

روش سخت یادگیری

# Ruby

راب سویرز، زد ای شاو

برگردان: محسن مصطفی جوکار

انتشارات پندار پارس

سرشناسه	: شاول، زد Shaw, Zed
عنوان و نام پدیدآور	: روش سخت یادگیری Ruby / راب سوبرز، زدای. شاول؛ برگردان محسن مصطفی جوکار.
مشخصات نشر	: تهران : پندار پارس، ۱۳۹۶.
مشخصات ظاهری	: ۱۹۰ص.
شابک	: 978-600-8201-32-8 : ۱۵۰۰۰۰ ریال
وضعیت فهرست نویسی	: فیبا
یادداشت	: عنوان اصلی: Learn Ruby the hard way, 2013.
موضوع	: روبی ( زبان برنامه نویسی کامپیوتر )
موضوع	: Ruby (Computer program language)
موضوع	: نرم افزار کاربردی -- طراحی و توسعه
موضوع	: Application software -- Development
شناسه افزوده	: سوبرز، راب
شناسه افزوده	: Sobers, Rob
شناسه افزوده	: مصطفی جوکار، محسن، -، ۱۳۶۵، مترجم
رده بندی کنگره	: ۷۲/۷۶۰۴ ۱۳۹۶ ۲ش ۹ر /
رده بندی دیویی	: ۰۰۵/۱۱۷
شماره کتابشناسی ملی	: ۴۶۴۳۲۲۵

#### انتشارات پندار پارس



دفتر فروش: انقلاب، ابتدای کارگر جنوبی، کوی رشتچی، شماره ۱۴، واحد ۱۶ [www.pendarepars.com](http://www.pendarepars.com)  
 تلفن: ۶۶۵۷۲۳۳۵ - تلفکس: ۶۶۹۲۶۵۷۸ همراه: ۰۹۱۲۲۴۵۲۳۴۸ [info@pendarepars.com](mailto:info@pendarepars.com)



نام کتاب	: روش سخت یادگیری Ruby
ناشر	: انتشارات پندار پارس
تالیف	: راب سوبرز، زد ای شاول
برگردان	: محسن مصطفی جوکار
چاپ نخست	: فروردین ۹۶
شمارگان	: ۵۰۰ نسخه
طرح جلد	: سارا یعسوبی
چاپ، صحافی	: روز

قیمت: ۱۵۰۰۰ تومان      شابک: ۹۷۸-۶۰۰-۸۲۰۱-۳۲-۸



\* هرگونه کپی برداری، تکثیر و چاپ کاغذی یا الکترونیکی از این کتاب بدون اجازه ناشر تخلف بوده و پیگرد قانونی دارد \*

## فهرست

۲	پیش‌گفتار
۵	تمرین شماره صفر: نصب
۵	Mac OSX
۶	OSX: آنچه که باید ببینید
۷	Windows
۸	Windows: آنچه که باید ببینید
۹	Linux
۱۱	هشدار برای مبتدی‌ها
۱۳	تمرین ۱: نخستین برنامه خوب
۱۴	تمرین اضافی
۱۵	تمرین ۲: توضیحات و کاراکترهای پوند
۱۶	تمرین ۳: اعداد و ریاضی
۱۸	تمرین ۴: متغیرها و نام‌ها
۱۹	تمرین اضافی
۲۰	تمرین ۵: متغیرهای بیشتر و چاپ کردن
۲۱	تمرین اضافی
۲۲	تمرین ۶: رشته‌ها و متن
۲۴	تمرین ۷: چاپ بیشتر
۲۶	تمرین ۷: چاپ، چاپ، چاپ
۲۶	تمرین اضافی
۲۷	تمرین ۹: چاپ، چاپ، چاپ
۲۷	تمرین اضافی
۲۸	تمرین ۱۰: چی بود؟
۲۹	تمرین اضافی
۳۰	تمرین ۱۱: پرسش پرسیدن
۳۲	تمرین ۱۲: کتابخانه‌ها
۳۲	صبر کنید! ویژگی‌ها نام دیگری هم دارند
۳۳	تمرین ۱۳: پارامترها، آنپک کردن، متغیرها
۳۵	تمرین ۱۴: اعلان و رد کردن
۳۷	تمرین ۱۵: اعلان و رد کردن
۴۰	تمرین ۱۶: خواندن و نوشتن فایل‌ها
۴۳	تمرین ۱۷: فایل‌های بیشتر
۴۵	تمرین ۱۸: نام‌ها، متغیرها، کد، توابع
۴۹	تمرین ۱۹: توابع و متغیرها
۵۱	تمرین ۲۰: توابع و متغیرها
۵۳	تمرین ۲۱: توابع می‌توانند چیزی را برگردانند

