

کتابخانه پایتون

زمستان ۴۰۴

کتابخانه پایتون چیست ؟

کتابخانه‌ها (Libraries) را می‌توان به یک جعبه ابزار تشبیه کرد و هر بریده کد در آن – مثلاً کلاس‌ها یا ماژول‌ها – نقش یک ابزار را ایفا می‌کنند. بنابراین زمانی که تصمیم به ساخت اثری دیجیتال می‌گیرید، به جای اینکه تمام کارها را با دست انجام دهید، به سراغ این جعبه ابزار می‌روید و ابزارهای ضروری را از داخل آن برمی‌دارید. در نتیجه، کتابخانه‌ها در زمان و انرژی شما صرفه‌جویی و شما را از ابداع کردن دوباره چرخ بی‌نیاز می‌کنند.

اما اهمیت کتابخانه‌ها در پایتون چیست و چرا باید از آن‌ها استفاده کنیم؟ زبان برنامه نویسی پایتون معمولاً به چاقوی سوئیسی تشبیه می‌شود، زیرا گستره وسیعی از توابع گوناگون دارد که به شما در رسیدگی به هر نیاز و وظیفه‌ای کمک می‌کنند. بخش اعظمی از این گستردگی توابع و کاربردها نیز به لطف کتابخانه‌هایی بوده که عمدتاً به صورت رایگان و متن‌باز در اختیار توسعه‌دهندگان قرار گرفته‌اند.





بهترین کتابخانه های پایتون

کتابخانه Requests

کاربرد اصلی: تسهیل مدیریت درخواست‌های HTTP برای ارتباط بهینه و آسان در بستر وب یکی از محبوب‌ترین و کاربرپسندترین گزینه‌ها برای مدیریت درخواست‌های HTTP به حساب می‌آید. مهم‌ترین ویژگی Requests، سادگی آن است که به توسعه‌دهندگان اجازه می‌دهد بدون سر و کله زدن با پیچیدگی‌های پروتکل‌های HTTP، با سرویس‌های تحت وب تعامل و داده‌های مورد نیاز را دریافت کنند.

اساساً اگر با رابط‌های برنامه‌نویسی تحت وب (Web APIs) سر و کار دارید یا باید درخواست‌های HTTP را در پروژه پایتون خود مدیریت کنید، هیچ کتابخانه دیگری نخواهید یافت که به اندازه Requests بهینه و متمرکز بر بنیادین‌ترین قابلیت‌ها باشد. مهم‌ترین ویژگی‌های Requests را در ادامه می‌خوانید:

- ارسال بی‌دردسر درخواست‌های HTTP
- قابلیت‌های پیشرفته مانند پشتیبانی از آپلود فایل‌های چندگانه، استریم دانلودها و اعتبارسنجی SSL
- رسیدگی خودکار به فرایند رمزگشایی و غیر فشرده‌سازی
- مدیریت بهینه اتصالات با قابلیت‌هایی مانند Keep-Alive و Connection Pooling که روی بهبود پرفورمنس و مصرف بهینه منابع متمرکز دارند
- انعطاف‌پذیر و قابل شخصی‌سازی بسته به نیازهای پروژه
- پشتیبانی کامل از دامنه‌ها و URL های بین‌المللی



کتابخانه FastAPI

• کاربرد اصلی: ارائه یک فریم ورک* مدرن و پرفورمنس بالا برای ساخت رابط برنامه نویسی با نوعدهی استاندارد پایتون فریم ورک چیست؟

فریم ورک (Framework) را می توان ساختار و چارچوبی توصیف کرد که نقش فونداسیون را در فرایند توسعه اپلیکیشن ایفا می کند. با استفاده از فریم ورک دیگر لازم نیست کدهای اپلیکیشن را کاملاً از نو بنویسید و از سوی دیگر به مجموعه ابزارها و عناصری دسترسی دارید که سرعت توسعه را بالا می برند. بنابراین فریم ورک مثل قالبی است که می توانید با استفاده از آن – و حتی دستکاری آن – پیش نیازهای پروژه را برآورده کنید.

فریم ورکها بر مبنای زبانهای برنامه نویسی گوناگون ساخته می شوند. از جمله فریم ورکهای بسیار محبوب امروزی می شود به موارد زیر اشاره کرد:

Django

Flutter

Angular

PyTorch

Springboot

React Native

Apache Spark

Ionic

توسعه دهندگان به کمک این فریم ورکها، نرم افزارهایی قدرتمند با مجموعه ای غنی از قابلیت های گوناگون می سازند.

FastAPI که خیلی زود به یکی از محبوبترین کتابخانه های پایتون تبدیل شد، فریمورکی سریع و مدرن است که برای ساخت API با صرف کمترین انرژی ممکن طراحی شده. آنچه این کتابخانه را واقعا جذاب می کند، تاکیدی است که بر سرعت، سهولت استفاده و تولید خودکار اسناد تعاملی دارد. از سوی دیگر به لطف طراحی نوآورانه، FastAPI فرایند کدنویسی هم با سرعت بسیار بالایی پیش می رود.

