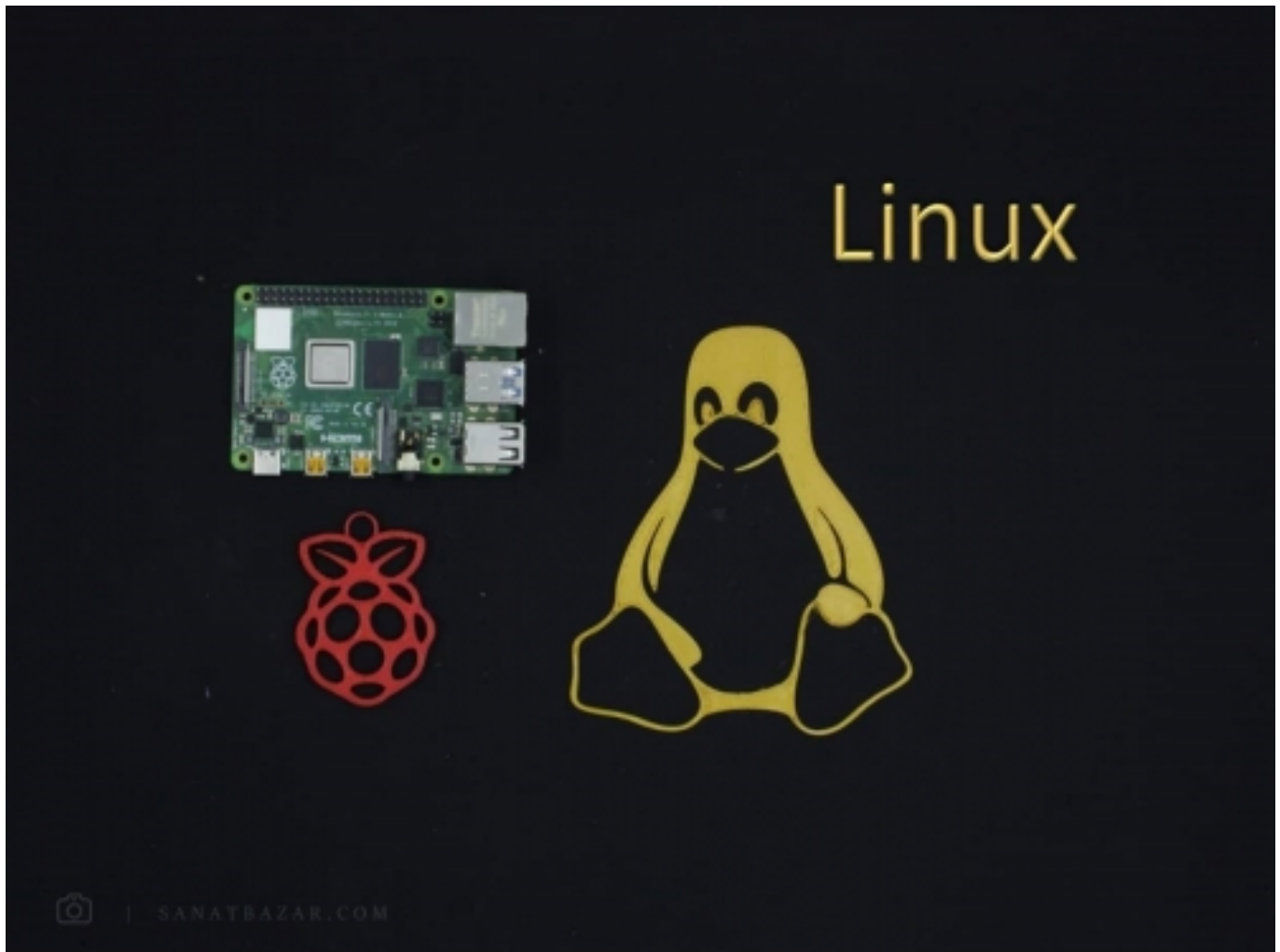


معرفی دستورات کاربردی لینوکس برای کار با رزبری پای اکامل و کاربردی



تعجب نکنید اگر بگم توسط لینوکس محاصره شدیم! سیستمعامل اندروید، ست آپ باکسها، دستگاههای ATM، سیستمهای هوشمند خودرو، سرورها و خیلی چیزهای دیگر مبتنی بر لینوکس اند. در اکثر حوزههای برنامه نویسی و مهندسی به خصوص وب سرور، مثل سیستمهای خانگی خیلی خبری از ویندوز نیست. حتی وقتی با چند تا از دوستانم که برای ادامه تحصیل به خارج از کشور ایلای کردن صحبت می کردم، فهمیدم دانشجویها در اکثر دانشگاه های مطرح دنیا فقط از لینوکس استفاده می کنند. پس کم کم به اهمیت این سیستمعامل پی بردم و شروع کردم به یادگیری. شاید اگر مثل من فقط تجربه ی کار با ویندوز را داشته باشید، شروع لینوکس سخت و پیچیده به نظر بیاد. (مخصوصاً با دیدن صفحه ی سیاه رنگی که کلی نوشته بالا و پایین می شه و ما رو به یاد هکرها تو فیلم پلیسیا می اندازد!) اما وقتی با لینوکس آشنا بشید، می بینید مثل همین پنگوئن ساده و نازه!

این مجموعه ی آموزشی به رزبری پای و پروژه های مربوط به آن می پردازد، اما باتوجه به این که اصول و دستورات لینوکس در تمام زمینه ها مشابه است، برای همه ی علاقه مندان به یادگیری می تواند مفید باشد. پس اگر قصد کار کردن با رزبری پای و اینترنت اشیا را دارید، یادگیری لینوکس از نون شب هم برای شما واجب تره! اگر هم با رزبری پای سر و کار ندارید و می خواهید لینوکس یاد بگیرید، باز هم این مطلب برای شماست. اصلاً اگر الکی این صفحه را باز کردید، قسمت بوده لینوکس یاد بگیرید. ضرر نمی کنید!!

در این بخش قصد دارم دستورات مهم و پایه ای برای کار با رزبری پای و لینوکس را به طور کامل به شما معرفی کنم. پس از تسلط بر مباحث اصلی، برای یادگیری بیشتر می توانید به کتابها و سایت های مرتبط مراجعه کنید. پس در ادامه با من همراه باشید تا هر آنچه در مورد لینوکس و کار با آن نیاز دارید، با مثال و به زبان ساده یاد بگیریم.

لینوکس چیست؟

وقتی عبارت سیستمعامل را می شنویم، ناخودآگاه به یاد ویندوز می افتیم. تعجبی هم ندارد، زیرا روی اکثر کامپیوترها و لپتاپها به صورت پیش فرض ویندوز

نصب شده است. در چند سال اخیر با فراگیر شدن استفاده از تبلت و موبایل و محصولات مختلف Apple، گوش‌های ما با نام سیستم‌عامل‌های دیگری مانند Android، IOS و OS X آشنا شد. حالا لینوکس هم مثل اسم‌های بالا یک نوع سیستم‌عامل است. یکی از پرکاربردترین آن‌ها! لینوکس در همه‌جا هست، از سیستم‌های شبکه تا خودرو و لوازم خانگی هوشمند، سیستم‌های اینترنت اشیا و حتی اندروید که بر پایه‌ی لینوکس نوشته شده!

حالا اولین سوال که پیش می‌آید این است:

چرا لینوکس انقدر مهم و محبوب شده؟

اولین دلیل می‌تواند منبع‌باز بودن آن باشد. حالا سوال دوم:

منبع‌باز یعنی چی؟

به زبان ساده، نرم‌افزار منبع‌باز نرم‌افزاری است که کد اصلی آن (Source Code) به‌صورت رایگان در اختیار عموم قرار داده شده و هرکس می‌تواند باتوجه به نیاز و توانایی خود آن را تغییر و نسخه‌ی جدیدتری را در اختیار بقیه قرار دهد. بنابراین وقتی می‌گوییم لینوکس منبع‌باز است، یعنی می‌توان آن را به‌صورت رایگان نصب کرد، تغییر داد و نسخه‌ی تغییر داده شده را منتشر کرد. این ویژگی باعث پویایی و توسعه‌ی سریع این دسته از نرم‌افزارها می‌شود. به‌طوری که به‌سرعت آپدیت شده و باگ‌های آن توسط برنامه‌نویسان شناسایی و برطرف می‌شود.

حالا برگردیم به سوال اول. ویژگی دوم که باعث محبوبیت لینوکس شده، تسلط و کنترل شما بر سیستم‌عامل است. دلیل آن کار با دستورات و کدها به جای آیکون‌ها و فضای گرافیکی محدود شده است. البته برای لینوکس هم مانند ویندوز، نسخه‌ی دسکتاپ منتشر شده اما قدرت اصلی آن در Command-Line و دسترسی به جزئیات سیستم‌عامل است.

اگر قصد دارید این سیستم‌عامل را دانلود و نصب کنید، با توزیع‌های بسیار زیادی از آن روبه‌رو می‌شوید که هرکدام در یک حوزه کاربرد بیشتری دارد. این نکته با توجه به Open Source بودن لینوکس خیلی هم عجیب نیست. همان‌طور که گفتیم هر شخص و گروهی می‌تواند نسخه‌ی جدیدی از لینوکس را طراحی و منتشر کند. توصیه می‌کنم در قدم اول خودتان را خیلی درگیر انواع توزیع‌ها نکنید. کافیس‌ت با اصول و نحوه‌ی کار با لینوکس آشنا شوید و بعداً بر اساس نیاز و حوزه‌ی کاری خود، توزیع مناسب را انتخاب کنید. بنابراین نگران نباشید، نحوه‌ی کار با همه‌ی توزیع‌ها مشابه بوده و فقط محیط گرافیکی و ویژگی‌هایی که ارائه می‌کنند باهم متفاوت است. در این‌جا آموزش را روی رزبری پای و رزبین که بر پایه‌ی توزیع دبیان نوشته شده جلو می‌بریم، اما شما می‌توانید هر نسخه از لینوکس که می‌خواهید روی برد یا کامپیوتر خود نصب و در ادامه‌ی آموزش با من همراه باشید. البته اگر از کامپیوتر استفاده می‌کنید، در ابتدا بهتر است لینوکس را به‌صورت مجازی (Virtual) روی سیستم خود نصب و پس از آشنایی، از آن به‌عنوان سیستم‌عامل اصلی استفاده کنید.

در این آموزش فرض شده که رزبری پای شما از قبل دارای سیستم‌عامل است. در غیر این صورت برای نصب سیستم‌عامل می‌توانید به آموزش راه‌اندازی رزبری پای ۴ با نصب سیستم‌عامل رزبین مراجعه کنید.

معرفی دستورات و مفاهیم پایه‌ای و کاربردی برای شروع کار با لینوکس

پس از این که به اهمیت لینوکس پی‌بردم، تصمیم گرفتم با جست‌جو در صفحات مختلف و بررسی کتاب‌های متنوع، یادگیری این سیستم‌عامل را شروع کنم. (خیلی به کلاس رفتن و صرف هزینه و وقت زیاد برای یادگیری محدود مطالب اعتقاد ندارم). در ابتدا مسلماً با سیل عظیمی از اطلاعات روبه‌رو شدم. وقتی هیچ آشنایی اولیه‌ای با یک مطلب نداشته باشید، قطعاً این که از کجا شروع کنید خیلی در فرآیند یادگیری و علاقه‌ی شما تاثیرگذار خواهد بود. بنابراین از آن جایی که با این حس سردرگمی و مطالعه‌ی زمان‌بر آشنا هستم، تصمیم گرفتم اطلاعاتم در این زمینه را به‌صورت کاربردی و خلاصه با شما به اشتراک بگذارم تا در عین استفاده‌ی بهینه از وقت، بتوانید هر آنچه برای شروع کار با لینوکس نیاز دارید در این بخش پیدا کنید. قطعاً نمی‌توان گفت با یادگیری این دستورات، لینوکس کار حرفه‌ای خواهید شد اما تسلط بر آن‌ها برای کار با رزبری پای کافی است.