تمرین ۲۱: تاکنون چه چیزهایی یاد گرفتید؟	۵۶
تمرین ۲۳: خواندن یکسری کد	۵۸
تمرین ۲۴: تمرین بیشتر	۶۰
تمرین ۲۵: بازهم تمرین بیشتر	۶۲
تمرین ۲۶: تبریک می‌گوییم، امتحان کنید!	۶۶
تمرین ۲۷: منطق حفظ کردن	۶۸
اصطلاح درستی	۶۹
جداول درستی	۶۹
تمرین ۲۸: تمرین عبارتهای منطقی	۷۲
تمرین ۲۹: If چیست	۷۵
تمرین ۳۰: If و Else	۷۷
تمرین ۳۱: تصمیم‌گیری	۷۹
تمرین ۳۲: حلقه‌ها و آرایه‌ها	۸۲
تمرین ۳۳: حلقه‌های While	۸۵
تمرین ۳۴: دسترسی به عناصر آرایه‌ها	۸۸
تمرین ۳۵: شاخه‌ها و توابع	۹۱
تمرین ۳۶: طراحی و اشکال‌یابی	۹۴
قوانینی برای If-Statements	۹۴
قوانینی برای حلقه‌های For	۹۵
نکاتی برای اشکال‌یابی	۹۵
مشق شب	۹۵
تمرین ۳۷: بررسی سمبول	۹۷
واژگان کلیدی	۹۷
انواع داده	۹۸
تمرین ۳۸: خواندن کد	۱۰۱
تمرین اضافی	۱۰۲
تمرین ۳۹: انجام چیزهایی برای آرایه	۱۰۳
تمرین ۴۰: دیکشنری‌ها، آه دیکشنری‌های دوست داشتنی	۱۰۶
تمرین ۴۱: Gothons From Planet Percal #25	۱۰۹
تمرین ۴۲: Gothons طبقه‌بندی می‌شوند	۱۱۷
تمرین ۴۳: یک بازی بسازید	۱۲۴
تمرین ۴۴: ارزیابی بازی شما	۱۲۵
سبک تابع	۱۲۵
سبک کلاس	۱۲۶
سبک کد	۱۲۶
توضیحات خوب	۱۲۷
ارزیابی بازی شما	۱۲۸
تمرین ۴۵: یک...است، یک...دارد، Classes و Objects	۱۲۹

۱۳۳	تمرین ۴۶: یک اسکلت پروژه .....
۱۳۳	محتویات اسکلت: Linux/OSX .....
۱۳۴	نصب Gems .....
۱۳۵	استفاده از اسکلت .....
۱۳۶	امتحان لازم .....
۱۳۷	تمرین ۴۷: تست خودکار .....
۱۳۷	نوشتن یک مورد برای تست (Test Case) .....
۱۳۹	دستورالعمل‌های تست کردن .....
۱۴۱	تمرین ۴۸: ورودی پیشرفته کاربر .....
۱۴۱	واژه‌نامه بازی ما .....
۱۴۲	شکستن یک عبارت .....
۱۴۲	ساختار واژگان .....
۱۴۳	اسکن کردن ورودی .....
۱۴۳	استثناها و اعداد .....
۱۴۴	چیزی که باید تست کنید .....
۱۴۵	نکات طراحی .....
۱۴۷	تمرین ۴۹: ساخت جملات .....
۱۴۸	تطبیق و زیرچشمی نگاه کردن .....
۱۴۹	دستور زبان جملات .....
۱۵۱	دستور زبان جملات .....
۱۵۳	تمرین ۵۰: نخستین وب سایت شما .....
۱۵۳	نصب Sinatra .....
۱۵۴	ساخت یک پروژه ساده "Hello World" .....
۱۵۵	چه خبره .....
۱۵۶	چه خبره .....
۱۵۶	ایجاد قالب‌های اساسی .....
۱۵۹	تمرین ۵۱: گرفتن ورودی از یک مرورگر .....
۱۵۹	وب چگونه کار می‌کند .....
۱۶۰	مرورگر .....
۱۶۰	آدرس .....
۱۶۱	ارتباط .....
۱۶۱	درخواست .....
۱۶۲	سرور .....
۱۶۲	پاسخ .....
۱۶۲	فرم‌ها چگونه کار می‌کنند .....
۱۶۴	ساخت فرم‌های HTML .....
۱۶۶	ایجاد یک قالب طرح‌بندی .....
۱۶۷	نوشتن تست‌های خودکارسازی شده برای فرم‌ها .....

---

---

۱۷۱.....	تمرین ۵۲: شروع بازی تحت وب شما
۱۷۱.....	اصلاح بازی تمرین ۴۲
۱۷۶.....	نشست‌ها و ردیابی کاربران
۱۷۷.....	ساخت یک موتور
۱۷۹.....	امتحان پایانی شما
۱۸۱.....	گام‌های بعدی
۱۸۲.....	نصیحتی از یک برنامه‌نویس قدیمی

به روش سخت یاد گرفتن روبی خوش آمدید. این ترجمه‌ای از "روش سخت یاد گرفتن پایتون" برای آموزش روبی به مبتدیان است. این کتاب به همان سبک کتاب پایتون است و محتویات تقریباً مشابه هم هستند اما این کتاب زبان روبی را به شما آموزش می‌دهد. در صورت وجود هرگونه اشکال به آدرس [help@learncodethehardway.org](mailto:help@learncodethehardway.org) ایمیل بزنید.