علاوه بر این ها FastAPI، با فریمورک های پردازش ناهمگام سازگاری دارد و بنابراین می تواند سطوحی بالا از همروندی (Concurrency) را امکان پذیر کند که گزینه ای معرکه برای اپلیکیشن های میزبان ترافیک بالا به حساب می آید. در مجموع FastAPI اثبات کرده که کتابخانه ای قدرتمند، بهینه و رضایت بخش است و به درد هرکسی می خورد که وارد جهان توسعه API وب با پایتون شده است. مهمترین ویژگی های FastAPI را در ادامه می خوانید:

- پرفورمنس بالا
- سهولت در یادگیری و استفاده با رابط کاربری آسان و اسناد پر جزئیات
- سندسازی تعاملی خودکار با Swagger UI و ReDoc
- قابلیت های اعتبارسنجی و سریال سازی که کیفیت کد را بالا می برند
- پشتیبانی از پردازش ناهمگام برای رسیدگی به درخواست های گوناگون و بهبود عملکرد و بهینگی
- گسترش پذیری و یکپارچه سازی آسان با سایر کتابخانه ها و دیتابیس های پایتون
- بهره گیری از قابلیت های مدرن پایتون و پایبندی به استانداردهای توسعه به روز



کتابخانه Tkinter

کاربرد اصلی: ارائه راهی ساده برای ساخت رابط کاربری گرافیکی (Graphical User Interface | GUI)

اگر بخواهید با پایتون رابط گرافیکی یا GUI بسازید، Tkinter قطعاً یکی از آن کتابخانه‌هایی است که باید دم دست باشد. این کتابخانه را می‌توان جعبه ابزاری استاندارد برای طراحی رابط کاربری گرافیکی توصیف کرد که راهی سرراست برای ساخت اپلیکیشن‌ها در دستاپ در اختیارتان می‌گذارد.

در واقع بزرگترین مزیت Tkinter همین سادگی و دسترس‌پذیری فراوان است که اجازه می‌دهد بدون اینکه متخصص GUI باشید، خیلی ساده شروع به ساخت اپلیکیشن کنید. برای مثال با تنها چند خط کد می‌توانید پنجره بسازید و دکمه‌ها، کادرهای متنی و سایر عناصر گرافیکی استاندارد را اضافه کنید. نکته مهم دیگر اینکه Tkinter به صورت پیش‌فرض و با نصب پایتون به دستانتان می‌رسد. بنابراین هر لحظه که اراده کنید به آن دسترسی خواهید داشت.

- گزینه‌ای ایده‌آل برای طراحی آسان رابط کاربری گرافیکی
- نصب همراه با پایتون به صورت پیش‌فرض
- جامعه بزرگ کاربران و دسترسی آسان به محتوای آموزشی
- بهترین انتخاب برای تازه‌کارانی که می‌خواهند برنامه‌نویسی GUI را درک کنند
- دسترسی به گستره وسیعی از ویجت‌های قابل شخصی‌سازی مانند دکمه‌ها، برچسب‌ها و کادرهای متنی
- سازگار با ویندوز، macOS و لینوکس
- پشتیبانی از مدل برنامه‌نویسی رویدادمحور که استاندارد در رابط‌های گرافیکی مدرن است



کتابخانه PyGame

کاربرد اصلی: تسهیل فرایند بازی‌سازی و ساخت اپلیکیشن‌های چندرسانه‌ای با پایتون
برای خیلی از بازی‌سازانی که چه به صورت مستقل کار می‌کنند و چه از شرکت‌های بازی‌سازی بزرگ سر درآورده‌اند، کتابخانه PyGame بخشی مهم از ماجراجویی آن‌ها در این حوزه به حساب می‌آید. PyGame مجموعه‌ای از ماژول‌های کاربردی برای پایتون است که فرایند بازی‌سازی را بی‌نهایت آسان می‌کنند، حتی برای آن‌ها که تازه شروع به یادگیری برنامه‌نویسی کرده‌اند. در واقع کتابخانه‌ای در اختیارتان قرار می‌گیرد که در عین سادگی، بسیار قدرتمند است و اجازه می‌دهد بدون غرق شدن در کدهای پیچیده، روی پیاده‌سازی ایده‌های خلاقانه تمرکز کنید.