قبل از معرفی دستورات بهتر است ابتدا با مفاهیم زیر آشنا شویم:

- **Kernel:** کرنل را می‌توان قلب سیستم‌عامل نامید. در واقع کرنل یا هسته‌ی سیستم‌عامل ارتباط بین نرم‌افزار و سخت‌افزار را برقرار و عملکرد سخت‌افزار را کنترل می‌کند. در حقیقت لینوکس کرنل و Ubuntu، Raspbian، RedHAT سیستم‌عامل‌اند. (کرنل رو میشه گفت نسخه‌ی ساده شده و پایه‌ای سیستم‌عامل برای ارتباط با سخت‌افزاره)
- **Shell:** یک رابط بین User و کرنل است که دستورات نوشته شده توسط کاربر را اجرا و نتیجه را نمایش می‌دهد. Shell دارای انواع مختلفی است که در رزبین به‌صورت پیش‌فرض از Bash استفاده می‌شود. مزیت Shell نسبت به رابط‌های گرافیکی، قدرت و تسلط بالای شما روی کرنل و سیستم‌عامل است اما در مقابل نیاز دارید دستورات لازم برای کار با آن را حفظ کنید (درواقع به هر رابط کاربری برای کار با سیستم‌عامل Shell گفته می‌شه اما این عبارت برای رابط‌های متنی Command-Line رایج‌تره). منظور ما از دستورات لینوکس همین دستورات کار با Shell است که در ادامه با آن‌ها آشنا می‌شویم. قطعاً کسی نمی‌تواند همه‌ی دستورات را حفظ کند. بنابراین کفایت دستورات پایه‌ای را به‌خاطر سپرده و بقیه را در صورت نیاز در مراجع جست‌وجو کنید.
- **Terminal:** به محیط و برنامه‌ی لازم برای نمایش و راه‌اندازی Shell، ترمینال می‌گویند. ترمینال یک رابط متنی برای دریافت ورودی و خروجی است و Shell دستورات نوشته شده در آن را اجرا می‌کند. (پس ترمینال اون صفحه سیاه‌س، Shell دستورایی که توش می‌نویسین رو اجرا می‌کنه و کرنل هم ارتباط بین

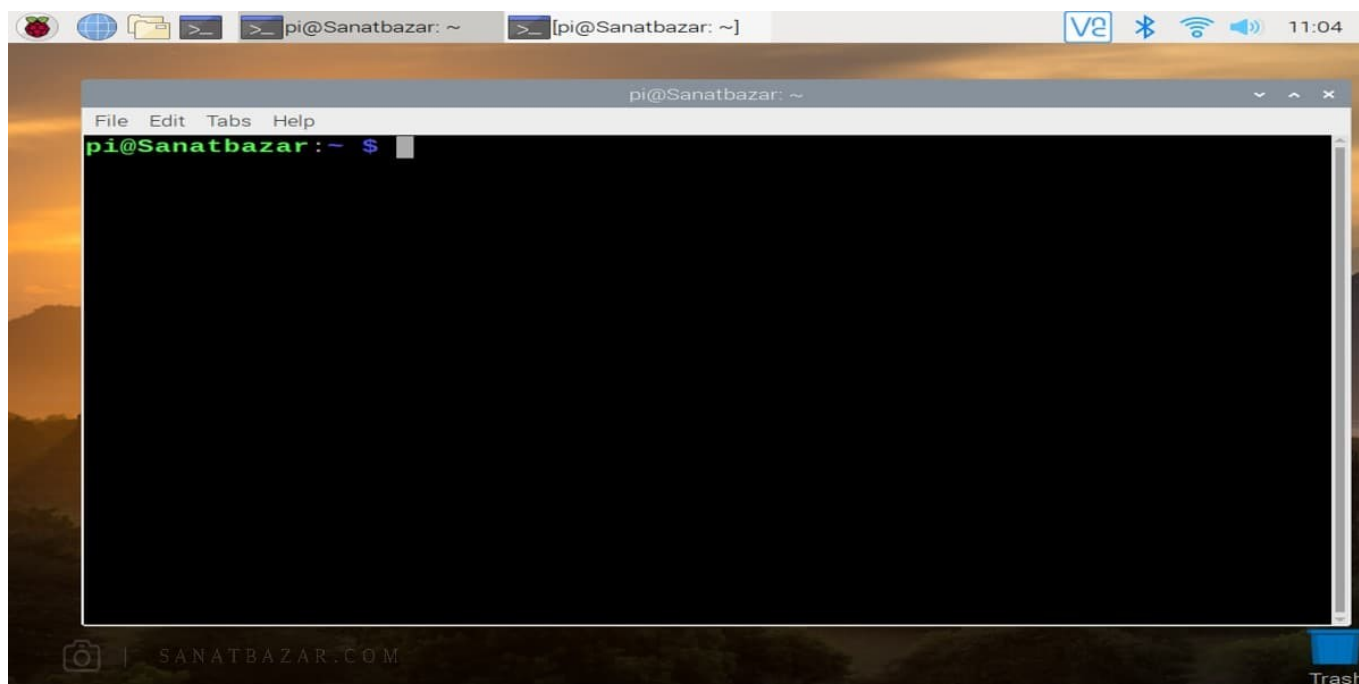
سیستم-عامل و سخت افزار رو برقرار می‌کنه

در رزبین برای ورود به Shell و Terminal کافیه از نوار ابزار روی آیکون Terminal کلیک یا از ALT+CTRL+T استفاده کنید.

• Prompt: محل تایپ دستورات شما در ترمینال را نشان می‌دهد. در اکثر توزیع‌های لینوکس، Prompt به صورت زیر است:

```
User@Hostname: Directory $
```

همانطور که در تصویر زیر مشاهده می‌کنید، در اینجا نام User نام پیش فرض pi و نام Hostname، sanatzbazar است. دایرکتوری نیز ~ می‌باشد.



• دایرکتوری (Directory)

دایرکتوری در لینوکس معادل پوشه (Folder) در ویندوز است. در واقع یک دایرکتوری می‌تواند حاوی فایل‌ها یا دایرکتوری‌های دیگر باشد. این دسته از فایل‌ها با رنگ آبی در لینوکس نشان داده می‌شوند. با توجه به این که در محیط خط فرمان خبری از رابط گرافیکی نیست، برای دسترسی به هر فایل و دایرکتوری باید آدرس آن را بنویسیم. اجازه بدید نحوه آدرس دهی را با یک مثال ببینیم:

```
/home/pi
```

در اینجا ما در فولدر pi که خودش در فولدر home قرار دارد، هستیم. حال با انتخاب فایل Documents از pi، Current Directory ما به /home/pi/Documents/ تغییر می‌کند.

چند تا دایرکتوری مهم و معروف در لینوکس هست که با آن‌ها زیاد سروکار دارید:

/: دایرکتوری root که تمامی دایرکتوری‌ها در آن قرار دارند. در لینوکس همه چیز از / شروع می‌شود. (پس اگه فایل‌ها و آدرس‌ها رو در لینوکس مثل یک درخت ببینیم، اولین فایل که همه از اون رشد می‌کنن، دایرکتوری روت)

نکته: در لینوکس دو نوع root داریم. دایرکتوری root و کاربر root. کاربر root مانند user administrator در ویندوز عمل می‌کند که خود دارای یک Home Directory با نام root می‌باشد و با / متفاوت است.

/home/: مشابه My Documents در ویندوز، اطلاعات مربوط به هر کاربر سیستم عامل در این پوشه قرار می‌گیرند. این دایرکتوری هم به اختصار با ~ نشان داده می‌شود.

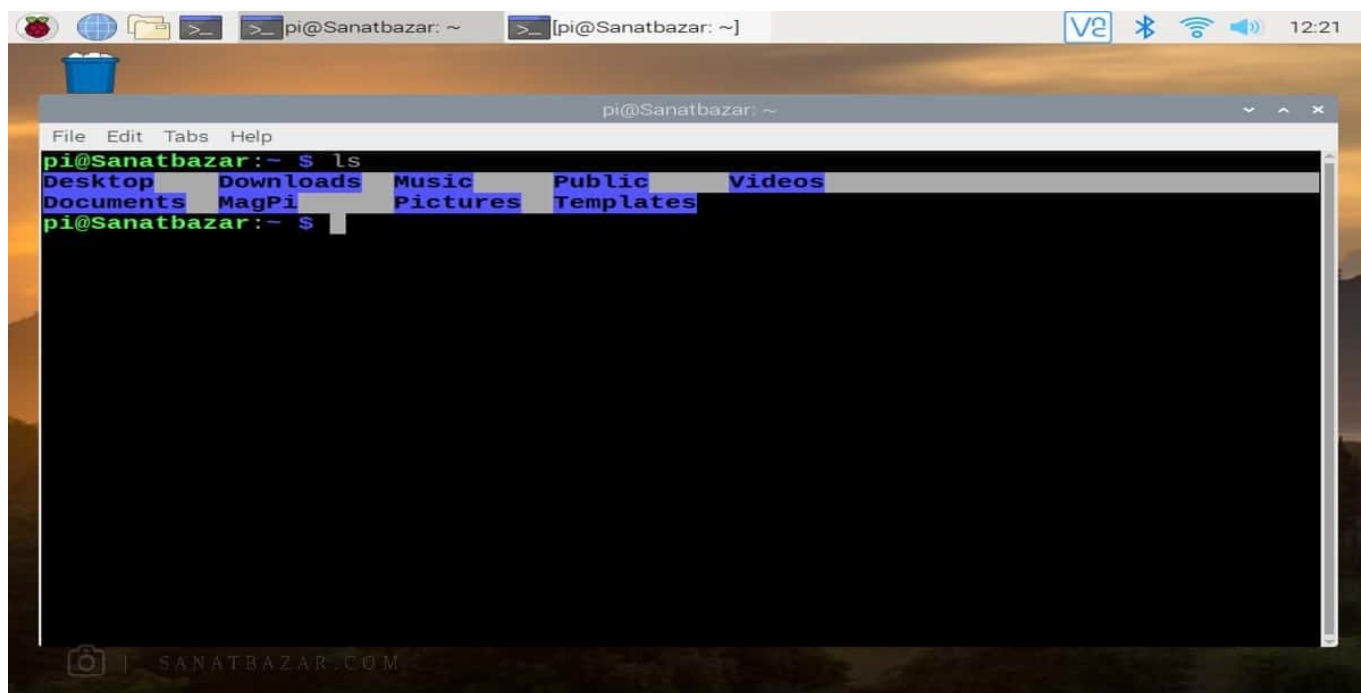
/bin/: فایل‌ها و کدهای مربوط به دستورات لینوکس در این پوشه قرار دارد. مثلاً کد نوشته شده برای اجرای ls و cd ...

/boot/: حاوی فایل‌های مربوط به بوت سیستم عامل

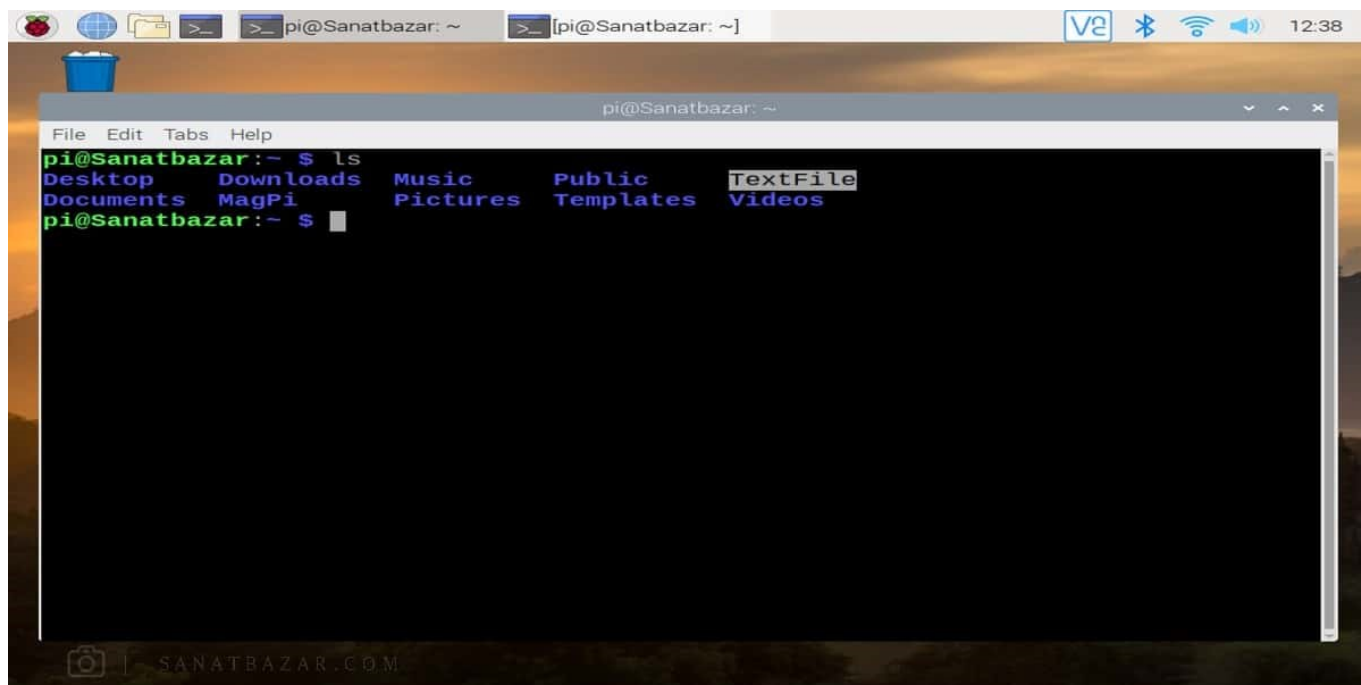
/dev: شامل Device File ها است مثل رابطهای سریال و USB برد.

/etc: این پوشه شامل تمامی configuration File های دستگاه شماست. Configuration File ها اطلاعات و تنظیمات لازم برای نحوه اجرای برنامههای سیستم هستند.

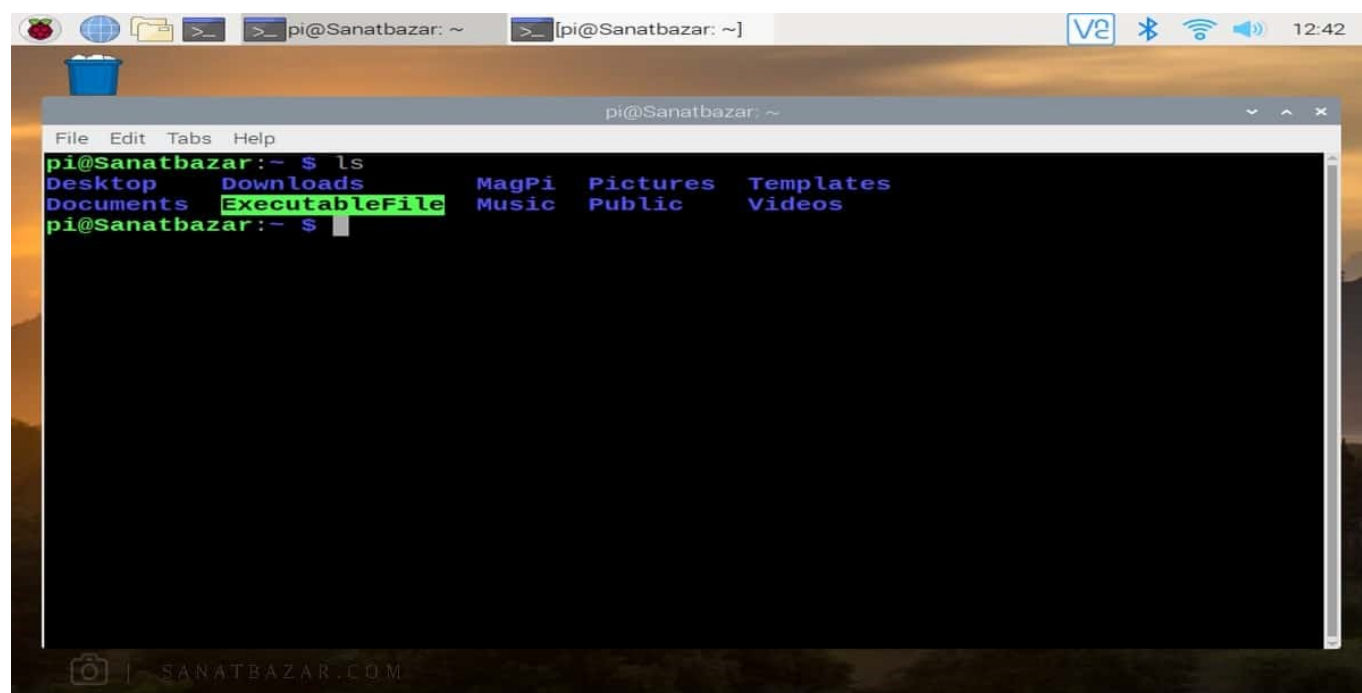
/media/pi: در صورتی که فلش مموری یا هارد اکسترنال را به رزبری پای متصل کنید، محتویات آنها در این آدرس قرار می گیرند.