## پیش‌گفتار مترجم

کتاب "سخت‌ترین روش یاد گرفتن روبی" برعکس نامش راحت‌ترین روش یاد گرفتن این زبان است. نویسنده اصلی که خود یک برنامه‌نویس با سابقه است این نام را عمداً انتخاب کرده. اما چرا روبی؟ برای پاسخ دادن به این پرسش می‌توان دلایل زیادی آورد؛ اما من تنها به چند مورد آن بسنده می‌کنم:

۱. روبی مستقل از پلتفرم است، یعنی برنامه شما روی تمام سیستم‌های عامل قابل اجرا است.
۲. در روبی هر چیزی یک شیء محسوب می‌شود، پس برنامه‌نویسی شیء‌گرا در روبی عالی است.
۳. نگاهی به تکنولوژی‌های وب بیاندازید، حتماً به چیزی به اسم Ruby On Rails برخورد می‌کنید و باید بدانید که یادگرفتن روبی نخستین گام برای یادگیری Ruby On Rails است.
۴. روبی آینده است و مطمئن باشید با یادگیری این زبان، شغل مناسبی پیدا می‌کنید.
۵. همان‌گونه که می‌دانید حرفه‌ای همچون تست نفوذ برای خیلی‌ها جالب است و یکی از معروف‌ترین برنامه‌ها در زمینه هک، ابزار Metasploit است که با زبان روبی نوشته شده است. پس برای نوشتن ماژول یا توسعه یک ماژول برای Metasploit به زبان روبی نیاز دارید.
۶. روبی زیبا و ظریف است.

با این کتاب نمی‌توانید به یک برنامه‌نویس حرفه‌ای در روبی تبدیل شوید اما این کتاب می‌تواند شروع مناسبی برای یادگیری این زبان و خواندن کتاب‌های دیگر روبی باشد.

## پیش‌گفتار

### روش سخت، آسان‌ترین است

این کتاب ساده به معنای فراهم کردن شروع برنامه‌نویسی برای شماست. عنوان کتاب می‌گوید که این روش سخت یادگرفتن کدنویسی است اما در واقع این‌گونه نیست. این تنها روش "سخت" است چون این روشی است که مردم برای آموزش همه چیز از آن استفاده می‌کنند. با کمک این کتاب، چیزهای ساده که همه برنامه‌نویسان برای یادگرفتن یک زبان نیاز به انجام آن دارند را به طرز باورنکردنی انجام خواهید داد:

۱. هر تمرین را انجام دهید.

۲. هر نمونه را دقیقاً بنویسید.

۳. آنرا اجرا کنید.

خودشه. در آغاز این بسیار دشوار خواهد بود اما آنرا انجام دهید. اگر این کتاب را بخوانید و هر تمرین را برای یک یا دو ساعت در شب انجام دهید یک پایه خوب برای حرکت به کتاب‌های دیگر خواهید داشت. ممکن است از این کتاب واقعا "برنامه‌نویسی" را یاد نگیرید اما مهارت‌های اساسی که برای آغاز یادگیری یک زبان به آن نیاز دارید را فرا خواهید گرفت.

وظیفه این کتاب یاد دادن سه مهارت ضروری است که یک برنامه‌نویس در آغاز نیاز به دانستن آن دارد: خواندن و نوشتن، توجه به جزئیات، کشف تفاوت‌ها.

### خواندن و نوشتن

اگر یک مشکل تایپی داشته باشید آنگاه یک مشکل یادگیری کد خواهید داشت و این، به طرز واضحی ابلهانه به نظر می‌رسد. به‌ویژه اگر یک مشکل تایپی که مربوط به کاراکترهای نسبتاً عجیب و غریب در کد منبع است داشته باشید. بدون این مهارت ساده، حتی قادر به یادگیری اساسی‌ترین چیزها در مورد اینکه نرم‌افزار چگونه کار می‌کند نخواهید بود.

نوشتن کدهای نمونه و اجرای آنها به شما در یادگیری نام سمبل‌ها، آشنا شدن با تایپ کردن آنها و خواندن آن زبان کمک خواهد کرد.

### توجه به جزئیات

توجه به جزئیات، یک مهارت است که برنامه‌نویسان خوب را از برنامه‌نویسان بد جدا می‌کند. در واقع، این چیزی است که در هر حرفه‌ای خوب را از بد جدا می‌کند. بدون توجه به کوچکترین جزئیات



کارتان، عناصر کلیدی آنچه که ایجاد کرده‌اید را از دست خواهید داد. در برنامه‌نویسی این‌گونه است که با اشکالات و مشکلات استفاده از سیستم مواجه می‌شوید.

با تجربه‌ای که از این کتاب به دست می‌آورید و با کپی کردن دقیق هر نمونه، مغز خود را به تمرکز بر روی جزئیات آنچه که انجام می‌دهید آموزش خواهید داد.

### کشف تفاوت‌ها

یک مهارت بسیار مهم - که بیشتر برنامه‌نویسان در طول زمان آنرا توسعه می‌دهند - توانایی متوجه شدن تفاوت‌های میان چیزها به صورت بصری است. یک برنامه‌نویس با تجربه می‌تواند دو تکه از کد را که تفاوت بسیار کمی با هم دارند بردارد و بی‌درنگ به نقطه شروع تفاوت اشاره کند. برنامه‌نویسان ابزارهایی را اختراع کرده‌اند که این کار را، حتی ساده‌تر می‌کند اما ما از هیچکدام از آنها استفاده نخواهیم کرد. نخست مغز خود را به روش سخت آموزش دهید سپس می‌توانید از ابزارها استفاده کنید.