- تسهیل فرایند بازی‌سازی با ارائه فریمورکی سراسر است
- دسترسی به ابزارهای لازم برای مدیریت گرافیک، صدا و سایر عناصر کلیدی بازی
- جامعه‌ای بزرگ از کاربران و دسترسی آسان به انبوهی محتوای آموزشی
- سازگاری با ویندوز، macOS و لینوکس
- امکان شخصی‌سازی و گسترش قابلیت‌ها بسته به نیازهای پروژه
- دسترسی به سیستمی بهینه برای مدیریت رویدادهایی که برای تعاملات بازی ضروری هستند
- دسترسی به انبوهی دیگر از کتابخانه‌های پایتون برای گرافیک، صدا و مکانیک‌های بازی



کتابخانه NumPy

کاربرد اصلی: ارائه گستره وسیعی از عملیات‌ها و محاسبات عددی برای پایتون

اغراق نیست اگر بگوییم که کتابخانه NumPy باید سنگ بنای هر پروژه‌ای باشد که نیاز به محاسبات عددی از طریق پایتون دارد. اما آنچه این کتابخانه را از باقی گزینه‌ها متمایز می‌کند، آرایه‌آبجکت‌ها است. آرایه‌های NumPy از لیست‌های سنتی پایتون بهینه‌تر و قدرتمندتر ظاهر می‌شوند و عملیات‌های ریاضیاتی را به سطحی تازه می‌برند.

از سوی دیگر باید به انعطاف‌پذیری NumPy نیز اشاره کرد و فرقی ندارد که روی جبر خطی کار می‌کنید یا تبدیل فوریه یا تبدیل اعداد تصادفی، در هر صورت NumPy به کمک‌تان می‌آید. NumPy ضمناً به سادگی با مجموعه‌ای از کتابخانه‌های پردازش علمی ادغام می‌شود و بنابراین نقش فونداسیون را برای ابزارهای سطح بالاتر مثل Pandas و Matplotlib و همین‌طور Scikit-Learn ایفا می‌کند.

- ارائه آرایه آبجکت N-بعدی که بسیار سریع و بهینه ظاهر می‌شود
- ارائه ابزارهای متعدد برای عملیات‌های ریاضیاتی، جبری و آماری
- امکان یکپارچه‌سازی با بسیاری از کتابخانه‌های علمی و تحلیلی پایتون
- کاربرد گسترده در پردازش علمی
- امکان پیاده‌سازی عملیات‌های برداری کردن یا Vectorization که کد را خواناتر و سریع‌تر می‌کنند
- پشتیبانی گسترده از سوی جامعه کاربران
- کاربردی در حوزه‌های مختلف نظیر امور مالی، مهندسی و تحقیق علمی



کتابخانه Pandas

کاربرد اصلی: ارائه ساختمان‌های داده سریع و انعطاف‌پذیر برای تسهیل کار با داده‌های ساختاریافته و سری زمانی

ابزاری بسیار قدرتمند و ضروری برای دستکاری و تحلیل داده به حساب می‌آید. این کتابخانه با ساختمان‌های داده‌ای مانند **DataFrames** و **Series** زندگی را برای شما آسان‌تر خواهد کرد و از طرف دیگر اجازه می‌دهد انواع داده را – چه جدولی باشد، چه سری زمانی و چه هر ساختمان دیگری – به شکلی بهینه پردازش کنید.

انعطاف‌پذیری در دستکاری داده هم نکته دیگری است که نباید فراموش شود. از پاکسازی و دگرگون‌سازی داده با عملگرهای پایتون گرفته تا عملیات‌های پیچیده ادغام داده، **Pandas** کاری که می‌کند که همه وظایف به شکلی سرراست و بهینه پیش بروند. این را نیز باید افزود که **Pandas** با برخی کتابخانه‌های آمار و دیتا ساینس مانند **NumPy** و **SciPy** ادغام شده و با ابزارهای مصورسازی داده مانند **Matplotlib** نیز سازگاری دارد.

- دستکاری بهینه داده به کمک ساختمان‌های داده‌ای مانند **DataFrames** و **Series**
- دسترسی به ابزارهای گسترده برای پاکسازی، دگرگون‌سازی و ادغام داده
- یکپارچه‌سازی شده با کتابخانه‌هایی مانند **NumPy** و **SciPy**
- سازگاری با انواع فرمت‌ها برای ایمپورت و اکسپورت کردن داده
- ابزارهای قدرتمند برای تحلیل داده و آمار توصیفی
- ابزارهای بی‌نظیر برای مدیریت و تحلیل داده سری زمانی



کاربرد اصلی: تسهیل وظایف مربوط به پردازش تصویر و بینایی کامپیوتری در پایتون یکی از مهم‌ترین اجزای بسیاری از پروژه‌های پردازش تصویر به حساب می‌آید. این کتابخانه عظیم، گستره وسیعی از ابزارهای ضروری برای پردازش تصویر، ضبط ویدیو و آنالیز در اختیارتان می‌گذارد. مهم‌ترین نکته نیز، انعطاف‌پذیری بالا و تنوع ابزارهایی است که هم نیازهای ساده را پوشش می‌دهند و هم وظایف پیچیده‌تر در حوزه بینایی کامپیوتری. از نقاط قوت برجسته OpenCV می‌شود به امکان پردازش در لحظه تصاویر و ویدیوها اشاره کرد و چه به دنبال تشخیص اشیا باشید، چه تشخیص چهره و چه تشخیص حرکت، به تمام ابزارهای ضروری برای پیاده‌سازی وظایف خود دسترسی دارید.