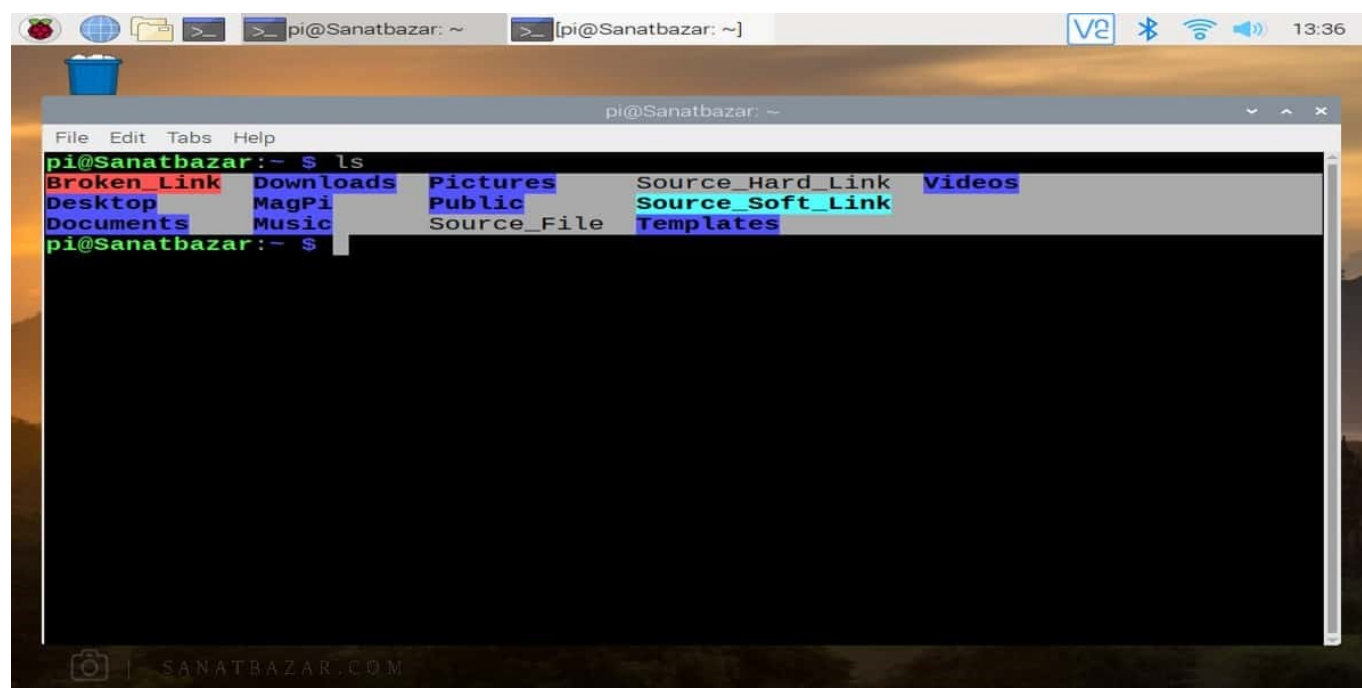
○ فایل های متنی و معمولی: فایل های text که می توانند شامل متن و کد باشند. معادل فرمت txt. در ویندوز. این فایل ها بی رنگ هستند.



○ فایل های اجرایی (Executable Files): فایل ها و برنامه هایی که می توانند توسط سیستم عامل اجرا شوند. این دسته از فایل ها با رنگ سبز نشان داده می شود.



○ فایل‌های لینک (Link Files): این فایل‌ها مشابه Shortcut در ویندوز عمل می‌کنند و شامل دو نوع Hard Links و Soft Links هستند. همان‌طور که از نام آن‌ها می‌توان حدس زد، لینک‌ها شما را به فایل اصلی که از روی آن ساخته شده‌اند، ارجاع داده و هر تغییری روی مرجع، به‌طور خودکار روی آن‌ها نیز اعمال می‌شود. هاردلینک در صورت حذف فایل اصلی همچنان قابل استفاده و سالم‌اند اما برخلاف آن‌ها اگر فایل مرجع دچار مشکل یا حذف شود، سافت‌لینک‌ها دیگر قابل استفاده نیستند و به آن‌ها Broken links گفته می‌شود. رنگ هاردلینک مشابه رنگ فایل مرجع و رنگ سافت‌لینک آبی روشن است. همچنین Broken Link‌ها با رنگ قرمز نشان داده می‌شوند.



○ فایل دیوایس (Device File): اطلاعات مربوط به دستگاه‌ها و ماژول‌هایی که به برد و سیستم شما متصل می‌شوند در این فایل‌ها قرار می‌گیرند (ارتباط سریال و USB و ...). این فایل‌ها با رنگ زرد نمایش داده می‌شوند.

```

pi@Sanatbazar: /dev
File Edit Tabs Help
cpu_dma_latency mmcblk0 serial1 tty3 tty57 vcsa2
cuse mmcblk0p1 snm tty30 tty58 vcsa3
disk mmcblk0p2 snd tty31 tty59 vcsa4
fb0 net stderr tty32 tty6 vcsa5
fd network_latency stdin tty33 tty60 vcsa6
full network_throughput stdout tty34 tty61 vcsa7
fuse null tty tty35 tty62 vcsa8
gpiochip0 ppp tty0 tty36 tty63 vcsa9
gpiochip1 ptmx tty1 tty37 tty7 vcsa10
gpiochip2 pts tty10 tty38 tty8 vcsa11
gpiomem ram0 tty11 tty39 tty9 vcsa12
hidraw0 ram1 tty12 tty4 ttyAMA0 vcsa13
hidraw1 ram10 tty13 tty40 ttyprintk vcsa14
hwrng ram11 tty14 tty41 uhid vcsa15
initctl ram12 tty15 tty42 uinput vcsa16
input ram13 tty16 tty43 urandom vcsa17
kmsg ram14 tty17 tty44 v4l vchi
log ram15 tty18 tty45 vchiq video0
loop0 ram2 tty19 tty46 vcio video1
loop1 ram3 tty20 tty47 vc-mem video2
loop2 ram4 tty21 tty48 vcs watchdog
loop3 ram5 tty22 tty49 vcs1 watchdog0
loop4 zero
pi@Sanatbazar: /dev $

```

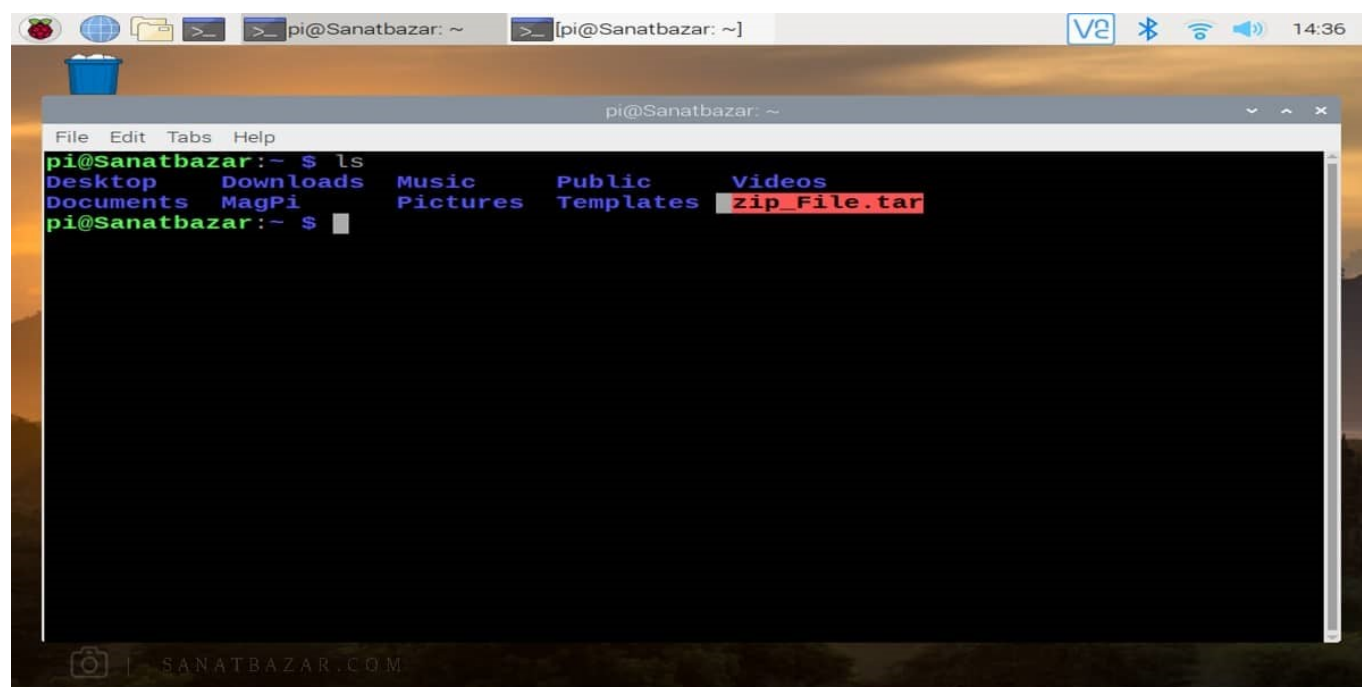
○ فایل‌های گرافیکی: فایل‌های تصویری مانند عکس‌ها و فیلم‌ها. این دسته از فایل‌ها با رنگ صورتی نشان داده می‌شوند.

```

pi@Sanatbazar: ~
File Edit Tabs Help
pi@Sanatbazar:~ $ ls
2019-09-18-142618_1024x768_scrot.png Documents MagPi Pictures Templates
Desktop Downloads Music Public Videos
pi@Sanatbazar:~ $

```

○ فایل‌های فشرده (Archive Files): فایل‌های فشرده و زیپ‌شده که مانند Broken Links با رنگ قرمز نمایش داده می‌شوند.



پس از آشنایی با مفاهیم فوق، دستورات مهم و اساسی لینوکس را یک‌به‌یک بررسی می‌کنیم. برای یادگیری بهتر، می‌توانید Command-Line را اجرا کرده و همراه با من دستورات را Run کنید.

pwd: این دستور آدرس دایرکتوری که شما الان در آن هستید (Current Directory) را نشان می‌دهد. به‌عنوان مثال، همانطور که گفتیم ~ نماد Home Directory برای هر کاربر است و طبق انتظار ما، دستور pwd آدرس Home/Pi را نمایش می‌دهد. Pi در اینجا نام کاربری است که ما با آن وارد سیستم شده‌ایم. (پس یعنی من الان کجام؟)

whoami: با اجرای این دستور، نام کاربری شما نمایش داده خواهد شد. به‌صورت پیش‌فرض این نام pi است.

```
$ whoami
```

mkdir/rmdir: برای ایجاد دایرکتوری (همون create new folder) از دستور mkdir استفاده می‌کنیم. برای مثال فرض کنید می‌خواهیم در Home Directory پوشه‌ای با نام sanatbazar ایجاد کنیم. برای این کار دستور زیر را می‌نویسیم:

```
mkdir sanatbazar $
```

برای ایجاد چند فایل می‌توانید از space استفاده کنید:

```
$ mkdir sanatbazar1 sanatbazar2
```

همان‌طور که حدس زدید، برای پاک کردن یک پوشه نیز دستور rmdir را اجرا می‌کنیم. بنابراین برای پاک کردن پوشه‌ی فوق باید `rmdir sanatbazar` را تایپ کنیم.

man: دستور man یکی از دستورات بسیار کارآمد برای کاربران لینوکس به‌خصوص تازه‌کار هاست. چرا؟ این دستور درواقع همان راهنما و Help سیستم‌عامل است. پس با آن می‌توانید اطلاعات و طرز استفاده از هر دستوری را که می‌خواهید، خیلی جامع و روان مطالعه کنید. مثلاً برای دستور rmdir عبارت `$ man rmdir` را اجرا کنید تا صفحه‌ی راهنما برای شما باز شود. در این صفحه با کلیدهای جهت روی کیبورد می‌توانید متن را بالا و پایین کرده و با کلید Q از آن خارج شوید. اگر می‌خواهید عملکرد هر دستور را خیلی خلاصه‌تر بخوانید، می‌توانید از `$ whatis` و نام دستور مورد نظر استفاده کنید.

date: نمایش تاریخ و ساعت

```
$ date
```

clear: زمانی که دستورات متعددی را وارد کرده‌اید، با استفاده از `$ clear` می‌توانید صفحه‌ی Command-Line را پاک کنید.

cd: برای تغییر دایرکتوری که همان معادل با کلیک کردن و بازکردن فولدرها در ویندوز است، از cd استفاده می-کنیم. برای این کار فرض کنید می-خواهیم از آدرس ~ به sanatzbazar/~ برویم. یعنی فایل sanatzbazar را از Home Directory انتخاب کنیم:

```
cd sanatzbazar $
```

در اینجا چون sanatzbazar در ~ بود، فقط کافیست که نام آن را مقابل cd بنویسیم. اما اگر این پوشه در آدرس دیگری و خارج از ~ قرارداشت، باید آدرس آن را بنویسیم. مثلاً فرض کنید ما در ~ و sanatzbazar در ~/documents باشد. برای این کار باید دستور cd Documents/sanatzbazar \$ را بنویسیم.