چنانچه هر یک از این تمرین‌ها را انجام می‌دهید و هر یک از آنها را تایپ می‌کنید، اشتباه‌هایی مرتکب خواهید شد. این گریزناپذیر است، حتی برنامه‌نویسان باتجربه هم، چنین اشتباه‌هایی می‌کنند. کار شما، مقایسه آنچه که نوشته‌اید با آنچه که نیاز است و برطرف کردن تمام تفاوت‌ها است. با انجام این کار به خودتان آموزش می‌دهید که متوجه اشکال‌ها و دیگر مشکل‌ها باشید.

### Copy-Paste نکنید

باید هر کدام از این تمرین‌ها را به صورت دستی تایپ کنید. اگر کپی و پیست کنید ممکن است حتی آنها را انجام هم ندهید. هدف این تمرین‌ها آموزش دادن دست‌ها، مغز و ذهن شماست که چگونه کد را بخوانید، بنویسید و ببینید. اگر کپی-پیست کنید خودتان را گول زده‌اید و درس‌ها اثربخشی ندارند.

### توجه به تمرین و پشتکار

اگر شما برنامه‌نویسی را یاد می‌گیرید من هم مشغول آموختن گیتار زدن هستم. هر روز به مدت دست‌کم ۲ ساعت گیتار زدن را تمرین می‌کنم. دست‌کم یک ساعت گام، آکورد و آرپژ را می‌زنم و سپس تئوری موسیقی، گوش دادن، آهنگ و هر چیز دیگری که بتوانم را یاد می‌گیرم. برخی از روزها در مورد گیتار و موسیقی به مدت ۸ ساعت مطالعه می‌کنم چون آنرا دوست دارم و برایم جالب است. برایم تمرین، تکراری عادی است و تنها، چگونگی یاد گرفتن چیزی است. می‌دانم برای اینکه هر چیزی را به خوبی یاد بگیرید باید هر روز تمرین کنید حتی اگر آنرا به خوبی نتوانید انجام دهید (که بیشتر همین‌گونه است) و یا آن سخت باشد. به تلاش ادامه دهید و در پایان، آن آسان‌تر و سرگرم‌کننده می‌شود.

همان‌گونه که این کتاب را مطالعه می‌کنید و برنامه‌نویسی را دنبال می‌کنید به یاد داشته باشید که انجام دادن هرچیزی در آغاز سخت است. شاید از آن دسته افرادی باشید که از شکست وحشت دارند بنابراین در نخستین نشانه از سختی، منصرف می‌شوند. شاید هرگز انضباط شخصی را یاد نگرفته باشید بنابراین نمی‌توانید هر چیزی که "خسته کننده" است را انجام دهید. ممکن است برخی از افراد به شما گفته باشند که "استعداد" دارید بنابراین تلاشی در مورد هرچیزی که از نظر شما احمقانه باشد یا غیرعادی نباشد را انجام نمی‌دهید. ممکن است رقابت را دوست داشته باشید اما شاید مقایسه کردن خودتان با کسی مانند من که بیش از ۲۰ سال است برنامه‌نویسی می‌کند ناعادلانه باشد.

هرچیزی که دلیل شما برای تسلیم شدن است را کنار بگذارید و به تلاش ادامه دهید. خودتان را مجبور کنید. اگر کارهای اضافی دارید که نمی‌توانید انجام دهید یا یک درس را متوجه نمی‌شوید از آن چشم‌پوشی کنید و بعدها به آن برگردید. تنها به تلاشتان ادامه دهید چون با برنامه‌نویسی چیزهای بسیار عجیب و غریبی روی می‌دهد.

در آغاز، هیچ چیزی را درک نمی‌کنید. آن عجیب و غریب خواهد بود درست مانند یادگرفتن هر زبان انسانی دیگر. با کلمات کشمکش خواهید داشت و نمی‌دانید که سمبول‌ها چه هستند و آن بسیار گیج کننده خواهد بود. سپس یک روز برای مغز شما رویدادی رخ می‌دهد و ناگهان "آن‌ها را خواهید فهمید". اگر انجام تمرین‌ها را ادامه دهید و تلاش کنید که آنها را درک کنید موفق خواهید شد. ممکن است یک استاد برنامه‌نویسی نشوید اما دست‌کم درک خواهید کرد که برنامه‌نویسی چگونه کار می‌کند.

اگر منصرف شوید آنگاه هرگز به این نقطه نمی‌رسید. به نخستین چیز گیج کننده (که در آغاز همه چیز گیج کننده است) که برخورد کنید توقف می‌کنید. اگر به تلاش تکیه کردن و تلاش برای درک و خواندن در مورد آن ادامه دهید، در پایان آنرا درک خواهید کرد.

اما اگر همه این کتاب را خواندید و هنوز هم نمی‌دانید که چگونه برنامه بنویسید، دست‌کم تلاشی انجام داده‌اید. می‌توانید بگویید که تلاشتان را کرده‌اید و جوابی نگرفته‌اید اما دست‌کم تلاشتان را کرده‌اید. می‌توانید به آن افتخار کنید.

## تمرین شماره صفر: نصب

این تمرین هیچ کدنویسی ندارد. این تمرین با انجام اینکه کامپیوترتان بتواند روبی را اجرا کند به سادگی کامل می‌شود. باید این دستورها را دقیقاً دنبال کنید.

