

# کیبرد آزاد

در دفاع از آزادی کیبرد

## راهنمای شروع و پیشرفت و حرفه ای شدن در زبان برنامه نویسی پایتون



سوالی که خیلی زیاد از من پرسید می شه که اینه که «کدوم زبان برنامه نویسی رو یاد بگیرم؟» یا «چجوری زبان فلان رو یاد بگیرم؟» و موارد مشابه. این مطلب تصورش اینه که شما پایتون رو انتخاب کردین و علاقمند هستین توش پیش برین. انتظار قبل از اون دو تا لینک قبلی رو خونده باشین و بدونین که واقعا علاقمند هستین پایتون رو یاد بگیرین و بدونین که آدم ها به شیوه های مختلفی چیز یاد می گیرن و یک راهنمای کلی برای همه نداریم. اگر با اینها اوکی هستین بیاین یادگیری پایتون رو شروع کنیم.

### شروع ایده اصلی

در مورد پایتون کتاب و راهنما خیلی خیلی زیاده. اگر با چیزهایی مثل کتاب های خوندنی راحتتر هستین به نظرم «یاد گرفتن پایتون به روش سخت» منبع کاملا خوبیه؛ بخصوص بخش های آزاد و آنلاینش برای یادگیری اولیه کاملا کافیه. اگر اهل ویدئو باشین و متن و تمرین حل کردن با استاد هم دوره یادگیری برنامه نویسی با پایتون یا ترجمه ما ازش در گوتو کلس به اسم دوره یادگیری فارسی برنامه نویسی با پایتون گوتو کلس گزینه خوبیه. اگر زبان بلدین منطقا انگیزی خوندن بهتره. یادتون باشه در

این بخش دارین خیلی مقدماتی درکی از یک زبان برنامه نویسی و شکلش پیدا می کنین. لازم نیست مته به خشخاش بذارین و لازم هم نیست برنامه نویس بزرگی بشین. نکته اینه که با مفهوم برنامه نویسی آشنا بشین و دستورزبان و مفاهیم پایه ای پایتون رو درک کنین؛ که بسیار هم ساده هستن!

در این مرحله اگر از یک زبان برنامه نویسی دیگه میاین و در حال حاضر زبون خودتون رو بلدین، پیشنهاد می کنم به شکل عمومی دنبال python tutorial بگردین و چند تا رو باز کنین و اونو که دوست دارین رو کامل بخونین. اینجوری خیلی زود سینتکس جذاب پایتون رو یاد می گیرین و می تونین برین مرحله بعدی.

**نکته:** بعضی ها در اولین برخورد با پایتون با یک سورپریز بزرگ روبرو می شن: پایتون از { و } برای مشخص کردن بلوک ها استفاده نمی کنه و به راحتی از اسپیس یا تب برای اینکار کمک میگیره. گفته می شه اولین شروع هرکس که از زبان دیگه اومده به پایتون، با یک فرار همراه و بار دوم احوالپرسی از پایتون است که عاشق این زبون ساده و کم ادعا می شه (:

## پنج نکته برای دوری

**جان سونمز** پنج نکته رو می گه که باید ازش دوری کنین و یکی از اصلی ترین مراجع این مطلب هم اونو تکرار می کنه و من کی باشم که تکرارشون نکنم؟

