیم اگر تجربه‌ای در نصب پکیج‌های **پایتون** نداشته باشید، فرآیند نصب ممکن است کمی مشکل باشد. Thonny به‌عنوان یک استاندارد به نسخه جدید Raspbian اضافه‌شده و برای انجام پروژه‌های رزبری‌پای مناسب است. لینک دسترسی:

<http://thonny.org>

4. Visual Studio Code

Visual Studio Code هنوز هم انتخاب اول بسیاری از توسعه‌دهندگان است. Visual Studio Code به دلیل ویژگی‌های قدرتمند، افزونه‌های کاربردی و انجمن‌های فعال نزد توسعه‌دهندگان محبوب است. مایکروسافت پس از روی کار آمدن Satya Nadella موضع دوستانه‌تری نسبت به دنیای متن‌باز اتخاذ کرد و نه تنها تصمیم گرفت سورس کد بسیاری از محصولات خود را به شکل رایگان روی گیت‌هاب قرار دهد، بلکه محصولات ارزشمندی نیز برای دنیای متن‌باز و اندروید ارائه کرد. مایکروسافت برای متن‌باز کردن سورس‌کدهای ویژوال استودیو چندان تمایل نداشت، اما در سال 2015 میلادی تصمیم گرفت سورس کدهای یکی دیگر از محصولات پرکاربرد دنیای برنامه‌نویسی یعنی Visual Studio Code را متن‌باز کرده و در ادامه کدهای اصلی این نرم‌افزار را نیز متن‌باز کند. همین مسئله باعث شد تا Visual Studio Code یا به‌اختصار Code نزد توسعه‌دهندگان بسیار محبوب شده و به یکی از انتخاب‌های اصلی توسعه‌دهندگان **پایتون** تبدیل شود. در نتیجه توسعه‌دهندگان با نصب افزونه **پایتون** به راحتی می‌توانند از

Code استفاده کنند. Code، ابزار اشکالزدایی خود را دارد، با تمام ابزارهای کنترل سورس‌کدها قابل ادغام بوده و از تورفتگی کدها پشتیبانی می‌کند. این ابزار همچنین به یک ترمینال داخلی مجهز شده و طیف بسیار گسترده‌ای از افزونه‌ها برای آن ارائه شده است. افزونه‌هایی که رایگان قابل استفاده هستند. توسعه‌دهندگان می‌توانند تنظیمات Code را مطابق با نیاز خود پیکربندی کرده و از پائل تنظیمات قدرتمندی که Code در اختیارشان قرار می‌دهد برای سفارشی‌سازی استفاده کنند. تنظیمات هر بخش به شکل تفکیک شده و متناظر با الگوی درون فایل settings.json در اختیار توسعه‌دهندگان قرار دارد. Code همانند ویرایشگر اتم بر پایه پروژه الکترون ساخته شده و در نتیجه یک ابزار چند سکویی است. اما با اتم تفاوت‌هایی دارد. اول آن که بزرگ‌تر و کامل‌تر از اتم بوده و دوم آن که از ویژگی Intellisense پشتیبانی می‌کند. Visual Studio Code از پشتیبانی کامل مایکروسافت برخوردار است. زمانی که از **پایتون** در Code استفاده می‌کنید، محیط توسعه یکپارچه پنجره‌های پاپ‌آپی را ظاهر می‌کند که به مستندات اشاره دارند که کلاس‌ها و متدها را توصیف می‌کنند.
لینک دسترسی:

[/https://code.visualstudio.com](https://code.visualstudio.com)

5. Geany

Geany، یک محیط توسعه یکپارچه **پایتون** است که انریکه توگلر، برنامه‌نویس مشهور به زبان C و ++C آن را نوشته است. محیطی که نخستین بار در اکتبر 2005 میلادی منتشر شد. Geany، ویرایشگر سبک و چند سکویی است که ویژگی‌های اصلی یک محیط توسعه یکپارچه در آن قرار گرفته است. نسخه تحت ویندوز این ویرایشگر تنها 16 مگابایت است. از ویژگی‌های شاخص این محیط توسعه می‌توان به برجسته کردن متن و شماره‌گذاری خطوط، تکمیل خودکار دستورات، بستن خودکار براکت‌ها، بستن خودکار تگ‌های HTML و XML، جمع‌کردن کدها (Code Folding)، ناوبری کدها (Code Navigation) و... اشاره کرد.
لینک دسترسی:

<https://www.geany.org/Download/Releases>

6. Reodeo

Reodeo، یک محیط توسعه یکپارچه با کاربری خاص است. به عبارت دقیق‌تر، این محیط توسعه یکپارچه برای کار روی پروژه‌های یادگیری ماشین و علم داده‌ها استفاده می‌شود و از کرنل IPython استفاده می‌کند. Reodeo، روند جست‌وجو، مقایسه و تعامل با دیتافریم‌ها و نمودارها را تسهیل کرده، از قابلیت تکمیل خودکار، برجسته‌سازی ترکیب نحوی و پشتیبانی توکار از IPython که فرآیند کدنویسی را سریع‌تر می‌کنند، پشتیبانی کرده و لینک‌های دسترسی سریع به مستندات **پایتون** را ارائه می‌کند. محیط توسعه Reodeo برای پژوهشگرانی که سابقه کار با زبان R و محیط توسعه یکپارچه RStudio را دارند، مفید است.
لینک دسترسی:

[/http://rodeo.yhat.com](http://rodeo.yhat.com)

7. SPYDER

Spyder را پیر ریبه، توسعه‌دهنده مشهور طراحی کرده است. نسخه پایدار این محیط توسعه یکپارچه در 24 نوامبر 2018 منتشر شد. Spyder به زبان **پایتون** نوشته شده و خصایص یک محیط توسعه یکپارچه را دارد. از جمله این خصایص می‌توان به منبع باز بودن، پشتیبانی از انشعاب افقی و عمومی، تعریف Goto، تکمیل و تحلیل کدها (ویژه الگوریتم‌های یادگیری ماشین)، دیباگ کردن بلادرنگ در زمان اجرا (حتی زمانی که توسعه‌دهنده در حال تایپ کدها است)، خطاها نشان داده می‌شود، نمایش اسنادی در ارتباط با کلاس‌ها یا توابعی که در کنسول یا ویرایشگر فراخوانی می‌شوند، قابلیت جست‌وجوی متغیر که به توسعه‌دهنده اجازه می‌دهد متغیرهایی را که در زمان اجرای فایل از رابط کاربری گرافیکی ساخته شده‌اند، جست‌وجو و ویرایش کند و... اشاره کرد. Spyder می‌تواند کتابخانه‌های علمی همچون Numpy، Scipy و Matplotlib و نمونه‌های مشابه را یکپارچه کند. این محیط توسعه زمانی که در قالب یک کنسول تعاملی برای ساخت و آزمایش برنامه‌های علمی و عددی و همچنین اسکریپت‌هایی که برای ساخت آن‌ها از کتابخانه‌های Scipy، Numpy و Matplotlib استفاده می‌شود، به کار گرفته شود، بهترین عملکرد را خواهد داشت.

ارائه مستندات زیاد و کاربردی از نقاط قوت این محیط توسعه هستند.
لینک دسترسی:

<https://pythonhosted.org/spyder/installation.html>

مطلب پیشنهادی



چه زمانی پایتون نمی‌تواند پاسخگوی مسئله شما باشد؟
زبان پایتون برای چه کارهایی مناسب نیست

Eric .8

Eric محیط توسعه یکپارچه کارآمدی که برای برنامه‌نویسی‌های سطح بالا بهترین عملکرد را دارد. تمرکز اصلی Eric روی زبان پایتون است، اما از زبان رویی و سایر زبان‌های برنامه‌نویسی به‌خوبی پشتیبانی می‌کند. Eric، یک محیط توسعه یکپارچه چند سکویی بوده که از ویژگی‌های کاربردی همچون تکمیل خودکار دستورات، باز و بسته کردن کدها و تطابق آکولادهای باز و بسته با یکدیگر پشتیبانی می‌کند. ممکن است رابط کاربری Eric کمی شلوغ به نظر برسد، اما به راحتی می‌توانید از آن به‌عنوان یک ویرایشگر قدرتمند استفاده کنید. Eric یک دیباگر قدرتمند دارد و برای کدنویسی‌های جدی یک ویرایشگر آماده‌هگزا، مرورگر SQL و ابزاری برای طراحی آیکون‌ها در اختیاران قرار می‌دهد. Eric از آزمایش واحد (Unit Tests) پشتیبانی کرده و به‌خوبی قادر است برنامه‌های چندرسمانی و چندپردازشی را اشکال‌زدایی کند. Eric از سامانه‌های کنترل نسخه SVN و Mercurial به شکل بومی و از گیت با نصب افزونه پشتیبانی کرده و شبیه Visual Studio Code از همکاری بلادرنگ میان اعضای تیم‌ها برای ویرایش کدها پشتیبانی می‌کند. اگر به دنبال یک محیط توسعه قدرتمند با مجموعه‌ای غنی از ویژگی‌ها برای زبان پایتون هستید، Eric باید انتخاب اول شما باشد.

تاریخ انتشار:

16 فروردین 1398

نشانی منبع:

<https://www.shabakeh-mag.com/workshop/programming/14711/%D8%A8%D9%87%D8%AA%D8%B1%DB%8C%D9%86-%D9%85%D8%AD%DB%8C%D8%B7%E2%80%8C%D9%87%D8%A7%DB%8C-%D8%AA%D9%88%D8%B3%D8%B9%D9%87-%DB%8C%DA%A9%D9%BE%D8%A7%D8%B1%DA%86%D9%87->

%D9%BE%D8%A7%DB%8C%D8%AA%D9%88%D9%86-%D8%AF%D8%B1-
%D8%B3%D8%A7%D9%84-2018