این آموزش فرض می‌کند که هم‌اکنون از روبی نسخه 1.9.2 استفاده می‌کنید.

ممکن است هم‌اکنون روبی بر روی سیستم شما نصب شده باشد. یک کنسول باز کنید و دستور زیر را اجرا کنید:

```
$ ruby -v
ruby 1.9.2
```

اگر هم‌اینک روبی بر روی سیستم شما نصب نشده است، بدون در نظر گرفتن اینکه از کدام سیستم عامل استفاده می‌کنید به شدت توصیه می‌کنم که از Ruby Version Manager (RVM) برای نصب آن استفاده کنید.

## Mac OSX

برای کامل کردن این تمرین، گام‌های زیر را دنبال کنید:

۱. به آدرس <https://wiki.gnome.org/Apps/Gedit#Download> بروید، ویرایشگر متن Gedit را دریافت و آنرا نصب کنید.
۲. برای اینکه بتوانید به راحتی به آن دسترسی داشته باشید Gedit (ویرایشگر) را در Dock قرار دهید.
  - (a) برنامه gedit را اجرا کنید بنابراین می‌توانیم برخی از پیش‌فرض‌های احمقانه آنرا تغییر دهیم.
  - (b) از منوی برنامه gedit قسمت Preferences را باز کنید و برگه Editor را انتخاب کنید.
  - (c) Tab width را به ۲ تغییر دهید.
  - (d) Insert spaces instead of tabs را انتخاب کنید (اطمینان یابید که مارک‌دار شده است).
  - (e) همچنین "Automatic indentation" را فعال کنید.
  - (f) برگه View را برای فعال کردن "Display line numbers" باز کنید.
۳. برنامه "Terminal" را پیدا کنید. آنرا جست‌وجو کنید. پیدایش خواهید کرد.
۴. همچنین برنامه "Terminal" را در Dock قرار دهید.

۵. برنامه "Terminal" را اجرا کنید. چنان خوب به نظر نمی‌رسد.
۶. در برنامه "Terminal" دستور irb (روبی تعاملی) را اجرا کنید. در Terminal می‌توانید همه چیز را تنها با نوشتن نام آنها و زدن کلید Enter اجرا کنید.
- (a) اگر irb را اجرا کردید و وجود نداشت (irb تشخیص داده نشد) آنگاه آنرا با استفاده از Ruby Version Manager (RVM) نصب کنید.
۷. کلیدهای CTRL-D (^D) را فشار دهید و از irb خارج شوید.
۸. باید به اعلان دستوری مانند به آنچه پیش از تایپ کردن irb وجود داشت برگردید. دلیلش را پیدا کنید.
۹. نحوه ساختن یک دایرکتوری در Terminal را یاد بگیرید. برای کمک، جست‌وجوی آنلاین را انجام دهید.
۱۰. نحوه وارد شدن به دایرکتوری که ایجاد کرده‌اید را از طریق Terminal یاد بگیرید. دوباره جست‌وجوی آنلاین را انجام دهید.
۱۱. از ویرایشگر مورد نظرتان برای ساخت یک فایل در این دایرکتوری استفاده کنید. فایل را ایجاد کنید، "Save" یا "Save As" را بزنید و دایرکتوری که ایجاد کرده‌اید را انتخاب کنید.
۱۲. با استفاده از کیبورد و سوئیچ کردن میان پنجره‌ها به Terminal برگردید. اگر آنرا بلد نیستید یاد بگیرید.
۱۳. به Terminal برگردید، ببینید آیا می‌توانید دایرکتوری را برای فایلی که به تازگی ایجاد کرده‌اید لیست کنید. وب را برای نحوه لیست کردن یک دایرکتوری جست‌وجو کنید.

**نکته:** اگر با Gedit مشکل دارید که معمولاً ممکن است به دلیل کیبورد غیرانگلیسی باشد پیشنهاد می‌کنم به جای آن Textwrangler که در آدرس زیر است را امتحان کنید:

<http://www.barebones.com/products/textwrangler>

## OSX: آنچه که باید ببینید

در اینجا چیزی که در ترمینال کامپیوترم انجام داده‌ام را می‌بینید. کامپیوتر شما متفاوت خواهد بود بنابراین می‌توانید تفاوت‌ها میان آنچه که من انجام داده‌ام و آنچه که شما باید انجام دهید را پیدا کنید.

Last login: Sat Apr 24 00:56:54 on ttys001

~ \$ irb

```

ruby-1.9.2-p180 :001 >
ruby-1.9.2-p180 :002 > ^D
~ $ mkdir mystuff
~ $ cd mystuff
mystuff $ ls
# ... Use Gedit here to edit test.txt....
mystuff $ ls
test.txt
mystuff $