در لینوکس دو نوع آدرس دهی داریم. آدرس دهی مطلق و نسبی. در روش مطلق، آدرس کامل فایل را می-نویسیم. مثلاً در مثال بالا فرض کنید می-خواهیم از ~/Documents/sanatzbazar/ به ~/Documents/ برگردیم. در آدرس دهی مطلق باید از دستور cd ~/Documents \$ و در نسبی از عبارت زیر استفاده کنیم.

```
cd .. $
```

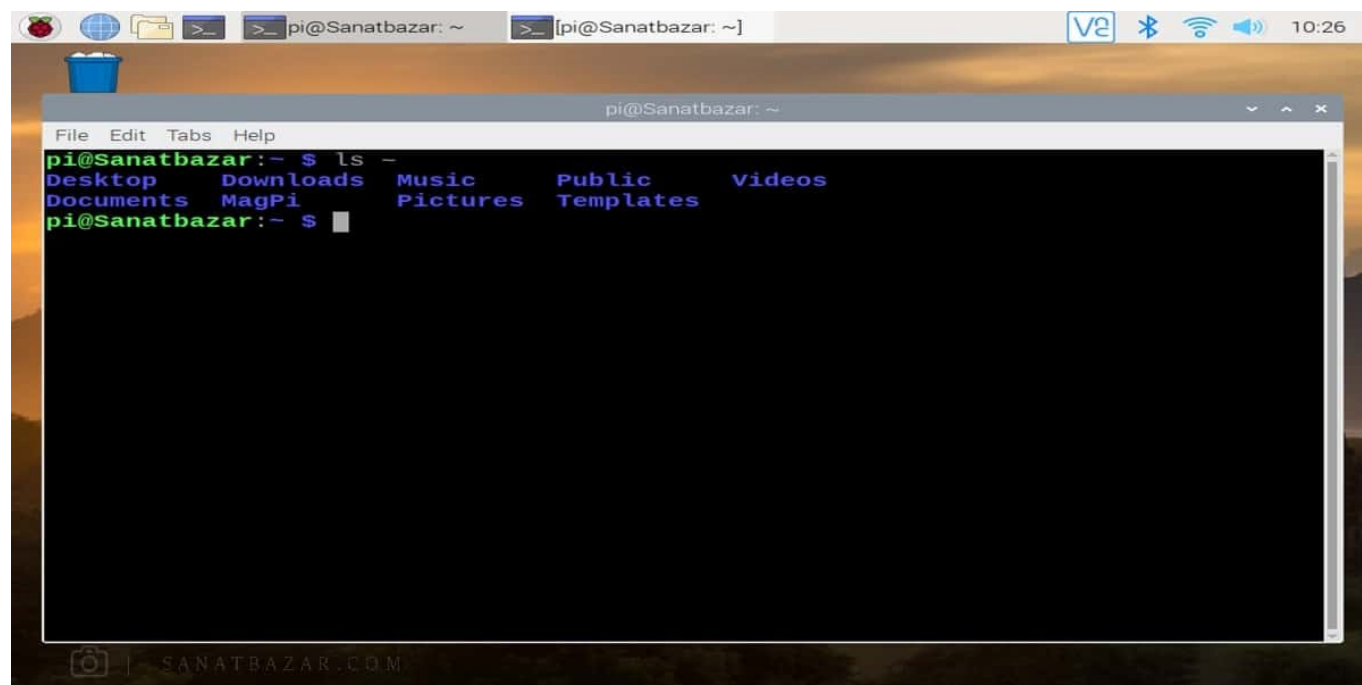
در واقع اگر فایل ها و آدرس ها را در لینوکس به شکل یک درخت ببینیم، دو نقطه (..) یعنی شاخه ی قبلی. حالا تمرین: به صورت نسبی از ~/Documents/ به Desktop/ بروید.

جواب: cd ../Desktop/ \$ یعنی یک شاخه برو عقب (به ~) و بعد وارد Desktop شو. همچنین ./ یعنی current directory (آدرسی که هر لحظه در آن هستید). از این آدرس دهی بیشتر برای کپی و انتقال فایل ها استفاده می-شود.

نکته: پس از نوشتن چند حرف اول هر فایل با زدن Tab، بقیه ی نام آن به صورت خودکار نوشته می-شود. برای مثال در ~ با نوشتن cd De Tab می-بینید که Desktop نشان داده خواهد شد. اگر حروف نوشته شده بین چند اسم مشترک باشد با دوبار زدن Tab تمام آن ها نمایش داده خواهد شد.

ls: پرکاربردترین دستورهای لینوکس همین ls و cd هستند. این دستور محتویات درون هر فولدر را لیست می-کند. در واقع با ls می-توانید ببینید در هر فولدر چه فایل ها و دایرکتوری هایی وجود دارد. برای کار با این دستور کافیست آدرس یا نام دایرکتوری مورد نظرتان را مقابل آن بنویسید. به عنوان مثال می-خواهیم محتویات داخل Home Directory را ببینیم:

```
$ ls ~
```



هر دستور در لینوکس دارای کلیدهای مختلفی است که با - مشخص شده و عملکرد این گزینه ها در man آن دستور نوشته شده است. در واقع با استفاده از کلیدها (که به آن ها Switch می-گویند) می-توانید از قابلیت های اضافی آن دستور استفاده کنید. به عنوان مثال اگر ls را به صورت ls -la استفاده کنید، فایل های Hidden هر فولدر نیز نشان داده خواهد شد.

```
$ ls -a ~
```

همانطور که مشاهده می‌کنید، نام فایل‌های Hidden با . شروع شده و برای ایجاد این فایل‌ها کفایت اول اسم آن‌ها . بگذارید (مثلاً به جای test از test. استفاده کنید). همچنین می‌توانید دو سویچ را به صورت همزمان استفاده کنید. مثلاً \$ ls -s که سایز هر فایل را نشان می‌دهد را همزمان با -a به صورت \$ ls -las اجرا کنید.

```
pi@Sanatbazar: ~
File Edit Tabs Help
pi@Sanatbazar:~ $ ls -a -
.      .bashrc  Documents  MagPi     Public     .Xauthority
..     .cache   Downloads  Music     Templates  .xsession-errors
.bash_history  .config  .gnupg     Pictures  Videos     .xsession-errors.old
.bash_logout  Desktop  .local     .profile  .vnc
pi@Sanatbazar:~ $
```

tree: مشابه دستور ls محتویات هر پوشه و زیر مجموعه‌های آن را به صورت درختی نشان می‌دهد. برای مثال بالا:

```
$ tree ~
```

```
pi@Sanatbazar:~ $ tree -
/home/pi
├── Desktop
├── Documents
│   └── test
├── Downloads
├── MagPi
│   └── MagPi83.pdf
├── Music
├── Pictures
├── Public
├── Templates
└── Videos

9 directories, 2 files
pi@Sanatbazar:~ $
```

cp: کپی کردن فایل‌ها و دایرکتوری‌ها. به عنوان مثال می‌خواهیم فایل متنی test را با نام test2 کپی کنیم. برای این کار از دستور \$ cp test test2 استفاده می‌کنیم. در مثال بعدی می‌خواهیم test را در آدرس ~/Desktop کپی کنیم:

```
$ cp test ~/Desktop/test2
```

mv: این دستور معادل Cut در ویندوز عمل می‌کند و طرز استفاده از آن مشابه cp است. همچنین از این دستور برای Rename کردن یک فایل نیز استفاده می‌شود. برای این کار فایل مورد نظرتان را با نام دیگری در همان آدرس Cut کنید. به عنوان مثال \$ mv test sanatbazar نام فایل test را به sanatbazar تغییر می‌دهد.

rm: معادل delete در windows است. نکته‌ی مهم در مورد این دستور غیر قابل برگشت بودن آن است. در واقع باید گفت این دستور معادل Shift+Delete است. پس برای استفاده از این دستور دقت کنید!

اگر دستور rm را با سوییچ -i استفاده کنید، قبل از حذف از شما سوال می‌شود که آیا از پاک کردن این فایل مطمئنید (مشابه ویندوز)، اگر بله y را تایپ و enter را بزنید. همچنین برای حذف یک دایرکتوری حاوی فایل، اگر از دستور \$ rmdir استفاده کنید با خطا مواجه می‌شوید. در واقع rmdir برای حذف دایرکتوری‌های خالی به کار می‌رود. برای این کار باید از rm با سوییچ -r استفاده کنید.

نکته: در لینوکس کارکترهایی وجود دارد که کار با بعضی دستورات را برای ما ساده‌تر می‌کند. به این کارکترها Wildcard گفته می‌شود. برای درک راحت‌تر، آن‌ها را با مثال بررسی می‌کنیم. فرض کنید در پوشه‌ی Desktop فایل‌های زیر موجود است:

Test1.jpeg و Test1.txt، Test2.txt، Test3.txt، Test1a.txt، Text1.png

*: به معنای هیچ یا همه. این کارکتر می‌تواند به جای چندین کارکتر بنشیند.

```
$ rm *.txt
```

این دستور تمام فایل‌هایی که به .txt ختم شوند را پاک می‌کند. پس * در اینجا تعیین می‌کند که به جز .txt باقی حروف هرچه باشد مهم نیست.

```
$ rm ~/Desktop/*
```

در دستور بالا، rm روی تمام فایل‌های موجود در Desktop اجرا می‌شود.

?: مشابه * است اما فقط به جای یک حرف می‌تواند بنشیند. برای مثال:

```
$ rm Test?.txt
```

فایل‌های Test1.txt و Text2.txt را حذف می‌کند. توجه کنید که تعداد حروف در این جا مهم است. برای مثال \$ rm Test??.txt فقط Test1a.txt را پاک می‌کند.

{ }: آکولاد انتخاب ما را به چند کلمه محدود می‌کند. مثال:

```
$ rm Test1.{png,jpeg}
```

این دستور یعنی فایل‌های با نام Test1. که دارای پسوند png یا jpeg باشند. پس اینجا فایل‌های Test1.png و Test1.jpeg حذف و بقیه باقی می‌مانند.

[]: براکت مشابه آکولاد عمل می‌کند اما فقط برای یک حرف. پس:

```
$ rm Test[1,3].txt
```

این دستور Test1.txt و Test3.txt را حذف می‌کند. در واقع هر براکت جای یک حرف می‌نشیند. Te[s,x]t[1,2].txt یعنی یکی از حالت‌های Test1.txt، Test2.txt، Text1.txt و Text2.txt

[!]: علامت تعجب به معنای نفی.

```
$ rm Test[!1]*
```

تمام فایل‌هایی که پس از Test عدد 1 را ندارند، پاک می‌کند. توجه کنید که با زدن * در انتها تعیین کردیم تعداد و نوع بقیه‌ی حروف مهم نیستند.

shutdown: این دستور برای خاموش کردن سیستمعامل استفاده می‌شود. به فقط سیستمعامل! برد شما همچنان دارای تغذیه است و برای قطع آن باید منبع تغذیه را قطع کنید (در کامپیوترها نیازی به این کار نیست!). نکته‌ی جالب درباره‌ی shutdown، سوییچ‌های متنوع آن است که به تعدادی از آن‌ها اشاره می‌کنیم:

```
$ shutdown
```

این دستور سیستمعامل را پس از یک دقیقه خاموش می‌کند.

```
$ shutdown -r
```

سیستمعامل را پس از یک دقیقه Restart می‌کند.

```
$ shutdown -h now
```

سیستمعامل را فوراً خاموش می‌کند. به جای این دستور می‌توان از \$ poweroff هم استفاده کرد.

```
$ shutdown -r now
```

سیستمعامل را فوراً Restart می‌کند. به جای این دستور می‌توان از \$ reboot هم استفاده کرد.

```
$ shutdown -h 10:15&
```

سیستمعامل را در ساعت 10:15 دقیقه خاموش می‌کند.

