Línea Línea

Apuntes Vladimir Farrera Vera

Git y GitHub

Linea de comandos para Z Shell o Bash Shell





Aprendido en Platzi

Escuela de Desarrollo web

Línea Línea

**Command Line Cheat Sheet**

**Descriptores de rutas :**

**/**  Ruta raíz del sistema

**.** Directorio actual

**..** Directorio anterior

**~** Directorio home del usuario

**Atajos de teclado :**

**CTRL - C**  Termina el proceso de un comando en la terminal

**CTRL - D**  Termina el input de un comando

**CTRL - A**  Avanza al inicio de linea

**CTRL - E** Avanza al final de linea

**CTRL - L** Limpia la pantalla de la terminal

Nota : Los siguientes comandos no funcionan en Windows debido a que los comandos power Shell son diferentes 😔

**Operaciones con directorios :**

**pwd**  Imprime el directorio actual

**mkdir** ejemplo1 Crea el directorio con nombre “ejemplo1” (una carpeta)

**cd** ejemplo1 Cambia al directorio con nombre “ejemplo1”

**cd** ../..  Cambia dos directorios anteriores al actual

**cd** Cambia al directorio home del usuario

**ls** Muestra archivos y directorios

**tree** /ruta Muestra todos los archivos y directorios anidados dentro de la ruta a cualquier profundidad (No funciona en Mac que usa Z Shell 😭).

**tree** -L 2 . Muestra todos los archivos y directorios anidados dentro de la ruta actual a dos niveles de profundidad (No funciona en Mac que usa Z Shell 🥲).

**Operaciones de ls :**

**-a**  Muestra todo (incluyendo archivos ocultos)

**-R** Muestra una lista de manera recursiva (Muestra todos los archivos que están en cada carpeta en la que estas actualmente o apuntando)

**-r** Muestra listado de forma inversa

**-t** Muestra los últimos modificados

**-s** Muestra ordenando por tamaño

**-l** Muestra usando un formato largo

**Manipulación de archivos y directorios :**

**touch** nuevofile Crea un nuevo archivo llamado “nuevofile”

**file** mi\_archivo Muestra las características de mi\_archivo

**cp** file1 /destino Copia el archivo file1 a la ruta /destino

**cp** -r dir1 dir1\_cp Copia el directorio dir1 y su contenido a uno nuevo llamado dir1\_cp

**mv** file1 ./destino Mueve el archivo file1 a la ruta /destino (Puede o no llevar el punto 🤷🏾‍♂️ debido a qué solo indica que partir de la carpeta actual)

**mv** file1 .. Mueve el archivo file1 un directorio atrás

**mv** file1 ok\_file Renombre el archivo file1 al nombre ok\_file

**rm** file1 Elimina el archivo file1 (Cuidado con lo que borras 🥶 ya no aparece en el basurero)

**rm -ri** dir1 Elimina el directorio dir1 y su contenido de forma interactiva (Solo te pregunta si estas seguro e indicas si o no, 🤔)

**ln -s** file link\_name Crea un link symbolic al archivo (Sirve como para darle un apodo al archivo y puedes manipularlo llamándolo con el apodo 🤪)

**open** filename Abre el archivo con el programa predeterminado (Para MacOS que usa Z Shell 🤩)

**xdg-open** filename Abre el archivo con el programa predeterminado (Mayoria de sistemas Linux 😎)

**Manipulación de archivos de texto :**

**head** file.txt Muestra las primeras 10 lineas de texto del archivo file.txt

**head -n 20** file.txt Muestra las primeras 20 lineas de texto del archivo file.txt (Tu indicas cuantas lineas mostrar 🤪 )

**tail** file.txt Muestra las ultimas 10 lineas de texto del archivo file.txt

**tail -n 20** file.txt Muestra las ultimas 20 lineas de texto del archivo file.txt (Tu indicas cuantas lineas mostrar 🤪 )

**less** file.txt Explorar el contenido de archivo con paginación (para salir del visor de texto solamente presiona la tecla “q” 😃)

**cat** file1 Muestra el contenido completo del texto

**cat** file1 file1 Concatena el contenido de file1 y file2 a la salida de la terminal (Te muestra en terminal el contenido completo primero file1 y luego file2 seguido 👷🏾🧕🏽)

**Exploración de comandos y ayuda dentro de la terminal :**

**man** command Muestra el manual de usuario del comando

**help** command Muestra ayuda para el comando (Solo funciona si la Shell es Bash 😔)

**which** command Muestra la ruta de donde se encuentra el ejecutable del comando 🕵🏽

**whatis** command Muestra brevemente la función del comando

**alias** aliasname = “command” Crea un alias para el comando (un apodo temporal para el comando por si lo usas mucho 😯)

Ejemplo :

**alias** l = “ls - la”

Crea un alias para el comando ls con opciones llamado l 😮

**Wildcars**

**\*** Coincide con cualquier caracter

**?** Coincide con cualquier caracter individual

**[caracteres]**  Coincide con cualquier carácter que sea un miembro del conjunto de caracteres

**[!caracteres]**  Coincide con cualquier carácter que no sea miembro del conjunto caracteres

**[[:clase:]]**  Coincide con cualquier carácter de la clase

Wildcards :

