



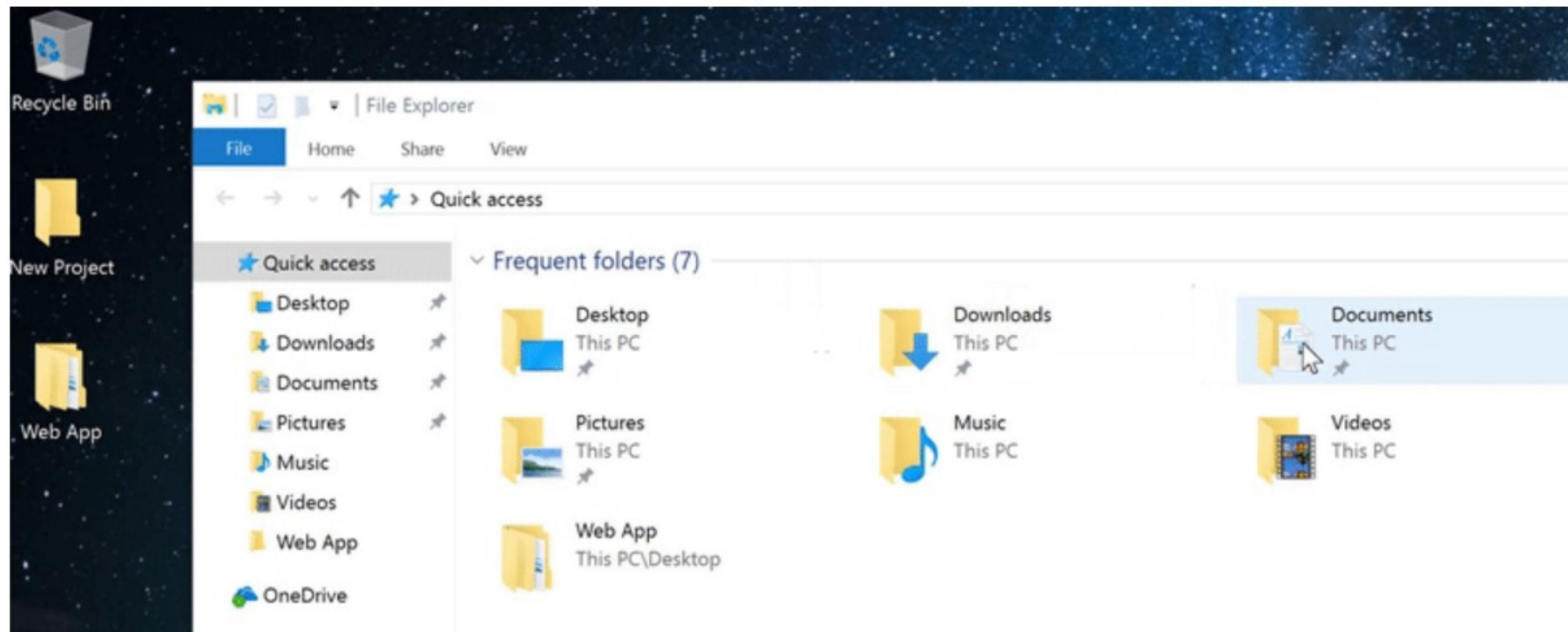
¿Qué es Linux?

Antes de determinar qué hace que Linux sea una gran adición a su repertorio de habilidades y conocimientos, definámoslo. Linux es un sistema operativo que se ejecuta en un sistema informático o hardware (ordenador). Un sistema operativo es un software que permite instalar y ejecutar otros programas como procesadores de texto o navegadores web en un ordenador.

Su teléfono celular, tableta, portátil y computadora de escritorio necesitan un sistema operativo para ejecutar el software (a menudo llamado aplicaciones) que desea utilizar. Esto significa que Linux no está limitado sólo a computadoras de escritorio y portátiles. Ampliaremos este punto a medida que avancemos con nuestra discusión.

¡La razón principal por la que aprender Linux es útil es que Linux se utiliza en casi todas partes! Linux se utiliza en computadoras de escritorio y portátiles, servidores web, dispositivos móviles (Android), tecnología de nube pública (i.e., Google, Amazon, etc.), Chromebooks y redes (i.e., redes Cisco). Aunque es posible que nunca haya utilizado Linux en un ordenador de escritorio o portátil, **es probable que esté utilizando otra tecnología que funciona con Linux**, por ejemplo un teléfono móvil Android, una tableta o páginas web populares como Facebook, Twitter o Amazon. De hecho, se estima que más de la mitad de las páginas web en Internet se generan a partir de servidores Linux.