برای لغو عملیات فوق می‌توانید از سوییچ c استفاده کنید. یعنی \$ shutdown c

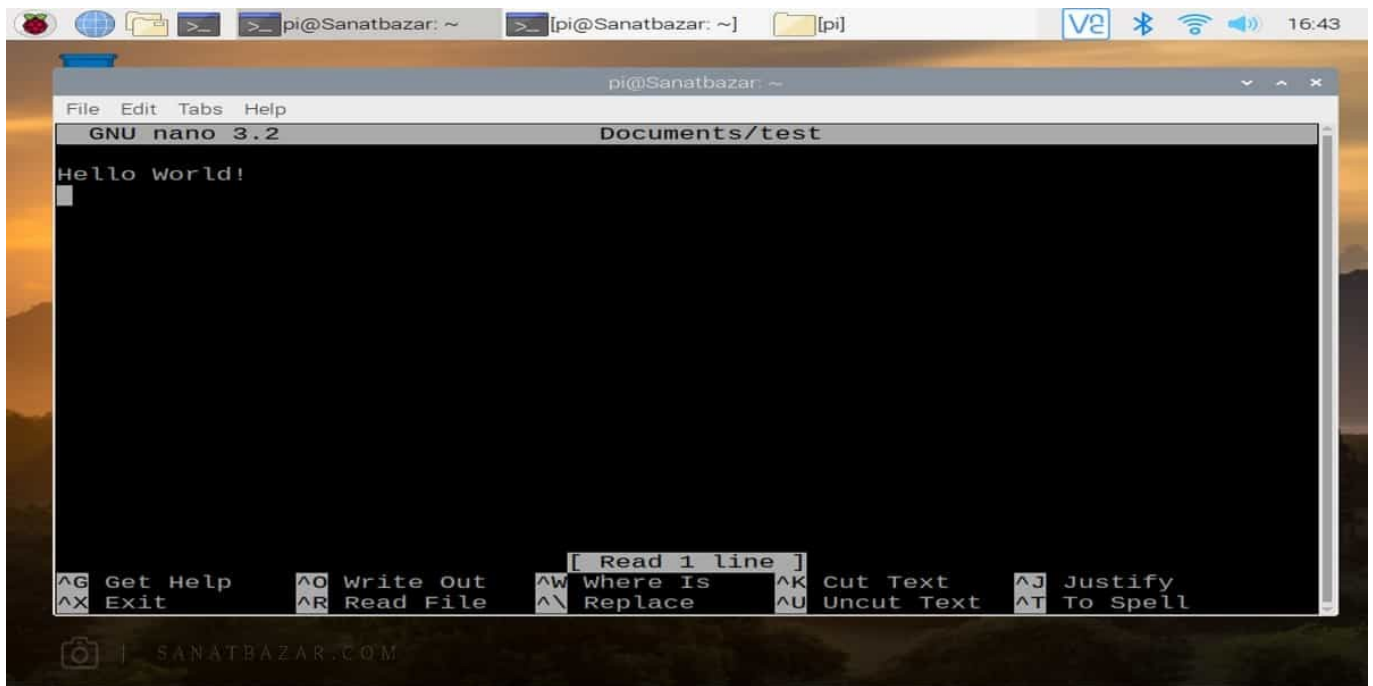
همین حالا یکی از دستورهای shutdown را اجرا کنید تا اهمیت دستور بعدی را متوجه شوید.

sudo: در لینوکس کاربران متعددی با سطح دسترسی مختلف به سیستمعامل می‌توان تعریف کرد. در این میان کاربر root بیشترین دسترسی را داشته و می‌تواند سایر کاربران و سیستمعامل را مدیریت کند. اگر دستور shutdown را اجرا کرده باشید، مطمئناً با خطا مواجه شده‌اید. زیرا کاربران عادی توانایی خاموش و restart کردن سیستم را ندارند. این محدودیت در بسیاری از دستورات و حتی فایل‌ها نیز دیده می‌شود. می‌توان فایلی را ایجاد کرد که سایر کاربران توانایی حذف، انتقال و تغییر آن را نداشته باشند. برای استفاده از توانایی root دو راه دارید: یا با کاربر root از سیستمعامل استفاده کنید یا با sudo دستورات را اجرا کنید. رمز این کاربر به صورت پیش‌فرض همان رمزی است که پس از نصب سیستمعامل وارد کرده‌اید. بنابراین اگر از رزبین استفاده می‌کنید و رمز را تغییر نداده‌اید، رمز همان raspberry خواهد بود. برای ورود به حساب root می‌توانید از دستور زیر استفاده کنید:

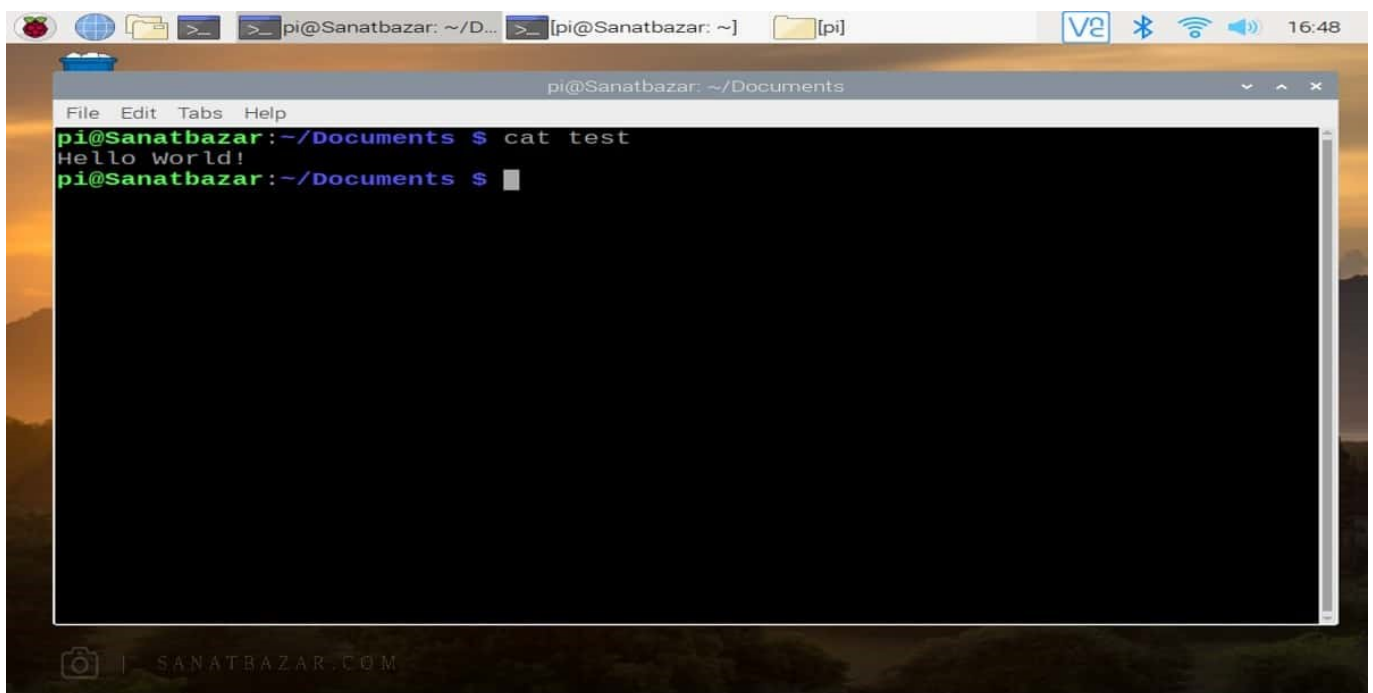
```
$ sudo -i
```

پس از وارد کردن رمز prompt شما از \$ به # تغییر می‌کند و می‌توانید از مزایای root استفاده کنید. برای خروج از این حساب هم کافیست \$ exit را وارد کنید. راه دوم استفاده از دستور sudo در ابتدای سایر دستورات است. مثلاً به جای \$ shutdown از \$ sudo shutdown استفاده کنید. در این صورت بعد از وارد کردن رمز، دستور شما بدون خطا اجرا می‌شود. کلاً اگر قبل از هر عملی در لینوکس sudo بنویسید، این دستور از طرف root اجرامی‌شود.

nano: ایجاد فایل متنی یا text. این دستور بسیار کاربردی کار یک ویرایشگر متنی را انجام می‌دهد. در واقع برای برنامه‌نویسی و تایپ هر چیزی در لینوکس از این دستور می‌توانید استفاده کنید. برای مثال می‌خواهیم عبارت Hello World! را در یک فایل متنی با اسم test بنویسیم. برای این کار ابتدا دستور nano test \$ را اجرا می‌کنیم. اگر می‌خواهید این فایل را در آدرس به جز Current Directory ایجاد کنید کافیست به جای test آدرس آن را بنویسید. (/testآدرس nano \$) در پنجره‌ی باز شده عبارت Hello World! را نوشته و برای ذخیره CTRL+X و سپس برای تایید ذخیره y و Enter را می‌زنیم. اگر قبل از زدن Enter از ذخیره منصرف شدید CTRL+C را بزنید تا این کار کنسل شود. حال اگر دوباره nano test \$ را اجرا کنید، عبارت Hello World! را خواهید دید. اگر یک فایل متنی را با دستور \$ nano sudo ایجاد کنید، سایر کاربران به صورت عادی توانایی تغییر، حذف و انتقال آن را ندارند.



cat: برای خواندن فایل متنی نوشته شده از cat استفاده کنید. بنابراین اگر `cat test` را اجرا کنید، عبارت `Hello world!` نمایش داده می شود.



less: گاهی متن آنقدر طولانی است که خواندن آن با زدن `cat` بسیار سخت خواهد بود. برای این متن ها بهتر است از دستور `$ less` استفاده کنید. با این دستور با زدن کلید جهت ها متن شما خط به خط و با زدن `page up` و `page down` به صورت صفحه به صفحه نشان داده می شود.

نام فایل متنی `$ less`

:Redirections

عملگر Pipe (`|`): معمولاً با نگه داشتن کلید `Shift` و کلید بالای `Enter` نوشته می شود و خروجی یک دستور را به عنوان ورودی دستور دیگر اعمال می کند.

`$ دستور دوم | دستور اول`

>: این عملگر نتیجه‌ی یک دستور را در فایل موردنظر ذخیره می‌کند. مثلاً:

```
$ echo hello > test
```

Hello در فایل متنی test ذخیره می‌کند. با این عملگر هرچه از قبل در test بوده حذف و فقط hello ذخیره می‌شود.

<: مشابه > است با این تفاوت که محتویات قبلی را حذف نکرده و محتویات جدید را به انتهای آن اضافه می‌کند.

head: 10 خط اول هر صفحه را نمایش می‌دهد. برای مثال می‌خواهیم محتویات فایل /var/log/syslog/ را که مثل جعبه سیاه هواپیما تمام فعالیت‌های سیستم‌عامل را ذخیره می‌کند، بخوانیم. برای این کار نیاز به دسترسی root دارید. پس:

```
$ sudo head /var/log/syslog
```

با سوییچ n می‌توانید تعداد خط‌های لازم برای نمایش را تغییر دهید.

```
$ sudo head /var/log/syslog -n 15
```

نمایش 15 خط اول فایل

tail: مشابه head اما خطوط پایانی هر فایل را نمایش می‌دهد.

grep: دستور کاربردی برای جست‌وجوی یک عبارت خاص در فایل‌های متنی. زمانی که در میان متن‌های طولانی، دنبال یک عبارت خاص می‌گردید، این دستور به شما کمک خواهد کرد:

```
$ grep overclock /boot/config.txt
```

در مثال بالا، عبارت overclock را در config.txt جست‌وجو می‌کنیم. این دستور به بزرگی و کوچکی حروف حساس نیست و برای اعمال این حساسیت باید از سوییچ i باید استفاده کنید. همچنین اگر مثلاً کلمه‌ی is را جست‌وجو کنید، عبارت‌های this، list و غیره را هم برمی‌گرداند. برای جلوگیری از این کار باید is را در " " قرارداده و از سوییچ w نیز استفاده کنید.

ehco: این دستور مانند print در زبان‌های برنامه‌نویسی عمل کرده و هر عبارتی که مقابل آن نوشته‌شود، برمی‌گرداند.

```
$ echo -e "first \nsecond"
```

عبارات را خط‌به‌خط نمایش می‌دهد.

```

pi@Sanatbazar:~$ echo salam! khubi?
salam! khubi?
pi@Sanatbazar:~$ echo -e "salam! \\nkhubi?"
salam!
khubi?
pi@Sanatbazar:~$

```

file: در ویندوز نوع هر فایل با پسوند آن فایل مشخص می‌شود. مثلاً در پسوند یک فایل متنی عبارت .txt و در عکس‌ها پسوندهایی مثل .jpg دیده می‌شود. در لینوکس اوضاع متفاوت است و هرجا که پسوندی دیدید، این پسوند فقط برای اطلاع کاربر نوشته شده است (البته موارد استثنا هم وجود دارد). یعنی شما می‌توانید یک فایل متنی را با پسوند .jpg بدون هیچ مشکلی ذخیره کنید. در واقع لینوکس با تحلیل محتوا و هدر فایل‌ها نوع آن‌ها را تشخیص می‌دهد. برای تشخیص نوع فایل می‌توانید از دستور زیر استفاده کنید:

\$ file نام فایل

برای مثال در اینجا می‌خواهیم نوع فایل test را ببینیم:

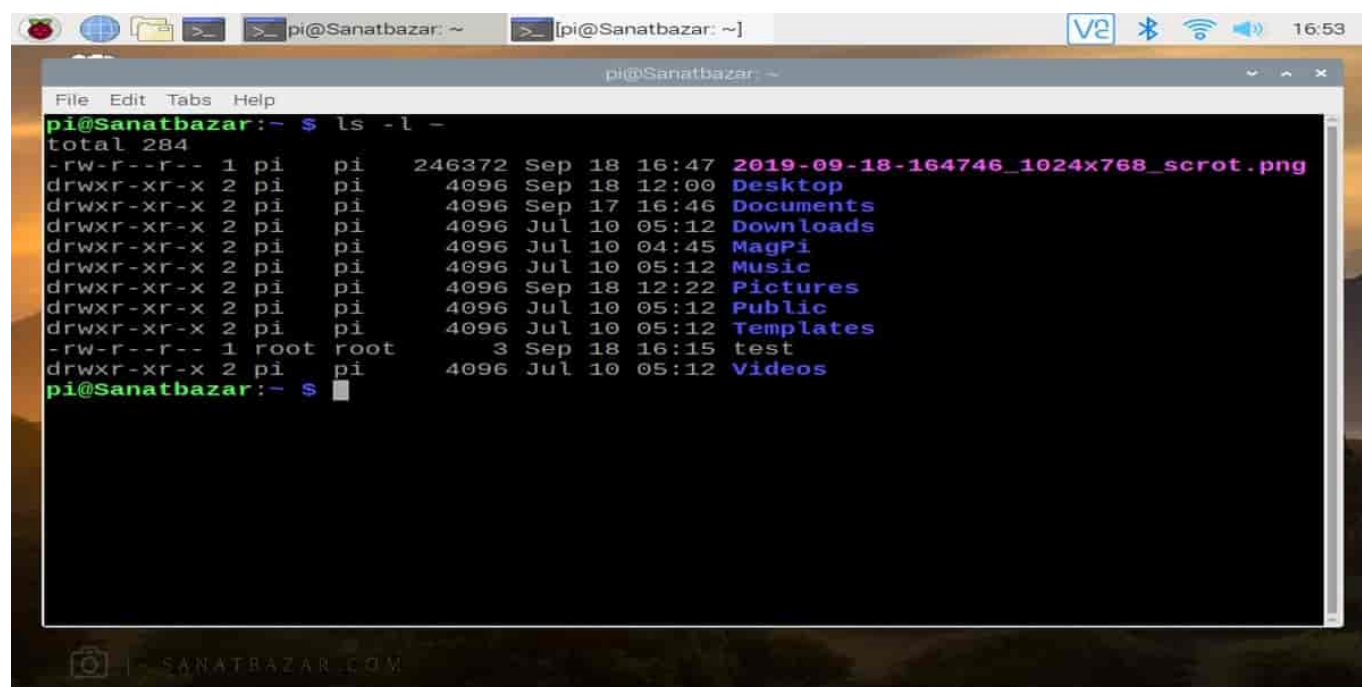
```

pi@Sanatbazar:~$ file test
test: ASCII text
pi@Sanatbazar:~$

