

103.1 Trabajar en la línea de comandos - Ejercicios

Ejercicio 3.1.1: Ejecutando Comandos Básicos y Usando Opciones

- **Objetivo:** Ejecutar comandos simples y experimentar con sus opciones cortas y largas.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal.
 2. **Comando sin opciones ni argumentos:** Ejecuta `pwd` (Print Working Directory). Te muestra el directorio actual.
 3. **Comando con opciones cortas:** Ejecuta `ls -l`. Muestra el contenido del directorio actual en formato largo.
 4. **Comando con múltiples opciones cortas agrupadas:** Ejecuta `ls -la`. Muestra todos los archivos (incluidos ocultos, `-a`) en formato largo (`-l`). Es lo mismo que `ls -l -a`.
 5. **Comando con opciones largas:** Ejecuta `ls --long --all`. Es lo mismo que el paso anterior.
 6. **Comando con argumento:** Ejecuta `ls /home`. Lista el contenido del directorio `/home`.
 7. **Comando con opciones y argumentos:** Ejecuta `ls -lh /tmp`. Lista el contenido de `/tmp` en formato largo (`-l`), con tamaños legibles por humanos (`-h`).

Ejercicio 3.1.2: Identificando Tipos de Comandos y Explorando PATH

- **Objetivo:** Distinguir entre comandos internos y externos y ver cómo la shell los localiza.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal.
 2. **Identifica el tipo de `cd`:** Ejecuta `type cd`. Debería decir que es un "shell builtin" (comando interno de la shell).
 3. **Identifica el tipo de `ls`:** Ejecuta `type ls`. Debería decir que es un alias o, si no hay alias para `ls`, que es un programa externo y mostrar su ruta (ej: `/usr/bin/ls`).
 4. **Identifica el tipo de `echo`:** Ejecuta `type echo`. Puede ser tanto un comando interno como un programa externo. La shell ejecuta primero el interno por eficiencia.
 5. **Visualiza la variable `PATH`:** Ejecuta `echo $PATH`. Verás una lista de directorios separados por dos puntos (`:`). Estos son los directorios que la shell busca en orden para encontrar ejecutables.
 6. **Simula la búsqueda de un comando:** Si ejecutaras `grep`, la shell lo buscaría en cada directorio listado en `PATH` hasta encontrar el primero llamado `grep` (generalmente en `/usr/bin` o `/bin`).

Ejercicio 3.1.3: Usando el Historial de Comandos y Completado por Tabulación

- **Objetivo:** Utilizar el historial de comandos y el completado por tabulación para mejorar la eficiencia.

- **Desarrollo Paso a Paso:**

1. Abre una terminal.
2. **Ejecuta algunos comandos:** Ejecuta `date`, `cal`, `pwd`, `ls -l /etc`.
3. **Navega el historial:** Presiona la tecla de flecha Arriba repetidamente para ver los comandos anteriores. Presiona Abajo para volver a los más recientes.
4. **Re-ejecuta el último comando:** Presiona flecha Arriba una vez y luego Enter, o simplemente escribe `!!` y Enter.
5. **Busca y re-ejecuta un comando reciente:** Escribe `!p` y Enter. Esto ejecutará el último comando que empezó con `p` (probablemente `pwd`).
6. **Muestra el historial completo:** Ejecuta `history`. Anota el número de un comando que ejecutaste (ej: el número de `cal`).
7. **Re-ejecuta un comando por su número:** Escribe `! (ej: !10) y Enter.`
8. **Practica el completado por tabulación:**
 - Escribe `cd /e` y presiona Tab. Debería completarse a `/etc/`.
 - Escribe `ls /etc/pa` y presiona Tab. Te mostrará las opciones que empiezan con "pa" en `/etc/`. Presiona Tab de nuevo para ver la lista si hay varias. Escribe `sswd` (para `passwd`) y presiona Tab. Debería completarse a `/etc/passwd`.
 - Escribe `export MI_VARIA` y presiona Tab. Si has creado una variable que empieza así, la completará.
 - Escribe `apt in` y presiona Tab (en sistemas Debian/Ubuntu). Te mostrará los comandos que empiezan con `in` dentro de `apt`.

