

A01 - Redirecciones

[Información sobre el documento.....1](#)

[1. - Introducción..... 1](#)

[2. - Preguntas..... 2](#)

[3. - Solución..... 3](#)

Información sobre el documento

El objetivo de este documento es enseñar, compartir conocimientos para facilitar el aprendizaje. Este documento es mejorable, y será actualizado si es preciso.

Este documento puede ser utilizado para uso personal, no comercial, como se presenta, respetando una serie de **condiciones de uso basadas en Licencia Creative Commons**, como aparece en el logotipo, que se describe a continuación



- El **documento se proporciona como está**, por tanto *no se pueden realizar modificaciones*, ni en el **formato** ni en el **contenido**, ni trabajos derivados, sin la autorización expresa del autor.
- Se debe **mencionar al autor** del mismo, por supuesto sin modificar los enlaces o imágenes introducidas por éste en el documento.
- **No se puede utilizar** este material **con fines lucrativos, comerciales o cualquier uso que pueda proporcionar, directa o indirectamente, un beneficio económico de terceros**, sin la autorización expresa del autor.
- Si se desea publicar el documento en algún sitio web, se debe hacer a través de un **enlace al documento en el sitio del autor** (www.educatica.es o cursos.educatica.es)

Si se desea publicar como un recurso dentro de un sitio web, sin utilizar un enlace al material en el sitio web del autor, se debe solicitar autorización expresa y, en cualquier caso, referenciar el sitio web del autor (www.educatica.es).

Espero sea de provecho ;)

1. - Introducción

El objetivo de esta actividad es trabajar con redirecciones de salida, de error y de entrada en sistemas GNU/Linux con shell bash utilizando los comandos que ya hemos estudiado de GNU/Linux que ya hemos estudiado.

De esta forma, afianzamos el uso de redirecciones y el trabajo con los comandos de GNU/Linux. Si tienes dudas sobre los comandos puedes consultar los [contenidos de comandos GNU/Linux en el sitio web de educatica](#).



2. - Preguntas

Inicia una nueva terminal desde GNOME.

1. Crea un fichero llamado readMe.txt cuyo contenido sea el nombre del usuario actual. Tendrás que utilizar variables de entorno para obtener este valor.
2. Añade al fichero readMet.txt la fecha y hora actual
3. Añade al fichero readMe.txt las cuatro últimas líneas del fichero que almacena información de configuración de cuentas de usuario del sistema
4. Muestra el contenido del fichero readMet.txt
5. Almacena en el fichero listado.txt el listado de ficheros que hay en tu directorio personal
6. Añade al final del fichero listado.txt el contenido del fichero readMe.txt
7. Almacena en el fichero procesos.txt el listado de todos los procesos que se están ejecutando en el sistema.
8. Añade al fichero procesos.txt el nombre del usuario actual. Deberás utilizar una variable.
9. Añade al fichero procesos.txt la ruta del directorio personal del usuario actual. Deberás utilizar una variable.
10. Almacena en un fichero llamado usuarios.txt el listado de usuarios que actualmente tienen iniciada sesión en el sistema.
11. Añade al fichero usuarios.txt el contenido del fichero procesos.txt.
12. Añade al fichero usuarios.txt la cadena de texto Usuarios actuales
13. Añade al fichero usuarios.txt información de identificación del usuario actual.
14. Añade al fichero usuarios.txt información de identificación del usuario marinapg, si hubiera algún error redirige el error al fichero errores.txt.
15. Añade al fichero usuarios.txt información de identificación del usuario jesusrp, si hubiera algún error redirige el error al fichero errores.txt.
16. Almacena en un fichero llamado root.txt el contenido del directorio /root (no utilices sudo) y la salida de error redirigela a errores.txt
17. Utilizando el comando sudo, añade al fichero root.txt el contenido del directorio /root y redirige la salida de error al fichero errores.txt
18. Añade al fichero allErrors.txt el nombre del usuario actual.
19. Añade al fichero allErrors.txt la fecha y hora actuales
20. Añade al fichero allErrors.txt el contenido del fichero errores.txt
21. Borra el fichero errores.txt
22. Crea un directorio llamado como el usuario actual, dentro del directorio backup dentro del directorio raíz del sistema. Si se produce algún error, que se redirija en el fichero errores.txt
23. Copia el directorio /home y todo su contenido en el directorio /backup. Redirige la salida estándar al fichero backup.log y redirige la salida de error al fichero errores.txt



24. Añade al fichero backup.log el listado del directorio /backup y todo su contenido de forma recursiva con información extendida de permisos propietario y grupo.

3. -Solución

1. Crea un fichero llamado readMe.txt cuyo contenido sea el nombre del usuario actual. Tendrás que utilizar variables de entorno para obtener este valor.