```

## Windows

نکته: نوشته شده توسط zhmark

۱. به وب سایت <https://wiki.gnome.org/Apps/Gedit#Download> بروید و ویرایشگر متن gedit را دریافت و آنرا نصب کنید. برای انجام این کار نیازی به داشتن حساب کاربری مدیر ندارید.
۲. اطمینان یابید که با قرار دادن gedit در دستکاپ و/یا Quick Launch راحت هستید. هر دو گزینه در طول راهاندازی در دسترس هستند.
  - (a) برنامه gedit را اجرا کنید؛ بنابراین می‌توانیم برخی از تنظیمات پیش فرض احمقانه را تغییر دهیم.
  - (b) برگه Editor را انتخاب و Edit->Preferences را باز کنید.
  - (c) Tab width را به ۲ تغییر دهید.
  - (d) Insert spaces instead of tabs را انتخاب کنید (اطمینان یابید که علامت‌دار شده است).
  - (e) همچنین "Automatic indentation" را فعال کنید.
  - (f) برگه View را باز کنید و "Display line numbers" را فعال کنید.
۳. برنامه Terminal را پیدا کنید. در ویندوز آن Command Prompt نامیده می‌شود. روش دیگر این است که تنها cmd را اجرا کنید.
۴. برای راحتی، یک میانبر در دستکاپ و/یا Quick Launch بسازید.
۵. برنامه Terminal را اجرا کنید. چندان خوب به نظر نمی‌رسد.
۶. در برنامه Terminal دستور irb (روبی تعاملی) را اجرا کنید. در ترمینال می‌توانید همه چیز را تنها با نوشتن نام آنها و زدن کلید Enter اجرا کنید.

- (a) اگر irb را اجرا کنید و آن وجود نداشته باشد (irb تشخیص داده نشد) آنگاه با استفاده از Ruby Version Manager (RVM) آنرا نصب کنید.
۷. کلیدهای CTRL-Z (^Z) را فشار دهید و از محیط irb خارج شوید.
۸. باید به اعلانی شبیه به پیش از تایپ کردن irb برگردید. دلیلش را پیدا کنید.
۹. نحوه ساختن یک دایرکتوری در ترمینال را یاد بگیرید. برای کمک جست‌وجوی آنلاین را انجام دهید.
۱۰. نحوه وارد شدن به دایرکتوری که ایجاد کرده‌اید را از طریق ترمینال یاد بگیرید. دوباره جست‌وجوی آنلاین را انجام دهید.
۱۱. از ویرایشگر برای ساخت یک فایل در این دایرکتوری استفاده کنید. فایل را بسازید، "Save" or "Save As". را بزنید و دایرکتوری که ایجاد کرده‌اید را انتخاب کنید.
۱۲. از کیبورد برای سوئیچ کردن میان پنجره‌ها و برگشتن به ترمینال استفاده کنید. اگر آنرا بلد نیستید یاد بگیرید.
۱۳. به ترمینال برگردید، ببینید که آیا می‌توانید دایرکتوری را برای فایلی که ایجاد کرده‌اید لیست کنید. وب را برای نحوه لیست کردن یک دایرکتوری جست‌وجو کنید.

**هشدار:** ویندوز برای رومی بزرگترین مشکل است. گاهی رومی را نصب می‌کنید و یک کامپیوتر مشکلی ندارد اما بقیه کامپیوترها فاقد ویژگی‌های مهم هستند. اگر هرگونه مشکلی دارید، لطفاً وب سایت <http://rubyinstaller.org> را ببینید.

## Windows: آنچه که باید ببینید

```
C:\Documents and Settings\you>irb
ruby-1.9.2-p180 :001 >
ruby-1.9.2-p180 :001 > ^Z
C:\Documents and Settings\you>mkdir mystuff
C:\Documents and Settings\you>cd mystuff
... Here you would use gedit to make test.txt in mystuff ...
C:\Documents and Settings\you\mystuff>
<bunch of unimportant errors if you installed it as non-admin - ignore them - hit Enter>
C:\Documents and Settings\you\mystuff>dir
Volume in drive C is
Volume Serial Number is 085C-7E02
Directory of C:\Documents and Settings\you\mystuff
04.05.2010 23:32 <DIR> .
```

```
04.05.2010 23:32 <DIR> ..
04.05.2010 23:32 6 test.txt
1 File(s) 6 bytes
2 Dir(s) 14 804 623 360 bytes free
C:\Documents and Settings\you\mystuff<
```

احتمالا یک اعلان بسیار متفاوت، اطلاعات مربوط به روبي و چیزهای دیگر را خواهید دید اما این یک مفهوم دارد. اگر سیستم شما متفاوت است به ما اطلاع دهید آنرا درست خواهیم کرد.

## Linux

لینوکس یک سیستم عامل متنوع با یک دسته از راه‌های گوناگون برای نصب نرم‌افزار است. با فرض اینکه لینوکس را اجرا می‌کنید و می‌خواهید بدانید که چگونه می‌توانید بسته‌های نرم‌افزاری را نصب کنید. در اینجا دستورالعمل‌هایی برای این کار وجود دارد:

۱. به آدرس <https://wiki.gnome.org/Apps/Gedit#Download> بروید و ویرایشگر متن gedit را دریافت و نصب کنید.
۲. اطمینان یابید که می‌توانید به راحتی gedit را در منوی مدیر پنجره قرار دهید.
  - (a) برنامه gedit را اجرا کنید، بنابراین می‌توانیم برخی از تنظیمات پیش فرض احمقانه را تغییر دهیم.
  - (b) Preferences را باز کرده و برگه Editor را انتخاب کنید.
  - (c) Tab width را به 2 تغییر دهید.
  - (d) Insert spaces instead of tabs را انتخاب کنید (اطمینان یابید که علامت‌دار شده است).
  - (e) همچنین "Automatic indentation" را فعال کنید.
  - (f) برگه View را باز کنید و "Display line numbers" را فعال کنید.
۳. برنامه Terminal را پیدا کنید. در محیط GNOME آن Terminal، Konsole، یا xterm نامیده می‌شود.
۴. همچنین Terminal را در Dock قرار دهید.
۵. برنامه ترمینال را اجرا کنید. چندان خوب به نظر نمی‌رسد.
۶. در برنامه ترمینال irb (روبی تعاملی) را اجرا کنید. در ترمینال می‌توانید همه چیز را تنها با نوشتن نام آن‌ها و زدن کلید Enter اجرا کنید.