1. کتاب ها رو از اول تا آخر نخونین. در شروع کتاب ها رو از اول تا آخر نخونین. ما بخش کمی از کتاب ها رو یادمون می مونه و فقط کتاب خوندن بخش زیادی از انرژی شما رو می گیره. کتاب مال وقتی است که شما موضوعات رو بلدین و می خواین توش عمیق بشین.
2. بدون برنامه تو بحث شیرجه نزنین. آدم ها رو به دلایل مختلف جو می گیره. یکهو می خوان خدای چیزی بشن و یک عالمه کتاب دالود می کنن و شایدم دو تا ادیتور برنامه نویسی و عضو یکسری کانال و فروم و میلینگ لیست می شن و هی چیز می خونن و فیلم می بینن. این اشتیاق معمولا به همین سرعت هم فروکش می کنه. بدونین برای چی وارد بحث شدین و می خواین به کجا برسین و می خواین باهاش چیکار کنین.
3. حوزه خودتون رو مشخص کنین. دنیا وسیعه! حتی برنامه نویسی هم خیلی وسیعه و شاید براتون جالب باشه که حتی پایتون هم خیلی وسیعه! بهتره شما بدونین در چه حوزه ای می خواین کار کنین و از اون شروع به یادگیری کنین. اگر می خواین اپ تحت وب درست کنین لازم نیست با numpy شروع کنین و اگر می خواین محاسبات انجام بدین شروع کردن یا اصرار روی scikit اشتباهه. از یک جا شروع کنین و توش پیش برین.
4. دو (یا بیشتر!) چیز رو همزمان یاد نگیرین. این اشتباه با مورد ۲ شباهتهایی داره. آدم ها یکهو می خوان «هکر» بشن و شروع می کنن شبکه و سیستم عامل و برنامه نویسی و انگلیسی و آلمانی و روسی و نقد فیلم و ادبیات کلاسیک رو با هم خوندن و یاد گرفتن. از یک جا شروع کنین و وقتی در حد معقولی پیش رفتین برین سراغ چیز بعدی. این مساله حتی داخل پایتون هم صدق می کنه. اگر دارین جنگو یاد می گیرین نیازی نیست همزمان شروع به خوندن فلسف هم بکنین. یکی رو پیش برین در حد معقول و بعد فرصت برای بعدی هست.
5. بدون تجربه کردن پیش نرین. بعضی چیزها برای آدم ها لذت بخشه. من از دیدن فیلم بازی شطرنج خیلی خوشم می یاد و در نتیجه شش ماه فیلم دیدم بدون اینکه بازی کنم؛ هیچ پیشرفتی هم نداشتم. حتما حین تئوری با عمل هم برخورد داشته باشین.

## شروع عملی پایتون

این مساله ممکنه با بخش اول که عنوانش «شروع اصلی» بود پیش بره. برای اینکار لازمه اول پایتون رو نصب کنین و چند برنامه خیلی خیلی ساده باهاش بنویسین. راه انداختن محیط توسعه یکی از قدم های مهم در یادگرفتن است؛ و اتفاقا گاهی سخت. خودتون رو با محیط کامندلاین و خط فرمان پایتون عادت بدین.

پایتون دو نسخه خیلی فعال داره که کمی با هم فرق دارن؛ پایتون نسخه دو و پایتون نسخه ۳. در این مورد همیشه ما رو دست می ندازن و ما هم باهاشون می خندیم. جدای از خنده شما اگر مردد هستین، پایتون ۳ رو یاد بگیرین.

در این سطح بازم می تونین چیزهایی مثل «یاد گرفتن پایتون از راه سخت» رو جلتون بذارین یا مجموعه ای از سوالات پروژه اولیر که براتون جالب و معقوله رو حل کنین. حتما خوبه سری به جامعه پایتون هم بزنین که بسیار مهربونن. مثلا خوندن زن پایتون بسیار توصیه می شه تا فلسفه پشت پایتون براتون روشن تر بشه. یادتون باشه هنوز دستتون داره گرم می شه ولی خیلی زود وقتشه که برنامه نویسی رو برای خودتون واقعی کنین.

## یک پروژه عملی بردارین

حالا باید تا حدی بدونین که کجا ایستادین و دنبال چی هستین. اگر دنبال وب هستین یک پروژه عملی برای خودتون تعریف کنین و با چیزی مثل جنگو شروع کنین. درگیر اسم و اینها نباشین. پروژه رو تعریف و شروع کنین. شاید خیلی زود پاکش کردین! اصلا مهم نیست. مثلا یک سیستم رای گیری آنلاین. یا سیستم حسابداری شخصی. یا دفترچه تلفن دوستانتون. یا فهرست کتاب ها و اینکه چی خوندین و چی قرض دادین یا هر چیز دیگه یا یه سیستم وبلاگ شخصی یا صورت غذاهایی که پختین و روش پختش. شاید یک برنامه مخصوص تنظیم کردن قرار ملاقات. موضوع خیلی مهم نیست. مهم اینه که خیلی زود چیزهایی که یاد میگیرین به شکل عملی در یک جایی استفاده بشه. اگر با جنگو کار می کنین کتاب Two Scoops of Django توصیه می شه. من خودم اخیرا به توصیه خودتون اینو خوندم و خوب بود برام. در این مرحله چیزهایی که یاد می گیرین رو نسبتا عمیق کنین. حوزه شما مشخص شده و باید حرفه ای تر بشین توش. اگر مثلا یادگیری ماشینی رو برداشتین یک مساله واقعی تعریف کنین و سعی کنین قدم به قدم حلش کنین. شاید مثلا تشخیص سگ از گربه قدم اول باشه و تشخیص طبقه بندی یک وبلاگ قدم دوم. شاید علاقمند به یک اپلیکیشن دسکتاپ باشین. بازم امکانش هست. مهم اینه که حوزه خودتون رو کشف کنین و بدونین که در کدوم خط دارین پیش می رین.