Para entender por qué Linux ofrece competencias únicas, examinemos la interfaz gráfica de usuario (GUI, *graphic user interface*). Esta es la interfaz que probablemente usted está usando en su smartphone, tableta o computadora hoy mismo. La GUI muestra iconos e imágenes que usted puede seleccionar para indicar a su dispositivo lo que desea hacer o utilizar. Detrás de la GUI, existe un código desarrollado por un programador. Cuando usted hace clic sobre un icono o carpeta en la GUI, se envía un comando al código que indica al sistema qué debe hacer.



Los ordenadores Linux utilizan una GUI, pero también poseen una herramienta más eficiente para llevar a cabo las mismas acciones que una GUI, la interfaz de línea de comandos (CLI, *command line interface*).



```
bsmith@localhost:~$ ls ~/Documents
```

```
School
```

```
Web_App
```

```
Work
```

```
adjectives.txt
```

```
alpha-first.txt
```

```
bsmith@localhost:~$
```

La línea de comandos de Linux es una interfaz basada en texto que acepta comandos que se escriben en ella. Estos comandos hacen que una acción se ejecute en el sistema operativo del equipo. Por supuesto, las ventanas y los iconos son fáciles de usar, sin embargo, la línea de comandos es a menudo el héroe cuando se trata de la administración del sistema y la solución de problemas, ya que proporciona una imagen clara de lo que el sistema está haciendo en cualquier momento dado.

Los sistemas operativos Linux vienen en muchas formas. Hay una variedad de distribuciones disponibles para adaptarse a las necesidades y demandas de muchos sectores de TI. Por ejemplo, los profesionales de la ciberseguridad pueden usar Linux Kali, los programadores y diseñadores pueden usar Linux Ubuntu, los usuarios habituales pueden usar Linux Mint y los servidores empresariales pueden funcionar con Red Hat Enterprise Linux.

Sintaxis de comandos básicos

Este módulo se ocupa exclusivamente de la CLI o interfaz de línea de comandos, en lugar de la GUI o interfaz gráfica de usuario con la que quizás esté más familiarizado. El terminal CLI es una poderosa herramienta y a menudo es el método principal utilizado para administrar dispositivos pequeños de bajo consumo, servidores de computación de gran capacidad en la nube, y mucho más. Una comprensión básica del terminal es esencial para diagnosticar y reparar la mayoría de los sistemas basados en Linux. Puesto que Linux se ha vuelto tan omnipresente, incluso aquellos que planean trabajar con sistemas que no utilizan el núcleo Linux pueden beneficiarse de tener una comprensión básica del terminal.

¿Qué es un comando? Un comando es un programa de software que, cuando se ejecuta en la CLI (interfaz de línea de comandos), realiza una acción en el ordenador. Cuando usted escribe un comando, el sistema operativo ejecuta un proceso para leer su entrada, manipular datos y producir resultados. Un comando ejecuta un proceso en el sistema operativo, que luego hace que el ordenador realice una tarea determinada.

Para ejecutar un comando, el primer paso es escribir el nombre del comando. Haga clic en el terminal de la derecha. Escriba `ls` (letras minúsculas **L** y **S**) y pulse **Enter**. Obtendrá un resultado parecido al del siguiente ejemplo:

```
sysadmin@localhost:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
```

Generalmente, el nombre del comando se basa en la tarea que hace o en lo que el programador que creó el comando cree que mejor describe la función del comando. Por ejemplo, el comando `ls` muestra una lista de información sobre archivos. Asociar el nombre del comando con algo mnemotécnico sobre lo que hace puede ayudarle a recordar los comandos más fácilmente.