```

همانطور که می‌بینید با این که برای این فایل پسوندی تعریف نکردیم، نوع آن Text شناسایی شده است. البته برای این که خودمان از نوع فایلی که ایجاد می‌کنیم آگاه باشیم، بهتر است پسوند آن را بنویسیم.

chmod: گفتیم که سطح دسترسی به هر فایل برای کاربران مختلف، متفاوت است. حالا از کجا بفهمیم این سطح چقدره و چگونه یک فایل را عمومی یا اختصاصی کنیم؟ دستور chmod بسیار مهم بوده و بعداً در انجام پروژه‌ها از آن استفاده خواهیم کرد. اما قبل از بررسی این دستور بگذارید برگردیم به \$ ls. این دستور یک سوییچ کاربردی به صورت -a دارد که اطلاعات مهمی را درباره‌ی فایل‌های لیست شده، نشان می‌دهد. برای مثال این دستور را روی ~ اجرا می‌کنیم:



```
$ ls -l ~
```

حتماً با دیدن نتیجه دستی بر محاسن کشیده و جامه دریدید که یا شیخنا این دیگه چیه؟ ولی نگران نباشید الان توضیح می‌دم که هرکدام از این عبارتها یعنی چی؟ در لینوکس فایل‌ها را از نظر سطح دسترسی به 3 دسته می‌توان تقسیم کرد:

- فایل‌های خواندنی (Readable) که به اختصار با r نشان می‌دهند و فقط می‌توان آن‌ها را خواند.
- فایل‌های نوشتنی (Writable) که به اختصار با w نشان می‌دهند و می‌توان در آن‌ها تغییر ایجاد کرد.
- فایل‌های قابل اجرا (Executable) که به اختصار با x نشان داده شده و مانند کدهای برنامه‌نویسی می‌توان آن‌ها را اجرا کرد.

حالا آیا یک فایل می‌تواند هم خواندنی باشد هم نوشتنی؟ یا هم اجرایی، هم نوشتنی و هم خواندنی؟ بله. هر فایل می‌تواند چند حالت را باهم داشته یا اصلاً هیچکدام را نداشته باشد!

حالا برگردیم و 10 کارکتر اول هر خط را باهم بررسی کنیم:

اولین کارکتر نوع فایل را مشخص می‌کند:

_ متن یا فایل معمولی

d دایرکتوری

l لینک

(c و b و s و p هم میتونن باشن که چون بهشون برنمی‌خوریم، اینجا بررسی نمی‌کنیم)

پس در مثال بالا برای test و فایل png کارکتر اول d و برای بقیه که دایرکتوری هستند، d تعریف می‌شود.

9 کارکتر بعدی به سه دسته‌ی 3 تایی تقسیم می‌شوند که readable و executable را برای 3 طیف کاربر group، user و other تعیین می‌کنند. پس لازمه دسته‌بندی جدیدی برای فایل‌ها معرفی کنیم. در لینوکس 3 نوع مالکیت (Ownership) برای فایل‌ها تعریف می‌شود:

User: کاربری که مالک اصلی فایل است. به صورت پیش فرض کسی که فایل را ایجاد کرده.

Group: می‌توانید تعدادی از کاربران را در یک گروه قرار داده و مالکیت فایل را به اعضای آن بدهید.

Other: همه‌ی کاربرها. در واقع این فایل عمومی است و هر کسی می‌تواند به آن دسترسی داشته باشد.

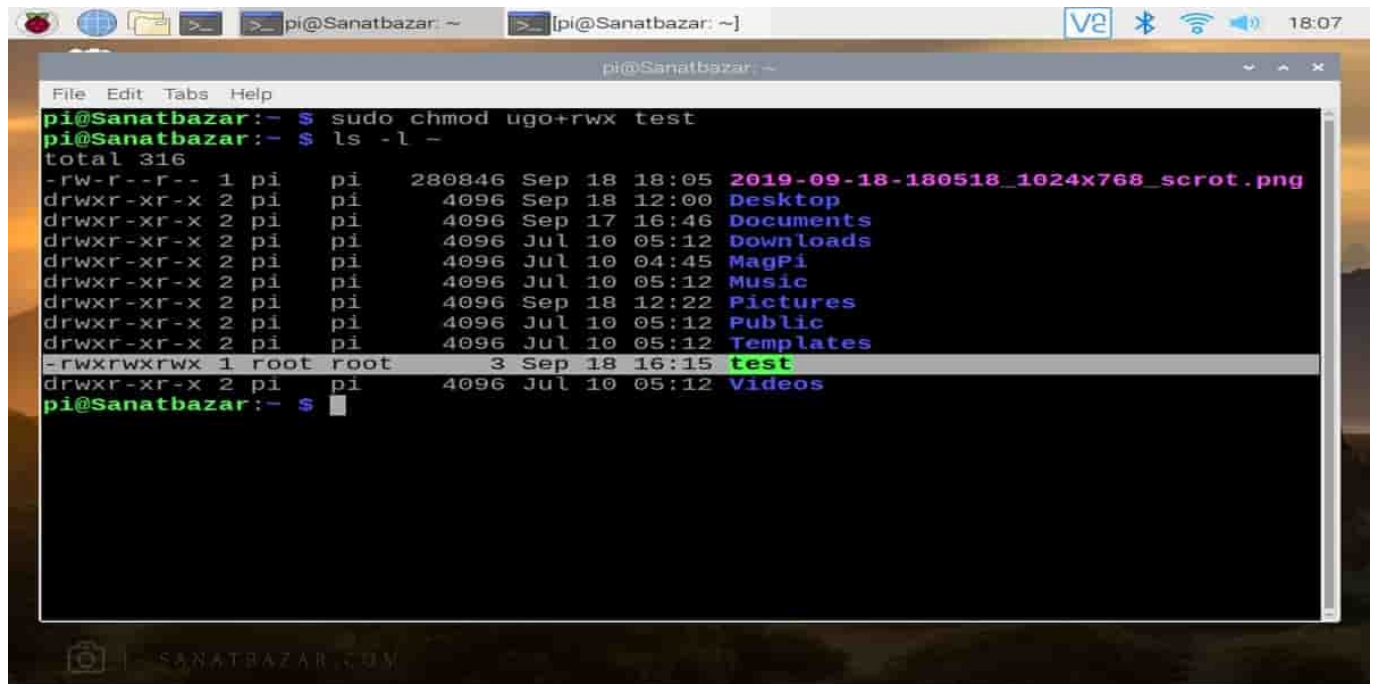
پس در این 9 کارکتر، 3 تای اول وضعیت 3 User، 3 تای دوم Group و 3 تای آخر Other را مشخص می‌کند. مثلاً برای فایل test، user توانایی Read و Write و

Group و Others فقط توانایی Read دارند.

عبارت‌های بعدی ساده‌است. عددی که بعد از کارکترهای گفته‌شده می‌بینید، تعداد Hard link های موجود از فایل را در سیستم نشان می‌دهد. دو عبارت بعدی هم نام User و Group مالک فایل را نمایش می‌دهند که برای test، چون با دستور \$ sudo nano آن را ایجاد کردیم هر دو root ثبت شده. (هر کاربر یک گروه یک نفره تشکیل می‌دهد که در صورت نیاز می‌توانید به تعداد اعضای گروه اضافه کنید) در قسمت‌های بعدی هم سایز فایل، تاریخ، زمان آخرین دسترسی و نام آن نوشته شده‌است.

حالا تازه بریم سراغ دستور chmod. با این دستور می‌خواهیم 9 کارکتری که به Permission ها معروفند، تغییر بدیم. در اینجا این کار را روی test انجام می‌دهیم: (اگر مالک فایل root نبود، نیازی به استفاده از sudo نداشتیم)

```
$ sudo chmod ugo+rwx test
```

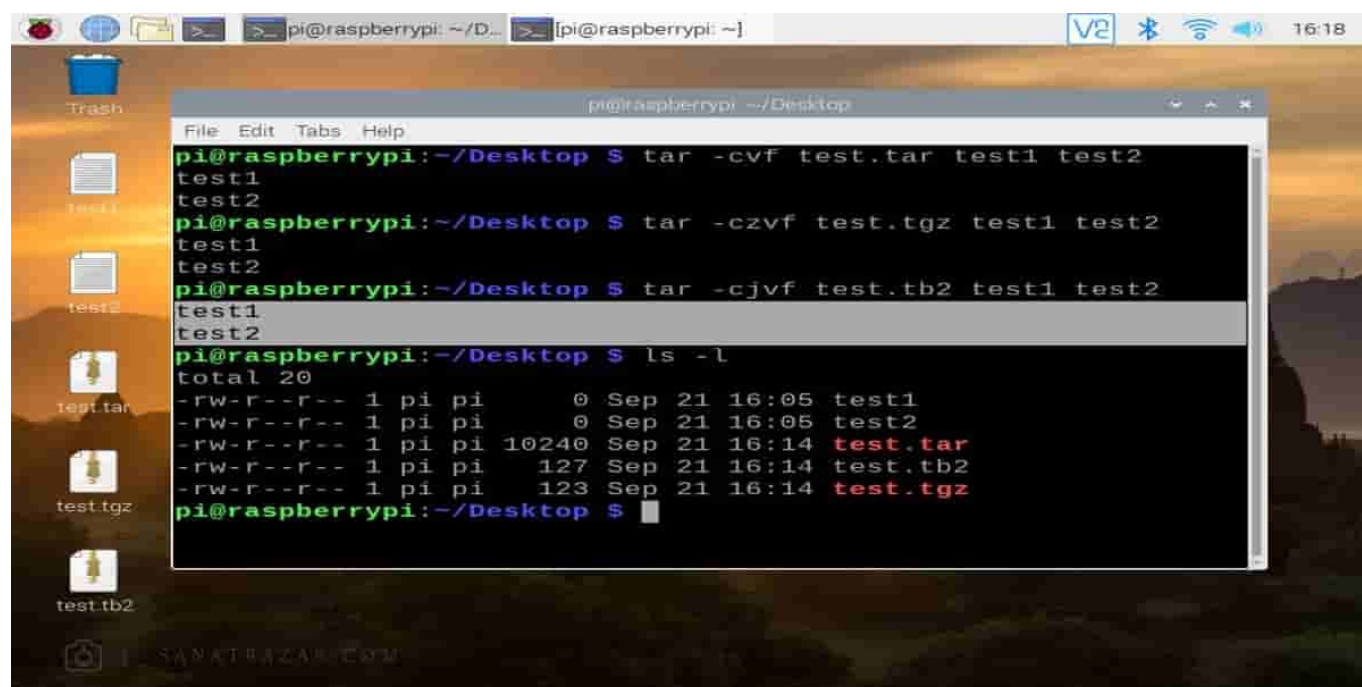


این دستور کلی‌ترین حالت را در نظر گرفته‌است. در اینجا به سه گروه (u(user)، g(group) و o(others)) اجازه‌ی w(write) و r(read) و x(execute) داده شده‌است. پس با حذف هر کدام از حروف می‌توانید دستور را مطابق نظرتان تغییر دهید. اگر به جای + از - استفاده کنید، دسترسی‌ها را از کاربران سلب می‌کنید.

tar: از این دستور برای ایجاد فایل‌های فشرده و زیپ استفاده می‌شود. مشابه دستورهای قبلی، tar را نیز در قالب مثال بررسی می‌کنیم. فرض می‌کنیم دو فایل test1 و test2 در Desktop وجود دارند:

```
$ tar -cvf test.tar test1 test2
```

برای ایجاد فایل فشرده با فرمت tar از سویچ c و f استفاده می‌کنیم. همچنین برای مشاهده‌ی نتیجه، می‌توان از سویچ v استفاده کرد. (به جای test.tar می‌توانستیم ~/Pictures/tets.tar را وارد کنیم تا فایل مورد نظر در Pictures ذخیره شود)



اگر از سویج v استفاده نمی‌کردیم، عبارت‌های test1 و test2 که هایلایت شده، نشان‌داده نمی‌شد.

برای ایجاد فایل فشرده با فرمت‌های gzip سویج z و با فرمت bz2 سویج j را به cvf اضافه می‌کنیم. bz2 نسبت به gzip و gzip نسبت به tar حجم کمتری دارند.