Ejercicio 3.1.4: Creando y Usando Aliases

- **Objetivo:** Crear y utilizar aliases temporales.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal.
 2. **Define un alias simple:** Ejecuta `alias ll='ls -lsa'`. Esto crea un alias llamado `ll` que ejecuta `ls -lsa`.
 3. **Usa el alias:** Ejecuta `ll`. Verás la salida de `ls -lsa`.
 4. **Lista tus aliases:** Ejecuta `alias`. Verás la lista de aliases definidos, incluyendo el que acabas de crear.
 5. **Define un alias con argumentos:** Ejecuta `alias homefiles='ls -l ~/.*'`. Este alias listará archivos ocultos en tu directorio home.
 6. **Usa el alias:** Ejecuta `homefiles`.
 7. **Elimina un alias:** Ejecuta `unalias ll`.
 8. **Verifica que se eliminó:** Ejecuta `alias`. `ll` ya no debería aparecer. Intenta ejecutar `ll`; debería fallar (`command not found`).
 9. **Nota:** Los aliases definidos de esta manera son temporales y solo existen en la shell actual. Para hacerlos permanentes, debes agregarlos a un archivo de configuración de

la shell como `~/ .bashrc` y luego "fuente" ese archivo (`source ~/ .bashrc`) o abrir una nueva terminal.

Ejercicio 3.1.5: Trabajando con Comillas y Escape de Caracteres

- **Objetivo:** Entender el efecto de las comillas simples, dobles y la barra invertida.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal.
 2. **Crea una variable con espacio:** Ejecuta `MI_TEXTO="Ho la Mundo"`.
 3. **Imprime la variable sin comillas:** Ejecuta `echo $MI_TEXTO`. Debería mostrar `Ho la Mundo`.
 4. **Imprime la variable con comillas dobles:** Ejecuta `echo "$MI_TEXTO"`. Debería mostrar `Ho la Mundo`. Las comillas dobles preservaron el espacio.
 5. **Crea una variable con caracteres especiales:** Ejecuta `MI_OTRA_VAR="$USER en $(hostname)"`. La variable contiene el valor de `$USER` y la salida del comando `hostname`.
 6. **Imprime la variable con comillas dobles:** Ejecuta `echo "$MI_OTRA_VAR"`. Verás el valor expandido (ej: `tu_usuario en tu_hostname`). Las comillas dobles permitieron la expansión.
 7. **Imprime la variable con comillas simples:** Ejecuta `echo '$MI_OTRA_VAR'`. Verás `$` y `$(hostname)` literales: `$MI_OTRA_VAR en $(hostname)`. Las comillas simples *no* permiten la expansión.
 8. **Escapa un espacio:** Crea un directorio con un espacio (¡solo para este ejercicio! No es buena práctica): `mkdir "Mi Directorio con Espacio"`.
 9. **Navega al directorio usando barra invertida:** Ejecuta `cd Mi\ Directorio\ con\ Espacio`. La barra invertida escapa cada espacio.
 10. **Navega al directorio usando comillas:** Ejecuta `cd "Mi Directorio con Espacio"`. Las comillas dobles protegen todo el nombre.
 11. **Limpia:** Ejecuta `cd ..` para salir del directorio. Luego `rmdir "Mi Directorio con Espacio"`.

Ejercicio 3.1.6: Obteniendo Ayuda en la CLI

- **Objetivo:** Usar `man`, `apropos` y `--help`.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal.
 2. **Obtén la página de manual para `ls`:** Ejecuta `man ls`. Lee la descripción (NAME), la sinopsis (SYNOPSIS) que muestra el formato de uso, y explora las opciones (DESCRIPTION). Presiona `q` para salir.
 3. **Busca páginas de manual relacionadas:** Ejecuta `apropos network`. Verás una lista de comandos cuyas man pages están relacionadas con "network".
 4. **Obtén ayuda rápida para un comando:** Ejecuta `cp --help`. Verás un resumen de uso y las opciones más comunes.

5. **Obtén ayuda para un comando interno:** Ejecuta `he lp cd`.
6. **Nota:** La cantidad de información en las man pages puede ser abrumadora al principio. No necesitas memorizarlo todo; lo importante es saber cómo acceder a la ayuda cuando la necesitas.