A tener en cuenta

Generalmente, los comandos distinguen entre mayúsculas y minúsculas. Por ejemplo `LS` es incorrecto y generará un mensaje de error, pero `ls` es correcto y se ejecutará normalmente.

```
sysadmin@localhost:~$ LS
-bash: /usr/games/LS: Permission denied
```

La mayoría de los comandos siguen un patrón de sintaxis simple:

```
comando [opciones...] [argumentos...]
```

En otras palabras, escriba un comando, seguido de las *opciones* y/o *argumentos* antes de presionar la tecla **Enter**. Generalmente, las opciones (*options*) alteran el comportamiento del comando y los argumentos (*arguments*) son elementos o valores sobre los que debe actuar el comando. Aunque hay algunos comandos en Linux que no son completamente consistentes con estas normas de sintaxis, la mayoría de los comandos usan esta sintaxis o alguna similar.

En el ejemplo anterior, el comando `ls` se ejecutó sin opciones ni argumentos. Cuando este es el caso, su comportamiento predeterminado es el de devolver una lista de los archivos contenidos en el directorio actual.


```
sysadmin@localhost:~$ ls
```

```
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
```

Argumentos

```
comando [opciones...] [argumentos...]
```

Un argumento (*argument*) se puede usar para especificar algo sobre lo que el comando debe actuar. Si al comando `ls` se le da el nombre de un directorio como argumento, obtendremos como resultado una lista del contenido de ese directorio. En el siguiente ejemplo, el directorio `Documents` se utilizará como argumento:

```
sysadmin@localhost:~$ ls Documents
School      alpha-second.txt  food.txt          linux.txt         os.csv
Work        alpha-third.txt   hello.sh          longfile.txt      people.csv
adjectives.txt alpha.txt          hidden.txt        newhome.txt       profile.txt
alpha-first.txt animals.txt        letters.txt       numbers.txt       red.txt
```

El resultado es una lista de los archivos incluidos en el directorio `Documents`.

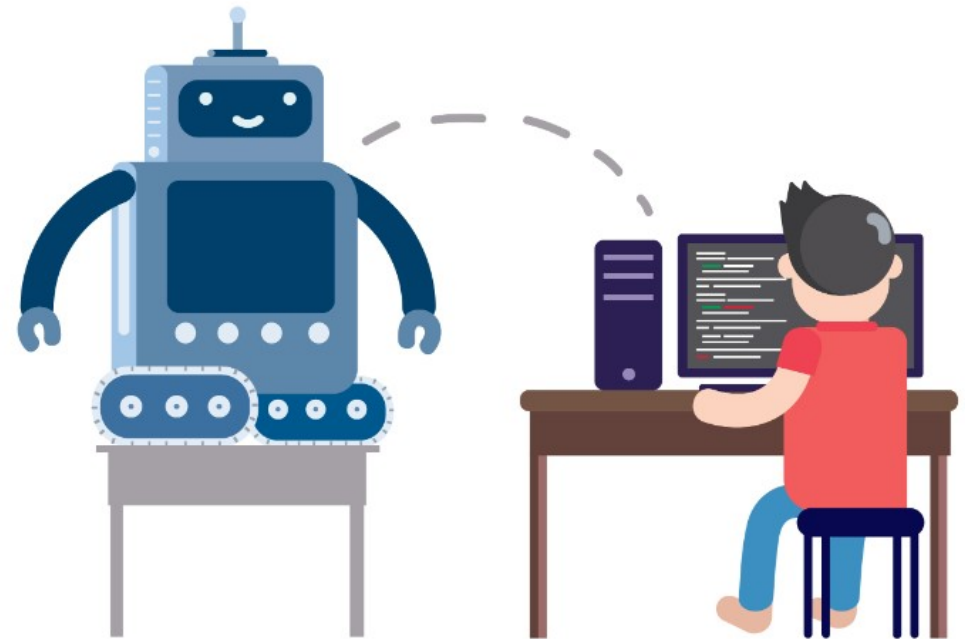
Debido a que Linux es de código abierto, contiene algunas funciones curiosas que han ido siendo agregadas por sus programadores y usuarios. Por ejemplo, el comando `aptitude` es una función de gestión de paquetes disponible en algunas versiones de Linux. Este comando acepta `mo` como argumento:

```
sysadmin@localhost:~$ aptitude moo  
There are no Easter Eggs in this program.
```

Este comando no solamente es lo que parece. ¡Siga leyendo para saber qué más hay detrás de este truco!

Linux is Open Source

Linux is developed by a
community, you view and
contribute to the source code!



«Linux se basa en código abierto (open source). Linux está diseñado y desarrollado en comunidad. ¡Usted puede acceder y contribuir a su código fuente (source code)!»

