



برنامه‌نویسان پایتون برای ساخت برنامه‌های کاربردی تحت وب و واسط‌های برنامه‌نویسی کاربردی (APIs) انتخاب‌های بسیار خوبی در اختیار دارند. Django، Weppy، Bottle و Flask تنها بخش کوچکی از چهارچوب‌های قدرتمندی هستند که فرآیند برنامه‌نویسی در پایتون را ساده کرده‌اند. اگر یک توسعه‌دهنده وب هستید و برای ساخت برنامه‌های تحت وب زبان برنامه‌نویسی پایتون را انتخاب کرده‌اید، باید بگوییم تصمیم هوشمندانه‌ای گرفته‌اید. ترکیب چهار عامل بلوغ و تکامل، کتابخانه‌ها و چهارچوب‌های قدرتمند، گسترش‌پذیری و انعطاف‌پذیری کم‌نظیر پایتون نسبت به تغییرات دنیای فناوری باعث شده تا برنامه‌نویسی تحت وب بیش‌ازپیش ساده شود. تا به امروز چهارچوب‌ها و کتابخانه‌های تحت وب متعددی برای پایتون ارائه شده‌اند که همین مسئله فرآیند انتخاب چهارچوب درست را دشوار کرده است. اما چگونه می‌توانیم از میان طیف گسترده‌ای از چهارچوب‌ها گزینه درست را انتخاب کنیم؟ در این مقاله سعی کرده‌ایم به بیانی ساده پاسخی برای این پرسش ارائه کنیم و 13 مورد از چهارچوب‌های پرقدرتی را که هر یک قابلیت‌های مختلفی برای ساخت برنامه‌های کاربردی تحت وب در اختیارتان قرار می‌دهند، معرفی خواهیم کرد.

### چهارچوب‌های بزرگ و قدرتمند تحت وب پایتون CubicWeb ... 1

CubicWeb، یک چهارچوب کاربردی وب معنایی با قابلیت استفاده مجدد و طراحی شی‌گرا است و از رویکردی جالب‌توجه پیروی می‌کند. ریک گرین، توسعه‌دهنده مشهور معتقد است: تاکید بیش‌ازاندازه روی مفاهیم انتزاعی و ایجاد بلوک‌هایی از کدها با قابلیت استفاده مجدد که مکعب‌ها (Cubes) نامیده می‌شوند، این چهارچوب را برای برخی از توسعه‌دهندگان بیش‌ازاندازه انتزاعی و غیرقابل‌تحمل کند. مکعب‌ها، مولفه‌های نرم‌افزاری هستند با سه ویژگی طرح (مدل داده‌ای)، موجودیت‌ها (منطق برنامه‌نویسی) و نماها که با کنار هم قرار دادن چند مکعب در کنار یکدیگر هر یک وظایف خاص خود را انجام می‌دهند. شما می‌توانید با ترکیب کدهایی که خود یا دیگران آن‌ها را نوشته‌اند و قابلیت استفاده مجدد دارند، نرم‌افزاری جدید ایجاد کنید. هسته مرکزی چهارچوب CubicWeb ملزومات اصلی را برای ساخت یک برنامه تحت وب ارائه می‌کند. این ملزومات به شرح زیر هستند:

- مخزنی برای ارتباطات داده‌ای و ذخیره‌سازی داده‌ها (Repository)
  - موتور تحت وب (Web Engine) برای مدیریت ارسال و درخواست بسته‌های پایه HTTP
  - توابعی برای ایجاد، خواندن، به‌روزرسانی و حذف که به‌اختصار CRUD نامیده شده و برای واکنش نشان دادن به رویدادها استفاده می‌شوند.
  - طرحی برای مدل‌سازی داده‌ها.
- CubicWeb همانند هر چهارچوب دیگری مشکلات خاص خود را دارد که می‌توان به موارد زیر اشاره کرد:
- برای تنظیم و مدیریت یک نمونه از کلاس CubicWeb باید از یک ابزار خط فرمان به همان شیوه‌ای استفاده کنید که در زمان کار با Django از آن استفاده می‌کنید.

- Cubicweb، از قابلیت برنامه‌نویسی غیرهمزمان/ناهمگام (Async) بومی پایتون 3 استفاده نمی‌کند. برای بهره‌وری از قابلیت برنامه‌نویسی غیرهمزمان باید از ماژول cubicweb.pyramid استفاده کنید تا در ادامه بتوانید از سازنده async و چهارچوب Pyramid به عنوان یک وب سرور استفاده کنید. سایر متدها و قابلیت‌های این چهارچوب بدون مشکل خاصی قابل‌استفاده هستند.

در برنامه CubicWeb برای واکنشی یا دستکاری داده‌های موجود می‌توان از RQL (سرنام Relation Query Language) استفاده کرد. RQL مسیری کاملاً تفکیک‌شده برای اتصال به منابع داده‌ای مختلف ارائه می‌کند. توسعه‌دهندگانی که محاوره‌های رشته‌ای را به شکل دستی می‌نویسند، ممکن است در ابتدا به راحتی نتوانند از این ترکیب نحوی استفاده کنند.

نصب CubicWeb آسان نیست، زیرا این چهارچوب وابستگی‌های زیادی دارد، نصب به صورت pip بهترین گزینه است، زیرا اجازه می‌دهد همه وابستگی‌ها دریافت شده و نصب شوند. در بیشتر موارد مجبور خواهید شد برخی از تنظیمات محیط کاربری محلی را به شکل دستی ویرایش کرده و تغییر دهید.

