

دوره آموزشی زبان پایتون

زبان پایتون یکی از پرطرفدارترین زبان‌های برنامه‌نویسی است. این زبان سینتکس ساده‌ای دارد و به راحتی و سرعت در توسعه چه در پروژه‌های بزرگ و چه در پروژه‌های کوچک معروفه.

مقدمات و مفاهیم اولیه

پایتون زبونی که روی سیستم‌عامل‌های مختلف نصب می‌شود و کد رو از روی منبع اجرا می‌کند. برخلاف زبان‌های سطح پایین‌تر کد رو کامپایل نمی‌کند و بعد هم‌اشار رو با هم اجرا کند، بلکه هم‌زمان با خواندن هر خط تفسیرش می‌کند و اجراش می‌کند. (توضیح اضافه‌تر!)

آموزش پایتون وبسایت [w3school](https://www.w3schools.com/python/) مفاهیم اولیه رو به خوبی توضیح میده. تا ته بخش [Modules](#) (منظور اولین درس مربوط به ماژول‌ها است و نه بخش ماژول‌ها) مطالعه‌اش کن.

مقاله‌های زیر رو در رابطه با شی‌گرایی توی پایتون بخون

- [Everything is an object](#)
- [mutable vs immutable objects](#)

البته پایتون یه زبان مولتی‌پارادایم هست و بعضی از ویژگی‌های زبان فانکشنال رو هم دارد مثل:

- <https://discord.gg/HDHnaS6Jsure>

مطالعه‌ی بیشتر: یکی از کاربردهای استفاده از کلوزر توی [دکوریتهای پایتونه](#). دکوریتهای شمشیر دولبه است و توی استفاده ازش باید خیلی احتیاط کرد. در همین حد که بدونی چیه و چجوری کار می‌کنه کافیه مطالعه کنی.

دکوریتهای

مثلا یکی از جاهایی که دکوریتور خیلی به درد میخوره اونجاست که می خوایم log بگیریم مثلا در یک سیستمی می خوایم ذخیره کنیم که کاربر کی وارد سیستم میشن و کی خارج میشن. من تو فایل decorator.py یک نمونه ازش پیاده کردم.

پکیجینگ

پکیجها در پایتون برای کمک به توسعه و کم کردن زمان توسعه استفاده می شوند. به جای این که کلی وقت بذاریم و همه کدهایی که پروژمون نیاز داره رو خودمون بنویسیم می تونیم از پکیجهای آماده ای استفاده کنیم که این کدها و فانکشنالیتیها رو پیاده کردن.

با نحوه استفاده از پکیجها و ابزار مدیریت پکیج پایتون آشنا شو.
در مورد [سمنتیکال ورژنینگ](#) هم بخون و یاد بگیر.

serialization

در پروژههای برنامه نویسی برای ذخیره و استفاده دوباره از دادهها و از همه مهمتر برای ارسال و دریافت دادهها به/از سایر سرویسها نیاز است تا روش ذخیره سازی و ارسال مشخص باشد تا هر دو طرف (کسی که بعد از دادهها استفاده می کند یا سرویس دیگری که از این دادهها استفاده می کند) بتوانند به راحتی این دادهها را بخوانند. به این کار serialize کردن داده می گویند و روش های مرسوم برای آن وجود دارد که به برنامه نویسان کمک می کند بدون تلاش برای مشخص کردن روش ارتباط با استفاده از این روشهای استاندارد این کار را انجام دهند.

[این مقاله در مورد روشهای مختلف سریالایزشن](#) رو بخون. (خیلی نیاز نیست تو این مقاله عمیق شی)

همچنین در مورد روشهای csv , yaml هم مطالبی رو پیدا کن و بخون که چی هستن. (به نظرت از هرکدوم از روشهای گفته شده چه زمانی خوبه که استفاده کنیم؟)

سریالایز کردن در پایتون:

سریالایز کردن یعنی اون **object** که داریم به یک فرمت استاندارد دربیاریم. حالا میتونه **pickle, CSV, yaml, json, ...** باشه.

خب **CSV** حالت جدولی داره و برای این ساختارهی جدولی مناسب هست و فایل رو نگاه کنیم قابل خوندن هست. فایل **json, yaml** هم ساختار قابل خوانایی دارند ولی **pickle** فرمت باینری داره و قابل خوندن واسه ما نیست ولی **load** شدن دیتا طبق تجربه من سریع تر هست.

تمرین

در این تمرین می‌خواهیم یک کلاینت ساده با استفاده از چیزهایی که تا به اینجا از زبان پایتون یاد گرفتیم بنویسیم.

یک کلاس ابسترکت به نام **BaseEntity** بسازید که دو تابع ابسترکت **to_json** و **to_csv** دارد.

سپس چند **Entity** دیگر مانند **City, Car, Student, Street, ...** تعریف کنید که از **BaseEntity** ارث‌بری کنند و توابع **to_csv**, **to_json** را به ازای هر کلاس طوری پیاده‌سازی کنید که با هر بار فراخوانی، به صورت رندوم و فرضی اطلاعات آن یک **Entity** را با فرمت **json**, **csv** بدهند.

سپس برنامه‌ای بنویسید که از ورودی نام یک **Entity** و فرمت مورد نظر و تعداد را بگیرد و در خروجی مورد مشخص شده را بفرستد. این برنامه باید در فایلی جداگانه باشد.

خروجی	ورودی
<pre>[{"city": "7O3Y8UHI", "district": 14}, {"city": "BHINQONU", "district": 20}] first_name,last_name,id,phone_number GOLCCJKX,N8QPR75U,1019,164-380-7640 PTFCYSXU,2JJTH17T,1267,675-102-6226</pre>	<pre>Street json 2 Student csv 2</pre>

در پوشه task1 کدها زده شده است.

Security

در زمینه‌ی امنیت کدهایی که می‌زنیم [این لینک](#) رو بخون.