- ارائه انواع ویژگی مختلف برای پردازش و دستکاری تصویر
- دسترسی به ابزارهای ضروری برای رسیدگی به وظایف پیچیده‌ای مانند تشخیص اشیا، تشخیص چهره و تحلیل حرکت
- امکان یکپارچه‌سازی با کتابخانه‌های یادگیری ماشین و یادگیری عمیق و بهبود پروژه‌های هوش مصنوعی
- دسترسی به کالکشنی عظیم از الگوریتم‌های بینایی کامپیوتر که تقریباً تمام نیازها را برطرف می‌کنند
- سازگاری با انواع سیستم عامل‌ها و پلتفرم‌ها
- تحت حمایت جامعه‌ای بزرگ از کاربرانی که به پیشرفت مداوم و ارائه منابع آموزشی کمک می‌کنند



کتابخانه TensorFlow

کاربرد اصلی: تسهیل فرایند ساخت و آموزش مدل‌های یادگیری ماشین، با تمرکز بر مقیاس‌پذیری و پرفورمنس بالا
اگر بخواهید با پایتون پروژه‌های یادگیری ماشین و یادگیری عمیق را پیش ببرید، کتابخانه TensorFlow بدون تردید یکی از بهترین انتخاب‌ها خواهد بود. این کتابخانه که توسط تیم Google Brain طراحی شده، اکوسیستمی انعطاف‌پذیر است که رسیدگی به انبوهی از وظایف مربوط به یادگیری ماشین را امکان‌پذیر می‌کند.

آنچه TensorFlow را واقعا جذاب می‌کند، ماهیت قدرتمند و مقیاس‌پذیری است که اجازه می‌دهد شبکه‌های عصبی بسیار پیچیده بسازید و آموزش دهید. علاوه بر این شاهد پشتیبانی از پردازش توزیعی هستیم که یعنی می‌توانید مدل‌های خود را روی چندین پردازنده یا پردازشگر گرافیکی آموزش دهید و سرعت کار را به مراتب بالاتر ببرید. بعد از این، اشاره به این ضروری است که TensorFlow با Keras هم یکپارچه‌سازی شده است. Keras یک رابط برنامه‌نویسی سطح بالا برای شبکه‌های عصبی است که هم ورود به این جهان را برای تازه‌کارها آسان می‌کند و هم به متخصصان اجازه می‌دهد با رابط‌های برنامه‌نویسی سطح پایین‌تر، عملیات‌های پیچیده‌تر را به اجرا درآورند.

در مجموع باید گفت که TensorFlow گام‌های بزرگی برای تقویت و دسترس‌پذیر کردن یادگیری ماشین و یادگیری عمیق برداشته است. این کتابخانه، قدرت، انعطاف‌پذیری و مقیاس‌پذیری را به صورت همزمان ارائه می‌دهد و ابزاری حیاتی برای هرکسی است که یادگیری ماشین را جدی می‌گیرد.



سایر کتابخانه ها

- فریمورک Aiohttp: ابزاری برای پیاده‌سازی قابلیت‌های کلاینت/سرور HTTP ناهمگام
- کتابخانه Seaborn: ابزاری برای بهبود مصورسازی داده در پایتون، به کمک رابط کاربری سطح بالایی برای ساخت عناصر گرافیکی آماری و اطلاعاتی
- کتابخانه Bokeh: ابزاری برای مصورسازی داده به صورت تعاملی در پایتون
- کتابخانه Sci-Kit Learn: مجموعه‌ای از ابزارهای ساده و بهینه برای استخراج داده و تحلیل داده با پایتون
- فریمورک PyTorch: فریمورکی پویا و نوآورانه برای یادگیری عمیق و محاسبات تنسور، همراه با پشتیبانی از شتاب‌دهی پردازشگر گرافیکی
- کتابخانه Keras: ابزاری برای تسهیل فرایند ساخت و آموزش مدل‌های یادگیری عمیق، به کمک یک رابط برنامه‌نویسی سطح بالا و کاربرپسند
- کتابخانه Theano: ابزاری برای بهینه‌سازی و ارزیابی میزان بهینگی اکسپرشن‌ها، به‌خصوص در آرایه‌های چندبعدی
- فریمورک LightGBM: ابزاری برای ارائه فریمورکی سریع، توزیع‌شده و پرفورمنس بالا و مناسب برای یادگیری ماشین در مقیاس بزرگ
- کتابخانه PyCaret: ابزاری برای بهبود جریان کاری در فرایند یادگیری ماشین و افزایش دسترسی‌پذیری و بهینگی از طریق اتوماسیون
- کتابخانه BeautifulSoup: ابزاری برای تسهیل فرایند اسکرپینگ اطلاعات از صفحات وب

فاریم ورک جنگو چیست؟

فریمورک چیست؟

فریمورک (Framework) چیزی نیست جز مجموعه‌ای از ماژول‌ها که کار توسعه را آسان‌تر می‌کند. فریمورک‌ها به شما این امکان را می‌دهند که به جای اینکه برنامه‌ها و وبسایت‌ها را از ابتدا ایجاد کنید، با استفاده از یک منبع موجود و آماده این کار را انجام دهید.