- فقدان موتور قالب‌بندی عمومی برای تولید فایل‌های HTML مشکل دیگری است، که توسعه‌دهندگان را مجبور می‌کند این فایل‌ها را خود تولید کنند. البته شما می‌توانید از یک سامانه قالب‌بندی ثالث شبیه Jinja2 یا ابزارهای دیگری شبیه چهارچوب Bootstrap HTML برای حل این مشکل استفاده کنید.

مستندات این چهارچوب همانند Web2py به اندازه‌ای طولانی‌اند که بیشتر شبیه یک کتاب هستند. در برخی موارد الگوها و رابط‌های برنامه‌نویسی کاربردی این چهارچوب به شکل غیرمعارف و پیچیده‌ای تشریح شده‌اند.

## Django ... 2

در یک دهه گذشته یا به عبارت دقیق‌تر از 13 سال پیش و درست از زمانی که چهارچوب Django به دنیای نرم‌افزار وارد شد، توانست به یکی از محبوب‌ترین چهارچوب‌های پایتون برای ساخت برنامه‌های کاربردی تحت وب تبدیل شود. قابلیت‌های کاربردی این چهارچوب باعث شده Django بیشتر برای ساخت برنامه‌های بزرگ استفاده شود تا برنامه‌های کوچک. در حالت ایده‌آل شما باید از پایتون 3.x استفاده کنید، تنها دلیلی که باعث می‌شود از نسخه 1.x استفاده کنید به برنامه کاربردی شما بازمی‌گردد که ممکن است بر پایه نسخه‌های قدیمی پایتون ساخته شده باشد. سرانجام پس از گذشت سال‌های متمادی تیم توسعه‌دهنده Django تصمیم گرفت همراه با عرضه نسخه 2.1 تغییرات بنیادینی در این چهارچوب به وجود آورد. تغییرات فوق باعث شده، نسخه جدید فقط با پایتون 3.4 به بعد کار کند. بخش عمده‌ای از محبوبیت Django به روند توسعه سریع نرم‌افزارها بازمی‌گردد، زیرا این چهارچوب شامل ماژول‌های متعددی است که اجازه می‌دهند در کوتاه‌ترین زمان ممکن از این ماژول‌ها برای توسعه برنامه‌های تحت وب استفاده کنید. مسیریابی، تجزیه و تحلیل آدرس‌های اینترنتی، اتصال به بانک اطلاعاتی، اعتبارسنجی فرم‌ها، محافظت در برابر حمله‌های اینترنتی و ارائه قالب‌های از پیش‌ساخته شده، از جمله مولفه‌های کاربردی Django هستند. امروزه، بیشتر سایت‌ها ابزارهایی را برای مدیریت کاربران در اختیار وب‌مسترها قرار می‌دهند. این مسئله باعث شده، تیم توسعه‌دهنده Django ابزارهای موردنیاز را برای مدیریت کاربران در قالب مجموعه عناصر استاندارد گرد هم آورده و در اختیار توسعه‌دهندگان قرار دهند. توسعه‌دهندگان می‌توانند به جای ساخت سامانه خاص خود برای پیگیری وضعیت حساب‌های کاربری، نشست‌ها، گذرواژه‌ها، ورود/خروج، مجوزهای مدیریتی و.... ویژگی‌های بومی Django را استفاده کنند. برای تسریع در روند انجام کارها، توسعه‌دهندگان می‌توانند قابلیت‌های کاربردی Django را به شکل as-in یا گسترش‌یافته استفاده کنند. تنظیمات و پیکربندی‌های پیش‌فرض درست و قابل‌اعتماد ارائه شده از سوی Django از برنامه‌های کاربردی در برابر تهدیدات و حمله‌های اینترنتی محافظت می‌کنند. زمانی که متغیری شبیه یک رشته با کدهای جاوااسکریپت یا HTML درون یک قالب صفحه (Page Template) قرار می‌گیرد، Django محتوای صفحه را به معنای واقعی کلمه پردازش نمی‌کند، مگر آن‌که به شکل صریح نمونه‌ای ایمن از متغیر تعریف کنید. این مکانیسم باعث کاهش حمله‌های اسکریپتی متداولی همچون Cross-Site Scripting می‌شود. زمانی که مجبور هستید فرآیند اعتبارسنجی فرم‌ها را انجام دهید، Django راهکار قدرتمندی پیشنهاد می‌کند و به شما اجازه می‌دهد از درون یک لایه ساده محافظتی CSRF فرآیند اعتبارسنجی فرم‌ها را انجام دهید. این تکنیک از برنامه شما در برابر حملات جعل درخواست فرا وبگاهی محافظت می‌کند. راهکار فوق نه تنها سطح ایمنی برنامه‌های وب را بهبود می‌بخشد، بلکه اجازه می‌دهد اعتبارسنجی تک‌تک فیلدها به شکل دقیق و ایمنی انجام شده و بازخوردهای دقیقی از خطاها همراه با جزئیات به دست آید. یکی از شاخصه‌های ارزشمند Django مستندات غنی و گسترده هستند که از زوایای مختلفی جنبه‌های مختلف این چهارچوب را بررسی کرده‌اند. کار با پایتون 3، پیاده‌سازی درست مباحث امنیتی، پیاده‌سازی درست مولفه‌های رایج به کار گرفته شده در یک برنامه وب (شبیه به نشست‌ها و صفحه‌بندی)، ایجاد نقشه سایت - همه و همه در سایت اصلی Django بررسی شده و نمونه‌های کاربردی مفیدی برای آن‌ها ارائه شده است. برای واسط‌های برنامه‌نویسی کاربردی (APIs) ارائه شده برای هر یک از لایه‌های کاربردی، مدل، نما و الگوها، جزئیات