**Complementado con el comando “ls”nos ayuda a buscar archivos de forma mas especifica**😮

Ejemplo

**ls \***.jpg Mostrar todos los .jpg

**ls** datos**?** Mostrar todos los archivos que tenga almenas esos caracteres

**ls** [ad]**\*** Mostrar todos los archivos que inicien con a o d

**Clases dentro de los Wildcars :**

**[:alnum:]** Coincide con cualquier carácter alfanumérico

**[:alpha:]** Coincide con cualquier carácter alfabético

**[:digit:]** Coincide con cualquier numero

**[:lower]** Coincide con cualquier letra mayúscula

**[:upper:]** Coincide con cualquier letra mayúscula

Ejemplo

**ls -d [:upper:]\*** Mostrar todos los archivos y carpetas que usen mayusculas, así mismo se usa la “-d” para indicar que también directorios 🤔

**Redireccion I/O y operadores de control :**

**Comando < archivo** Redirige el input de un comando hacia un archivo

**Comando > archivo**  Redirige la salida de un comando a un archivo (usarse con cuidado por que sobreescribe el sistema 😵)

**Comando >> archivo** Redirige la salida de un comando a un archivo. Si no existe lo crea (Cocatenamos y podemos seguir llenando el archivo 🤗)

**Comando 2> error.txt** Redirigie la salida de error de un comando al archivo error.txt

**Comando1 | Comando2** Redirige la salida del comando1 a la entrada del comando2

**Comando1; Comando2** Ejecuta de manera consecutiva

**Comando1 & Comando2** Ejecuta de manera asíncrona

**Comando1 && Comando2** Ejecuta el Comando2 si y solo si el comando1 se ejecuto de manera exitosa

**Comando1 || Comando2** Ejecuta el Comando1 o el Comando2

Ejemplo

**ls -lh > salida.txt** Guarda los resultados en un archivo de texto🤔

**ls dfjjf 2> error.txt** Guarda los errores que salen en consola

**Manejo de permisos y usuarios :**

**Tipo de modo :**

| Dueño | | | Grupo | | | World | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| rwx | | | r - x | | | r - x | | |
| 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |

**Modo Octal :**

| Octal | Binario | Permisos |
| --- | --- | --- |
| 0 | 000 | - - - |
| 1 | 001 | - - x |
| 2 | 010 | - w - |
| 3 | 011 | - w x |
| 4 | 100 | r - - |
| 5 | 101 | r w - |
| 6 | 110 | r w - |
| 7 | 111 | r w x |

**Modo simbolico :**

| Simbolo | Significado |
| --- | --- |
| u | Solo para el usuario |
| g | Solo para el grupo |
| o | Solo para otros (World) |
| a | Aplica para todos |

**chmod 755** mitexto Cambia los permisos a 755

**chmod u - r** mitexto Quita el permiso de lectura al archivo de usuario

**chmod u = rwx, go = r** mitexto Agrega permiso de lectura, escritura y ejecución al usuario y solo de lectura al grupo y otros

**id** Muestra el ID del usuario

**whoami** Muestra el nombre del usuario logueado

**su** username Cambia de usuario

**sudo** comando Ejecuta un comando como super usuario 💪🏽

**Variables de entorno :**

**env** Imprime las variables de entorno actuales

**echo $VAR**  Imprime la variable de entorno VAR

**$PATH** Variable donde están las rutas de los ejecutables

**$HOME** Directorio home del usuario 🤷🏾‍♂️

**export** $VAR = val Asigna la variable $VAR con el valor val 🙃

**Comandos de búsqueda :**

**find** <ruta> **-name** <nombre> Busca en una ruta y con un nombre de archivo

**find . -name** agenda Busca el archivo con el nombre agenda en el directorio actual

**grep** <regex> file Busca con expresiones regulares dentro de un archivo o salida de terminal

**grep -i** hola mensaje.txt Busca la palabra hola ignorando minúsculas y mayusculas

**Utilidades de red :**

**ifconfig** Muestra información de red

**ping** ip\_domain Consulta la disponibilidad del recurso

**curl** ip\_domain Consuta un recurso ya sea por dirección IP o por dominio y lo muestra en terminal

**wget** ip\_domain Descarga un recurso ya sea por dirección IP o por dominio

**traceroute** ip- domain Muestra la ruta del paquete a una IP o dominio

**netstat -i** Muestra las interfaces de red y su estado

**Comprimir archivos :**

**tar -czvf** <nombre>**.tar.gz** <nombre\_directorio>

**Ejemplo :**

**tar -czvf** mis\_archivos**.tar.gz** Documentos

**zip -r** copressed.zip Documents/toCompress/ (La -r se usa si es una carpta la que se comprimira 🤔)

**Descomprimir archivos :**

**tar -xzvf** <nombre>**.tar.gz** <nombre\_directorio>

Ejemplo :

**tar -xzvf** mis\_archivos**.tar.gz** Documentos

**unzip** compressed.zip

**Significado**

c : comprimir

x : descomprimir

z : especifica que tiene la extencion “tgz”o “.tar.gz”

v : Muestra lo que se hace

**Comprimir en zip**

**zip** comprimir

**unzip** Descomprimir



**GIT**