در این مرحله مهمه که مثل یک برنامه نویس پایتون فکر کنین. به زن پایتون برگردین و پپ ۸ که شیوه نامه نوشتن به پایتون است رو بخونین و سعی کنین شبیه یک پایتون نویس حرفه ای باشین - چه تو دستخط چه شیوه تفکر. گفته می شه زبانی که تفکر آدم رو عوض نکنه، زبان مفیدی نیست. به جای تلاش برای بزرگ تر کردن برنامه تون، رو این تمرکز کنین که پایتونی بنویسینش و زبان رو بفهمین. در این مرحله حتی شاید با بعضی کتابخونه ها آشنا بشین و مثلا خوبه مطمئن بشین که برای برنامه خودتون یونیت تست نوشتین. در همین قدم شما ممکنه از خیلی «حرفه ای» ها جلو بیافتین.

## حالا عمیق بشین

شما درک از زبان دارین و می تونین باهاش کار کنین درسته؟ حالا وقتشه که حرفه ای بشین و بهتر از هر کس زبان رو درک کنین. حداقل بهتر از ۸۰٪ آدم های دیگه. این بخش ها مهمترین اجزای داکيومنتهای رسمی پایتون هستن برای خونده شدن توسط کسی که می خواد زبان رو به تسخیر خودش در بیااره: 1, 2, 3, 4, 5, 6.1, 6.2, 26.4 و 27.3.

یک تمرین فوق العاده برای بهتر کردن دانش جدید و تثبیت کردنش اینه که کدهای قبلی رو ریفتور کنین؛ بازنویسی کد برای بهتر کردنش بدون تغییر در خروجی ها.

## پیشرفت کنین

حالا شما یک برنامه نویس کاملا قابل قبول هستین ولی هنوز می تونین بهتر بشین. مورد ۵ رو یادتون باشه و کارهای عملی دیگه ای رو شروع کنین. به کد بقیه نگاه کنین [مثلا پروژه بستون](#) و اونها رو بهتر کنین یا بهشون فیچر اضافه کنین. اینطوری برای خودتون رزومه می سازین. البته به جای اون پروژه می تونین سراغ کارهای آدم های حرفه هم برین، مثلا تمام کارهای [دیوید بیزلی](#) یا [کدهای کنت ریز](#) (مثلا به [tablib](#) نگاه کنین) پروژه های مفید برای عموم رو شروع کنین و پیش برین و البته هر جا که گیج می شین شروع کنین به یادداشت کردن چند تا سر فصل اصلی و بعد سرچ و خوندن در موردشون. یادتون باشه که [داکیومنت های خود پایتون](#) واقعا فوق العاده هستن. و حالا وقت خوندن کتاب هم شده:

- کتاب [Fluent Python](#)
- کتاب [Effective Python](#)

## نکته پایانی

از مسیر لذت ببرین. اگر منتظر رسیدن به مقصد هستین، برنامه نویسی کار شما نیست. آدم هایی که هی به نوک قله نگاه می کنن و افسوس می خورن معمولا توی دینای کامپیوتر رنج می کشن چون هیچ وقت به پایانش نخواهیم رسید. قدم به قدم پیش برین و از هر قدم لذت ببرین و جلو برین. به بقیه کمک کنین چون یکی از بهترین راه های حرفه ای شدن است. توی فروم ها وقتی سوالی می بینین دنبال جوابش بگردین، جوابش رو دقیق یاد بگیرین و با زبون خودتون برای طرف توضیح بدین. حوصله داشته باشین و هیچ وقت سعی نکنین در جایی باشین که بهترین هستین. همیشه سعی کنین کنار بهتر از خودتون بشینین و چیز یاد بگیرین و به زودی آدم فوق العاده ای خواهید شد. پایتون شیرین و بانمکه؛ امیدوارم کدهاتون رو توی گیت هاب ببینم.