کاملی ارائه شده است. به همان اندازه‌ای که Django قدرتمند است، پیچیدگی‌های این چهارچوب نیز بزرگ هستند و برنامه‌های نوشته شده با آن به داشتن مولفه‌های متحرک شهرت دارند. حتی برای ساخت یک برنامه ساده Django با چند مسیر باید زمان قابل توجهی را صرف پیکربندی کنید تا برنامه اجرا شود.

Django همانند چهارچوب CubicWeb دلایل خاص خود را برای این سطح از پیچیدگی‌ها دارد، برای مثال، قالب‌های صفحه نمی‌توانند از فراخوانی‌ها استفاده کنند. `{{user.name}}` را به عنوان مولفه‌ای در یک قالب می‌توانید ارسال کنید، اما نمی‌توانید همین‌کار را با `{{()user.get_name}}` انجام دهید.

Django از این تکنیک استفاده می‌کند تا مطمئن شود قالب‌ها به شکل غیرمستقیم نمی‌توانند کارهای در دست‌آفرین انجام دهند. اگر به عنوان یک توسعه‌دهنده با این سازوکار و محدودیت‌ها آشنایی نداشته باشید، ممکن است پیغام‌های خطای عجیبی در کدهای خود مشاهده کنید. درست است که برای غلبه بر این مشکل راه‌حلهایی وجود دارد، اما این راه‌حل‌ها روی عملکرد و کارایی تأثیرگذار خواهند بود. هسته کلیدی Django بر پایه همگام‌سازی استوار است. با این حال، می‌توانید از پروژه Django Channels برای اضافه کردن ویژگی همگام‌سازی به برنامه خود استفاده کنید. این پروژه، یک افزونه رسمی Django است که ضمن حفظ خطمشی Django برای مدیریت و اداره کردن همگام‌سازی ارتباطات و سوکت‌ها استفاده می‌شود.

## مطلب پیشنهادی



کتابی مختص افراد مرتبط با شبکه  
دانلود کنید: برنامه نویسی شبکه با پایتون

## Web2py ... 3

در دنیای رومی، Ruby on Rails یک چهارچوب وب خوب و قدرتمند است. ماسیمو دی پیررو (Massimo Di Piero)، استاد علوم کامپیوتر دانشگاه DePaul از چهارچوب فوق الهام گرفت تا چهارچوب وب قدرتمندی برای **پایتون** ایجاد کند؛ چهارچوبی ساده که به سادگی پیکربندی می‌شود. حاصل این کار خلق چهارچوب Web2py بود. مهم‌ترین و تأثیرگذارترین جاذبه Web2Py، محیط توسعه از پیش ساخته شده است. زمانی که نمونه‌ای از Web2Py را تنظیم می‌کنید، یک رابط تحت وب در اختیار خواهید داشت. رابطی که یک ویرایشگر آنلاین **پایتون** بوده و اجازه می‌دهد مولفه‌های کاربردی برنامه خود را از طریق این ویرایشگر پیکربندی کنید. Web2py به عنوان یک چهارچوب جامع، بدون وابستگی عمل می‌کند. این رویکرد باعث شده تا یادگیری و استقرار آن آسان شود. پس از دانلود و نصب، چهارچوب آماده بهره‌برداری است. توسعه‌دهندگان یک بانک اطلاعاتی، یک محیط توسعه یکپارچه مبتنی بر وب، یک وب سرور و یک API قدرتمند را همراه با تعدادی شی اصلی در اختیار خواهند داشت. توسعه‌دهندگان می‌توانند از طریق ماژول‌های **پایتون** یا قالب‌های HTML هر یک از مولفه‌های مدل، نما و کنترلر را ایجاد کنند. در سایت رسمی این چهارچوب مثال‌های جالبی وجود دارند که نحوه ساخت برنامه‌ها را با این چهارچوب تشریح کرده‌اند. با تحلیل این برنامه‌ها می‌توانید با منطق کار این چهارچوب آشنا شده و به سرعت از قالب‌ها برای ساخت برنامه‌های کاربردی خود استفاده کنید. توسعه‌دهندگان در حالت کلی کدهای منبع Web2Py را دانلود و نصب کرده و از آن‌ها استفاده می‌کنند. اما سازندگان Web2py برای کاربران نیمه‌فنی و ویندوز و مک نسخه‌هایی ارائه کرده‌اند که سرورهای مستقل هستند. با دانلود و اجرای هر یک از این نسخه‌ها می‌توانید یک وب سرور محلی با یک پیکربندی از پیش تعریف شده از Web2py در اختیار داشته باشید. رابط تحت وب Web2py با استفاده از Bootstrap 2.16.1 ساخته شده است. البته توجه داشته باشید که ویرایشگر درون- مرورگری ارائه شده از سوی Web2py را نباید جایگزینی برای یک محیط توسعه یکپارچه در نظر بگیرید. با این حال، قابلیت‌های مفیدی همچون نمایش شماره خط، های‌لایت کردن ترکیب نحوی **پایتون** و... را ارائه می‌کند. سیستم انتزاعی داده‌های Web2py از Django و دیگر ORM‌های (نگاشت رابطه به شی) الهام گرفته از Peewee متفاوت عمل می‌کند. این سامانه‌ها از کلاس‌های **پایتون** برای تعریف مدل‌ها استفاده می‌کنند. در Web2py شما از توابع ساده شبیه Define\_Table برای نمونه‌سازی مدل‌ها استفاده می‌کنید. البته این تفاوت‌ها بیشتر برای توسعه‌دهندگانی که تجربه کار با چهارچوب‌های دیگر را داشته‌اند، مشهود است. یک تابع مرتبط با بانک‌های اطلاعاتی رابطه‌ای اجازه می‌دهد دی‌گرامی از مدل‌ها تولید کنید؛ دی‌گرامی که عملکردی به مراتب بهتر از تجسم‌سازی مدل‌ها دارد. توسعه‌دهندگان، از رویکرد تجسم‌سازی برای نشان دادن این موضوع که مدل‌ها چگونه با یکدیگر در ارتباط هستند، استفاده می‌کنند. البته برای فعال‌سازی این ویژگی باید کتابخانه pygraphviz را نصب کنید. یکی از محدودیت‌های بزرگ Web2py به سازگاری آن با **پایتون** x.2 بازمی‌گردد. در نتیجه Web2Py نمی‌تواند از قابلیت