- (a) اگر irb را اجرا کنید و آن وجود نداشته باشد (irb تشخیص داده نشد) آنگاه با استفاده از Ruby Version Manager (RVM) آنرا نصب کنید.
۷. کلیدهای CTRL-D (^D) را فشار دهید و از محیط irb خارج شوید.
۸. باید به چیزی مانند پیش از تایپ کردن irb برگردید. دلیلش را پیدا کنید.
۹. چگونگی ایجاد کردن یک دایرکتوری در ترمینال را یاد بگیرید. برای کمک، اینترنت را جست‌وجو کنید.
۱۰. چگونگی وارد شدن به دایرکتوری از طریق ترمینال را یاد بگیرید. دوباره اینترنت را برای کمک جست‌وجو کنید.
۱۱. از ویرایشگر برای ساخت یک فایل در این دایرکتوری استفاده کنید. به‌طور معمول فایل را ایجاد خواهید کرد، "Save" or "Save As" را بزنید و دایرکتوری که ایجاد کرده‌اید را انتخاب کنید.
۱۲. با استفاده از کیبورد و سوئیچ کردن میان پنجره‌ها به ترمینال برگردید. اگر آنرا بلد نیستید یاد بگیرید.
۱۳. به ترمینال برگردید تا ببینید که آیا می‌توانید دایرکتوری را برای فایلی که به تازگی ایجاد کرده‌اید لیست کنید. اینترنت را برای چگونگی لیست کردن دایرکتوری جست‌وجو کنید.

### Linux: آنچه که باید ببینید:

```
$ irb
ruby-1.9.2-p180 :001 >
ruby-1.9.2-p180 :002 > ^D
$ mkdir mystuff
$ cd mystuff
# ... Use gedit here to edit test.txt ...
$ ls
test.txt
$
```

احتمالا یک اعلان بسیار متفاوت، اطلاعات در مورد رومی و چیزهای دیگر را خواهید دید اما این یک مفهوم دارد.



## هشدار برای مبتدی‌ها

این تمرین‌ها را انجام دهید. بسته به آشنایی شما با کامپیوترتان این تمرین‌ها ممکن است سخت باشد. اگر سخت است برای خواندن و مطالعه کردن زمان بگذارید و آنها را انجام دهید چون تا زمانی که این چیزهای بسیار ابتدایی برای شما دشوار است برنامه‌نویسی را به خوبی نمی‌توانید انجام دهید.

اگر یک برنامه‌نویس به شما گفت که از vim یا emacs استفاده کنید آنگاه به او بگویید "نه"، این ویرایشگرها برای زمانی هستند که به یک برنامه‌نویس بهتر تبدیل شدید. هم‌اکنون همه چیزی که احتیاج دارید یک ویرایشگر است که به شما اجازه دهد متن را در یک فایل قرار دهید. ما از gedit استفاده خواهیم کرد چون ساده است و بر روی تمام کامپیوترها یکی است. برنامه‌نویسان حرفه‌ای از gedit استفاده می‌کنند؛ پس آن به اندازه کافی برای آغاز خوب خواهد بود.

در پایان یک برنامه‌نویس به شما خواهد گفت که از Linux یا Mac OSX استفاده کنید. اگر برنامه‌نویس فونت‌ها و تایپوگرافی را دوست داشته باشد به شما می‌گوید که یک کامپیوتر Mac OSX بگیرید؛ اما اگر آنها کنترل و داشتن یک ریش دراز را دوست داشته باشند به شما می‌گویند که Linux را نصب کنید. باز هم از هر کامپیوتری که هم‌اینک دارید، استفاده کنید. تمام چیزی که احتیاج دارید، gedit، یک ترمینال و رومی است.

### نکته مترجم:

استفاده از PC همراه با سیستم عامل لینوکس خیلی بهتر از خرید یک کامپیوتر Mac با آن همه محدودیت است. کامپیوترهای Mac در واقع همان PC هستند که تنها از سیستم عاملی از خانواده BSD با نام OSX استفاده می‌کنند و چیزی جز اتلاف هزینه نیستند. با نگاه کردن به یک کامپیوتر Mac خواهید دید که تمام قطعات سخت‌افزاری همان قطعات PC هستند و تنها سیستم عامل آن متفاوت است. با داشتن یک PC امکان نصب تمام سیستم عامل را خواهید داشت و در صورت نیاز به ارتقاء یا برطرف کردن مشکل سخت‌افزاری، خودتان به راحتی می‌توانید این کار را انجام دهید اما در مورد محصولات Apple باید به Apple Store مراجعه کنید. برای دانستن اینکه چرا نباید از محصولات Apple استفاده کنید مقاله‌ای که توسط Richard Matthew Stallman نوشته شده و در آدرس <https://stallman.org/apple.html> وجود دارد را مطالعه کنید.