اینگونه است که وبسایت‌ها – حتی وبسایت‌های ساده‌ای که توسط یک شخص طراحی شده‌اند – همچنان می‌توانند عملکردهای پیشرفته‌ای مانند پشتیبانی از احراز هویت، پنل‌های مدیریت و ادمین، فرم‌های تماس، بخش نظرات، پشتیبانی از آپلود فایل و... داشته باشند. به عبارت دیگر، اگر می‌خواستید از ابتدا یک وبسایت را طراحی کنید، باید خودتان تمام این اجزاء را توسعه می‌دادید. اما با استفاده از یک فریمورک، این اجزاء از قبل ساخته شده‌اند، فقط باید آن‌ها را به‌درستی پیکربندی کنید تا با سایت شما مطابقت داشته باشند.

جنگو (Django) چیست؟

django یا جنگو، یک فریم‌ورک متن‌باز، رایگان و سطح بالای پایتون است که توسعه‌ی سریع و طراحی تمیز و کاربردی را برای برنامه‌های وب امکان‌پذیر می‌سازد. این فریم‌ورک که توسط توسعه‌دهندگانی باتجربه ساخته شده است، بسیاری از درس‌های توسعه وب را برطرف می‌کند؛ بنابراین بدون اینکه بخواهید همه‌چیز را از ابتدا کدنویسی کنید، می‌توانید بر روی نوشتن برنامه‌ی خود تمرکز کنید.

جنگو در گذشته و حال

فریم‌ورک Django توسط Adrian Holovaty و Simon Willison توسعه‌دهندگانی که در روزنامه Lawrence Journal-World در کانزاس کار می‌کردند، طراحی شد. در آن زمان، آدریان و سایمون با PHP کدنویسی می‌کردند. اما از آنجایی که وب‌سایت آن‌ها نیاز به به‌روزرسانی سریع داشت، رسیدن به مدل‌ها دشوار بود. آدریان و سایمون به چیزی نیاز داشتند که بتوانند به‌سرعت وب‌سایت را با آن بسازند. آن وقت بود که آن‌ها تصمیم گرفتند از پایتون استفاده کنند.

آن‌ها بعدها شروع به کار بر روی جنگو کردند تا کدنویسی را ساده‌تر و مؤثرتر کنند. در سال 2005، جنگو به عنوان یک پروژه‌ی متن‌باز به صورت عمومی منتشر شد. زمانی که جامعه جنگو شکل گرفت، توسعه‌ی آن آغاز شد و تعداد وب‌سایت‌هایی که از جنگو استفاده می‌کردند به سرعت افزایش یافت. برخی از بهترین برنامه‌های ساخته‌شده با جنگو عبارت‌اند از Spotify، Instagram، Youtube، Pinterest، Dropbox جدیدی به لیست سایت‌های ساخته شده با جنگو اضافه می‌شوند و شرکت‌های مختلفی نیز شروع به استخدام برنامه‌نویس جنگو می‌کنند. امروزه، جامعه‌ی Django بیش از 11000 توسعه‌دهنده از 166 کشور را با یکدیگر متحد کرده است.