Si no recordamos las variables de entorno, podemos consultarlas con el comando env.

```
javier@ssoo-100:~$ env
SHELL=/bin/bash
WINDOWID=0
QT_ACCESSIBILITY=1
XDG_CONFIG_DIRS=/etc:/etc/xdg:/usr/share
XDG_MENU_PREFIX=lxqt-
LANGUAGE=
LC_ADDRESS=es_ES.UTF-8
LC_NAME=es_ES.UTF-8
SSH_AUTH_SOCK=/tmp/ssh-aWpDE560bN6i/agent.1154
XDG_DATA_HOME=/home/javier/.local/share
XDG_CONFIG_HOME=/home/javier/.config
LXQT_SESSION_CONFIG=session
LC_MONETARY=es_ES.UTF-8
SSH_AGENT_PID=1214
XDG_SEAT=seat0
PWD=/home/javier
LOGNAME=javier
```

```
USER=javier
COLORFGBG=15;0
DISPLAY=:1
SHLVL=2
LC_TELEPHONE=es_ES.UTF-8
LC_MEASUREMENT=es_ES.UTF-8
```

Hay un par de variables que podemos usar para consultar el nombre del usuario actual, USER y LOGNAME. Vamos a utilizar USER.

```
javier@ssoo-100:~$ echo $USER > readMe.txt
javier@ssoo-100:~$ ls readMe.txt
readMe.txt
javier@ssoo-100:~$ cat readMe.txt
javier
```

Podemos ver que el fichero se ha creado correctamente y que su contenido es el nombre del usuario actual.

2. Añade al fichero readMe.txt la fecha y hora actual

```
javier@ssoo-100:~$ date >> readMe.txt
javier@ssoo-100:~$ cat readMe.txt
javier
mié 03 feb 2021 11:08:16 CET
```

3. Añade al fichero readMe.txt las cuatro últimas líneas del fichero que almacena información de configuración de cuentas de usuario del sistema

```
javier@ssoo-100:~$ tail -4 /etc/passwd >> readMe.txt
javier@ssoo-100:~$ cat readMe.txt
javier
mié 03 feb 2021 11:08:16 CET
vboxadd:x:998:1::/var/run/vboxadd:/bin/false
alfredoff:x:1001:1002:,,,:/home/alfredoff:/bin/bash
marinapg:x:1002:1003:,,,:/home/marinapg:/bin/bash
ramonam:x:1003:1004:,,,:/home/ramonam:/bin/bash
```

Usamos el comando tail con la opción -4 para mostrar solo las últimas cuatro líneas del fichero de configuración. La ruta del fichero de configuración de cuentas de usuario del sistema es /etc/passwd. Por último, usamos una doble redirección para añadir la salida estándar del comando al fichero readMe.txt

4. Muestra el contenido del fichero readMe.txt

```
javier@ssoo-100:~$ cat readMe.txt
javier
mié 03 feb 2021 11:08:16 CET
vboxadd:x:998:1::/var/run/vboxadd:/bin/false
alfredoff:x:1001:1002:,,,:/home/alfredoff:/bin/bash
marinapg:x:1002:1003:,,,:/home/marinapg:/bin/bash
ramonam:x:1003:1004:,,,:/home/ramonam:/bin/bash
```

5. Almacena en el fichero listado.txt el listado de ficheros que hay en tu directorio personal

```
javier@ssoo-100:~$ ls > listado.txt
javier@ssoo-100:~$ cat listado.txt
castelar
Descargas
Desktop
Documentos
Imágenes
listado.txt
Música
Plantillas
Público
readMe.txt
saludo.txt
Vídeos
```

Como nos dicen que almacenemos información, y no nos dicen que añadamos esta información al fichero, vamos a utilizar una redirección simple. Esta redirección es de sobrescritura, es decir si el fichero listado.txt existe se perderá la información que tuviera previamente.

6. Añade al final del fichero listado.txt el contenido del fichero readMe.txt

Como hasta ahora, con las redirecciones, podemos añadir o escribir en un fichero lo que un comando al ejecutarse mostraría en la salida estándar. Es decir, si somos capaces de mostrar información en la pantalla, entonces somos capaces de escribirla en un fichero.

Por tanto, tenemos que utilizar un comando capaz de mostrar el contenido del fichero readMe.txt y redirigir la salida al fichero listado.txt.

```
javier@ssoo-100:~$ cat readMe.txt >> listado.txt
javier@ssoo-100:~$ cat listado.txt
castelar
Descargas
Desktop
Documentos
Imágenes
listado.txt
Música
Plantillas
Público
readMe.txt
saludo.txt
Vídeos
javier
mié 03 feb 2021 11:08:16 CET
vboxadd:x:998:1::/var/run/vboxadd:/bin/false
alfredoff:x:1001:1002:,,,:/home/alfredoff:/bin/bash
marinapg:x:1002:1003:,,,:/home/marinapg:/bin/bash
ramonam:x:1003:1004:,,,:/home/ramonam:/bin/bash
```

