

Cadenas en Bash

1. `echo "El home del usuario ${USER} es:" $HOME`
 - Imprime en pantalla el mensaje "El home del usuario [nombre_usuario] es: [ruta_home]".
 - `${USER}` es una variable de entorno que almacena el nombre del usuario.
 - `$HOME` es la variable que almacena el directorio home del usuario.
2. `echo "abc def" | grep -o .`
 - Muestra cada carácter de la cadena "abc def" en una línea separada.
 - `grep -o .` busca y muestra cada coincidencia individualmente.
3. `for c in $(echo "zyx wvu" | grep -o .); do echo "$c"; done`
 - Recorre cada carácter de "zyx wvu" y lo imprime en una nueva línea.
 - `grep -o .` separa los caracteres.
4. `cadena="Brian Fox comenzó a codificar Bash el 10 de enero de 1988"`
 - Asigna la cadena a la variable `cadena`.
 - `echo ${#cadena}` muestra la longitud de la cadena.
5. `subcadena=${cadena:10:7} && echo $subcadena`
 - Extrae 7 caracteres desde la posición 10 de la variable `cadena` y los muestra.
6. `subcadena=${cadena:35} && echo $subcadena`
 - Extrae la parte de la cadena desde la posición 35 hasta el final.
7. `echo ${cadena: -4}`
 - Muestra los últimos 4 caracteres de `cadena`.
8. `echo ${cadena: -13:8}`
 - Extrae 8 caracteres empezando desde la posición -13 (contando desde el final).
9. `echo ${cadena: -13:-8}`
 - No es un rango válido en Bash, generará un error.
10. `echo ${subcadena^}`
 - Convierte la primera letra de `subcadena` en mayúscula.
11. `echo ${subcadena^^}`
 - Convierte toda `subcadena` a mayúsculas.
12. `echo ${cadena,}`
 - Convierte la primera letra de `cadena` a minúscula.
13. `echo ${cadena,,}`
 - Convierte toda `cadena` a minúsculas.

Comando sed

14. `echo "hola mundo" | sed "s/mundo/${USER}/"`
 - Reemplaza "mundo" por el nombre de usuario.
15. `ls -l $HOME > home.txt && sed 's/./texto/' home.txt`
 - Lista los archivos en el home del usuario y lo guarda en `home.txt`.
 - `sed 's/./texto/' home.txt` intenta reemplazar "." por "texto", pero solo muestra el resultado en la terminal.
16. `sed -i "s/${USER}/usuario/" home.txt && cat home.txt`
 - Modifica `home.txt` reemplazando el nombre del usuario por "usuario".
17. `ls -l $HOME > home.txt && sed -i.copia "s/./texto/" home.txt && cat home.txt`
 - Guarda una copia de `home.txt` como `home.txt.copia` antes de modificarlo.
 - Reemplaza "." por "texto" en `home.txt`.

Comando find

18. `cd $HOME/Downloads && find`
 - Muestra todos los archivos y carpetas dentro de Downloads.
19. `find -type f`
 - Busca y muestra solo los archivos en el directorio actual y subdirectorios.
20. `find -type d`
 - Busca y muestra solo los directorios en el directorio actual y subdirectorios.

21. `find -type f -name "*.txt"`
 - Busca archivos con extensión ".txt" en el directorio actual y subdirectorios.
22. `find $HOME/Desktop -name "*.txt"`
 - Busca archivos con extensión ".txt" en el escritorio del usuario.

Cadenas en Bash

1. `echo "El home del usuario ${USER} es:" $HOME`
 - Imprime en pantalla el mensaje "El home del usuario [nombre_usuario] es: [ruta_home]".
 - `${USER}` es una variable de entorno que almacena el nombre del usuario.
 - `$HOME` es la variable que almacena el directorio home del usuario.
2. `echo "abc def" | grep -o .`
 - Muestra cada carácter de la cadena "abc def" en una línea separada.
 - `grep -o .` busca y muestra cada coincidencia individualmente.
3. `for c in $(echo "zyx wvu" | grep -o .); do echo "$c"; done`
 - Recorre cada carácter de "zyx wvu" y lo imprime en una nueva línea.
 - `grep -o .` separa los caracteres.
4. `cadena="Brian Fox comenzó a codificar Bash el 10 de enero de 1988"`
 - Asigna la cadena a la variable `cadena`.
 - `echo ${#cadena}` muestra la longitud de la cadena.
5. `subcadena=${cadena:10:7} && echo $subcadena`
 - Extrae 7 caracteres desde la posición 10 de la variable `cadena` y los muestra.
6. `subcadena=${cadena:35} && echo $subcadena`
 - Extrae la parte de la cadena desde la posición 35 hasta el final.
7. `echo ${cadena: -4}`
 - Muestra los últimos 4 caracteres de `cadena`.
8. `echo ${cadena: -13:8}`
 - Extrae 8 caracteres empezando desde la posición -13 (contando desde el final).
9. `echo ${cadena: -13:-8}`
 - No es un rango válido en Bash, generará un error.
10. `echo ${subcadena^}`
 - Convierte la primera letra de `subcadena` en mayúscula.
11. `echo ${subcadena^^}`
 - Convierte toda `subcadena` a mayúsculas.
12. `echo ${cadena,}`
 - Convierte la primera letra de `cadena` a minúscula.
13. `echo ${cadena,,}`
 - Convierte toda `cadena` a minúsculas.