نکته: برای فایل gzip در نام‌گذاری از پسوند tb2 و برای bz2 از tb2 استفاده کردیم.

```
$ tar -tvf نام فایل فشرده
```

از این دستور برای لیست کردن محتویات فایل فشرده (همه‌ی فرمت‌ها) بدون Extract کردن آن استفاده می‌شود.

```
$ tar -xvf نام فایل فشرده
```

از این دستور برای Extract کردن فایل‌های فشرده (همه‌ی فرمت‌ها) استفاده می‌شود.

history: برای مشاهده‌ی دستوراتی که تا به حال اجرا کرده‌اید، می‌توانید از این دستور استفاده‌نمایید. همان‌طور که می‌بینید، کنار هر دستور، شماره‌ای نوشته‌شده که می‌توانید با وارد کردن آن به همراه ! آن دستور را اجرا کنید. مثلاً در اینجا با وارد کردن \$ 324 دستور cat test اجرا می‌شود. همچنین در خط فرمان می‌توانید با زدن کلید جهت بالا، به دستورات قبلی خود دسترسی پیدا کنید.

```

pi@Sanatbazar:~$ sudo shutdown
pi@Sanatbazar:~$ sudo shutdown -c
pi@Sanatbazar:~$ date
pi@Sanatbazar:~$ sudo shutdown -r
pi@Sanatbazar:~$ sudo shutdown -c
pi@Sanatbazar:~$ man shutdown
pi@Sanatbazar:~$ sudo shutdown -r now
pi@Sanatbazar:~$ ls -l -
pi@Sanatbazar:~$ chmod +x test
pi@Sanatbazar:~$ sudo chmod +x test
pi@Sanatbazar:~$ ls -l -
pi@Sanatbazar:~$ sudo chmod -x test
pi@Sanatbazar:~$ clear
pi@Sanatbazar:~$ history
pi@Sanatbazar:~$ ls -l -
pi@Sanatbazar:~$ nano test
pi@Sanatbazar:~$ history
pi@Sanatbazar:~$ nano test
pi@Sanatbazar:~$ cat test
pi@Sanatbazar:~$ clear
pi@Sanatbazar:~$ history
pi@Sanatbazar:~$ !325
cat test
hi
pi@Sanatbazar:~$

```

touch: یک فایل خالی و بدون پسوند ایجاد می‌کند. به عنوان مثال این فایل را با نام test می‌سازیم:

```
$ touch test
```

همچنین اگر این دستور را روی یک فایل که قبلاً ایجاد شده اجرا کنید، زمان آخرین دسترسی و تغییر آن به روزرسانی می‌شود. (این زمان در ls قابل مشاهده است) Passwd: با این دستور می‌توان رمز کاربران سیستم را تغییر داد. در صورتی که کاربران معمولی این دستور را وارد کنند، می‌توانند رمز خود، و در صورتی که root این دستور را اجرا کند، می‌تواند رمز سایر کاربران را نیز تغییر دهد.

```
$ passwd
```

تغییر رمز کاربر توسط خودش

```
$ sudo passwd pi
```

تغییر رمز کاربر pi توسط root

lsusb: مشاهده‌ی اطلاعات درباره‌ی پورتهای USB. برای مثال پس از اتصال کیبورد این دستور را اجرا می‌کنیم:

```
$ lsusb
```

```

pi@Sanatbazar: ~
$ lsusb
Bus 001 Device 006: ID 1c4f:0002 Sigma Micro Keyboard TRACER Gamma Ivory
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp. SMSC9512/9514 Fast Ethernet Adapter
Bus 001 Device 002: ID 0424:9514 Standard Microsystems Corp. SMC9514 Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub

```

lscpu: نمایش اطلاعات درباره‌ی پردازنده‌ی برد.

```
$ lscpu
```

df: نمایش اطلاعات درباره‌ی حافظه‌ی سیستم. با استفاده از سوئیچ `-h` واحدها به صورت مضربی از 1024 (MB) نمایش داده می‌شوند. حافظه‌ی قابل استفاده برای شما، همان root است که در ابتدا گفتیم با / نشان داده می‌شود.

```
$ df -h
```

```

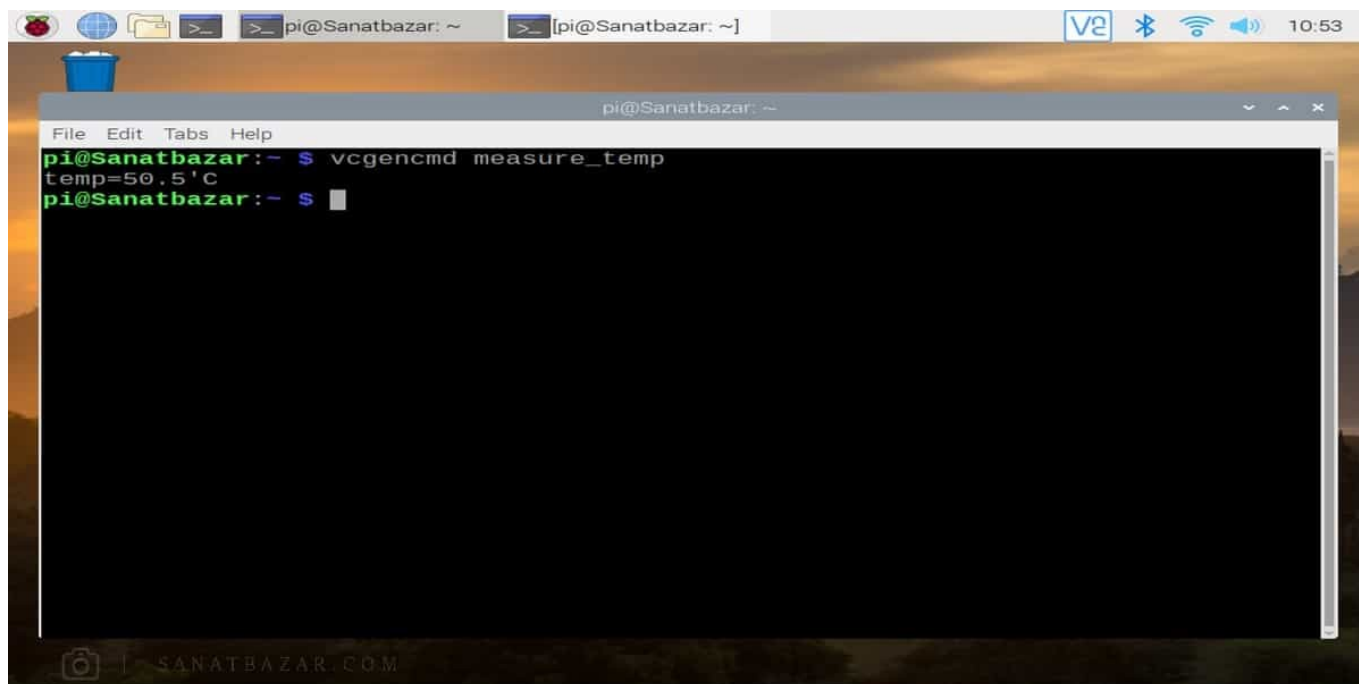
pi@Sanatbazar: ~
$ df -h

```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/root	7.0G	5.4G	1.3G	81%	/
devtmpfs	459M	0	459M	0%	/dev
tmpfs	464M	0	464M	0%	/dev/shm
tmpfs	464M	13M	451M	3%	/run
tmpfs	5.0M	4.0K	5.0M	1%	/run/lock
tmpfs	464M	0	464M	0%	/sys/fs/cgroup
/dev/mmcblk0p1	253M	40M	214M	16%	/boot
tmpfs	93M	0	93M	0%	/run/user/1000

vcgencmd measuer_temp: مشاهده‌ی دمای cpu رزبری پای

```
$ vcgencmd measure_temp
```



apt-get: دستوری فوق‌العاده مهم و کاربردی برای مدیریت پکیج‌ها. این دستور باید توسط **sudo** اجرا شود و دارای حالت‌های زیر است:

```
$ sudo apt-get update
```

با استفاده از دستور بالا اطلاعات سیستم شما نسبت به پکیج‌ها به‌روز رسانی می‌شود. به‌عنوان مثال اگر برای یک پکیج **update** جدیدی ارائه‌شود سیستم شما از آن آگاه‌شده و با دستور بعدی می‌توانید آن را به‌روزرسانی کنید.

```
$ sudo apt-get upgrade
```

به‌روزرسانی پکیج‌های نصب‌شده

```
$ sudo apt-get install نام پکیج
```

نصب پکیج مورد نظر شما

```
$ sudo apt-get-cache search نام پکیج
```

برای این که ببینید پکیج موردنظر شما برای نصب وجود دارد یا نه می‌توانید از دستور بالا استفاده کنید.

```
$ sudo apt-get remove نام پکیج
```

با استفاده از این دستور می‌توانید پکیج مورد نظرتان را حذف کنید. البته فایل‌های پیکربندی آن همچنان در سیستم شما باقی‌می‌ماند. برای پاک‌کردن پکیج و فایل‌های پیکربندی آن، می‌توانید از دستور بعدی استفاده کنید:

```
$ sudo apt-get purge نام پکیج
```

wget: این دستور برای دانلود فایل به‌صورت مستقیم از اینترنت با وارد کردن لینک آن به‌کار می‌رود.

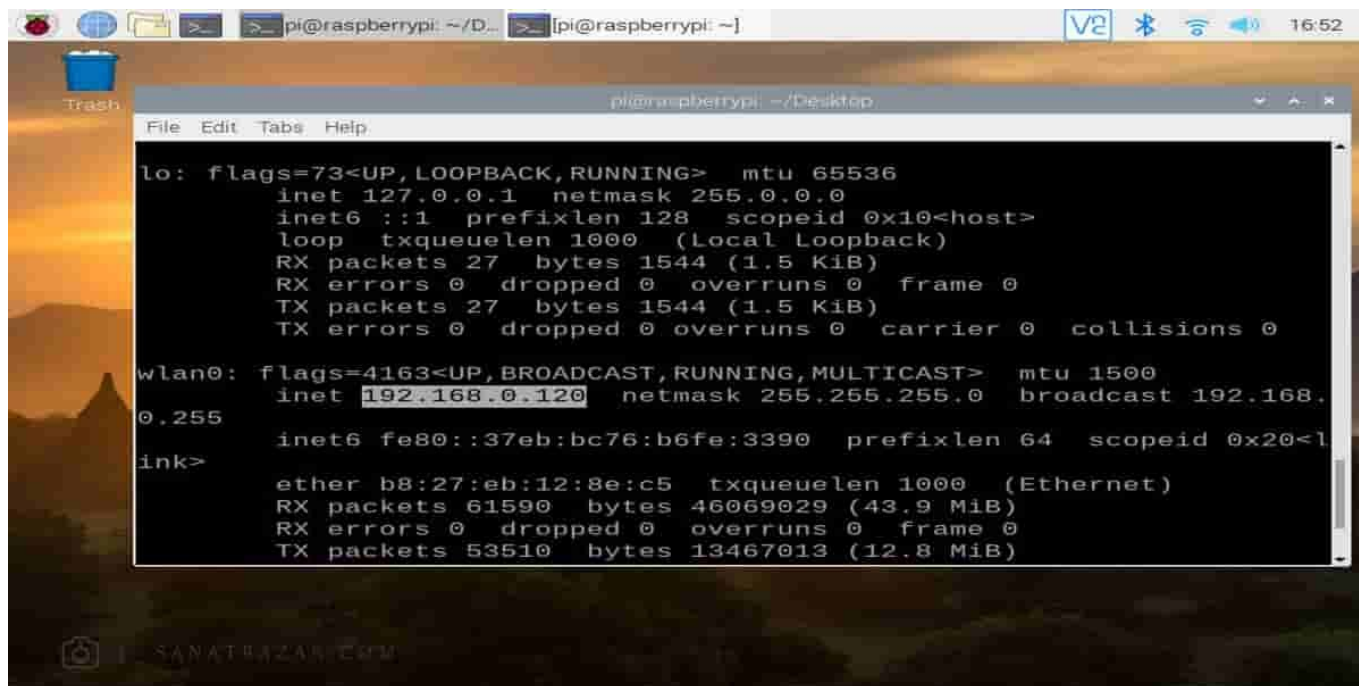
```
$ wget لینک دانلود
```

git: گیت یک سیستم کنترل و مدیریت تغییرات فایل‌ها و برنامه‌ها برای بهبود همکاری برنامه‌نویسان یک پروژه است. این سیستم قابلیت‌های زیادی دارد که از جمله آن‌ها می‌توان به اشتراک، توزیع و بازبانی کدهای نوشته‌شده به صورت عمومی و اختصاصی اشاره کرد. برای استفاده از گیت می‌توانید از سایت‌های github.com و gitlab.com و دستور \$ git در لینوکس استفاده کنید. از دستورات پرکاربرد گیت می‌توان به دستور زیر اشاره کرد:

```
$ git clone URL فایل مورد نظر در گیت
```

با استفاده از این دستور می‌توانید فایل موجود در URL گیت را دریافت کنید. برای آشنایی بیشتر با دستور \$ git می‌توانید Manual آن را با \$ man git مطالعه نمایید.

ipconfig: ifconfig معادل ipconfig در Command-Window ویندوز است و برای پیکربندی شبکه استفاده می‌شود. برای مثال در اینجا می‌خواهیم ip اختصاص داده شده به برد توسط مودم را پیدا کنیم:



برای نمایش IP می‌توانید از \$ ifconfig hostname هم استفاده نمایید.

ssh: از این دستور برای اتصال به یک سیستم عامل مبتنی بر لینوکس از راه دور با استفاده از پروتکل ssh استفاده می‌شود. نحوه‌ی استفاده از آن به صورت زیر است:

```
$ ssh Hostname یا ip @ نام کاربری
```

نام کاربری و Hostname همان عباراتی است که در Prompt نوشته شده است. به صورت پیش فرض نام کاربری رزبری پای، pi است و در بخش پیکربندی هم گفتیم که چطور Hostname خود را تغییر دهید (این کار را می‌توانید توسط raspi-config که در ادامه آن را معرفی می‌کنیم نیز انجام دهید). همچنین به جای Hostname می‌توانید IP رزبری پای را وارد کنید و به آن متصل شوید. مثال:

```
$ ssh pi@192.168.5.1
```

(به جای 192.168.5.1 باید IP رزبری خود را وارد کنید)

از این دستور می‌توانید به صورت برعکس برای دسترسی به یک سیستم لینوکسی از رزبری پای نیز استفاده کنید.

scp: با استفاده از scp می‌توانید هر فایل را با استفاده از پروتکل ssh به یک سیستم لینوکسی دیگر منتقل کنید.

```
$ scp Username@Hostnamr: مقصد نام و آدرس فایل در مبدا
```

برای مثال می‌خواهیم فایل test را از Desktop در کامپیوتر به Documents در رزبری پای انتقال دهیم:

```
$ scp ~/Desktop/test pi@192.168.1.5: /Documents/test
```

اگر در مقصد آدرسی را تعیین نکنید، فایل را به Home Directory انتقال می‌دهد.