async پایتون 3 استفاده کند. با این حال، تیم توسعه‌دهنده قصد دارد تا web2Py را با پایتون 3 سازگار کند. مستندات Web2Py به اندازه‌ای طولانی هستند که بیشتر شبیه یک کتاب هستند. این مستندات ضمن آن که نحوه کار با پایتون را تشریح می‌کنند، فرآیند استقرار و کار با Web2Py را نیز آموزش می‌دهند. برای مثال، چگونگی استفاده از JQuery (باندل شده با Web2Py) برای ساخت برنامه‌های AJAX در فصلی کامل آموزش داده شده است.

## Weppy ... 4

Weppy حد وسط Flask و Django است. زمانی که از Weppy برای ساخت یک برنامه کاربردی استفاده می‌کنید، سادگی و روان بودن Flask و دسترسی به ویژگی‌های قدرتمندی شبیه به لایه داده‌ها، احراز هویت و... را که Django ارائه می‌کند، احساس می‌کنید. Weppy برای ساخت برنامه‌های متنوعی که بسیار پیچیده یا خیلی ساده هستند، مناسب است. کدهای Weppy همانند کدهای Flask یا کدهای Bottle خوانا بوده و پیچیده نیستند. توسعه‌دهندگان تنها به چند دستورالعمل کاربردی برای ساخت و اجرای یک وبسایت ساده نیاز دارند. Weppy با سایر چهارچوب‌هایی که توسعه‌دهندگان را مجبور می‌کنند برای دسترسی به یکسری قابلیت‌ها افزونه‌ها یا پلاگین‌های خاصی را نصب کنند، عملکردی متفاوت دارد. برای مثال، Flask و Bottle فاقد یک ORM از پیش‌ساخته شده هستند، اما Weppy همراه با یک ORM که بر پایه پروژه PyDAL به جای گزینه محبوب‌تر SQLAlchemy ایجاد شده در اختیار توسعه‌دهندگان قرار دارد. Weppy دارای یک مکانیسم به‌کارگیری افزونه‌ها است، اما فهرست کوچکی از افزونه‌ها برای آن ارائه شده که خیلی کوچک‌تر از فهرست افزونه‌های Flask است. چهارچوب‌های سبک وزنی شبیه Weppy بیشتر برای ساخت RESTful APIs استفاده می‌شوند. Weppy به ارائه برخی از ویژگی‌های کاربردی به دور از پیچیدگی‌های خاص شهرت دارد و بیشتر در چهارچوب‌های بزرگ شاهد آن‌ها هستیم. ویژگی‌هایی شبیه مکانیسم اعتبارسنجی داده‌ها، دستکاری و مدیریت فرم‌ها، اعتبارسنجی کاربر و Response Caching از جمله این موارد هستند. عملکرد Weppy روی هر یک از این ویژگی‌ها مطلوب است. عملکرد Weppy در این حوزه با Django قابل مقایسه نیست، اما برای توسعه‌دهنده‌ای که به انجام کار اضافی نیاز ندارد، مفید است، زیرا توسعه‌دهنده هر زمان احساس کرد قابلیت‌ها باید گسترش پیدا کنند با کمی صرف وقت می‌تواند این کار را انجام دهد. قابلیت کاربردی مهم دیگر Weppy که بیشتر در چهارچوب‌های بزرگ شاهد آن هستیم، پشتیبانی از زبان‌های جهانی است. این کار از طریق تجزیه کردن درخواست مرورگر (سرپازره Accept-Language HTTP) انجام می‌شود. توسعه‌دهندگان چهارچوب Weppy در نظر دارند، پشتیبانی سطح پایین از async، سوکت‌ها و First-Class Entities را به نسخه 2 این چهارچوب اضافه کنند.