جنگو (Django) برای چه کارهایی مورد استفاده قرار می‌گیرد؟

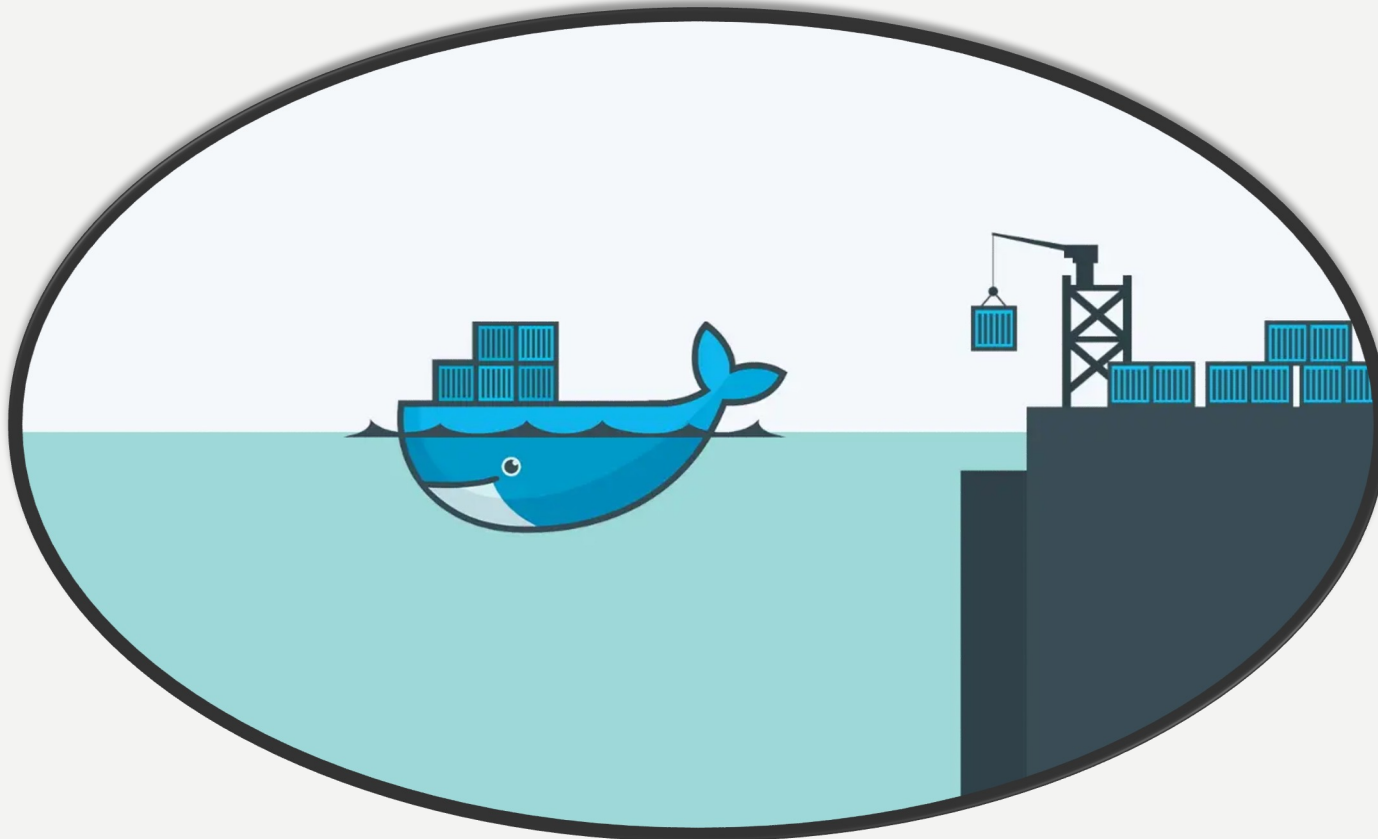
- سیستم‌های مدیریت ارتباط با مشتری (CRM)
- سیستم‌های مدیریت محتوا (CMS) برای استفاده‌ی داخلی و تجاری
- پلتفرم‌های ارتباطی
- موتورهای رزرو
- پلتفرم‌های مدیریت اسناد
- سیستم‌های تأیید هویت
- سیستم‌های فیلترینگِ دارای قوانین متغیر پویا و پارامترهای پیشرفته
- راهکارهای تجزیه و تحلیل داده‌ها و محاسبات پیچیده
- یادگیری ماشین



داگر چیست؟

Docker

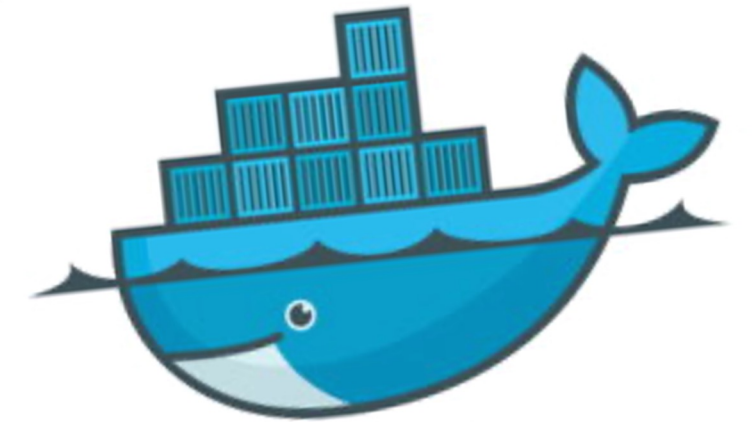
داکر یک پلتفرم برای ساخت، استقرار و مدیریت اپلیکیشن‌های کانتینر بندی شده است. امروزه داکر به ابزاری استاندارد برای توسعه‌دهندگان نرم‌افزار و مدیران سیستم بدل شده است Docker. رویکردی مبتکرانه برای اجرای سریع اپلیکیشن‌ها بدون تحت تاثیر قرار دادن باقی سیستم به حساب می‌آید. می‌توان یادگیری داکر را یکی از مهارت‌های مهم و حیاتی برای افراد فعال در حوزه DevOps دانست



داکر چیست ؟

به بیان ساده، داکر (Docker) یک پلتفرم نرم‌افزاری است که عملیات ساخت، اجرا، مدیریت و توزیع اپلیکیشن‌ها را ساده‌تر می‌کند. داکر این ساده‌سازی فرایند ایجاد اپلیکیشن‌ها را به وسیله مجازی‌سازی سیستم عامل کامپیوتری انجام می‌دهد که اپلیکیشن قرار است روی آن نصب و اجرا شود.