Opciones

```
comando [opciones...] [argumentos...]
```

Las opciones (*options*) se pueden utilizar para modificar el comportamiento de un comando. En la página anterior, el comando `ls` se utilizó para enumerar el contenido de un directorio. En el ejemplo siguiente, la opción `-l` se agrega al comando `ls` para obtener un resultado de “pantalla larga”, y proporcionar más información sobre cada uno de los archivos enumerados:

```
sysadmin@localhost:~$ ls -l
total 32
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Desktop
drwx----- 4 sysadmin sysadmin 4096 Dec 20 2017 Documents
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Downloads
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Music
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Pictures
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Public
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Templates
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Videos
```

Tenga en cuenta que, en el comando anterior, `-l` es la letra "L" minúscula.

A menudo, el carácter elegido para el comando es mnemotécnico de su propósito en inglés. Por ejemplo, la letra l para indicar *largo* (*long*) o r para *invertir* (*reverse en inglés*). De forma predeterminada, el comando `ls` imprime los resultados en orden alfabético, al agregar la opción `-r` se imprimirán los resultados en orden alfabético inverso.

```
sysadmin@localhost:~$ ls -r
Videos  Templates  Public  Pictures  Music  Downloads  Documents  Desktop
```

Se pueden usar varias opciones a la vez, ya sea como opciones separadas como en `-l -r` o combinadas como `-lr`. El resultado de los siguientes ejemplos sería el mismo:

```
ls -l -r
ls -rl
ls -lr
```

Como se ha explicado anteriormente, `-l` proporciona un formato de listado largo y `-r` invierte el listado. El resultado de usar ambas opciones será un listado largo en orden alfabético inverso:

Como se ha explicado anteriormente, `-l` proporciona un formato de listado largo y `-r` invierte el listado. El resultado de usar ambas opciones será un listado largo en orden alfabético inverso:

```
sysadmin@localhost:~$ ls -l -r
total 32
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Videos
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Templates
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Public
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Pictures
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Music
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Downloads
drwx----- 4 sysadmin sysadmin 4096 Dec 20 2017 Documents
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Desktop
sysadmin@localhost:~$ ls -rl
total 32
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Videos
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Templates
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Public
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Pictures
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Music
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Downloads
drwx----- 4 sysadmin sysadmin 4096 Dec 20 2017 Documents
drwx----- 2 sysadmin sysadmin 4096 Dec 20 2017 Desktop
```

Los comandos pueden utilizar muchas combinaciones de opciones y argumentos. Las posibilidades para cada comando serán únicas. ¿Recuerda los huevos de Pascua (*Easter Eggs*) del comando `aptitude`?

```
sysadmin@localhost:~$ aptitude moo
There are no Easter Eggs in this program.
```

Es posible modificar el comportamiento de este comando usando opciones. Vea lo que sucede cuando se agrega la opción `-v` (*verbose*):

```
sysadmin@localhost:~$ aptitude -v moo
There really are no Easter Eggs in this program.
```

Mediante la combinación de múltiples opciones `-vv`, podemos obtener una variedad de respuestas:

```
sysadmin@localhost:~$ aptitude -vv moo
Didn't I already tell you that there are no Easter Eggs in this program?
sysadmin@localhost:~$ aptitude -vvv moo
Stop it!
```

Recuerde que las varias opciones se pueden denotar por separado o combinadas:

```
aptitude -v -v moo
aptitude -vv moo
```


¡Siga añadiendo opciones  para ver cuántas respuestas únicas puede obtener!

Press the [Enter] key to begin...

```
sysadmin@localhost:~$ aptitude moo
```

There are no Easter Eggs in this program.

```
sysadmin@localhost:~$ aptitude -v moo
```

There really are no Easter Eggs in this program.

```
sysadmin@localhost:~$ aptitude -vv moo
```

Didn't I already tell you that there are no Easter Eggs in this program?

```
sysadmin@localhost:~$ aptitude -vvv moo
```

Stop it!

```
sysadmin@localhost:~$ aptitude -vvvv moo
```

Okay, okay, if I give you an Easter Egg, will you go away?

```
sysadmin@localhost:~$ aptitude -vvvv moo
```

Okay, okay, if I give you an Easter Egg, will you go away?

```
sysadmin@localhost:~$ aptitude -vvvvv moo
```

All right, you win.

```
sysadmin@localhost:~$
```