Python: پایتون 2 و 3 به صورت پیش فرض روی رزبین لینوکس نصب شده‌اند. برای ورود به shell پایتون کافیس `$ python` را وارد کنید. (`$ python3` برای پایتون 3)

همچنین برای نوشتن script و اجرای آن می‌توانید کد پایتون خود را در nano وارد کرده و سپس آن را با دستور زیر اجرا کنید.

`$ python` نام فایل متنی حاوی کد

برای اجرای پایتون 3 کافیس به جای `python`، `python3` را بنویسید.

`++G`: مشابه پایتون می‌توانید از زبان `++C` نیز استفاده کنید. در اینجا پس از نوشتن کدها توسط دستور `nano` لازم است دستورات زیر را اجرا کنید. فرض می‌کنیم کد خود را با نام `test.cpp` ذخیره کرده‌ایم. (برای کامپایل فایل `++c` لازم است کد خود را با پسوند `cpp` ذخیره کنید)

```
$ g++ test.cpp -o test2
```

فایل کامپایل شده را با نام `test2` ذخیره و با دستور زیر اجرا می‌کنیم:

```
$ ./test2
```

`adduser`: از این دستور برای اضافه کردن حساب کاربری جدید به سیستم عامل استفاده می‌شود. برای این کار از کد زیر استفاده می‌کنیم:

```
$ sudo adduser نام کاربر جدید
```

به عنوان مثال در اینجا کاربر `arvin` را ایجاد می‌کنیم:

```
$ sudo adduser arvin
```

سپس رمز عبور و اطلاعات تکمیلی را وارد می‌کنیم.

از این به بعد می‌توانید با کاربر جدیدی به جز `pi` نیز وارد برد شوید. همچنین در دایرکتوری `Home`، علاوه بر `arvin`، `pi` نیز اضافه شده‌است.

`deluser`: همانطور که حدس زدید از این دستور برای پاک کردن حساب کاربری استفاده می‌شود.

```
$ sudo deluser نام کاربر
```

نکته: با استفاده از این دستور، `Home directory` کاربر پاک نمی‌شود و فایل‌های آن همچنان باقی می‌ماند. برای پاک کردن کاربر و `Home directory` آن باید از سویچ `remove-home` استفاده کنید.

```
$ sudo deluser -remove-home نام کاربر
```

`Ctrl+C` و `Ctrl+Z`: در سیستم عامل‌های لینوکسی برای متوقف کردن برنامه‌ها از کلیدهای `Ctrl+C` و `Ctrl+Z` استفاده می‌شود. اما فرق این دو کلید چیه؟

`Ctrl+C`: در حین انجام پردازش یا اجرای برنامه‌ها در لینوکس، با فشار دادن کلیدهای `Control+C` فرمان متوقف شدن آن دستور یا برنامه صادر می‌شود. (در ویندوز تقریباً این کار را با `Alt+Ctrl+del` و `Task Manager` انجام می‌دهیم.)

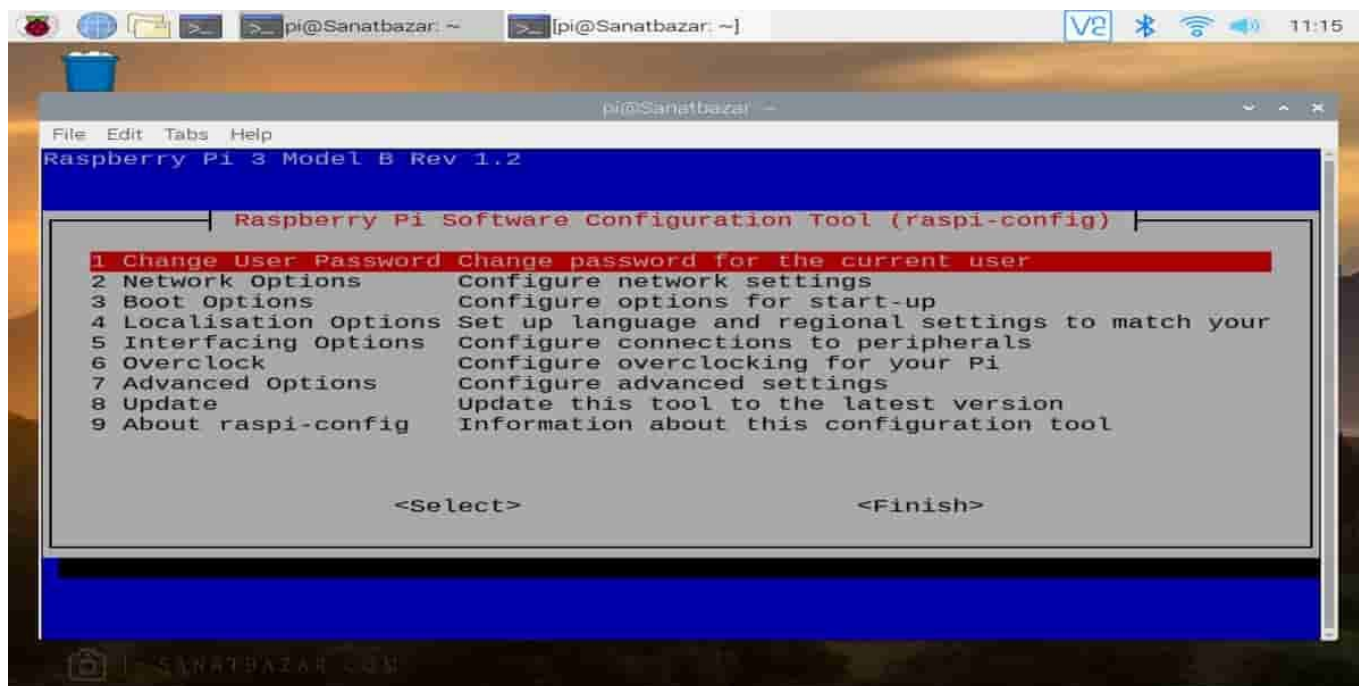
`Ctrl+Z`: برنامه را متوقف می‌کند اما کامل نمی‌بندد. در واقع می‌توانید دوباره برنامه را از هرجا که متوقف کردید، ادامه دهید. اصطلاحاً برنامه به پس زمینه فرستاده می‌شود. برای ادامه‌ی آن از دستور زیر استفاده می‌کنیم.

```
$ fg
```

raspi-config: با این دستور همان پیکربندی‌هایی که در محیط گرافیکی دیدیم و حتی بیشتر را می‌توانید انجام دهید. برای اجرای آن نیاز به دسترسی root و sudo دارید.

```
$ sudo raspi-config
```

با اجرای دستور بالا، این صفحه نمایش داده خواهد شد:



1- در بخش اول می‌توانید رمز عبور خود را تغییر دهید.

2- در بخش دوم تنظیمات شبکه شامل :

- تعیین Hostname
- اتصال به مودم Wi-Fi
- فعال یا غیرفعال کردن Predictable network interface name

3- تنظیمات Boot شامل :

- بوت به CLI یا GUI
- بوت پس از اتصال به شبکه
- Splash screen

4- تنظیمات محل، زمان و کیبورد

5- تنظیمات رابط‌های سریال و SSH و دوربین

6- تنظیمات Overclock

7- تنظیمات پیشرفته شامل:

- تخصیص پارتیشن اضافه به root directory. در نسخه‌ی اخیر رزبین این کار به صورت خودکار انجام می‌شود.
- تنظیمات Underscan و Overscan
- تقسیم RAM بین CPU و GPU
- تنظیمات رابط صوتی

- تنظیمات صفحه نمایش
- بزرگنمایی تصویر با دوبرابر کردن سایز هر پیکسل
- فعال و غیرفعال کردن کتابخانه‌ی گرافیکی OpenGL Driver (مشابه DirectX)
- فعال و غیرفعال کردن Composite Manager (برای اجرای برخی پکیج‌های زیباسازی دسکتاپ و گرافیکی)
- تنظیمات نمایش ویدئو برای رزبری پای 4 با توجه به دو خروجی HDMI

8- به روزرسانی

9- درباره‌ی دستور

قرار دادن دستور در Startup سیستم‌عامل:

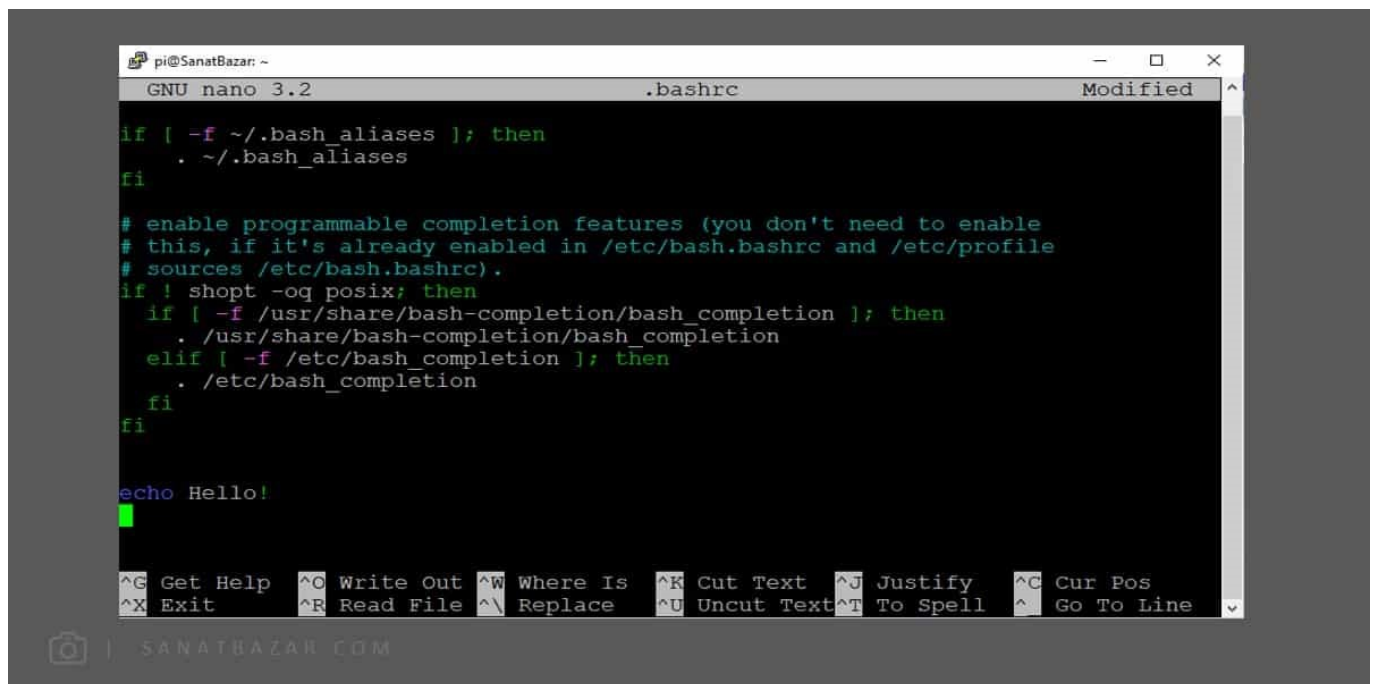
زمانی که پروژه‌ای را با رزبری پای انجام می‌دهید، برنامه‌ی شما باید این توانایی را داشته‌باشد تا به‌صورت خودکار پس از روشن‌شدن، اجرا شود. در غیر این صورت کاربر باید هر بار به برد متصل و به‌صورت دستی برنامه را اجرا کند که اصلاً منطقی نیست! برای این کار لازم است که برنامه‌ی خود را در فایل `bashrc` در `HomeDirectory` بنویسید. بنابراین به دایرکتوری ~ رفته و با استفاده از دستور `sudo`، فایل `bashrc` را بخوانید. سپس در انتهای متن می‌توانید دستور مورد نظر را بنویسید. برای مثال می‌خواهیم هر بار پس از روشن‌شدن، عبارت `Hello!` را ببینیم:

```
$ sudo nano .bashrc
```

و در انتهای فایل دستور زیر را وارد می‌کنیم:

```
echo Running at boot

echo Hello!
```



به‌صورت پیش‌فرض، دستورات قرار داده‌شده در `bashrc` پس از هر بار باز کردن Terminal یا برقراری ارتباط به‌صورت SSH اجرا خواهند شد. حالا برای این که این دستورات پس از boot شدن نیز اجرا شوند، دستور `echo Running at boot` را نیز اضافه می‌کنیم. از این پس، بعد از هر بار روشن‌شدن برد، دستور `$ echo Hello!` اجرا خواهد شد. به‌طور مشابه می‌توان دستور مورد نظر را در `/etc/rc.local/` قبل از `exit 0` و همراه با علامت `&` نیز نوشت.

```
echo Hello! &

exit 0
```

علامت `&` برای این است که دستور مورد نظر شما به‌صورت مستقل و موازی با عملیات boot اجرا شود. زیرا اگر این دستور طولانی باشد، boot شدن برد را با مشکل

روبه‌رو خواهدکرد.

خب تا اینجا با دستورات مهم و کاربردی لینوکس به‌خصوص برای کار با رزبری پای آشنا شدیم. با توجه به این که از این به بعد فقط با محیط Command-line کار می‌کنیم، به‌خاطر داشتن دستورات بررسی‌شده بسیار مهم است.

در بخش بعدی می‌خواهم روشی را به شما معرفی کنم که بدون نیاز به LCD و مانیتور و تنها با استفاده از لپ‌تاپ خود بتوانید با کابل یا به‌صورت بی‌سیم به دسکتاپ و Command-Line رزبری پای دسترسی داشته باشید. پس با من همراه باشید.

نظرات شما باعث بهبود محتوای آموزشی ما می‌شود. اگر این آموزش را دوست داشتید، همین‌طور اگر سوالی در مورد آن دارید، از شنیدن نظراتتان خوشحال خواهیم شد.