## Zope2 ... 5

Zope برای برنامه‌های سرور در مقیاس سازمانی طراحی شده است، نه برای ساخت RESTFUL APIs ساده یا وبسایت‌های پایه با رویکرد تعاملی. مستندات Zope، این چهارچوب را برای طراحان وب، توسعه‌دهندگان مولفه‌ها و افرادی که وظیفه یکپارچگی کدها را عهده‌دار هستند، توصیف کرده‌اند. برای مثال، سامانه مدیریت محتوای Plome از Zope به‌عنوان بستری برای انجام فعالیت‌های خود استفاده می‌کند. روال کار Zope به این ترتیب است که درخواستی را از وب دریافت کرده، پارامترهای درخواست را با اطلاعات موجود در بانک اطلاعاتی اشیا داخلی خود (ZODB) تطبیق داده و یک شی را با استفاده از دستورات GET یا پارامترهای POST اجرا می‌کند. در ادامه اگر شی خروجی تولید کند، این خروجی را برای کلاینت باز می‌گرداند. Zope از سیستم بانک اطلاعاتی اشیا برای ساده کردن وظایفی همچون تخصیص مجوز به اشیا، ارائه سلسله مراتب وراثتی برای اشیا، مدیریت تراکنش‌ها و بازگردانی (Rollback) اشیا به بانک اطلاعاتی استفاده می‌کند. نصب Zope به دلیل پیچیدگی و اندازه کمی وقت‌گیر و پیچیده بوده و شما نمی‌توانید به‌سادگی فایل‌های منبع را از حالت فشرده خارج کرده و درون زیرپوشه پروژه خود قرار دهید. برخی از فرآیندهای نصب، شامل کامپایل کردن ماژول‌های زبان C و نصب آن‌ها در ویندوز می‌شوند. متأسفانه فرآیند از بسته‌بندی خارج کردن فایل‌های باینری ویندوز برای Zope از سال 2010 تا به امروز تغییری نداشته و درک مستندات ارائه شده نیز کمی مشکل است. زمانی که Zope اجرا و به سرور متصل می‌شود، یک رابط کاربری تحت وب برای ساخت و ویرایش اشیا ZODB در اختیار توسعه‌دهنده قرار می‌دهد.



سایتون میانبری برای پایتون سریع!  
سایتون چیست و چگونه سرعت اجرای کدهای پایتون خود را افزایش دهیم

### چهارچوب‌های تحت وب سبک‌وزن پایتون Bottle ... 1

Bottle را می‌توان نسخه کوچک‌شده Flask توصیف کرد. Bottle از یک ریزچهارچوب نیز کوچک‌تر و فشرده‌تر است، به دلیل این‌که همه کدهای پایه Bottle درون یک فایل قرار دارند. با توجه به کوچک بودن این چهارچوب، توسعه‌دهندگان می‌توانند از آن در پروژه‌های دیگر یا برای اجرای سریع‌تر پروژه‌های کوچکی شبیه به RESTFUL APIs استفاده کنند. این چهارچوب فاقد هرگونه وابستگی خارجی است. Bottle چهارچوب کوچکی است، اما به دلیل قابلیت‌های کاربردی که ارائه می‌کند، برای ساخت برنامه‌های وب عادی که به هیچ وابستگی خارجی نیازی ندارند، مناسب است.

سیستم مسیریابی Bottle بر پایه نگاشت آدرس‌های اینترنتی به توابع کار می‌کند. مکانیسمی که با ترکیب نحوی Flask تقریباً یکسان است. در ارتباط با تنظیم مسیرها با محدودیت‌های جدی روبه‌رو نخواهید شد و می‌توانید مسیرها را به شکل پویا ایجاد کنید. توسعه‌دهندگان می‌توانند از اشیای موجود در چهارچوب Bottle به‌منظور دسترسی و ویرایش داده‌های مربوط به درخواست‌ها و واکنش‌ها، کوکی‌ها، متغیرهای پرس‌وجو، داده‌های درون یک فرم که از عمل POST دریافت شده‌اند، سربراره‌های HTTP و آپلود فایل‌ها استفاده کنند. Bottle از موتور قالب‌بندی HTML ساده خودش استفاده می‌کند.

### CherryPy ... 2

CherryPy، عمری نزدیک به 10 سال دارد، اما هنوز هم به دور از پیچیدگی‌ها قابل‌استفاده است. مستندات ارائه‌شده برای این چهارچوب نشان می‌دهند، سایت تفلیکس از این چهارچوب استفاده می‌کند. CherryPy همانند الگوهای معماری نرم‌افزار اجازه می‌دهد منطق برنامه تحت وب را از کدهای برنامه جدا کرده و متمایز از یکدیگر آن‌ها را مدیریت کنید. سازوکار چهارچوب CherryPy به این ترتیب است که برای نگاشت توابع برنامه به آدرس‌های اینترنتی یا مسیریابی باید کلاسی ایجاد کنید که این کلاس شامل فضاهای نام متعلق به اشیایی است که این اشیا به‌طور مستقیم به آدرس‌های اینترنتی که قرار است استفاده شوند، نگاشت می‌شوند. برای مثال، ریشه وب‌سایت با تابعی به نام index مشخص می‌شود. برای ارسال پارامترها به توابع از متغیرهایی استفاده می‌شود که با متدهای GET یا POST مدیریت و دستکاری می‌شوند. این چهارچوب بر پایه پارادایم برنامه‌نویسی شی‌گرا ایجاد شده و یک چهارچوب HTTP کاملاً پایدار است. برای یادگیری و استفاده از این چهارچوب نیازی نیست بر تمامی جنبه‌های آن اشراف داشته باشید، هرچند فرآیند یادگیری آن ساده و روان است.