واقعاً، داکر مجموعه‌ای از محصولات پلتفرم به عنوان یک سرویس (PaaS) است که از مجازی‌سازی در سطح سیستم عامل برای تولید بسته‌های نرم‌افزاری استفاده می‌کند. اولین نسخه از داکر در سال ۱۳۹۲ خورشیدی (۲۰۱۳ میلادی) منتشر شد. داکر می‌تواند یک اپلیکیشن و متعلقات آن را در یک نگهدارنده مجازی اجرا کند که روی لینوکس، ویندوز یا مک OS اجرا می‌شود. Docker با استفاده از زبان برنامه نویسی GO توسعه داده شده است. داکر اپلیکیشن‌های بسته‌بندی شده ایجاد می‌کند. به این بسته‌ها کانتینر (نگهدارنده) | (Container) گفته می‌شود. کانتینرها نقش اساسی در Docker ایفا می‌کنند.



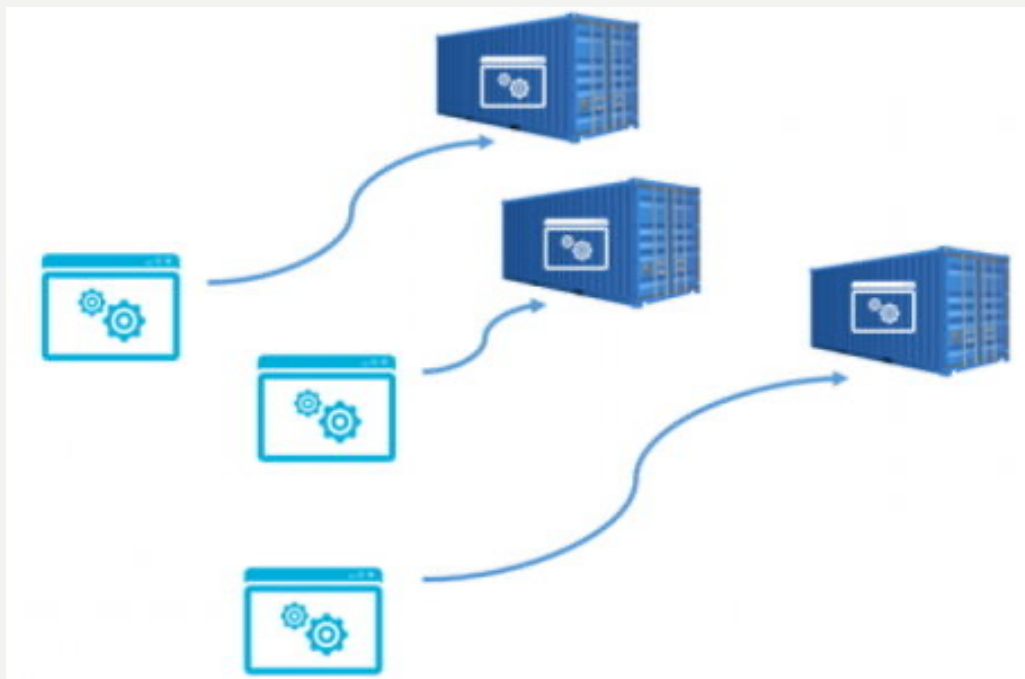
docker

اپلیکیشن محافظه‌بندی شده



کانتینر داکر (Docker Container) چیست ؟

نگهدارنده (محفظه Container) | یک واحد نرم‌افزاری استاندارد است که کدها و تمام متعلقات (Dependency) آن را بسته‌بندی می‌کند. به این ترتیب، اپلیکیشن در محیط‌های محاسباتی مختلف، سریع‌تر و با اطمینان بیشتر اجرا می‌شود. هر کانتینر یک محیط ایزوله شده را مشابه یک ماشین مجازی (Virtual Machine | VM) فراهم می‌کند. برخلاف ماشین‌های مجازی، کانتینرهای داکر یک سیستم عامل کامل را اجرا نمی‌کنند، بلکه هسته (Kernel) میزبان را به اشتراک می‌گذارند و مجازی‌سازی را در یک سطح نرم‌افزاری انجام می‌دهند. کان استفاده از Container ها هم برای اپلیکیشن‌های تحت ویندوز و هم برنامه‌های کاربردی مبتنی بر لینوکس در دسترس است. در واقع می‌توان برنامه‌های ویندوزی، لینوکسی و فایل‌های اجرایی (EXE) را در کانتینرهای داکر اجرا کرد.