در پایان هدف از این راه‌اندازی این است که چنانچه بر روی تمرین‌ها کار می‌کنید بتوانید سه چیز را به‌صورت بسیار قابل اعتماد انجام دهید:

۱. نوشتن تمرین‌ها با استفاده از gedit.

۲. اجرای تمرین‌هایی که نوشته‌اید.

۳. درست کردن آنها زمانی که مشکل دارند.

۴. تکرار.

هرچیز دیگری تنها شما را گیج می‌کند؛ پس برنامه را ادامه دهید.

## تمرین ۱: نخستین برنامه خوب

به یاد داشته باشید که برای یادگیری چگونگی نصب یک ویرایشگر متن، اجرای ویرایشگر متن، اجرای ترمینال و کار کردن با هر دوی آنها، زمان مناسبی را صرف تمرین صفر کردید. اگر آنها را انجام ندهاید پس ادامه ندهید. شانس خوبی را نخواهید داشت. این تنها باری است که یک تمرین را با هشدار دادن به شما که نباید از چیزی رد شوید یا جلوتر از خودتان حرکت کنید شروع می‌کنم.

```
puts "Hello World!"
puts "Hello Again"
puts "I like typing this."
puts "This is fun."
puts 'Yay! Printing.'
puts "I'd much rather you 'not'."
puts 'I "said" do not touch this.'
```

خطوط بالا را در یک فایل با نام ex1.rb بنویسید و ذخیره کنید. مهم است که بدانید رومی با یک فایل با پسوند .rb بهتر کار می‌کند. سپس در ترمینال، فایل را با اجرای دستور زیر اجرا کنید:

```
ruby ex1.rb
```

اگر این کار را به درستی انجام داده باشید باید خروجی مانند چیزی که در زیر نشان داده‌ام را ببینید. اگر چنین چیزی را نبینید آنگاه چیزی را به اشتباه انجام داده‌اید. نه، کامپیوتر اشتباه نمی‌کند.

### چیزی که باید ببینید:

```
$ ruby ex1.rb
Hello World!
Hello Again
I like typing this.
This is fun.
Yay! Printing.
I'd much rather you 'not'.
I "said" do not touch this.
```

```
$
```

ممکن است نام دایرکتوری را پیش از \$ ببینید که خوب است اما اگر خروجی دقیقا همان چیزی نیست که در بالا می‌بینید دلیلش را پیدا و آنرا برطرف کنید.

اگر یک خطا مانند زیر را می‌بینید:

```
ruby ex1.rb
ex1.rb:4: syntax error, unexpected tCONSTANT, expecting $end
```

```
puts "This is fun."
```

```
^
```

از آنجا که بسیاری از این اشتباه‌ها را خواهید داشت مهم است که بتوانید آنها را بخوانید. حتی من هم بسیاری از این اشتباه‌ها را مرتکب می‌شوم. بیایید خط به خط به آن نگاه کنیم.

۱. در اینجا برای اجرا کردن اسکریپت ex1.rb دستور را در ترمینال اجرا می‌کنیم.
۲. سپس رویی به ما می‌گوید که فایل ex1.rb در خط 4 یک خطا دارد.
۳. سپس این خط را برای ما چاپ می‌کند.
۴. سپس یک علامت <sup>^</sup> (هشتک) را در نقطه‌ای که خطا وجود دارد قرار می‌دهد.
۵. در پایان، یک خطای "syntax error" را چاپ می‌کند و به ما چیزی را در مورد آنچه که ممکن است خطا باشد می‌گوید. معمولاً این خطاها بسیار مرموز هستند و اگر آنها را در یک موتور جست‌وجو کپی کنید افراد دیگری را پیدا می‌کنید که پیش‌تر با این خطا مواجه شده بودند و می‌توانید راه برطرف کردن آنرا پیدا کنید.

## تمرین اضافی

همچنین تمرین اضافی هم دارید. تمرین اضافی شامل چیزهایی است که باید سعی کنید انجام دهید. اگر نمی‌توانید، از آن چشم‌پوشی کنید و دوباره برگردید.

برای این تمرین، کارهای زیر را انجام دهید:

۱. اسکریپت را طوری تغییر دهید که خط دیگری را چاپ کند.
۲. اسکریپت را طوری تغییر دهید که تنها یکی از خطوط را چاپ کند.
۳. یک کاراکتر # (octothorpe) را در ابتدای یک خط قرار دهید. این کاراکتر چه کاری انجام می‌دهد؟ سعی کنید بفهمید این کاراکتر چه کاری انجام می‌دهد.
۴. از این پس، نمی‌خواهم توضیح بدهم که هر تمرین چگونه کار می‌کند مگر اینکه یک تمرین متفاوت باشد.

---

**نکته:** کاراکتر octothorpe همچنین "پوند"، "هشتک"، "مش" یا نام‌های دیگر نامیده می‌شود. یک نام که با آن احساس راحتی می‌کنید را انتخاب کنید.

---