### Falcon ... 3

اگر پروژه شما در ارتباط با واسط‌های برنامه‌نویسی مبتنی بر REST-APIs است و کار مضاعف و غیرمرتبطی انجام نمی‌دهید، Falcon هر آن چیزی را که برای ساخت این مدل پروژه‌ها به آن نیاز دارید، در اختیارتان قرار می‌دهد. این چهارچوب بر پایه سریع و روان بودن طراحی‌شده و تقریباً به هیچ چیزی فراتر از کتابخانه‌های استاندارد وابسته نیست. یکی از دلایل اصلی که باعث شده کارشناسان نرم‌افزار برجسته روان و سریع بودن را به چهارچوب Falcon اختصاص دهند به تعداد خطوطی که برای ساخت این چهارچوب نوشته‌شده است، بازمی‌گردد. سازندگان این چهارچوب سعی کرده‌اند، با در نظر گرفتن موارد ضروری درون Falcon موارد اضافی را از این چهارچوب حذف کنند. توسعه‌دهندگان می‌توانند از چهارچوب Falcon برای توسعه پروژه‌های ساده یا پیچیده استفاده کنند. در این چهارچوب خطای 404 به شکل پیش‌فرض برای هر مسیری که به شکل صریح تعریف‌نشده نشان داده می‌شود. اما اگر توسعه‌دهنده‌ای تمایل داشته باشد خطاها را به شکل سفارشی به کاربر نشان دهد، به راحتی می‌تواند یکی از باندل‌های تولید استثنا (خطا) این چهارچوب شبیه به HTTPBadRequest یا استثناهای ژنریک Falcon همچون falcon.HTTPError را فراخوانی کند. تمرکز Falcon روی واسط‌های برنامه‌نویسی کاربردی است، این حرف به معنای آن است که شما برای ساخت برنامه‌های وب با استفاده از رابط کاربری استاندارد HTML با محدودیت‌هایی



روبه‌رو هستید. برای مثال، فرآیندهایی شبیه پردازش فرم و ابزارهای محافظتی CSRF تقریباً در دسترس نیستند و به جای آن Falcon از گزینه‌های ظریف‌تری برای گسترش‌پذیری قابلیت‌های خود استفاده می‌کند که این گسترش‌پذیری از طریق ساخت عناصر پیچیده حاصل می‌شود. در مقایسه با سایر چهارچوب‌ها مستندات ارائه‌شده برای Falcon کمتر است. راهنمای استفاده یک دستورالعمل گام‌به‌گام است که همه ویژگی‌های اصلی را بررسی کرده است.

## Flask ... 4

تقریباً هر کجا سخن از پایتون و چهارچوب‌های پایتون به میان می‌آید، به Flask نیز اشاره‌ای می‌شود، به این دلیل که Flask یک چهارچوب قابل‌درک، پایدار و بالغ است که به شکل گسترده‌ای استفاده می‌شود. اگر از Flask برای انجام یک پروژه تحت وب سبک‌وزن یا یک پروژه REST API استفاده کنید، تقریباً مسیر کار مشخص بوده و به‌ندرت ممکن است با مشکل خاصی روبه‌رو شوید. اگر در پایتون از Flask برای ساخت برنامه Hello World استفاده کنید، این برنامه را می‌توان در کمتر از 10 خط نوشت. Jinja2 سیستم قالب‌بندی HTML فلاسک بوده که به شکل گسترده استفاده می‌شود.



این سیستم به‌منظور ساده کردن فرآیند پردازش متن‌ها استفاده می‌شود. اما توسعه‌دهندگان می‌توانند Jinja2 را با هر یک از موتورهای قالب‌بندی که در نظر دارند، همچون Mustache جایگزین کنند. با توجه به این‌که شعار Flask سادگی است، تیم سازنده یکسری قابلیت‌های کاربردی را که سایر چهارچوب‌ها به آن تجهیز شده‌اند و عملکردی پیچیده دارند، از فلاسک حذف کرده است. برای مثال، هیچ لایه داده یا ORM وجود ندارد و هیچ قانونی مشخصی برای اعتبارسنجی فرم‌ها نیست. با این حال، خلاء عدم وجود این قابلیت‌ها را می‌توان با استفاده از افزونه‌ها جبران کرد و ده‌ها قابلیت کاربردی همچون کش‌کردن، اعتبارسنجی و اداره کردن فرم‌ها، ارتباط با بانک اطلاعاتی و... را به Flask اضافه کرد. مستندات Flask شفاف و ساده بوده، یک راهنمای شروع سریع در اختیار کاربر قرار داده و مجموعه مثال‌های کاربردی خوبی در ارتباط با نحوه کار با چهارچوب فوق ارائه می‌کند.