Comando sed

14. `echo "hola mundo" | sed "s/mundo/${USER}/"`
 - Reemplaza "mundo" por el nombre de usuario.
15. `ls -l $HOME > home.txt && sed 's/.txt/.texto/' home.txt`
 - Lista los archivos en el home del usuario y lo guarda en `home.txt`.
 - `sed 's/.txt/.texto/' home.txt` intenta reemplazar ".txt" por ".texto", pero solo muestra el resultado en la terminal.
16. `sed -i "s/${USER}/usuario/" home.txt && cat home.txt`
 - Modifica `home.txt` reemplazando el nombre del usuario por "usuario".
17. `ls -l $HOME > home.txt && sed -i.copia "s/.txt/.texto/" home.txt && cat home.txt`
 - Guarda una copia de `home.txt` como `home.txt.copia` antes de modificarlo.
 - Reemplaza ".txt" por ".texto" en `home.txt`.

Comando find

18. `cd $HOME/Downloads && find`
 - Muestra todos los archivos y carpetas dentro de Downloads.
19. `find -type f`

- Busca y muestra solo los archivos en el directorio actual y subdirectorios.
- 20. `find -type d`
 - Busca y muestra solo los directorios en el directorio actual y subdirectorios.
- 21. `find -type f -name "*.txt"`
 - Busca archivos con extensión ".txt" en el directorio actual y subdirectorios.
- 22. `find $HOME/Desktop -name "*.txt"`
 - Busca archivos con extensión ".txt" en el escritorio del usuario.
- 23. `find $HOME/Desktop -mmin 20 -type f`
 - Busca archivos en \$HOME/Desktop que hayan sido modificados exactamente hace 20 minutos.
- 24. `find $HOME/Desktop -mmin -20 -type f`
 - Busca archivos en \$HOME/Desktop modificados en los últimos 20 minutos.
- 25. `find $HOME/Desktop -mmin +20 -type f`
 - Busca archivos en \$HOME/Desktop modificados hace más de 20 minutos.
- 26. `find $HOME/Desktop -atime 20 -type f`
 - Busca archivos en \$HOME/Desktop que hayan sido accedidos hace 20 días.
- 27. `find $HOME/Desktop -type f -atime 1`
 - Busca archivos en \$HOME/Desktop accedidos en los últimos 1 día.
- 28. `find $HOME/Desktop -type f -atime -1`
 - Busca archivos en \$HOME/Desktop accedidos en menos de 1 día.
- 29. `mkdir -p $HOME/articulos && cd $HOME/articulos && touch articulo{1..25}.txt && touch venta-{1..5}.csv && mkdir -p cierre-caja_{1..12} && ls`
 - Crea el directorio articulos, 25 archivos .txt, 5 archivos .csv y 12 directorios cierre-caja #.
- 30. `find $HOME/articulos -type f -name "*.csv" -or -type d`
 - Busca archivos .csv y directorios dentro de articulos.
- 31. `find $HOME/articulos -type f -name "*.csv" -exec rm -f {} \;`
 - Elimina todos los archivos .csv dentro de articulos.

Comando `chown`

- 32. `chown root /home/{su-nombre-de-usuario}/articulos/cierre-caja_1`
 - Cambia el propietario del directorio cierre-caja_1 a root.
- 33. `chown root:root /home/{su-nombre-de-usuario}/articulos/cierre-caja_2`
 - Cambia el propietario y grupo de cierre-caja_2 a root.
- 34. `chown -R {su-nombre-de-usuario}:{su-nombre-de-usuario} /home/{su-nombre-de-usuario}/articulos/cierre-caja_2`
 - Cambia recursivamente el propietario y grupo de cierre-caja_2 al usuario actual.

Comando `chmod`

- 35. `chmod 750 /home/{su-nombre-de-usuario}/articulos/art*.txt`
 - Da permisos 750 a los archivos art*.txt.
- 36. `chmod -R 771 /home/{su-nombre-de-usuario}/articulos`
 - Cambia recursivamente los permisos del directorio articulos a 771.