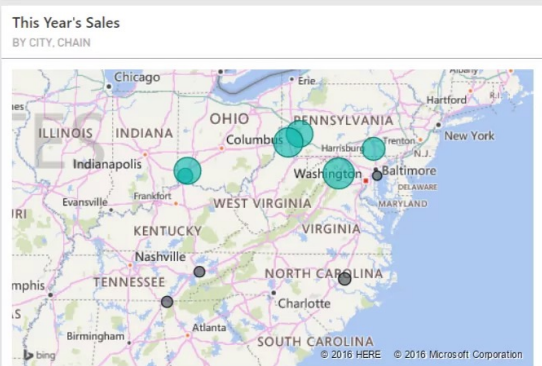
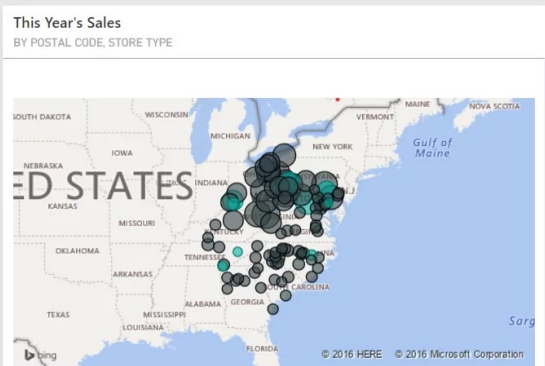
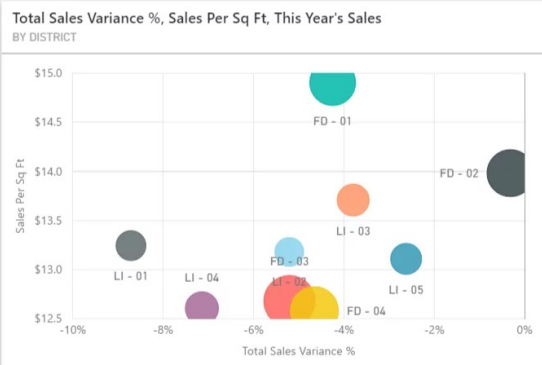
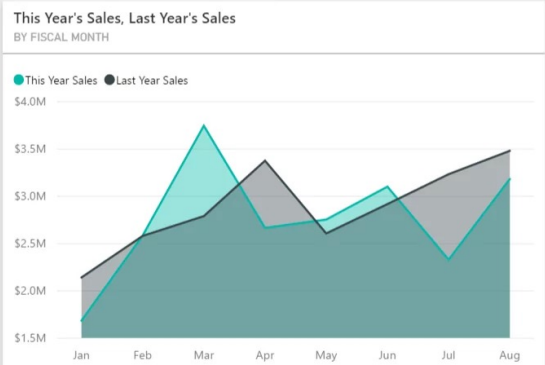
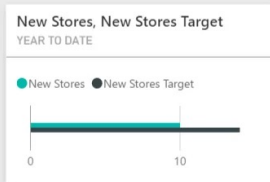
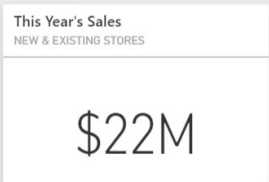
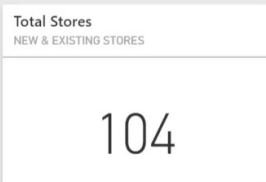
Power BI

Power BI چیست ؟

هوش تجاری مجموعه استراتژی‌ها و راهکارهای تکنولوژیکی است که در آن داده‌های سازمان تبدیل به اطلاعات کاربردی می‌شوند. هدف این فرآیند کسب اطلاعات کاربردی و مفید از داده‌های پیشین و حال حاضر سازمان است. پاور بی آی یک ابزار هوش تجاری و در واقع بخشی از این فرآیند بوده که توسط مایکروسافت توسعه داده شده است. این نرم‌افزار، پس از دریافت داده‌ها از منابع مختلف و آماده‌سازی آن‌ها برای تحلیل، به تحلیل‌گران برای ساخت گزارش‌های حاوی اطلاعات مصورسازی شده از داده‌های خام، کمک می‌کند.

مدیران با پایش مداوم گزارش‌های Power BI می‌توانند عملکرد سازمان را در بخش‌های مختلف ارزیابی و در راستای بهبود آن تلاش کنند. این گزارش‌ها در همه بخش‌های سازمان ایجاد می‌شوند. به طور مثال تصویر زیر نمونه‌ای از داشبورد شاخص‌های مدیریت منابع انسانی در نرم‌افزار پاور بی آی را نمایش می‌دهد.

Ask a question about your data



Ask a question about your data