## Pyramid ... 5

Pyramid، چهارچوبی کوچک و روان است که شباهت زیادی به Flask داشته و خیلی بیشتر از Falcon به Django شباهت دارد. توسعه‌دهندگان می‌توانند از چهارچوب Pyramid برای نشان دادن کدهای پایتون به‌عنوان REST API استفاده کرده یا از این چهارچوب به‌عنوان هسته مرکزی پروژه وی‌بی که نیازمند برنامه‌نویسی‌های سنگین است، استفاده کنند. در مستندات Pyramid چهارچوب فوق بدون خط‌مشی‌های صریح و قاطع در اختیار برنامه‌نویسان قرار دارد که همین موضوع چهارچوب فوق را از رقبای خود متمایز می‌کند. اما فارغ از خط‌مشی صریح به چه معنا است؟ به این معنا که برای چهارچوب اهمیتی ندارد شما از چه بانک اطلاعاتی یا از چه قالب زبانی استفاده می‌کنید. در بخشی از مستندات آمده است: «Pyramid مکانیسمی برای نگاشت آدرس‌های اینترنتی به کدها ارائه می‌کند و شما آزاد هستید از مولفه‌های ثالثی که متناسب با برنامه کاربردی‌تان است، استفاده کنید.» می‌توانید با صرف کمترین کار ممکن یک برنامه بر پایه Pyramid ایجاد کنید. همانند Flask و Bottle، یک برنامه Pyramid شامل یک فایل پایتون تکی است که این فایل جدا از فایل‌های خود چهارچوب است. یک تابع (API) مسیریابی موجود در Pyramid بیشتر از دوازده خط کد نخواهد بود. بیشتر محتویات درون فایل‌ها توضیحاتی در ارتباط با تنظیم سرور WSGI یا نحوه وارد کردن دستورات هستند. به‌طور پیش‌فرض، چهارچوب Pyramid همراه با یکسری عناصر عادی که درون هر برنامه وی‌بی پیدا می‌شود، در اختیار توسعه‌دهندگان قرار گرفته است. اما این عناصر به‌جای آن‌که راه‌حل کاملی در اختیاران قرار دهند، به شکل مولفه‌های متصل به یکدیگر قابل‌استفاده هستند. مدیریت نشست‌های کاربری یا حتی مکانیزم محافظتی CSRF از جمله این مولفه‌ها هستند. در نتیجه اگر Django در زمینه پشتیبانی از حساب‌های کاربری و

فرآیندهایی شبیه به ورود یا خروج عملکرد عالی و یکپارچه‌ای دارد، نباید انتظار داشته باشید Pyramid نیز این‌گونه عمل کند. توسعه‌دهندگان می‌توانند برای حل این مشکل از افزونه‌ها استفاده کنند. همین مسئله در ارتباط با مدیریت فرم‌ها و اتصال به بانک‌های اطلاعاتی نیز صدق می‌کند.

## Tornado ... 6

Tornado، چهارچوب کوچک دیگری که برای پروژه‌های ویژه استفاده می‌شود. توسعه‌دهندگان از این چهارچوب برای ساخت برنامه‌های شبکه نامتقارن استفاده می‌کنند، برای ساخت سرویس‌ها هم عملکرد قابل قبولی دارد و به یک برنامه کاربردی اجازه می‌دهد از طریق به‌کارگیری WebSockets در یک زمان ارتباطات شبکه‌ای متعدد باز و فعالی (زنده‌ای) ایجاد کند. چهارچوب Tornado شبیه Bottle یا Falcon همراه با یک سیستم قالب‌بندی از پیش‌ساخته شده برای تولید فایل‌های خروجی (HTML)، مکانیسمی برای پشتیبانی از زبان‌های بین‌المللی، مدیریت فرم‌ها، تنظیم کوکی‌ها، احراز هویت کاربر و مکانیسم امنیتی CSRF عرضه شده است. اما فاقد قابلیت‌هایی شبیه به اعتبارسنجی فرم‌ها و ORM بوده که بیشتر برنامه‌های وبی که در تعامل با کاربران هستند، به ویژگی‌های یادشده نیاز دارند. Tornado می‌تواند زیرساختی ایده‌آل برای برنامه‌هایی که باید کنترل دقیقی روی شبکه‌های نامتقارن داشته باشند، به وجود آورد. برای مثال، Tornado ضمن آن‌که یک سرور HTTP ارائه می‌کند، یک سرویس‌گیرنده HTTP نامتقارن نیز ارائه می‌کند. با این اوصاف باید گفت: Tornado چهارچوب خوبی برای ساخت ربات‌ها و خزنده‌های وبی است که محاوره‌هایی را به شکل موازی روی سایت‌های مختلف پیاده‌سازی کرده و داده‌های بازگشتی را دریافت می‌کنند. اگر در حال ساخت برنامه‌ای هستید که از پروتکل‌هایی به غیر از HTTP استفاده می‌کند، Tornado به این نیاز شما پاسخ می‌دهد.

## Web.py ... 7

Web.py ابتدا از سوی آرون شوارتز ایجاد شد. چهارچوبی که سایت ردیت بر پایه آن شکل گرفت و بعدها ردیت آن را با چهارچوب دیگری جایگزین کرد. تا به امروز افراد مختلفی از چهارچوب web.py نگهداری و پشتیبانی کرده‌اند که از آن جمله می‌توان به آناند چیتیوتی اشاره کرد. Web.py در حوزه طراحی عملکردی شبیه Bottle و Flask دارد. می‌توانید از این چهارچوب به‌عنوان یک اسکلت اولیه استفاده کرده و بدون محدودیت خاصی پروژه خود را بر مبنای آن ایجاد کنید. برای فراخوانی یک نمونه از Web.py باید فهرستی از آدرس‌های اینترنتی و توابع نگاشت را به این چهارچوب انتقال دهید. آدرس‌های اینترنتی می‌توانند شامل عبارت‌های با قاعده همراه با پارامترهای مربوط باشند. این مکانیسم اجازه می‌دهد تا داده‌ها را از آدرس‌های اینترنتی با فرمتی شبیه `users/RayB/` یا `article/451/` به چهارچوب وارد کنید.