## منابعی برای نگاه کردن

- [پپ ۸ به زبان آدمیزاد \(فارسی\)](#)
- [راهنمای هیچایکرها به پایتون مرجعی بزرگ از پایه تا پیشرفته](#)
- [ترجمه فارسی و دسته جمعی تینک پایتون، یکی از منابع خوب](#)

اشتراک گذاری:



این نوشته در مقاله با برچسب برنامه نویسی، پایتون، راهنمای شروع در فوریه 7, 2017 [https://jadi.net/2017/02/first-steps-python/] توسط منتشر شده است.

1 ورود ▾

8 نظر جادی . نت

چینش بر اساس بهترین ها ▾

پیشنهاد 2 به اشتراک گذاری

## به گفتگو بپیوند...

nima • ۲ ماه پیش

سلام ایا تو برنامتون ساختن 2 ipic رو هم دارین یا نه؟؟  
• پاسخ • به اشتراک گذاری >

yazdanpanah.a • ۲ ماه پیش

سلام جادی جان!

دستت درد نکنه. فقط دوتا مورد:

- ۱- فکر کنم لینک به در گوتو کلاس اشتباه باشه و میره تو edx
- ۲- یه سری از برویج که من قاعدتا جزئشونم نشستن کتاب رو بصورت آزاد ترجمه کردن ولی هنوز ویرایش نکردن گفتم شاید بدرد بخوره  
<https://github.com/ThinkPyt>  
• پاسخ • به اشتراک گذاری >

yazdanpanah.a ناظر • jadijadi • ۲ ماه پیش

مرسی از الان نگاه می کنم. و لینک رو هم اضافه می کنم. آره اون کتاب کار خوبی بود. حیف که من به مشارکتش نرسیدم (:

10:19 2017-02-10 <notifications@disqus.net> GMT+03:30:

• پاسخ • به اشتراک گذاری >

Hamid Shekarforoush • ۲ ماه پیش

man vase machine learning az ide spyder estefade mikonam. hamintor package anaconda.  
nasb rahati dare.  
• پاسخ • به اشتراک گذاری >

ahmad mianbandi • ۲ ماه پیش

خیلی ممنون جادی جان.  
منم مث دوستمون سوالم اینکه که به نظرت بهترین IDE برای پایتون (البته برای کسانی که مث من وب کار می کنن) چیه؟  
من تا الان با ویم کار می کنم. مشکلم باش خطاهای سینتکس و رفت و آمد بین فایل هاس.  
اگه پلاگینی برای حل مشکلم دارین خیلی ممنون میشم کمک کنید.  
• پاسخ • به اشتراک گذاری >

Zahed Zibae • ۲ ماه پیش

مرسی جادی جان به خاطر راهنمایی امیدوارم email ام به دستت رسیده باشه و مفید واقع شده باشه  
;  
• پاسخ • به اشتراک گذاری >

Saeid Dadkhah • ۲ ماه پیش

سلام خسته نباشید. ببخشید بهترین IDE برای زبان پایتون چیه؟ البته این سوال یکم سلیقه ای هست ولی برای کاربرد خاص اگر بخوایم محیطی طراحی سمت کاربر برای برنامه های دسکتاپ تولید کنیم. (شبیه ویژوال استودیو) کد نویسی درچه محیطی رو توصیه می کنید و چه کتابخونه و یا فریم

وروی رو پیسبهد سی دید.

با تشکر

• پاسخ • به اشتراک گذاری >

شاهین ک • Saeid Dadkhah • ۲ ماه پیش

<https://www.jetbrains.com/pycharm/> - PyCharm

<https://code.visualstudio.com/> - VsCode

• پاسخ • به اشتراک گذاری >

---

حریم خصوصی

دیسکاس را به وبگاه خود اضافه کنید افزودن دیسکاس افزودن

مشترک شوید

ت