چهارچوب Bottle از مکانیسم مشابهی پیروی کرده، اما راهکارهایی در اختیار برنامه‌نویس قرار می‌دهد تا مطمئن شود پارامترها با معیارهای مشخص شده منطبق هستند. Web.py قابلیت‌هایی برای مدیریت نشست‌ها و کوکی‌ها ارائه کرده و حتی یک مکانیسم ساده خروجی کش نیز در آن تعبیه شده است. Web.py یک سیستم قالب‌بندی HTML دارد که بسیار ابتدایی است و برای کارهای عادی مناسب است. این چهارچوب برای ساخت پویای فرم‌های HTML که بر پایه مکانیسم‌های اولیه اعتبارسنجی اولیه کار می‌کنند، ایده‌آل است. اگر در حال ساخت برنامه‌ای هستید که فرم‌های آن باید به شکل برنامه‌نویسی ساخته شود، شبیه یک مکانیسم اولیه جست‌وجو در بانک‌های اطلاعاتی این چهارچوب کمک فراوانی به شما می‌کند.

## Wheezy.web ... 8

Wheezy.web ترکیبی از چهارچوب‌های Flask/Bottle/Pyramid است. چهارچوبی که ضمن کوچکی و روان بودن روی سرعت و هماهنگی بالا تاکید دارد. مجموعه ویژگی‌های اصلی درون خود هسته چهارچوب قرار گرفته‌اند، اما سازندگان چهارچوب مجموعه‌ای از توابع کاربردی را برای آن ارائه کرده‌اند. Wheezy.web را نمی‌توانیم یک محصول واحد توصیف کنیم. Wheezy.web در تعامل با کتابخانه‌های دیگری که از سوی سازنده چهارچوب ساخته شده‌اند، کار می‌کند که هر یک از این کتابخانه‌ها برای ارائه سرویس‌های مختلفی متناسب با نیازهای یک برنامه کاربردی قابل استفاده هستند. برای مثال، کتابخانه Wheezy.http برای انجام بیشتر کارهای ضروری استفاده می‌شود، اما کتابخانه Wheezy.security تنها زمانی استفاده می‌شود که برنامه کاربردی باید فرآیند احراز هویت کاربر را انجام دهد. این روش به‌کارگیری کتابخانه‌ها در قالب یک مجموعه فرآیند طراحی با Wheezy را ساده می‌کند. شما می‌توانید کتابخانه فوق را از طریق PyPI یا از طریق `easy_install` نصب کنید تا همه بسته‌های مربوط نصب شود. `easy_install` به‌خوبی با پایتون 2.7 کار می‌کند، اما با پایتون 3.51 مشکلاتی دارد.

چهارچوب Wheezy.web روی دو مفهوم اصلی نگاشت مسیرها به توابع و مدیریت تغییر مسیرها (Rredirect) عملکردی عالی داشته و در عین حال ویژگی‌های کاربردی دیگری نیز در اختیاران قرار می‌دهد. برای مثال، هر مسیری که با تگ secure@ مشخص شده باشد، درخواست‌های HTTPS را قبول کرده و درخواست‌ها و واکنش‌ها را به سمت پروتکل HTTPS انتقال می‌دهد. ویژگی کاربردی دیگر، میان‌افزاری است که اجازه می‌دهد خطاهای HTTP و نحوه نمایش آن‌ها را سفارشی کنید. کتابخانه‌های دیگر Wheezy مجموعه‌ای غنی از قابلیت‌ها را در اختیاران قرار می‌دهند.

کتابخانه Wheezy.validation به توسعه‌دهنده اجازه می‌دهد مطمئن شود داده‌های ثبت‌شده در یک فرم با معیارهای خاص از پیش تعریف شده منطبق شده‌اند. برای مثال، نام کاربری و گذرواژه‌ها باید حداقل تعداد کاراکترها را داشته و ساده نباشند.

کتابخانه Wheezy.caching به شما اجازه می‌دهد با ذخیره کردن پاسخ‌هایی که تغییری روی آن‌ها اعمال نشده، سرعت پردازش‌ها را افزایش دهید.

کتابخانه Wheezy.captcha با کتابخانه تصویری PIL/Pillow **پایتون** یکپارچه شده و به یک برنامه کاربردی اجازه می‌دهد کپچاها را تولید کند

## تاریخ انتشار:

06 فروردین 1398

### نشانی منبع:

<https://www.shabakeh-mag.com/workshop/programming/14710/13-%DA%AF%D9%86%D8%AC%D8%B%8C%D9%86%D9%87-%D9%BE%D9%86%D9%87%D8%A7%D9%86-%D8%AF%D9%86%DB%8C%D8%A7%DB%8C-%D9%BE%D8%A7%DB%8C%D8%AA%D9%88%D9%86>