7. Almacena en el fichero procesos.txt el listado de todos los procesos que se están ejecutando en el sistema.

Utilizamos el comando ps con la opción -A para mostrar en pantalla todos los procesos en ejecución en el sistema. Redirigimos la salida estándar con redirección simple, puesto que no nos indican que añadamos al fichero en cuestión.

```
javier@ssoo-100:~$ ps -A > procesos.txt
javier@ssoo-100:~$ cat procesos.txt
  PID TTY          TIME CMD
    1 ?            00:00:01 systemd
    2 ?            00:00:00 kthreadd
    3 ?            00:00:00 rcu_gp
    4 ?            00:00:00 rcu_par_gp
    6 ?            00:00:00 kworker/0:0H-kblockd
    9 ?            00:00:00 mm_percpu_wq
   10 ?            00:00:00 ksoftirqd/0
```

8. Añade al fichero procesos.txt el nombre del usuario actual. Deberás utilizar una variable.

Para mostrar el nombre del usuario actual utilizaremos la variable \$USER que almacena el nombre del usuario con el que estamos trabajando. Para mostrar una cadena de texto usamos el comando echo y, ahora sí, utilizamos redirección doble para añadir la cadena de texto al final del fichero.

```
javier@ssoo-100:~$ echo $USER >> procesos.txt
javier@ssoo-100:~$ tail -3 procesos.txt
 5455 ?            00:00:00 kworker/u2:0-events_power_efficient
 5456 pts/0        00:00:00 ps
javier
```

Para comprobar que todo ha ido como esperabamos, vamos a mostrar las tres últimas líneas del fichero procesos.txt con el comando tail y la opción -3.

9. Añade al fichero procesos.txt la ruta del directorio personal del usuario actual. Deberás utilizar una variable.

Usamos el comando echo, de nuevo, para mostrar una cadena de texto en la salida estándar y redirigirla al fichero procesos.txt con redirección doble para añadir la salida al final del fichero.

```
javier@ssoo-100:~$ echo ~ >> procesos.txt
```

Podemos utilizar la variable ~ o bien la variable de entorno HOME. Ambas contienen la ruta del directorio personal del usuario actual.

```
javier@ssoo-100:~$ echo $HOME >> procesos.txt
```

10. Almacena en un fichero llamado usuarios.txt el listado de usuarios que actualmente tienen iniciada sesión en el sistema.

Para mostrar los usuarios que actualmente tienen iniciada sesión en el sistema utilizamos el comando who. Redirigimos la salida estándar al fichero usuarios.txt con redirección simple puesto que no nos indican que añadamos información en el fichero si existía.

```
javier@ssoo-100:~$ who > usuarios.txt
javier@ssoo-100:~$ cat usuarios.txt
javier  tty1      2021-01-28 10:37 (:0)
javier  tty3      2021-01-28 10:42
```

11. Añade al fichero usuarios.txt el contenido del fichero procesos.txt.

Para añadir el contenido del fichero procesos.txt al fichero usuarios.txt, mostramos el contenido del fichero procesos.txt en la salida estándar y redirigimos, on doble redirección, la salida estándar al fichero usuarios.txt

```
javier@ssoo-100:~$ cat procesos.txt >> usuarios.txt
javier@ssoo-100:~$ cat usuarios.txt
javier  tty1      2021-01-28 10:37 (:0)
javier  tty3      2021-01-28 10:42
  PID TTY          TIME CMD
   1 ?            00:00:01 systemd
   2 ?            00:00:00 kthreadd
   3 ?            00:00:00 rcu_gp
```

12. Añade al fichero usuarios.txt la cadena de texto Usuarios actuales

Usamos el comando echo para mostrar en pantalla un mensaje y lo redirigimos, con redirección doble, al fichero usuarios.txt

```
javier@ssoo-100:~$ echo Usuarios actuales >> usuarios.txt
javier@ssoo-100:~$ tail -2 usuarios.txt
/home/javier
Usuarios actuales
```

13. Añade al fichero usuarios.txt información de identificación del usuario actual.

Para mostrar información de identificación utilizamos el comando id. En este caso, como queremos mostrar información del usuario actual, no le pasamos ningún parámetro.

Redirigimos la salida estándar al fichero usuarios.txt con doble redirección para que se añada al final del fichero.

```
javier@ssoo-100:~$ id >> usuarios.txt
javier@ssoo-100:~$ tail -5 usuarios.txt
14526 pts/0    00:00:00 ps
javier
/home/javier
Usuarios actuales
uid=1000(javier) gid=1001(javier) grupos=1001(javier),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),119(lpadmin),1000(sambashare)
```

14. Añade al fichero usuarios.txt información de identificación del usuario marinapg, si hubiera algún error redirige el error al fichero errores.txt.

Para redirigir la salida de errores tenemos que añadir un 2 delante del signo de redirección sin dejar espacios, como aparece a continuación

```
javier@ssoo-100:~$ id pepe 2> errores.txt
javier@ssoo-100:~$ cat errores.txt
id: «pepe»: no existe ese usuario
```

Con esta redirección lo que estamos haciendo es enviar la salida de errores al fichero errores.txt. Sin embargo no se redirige la salida estándar.

```
javier@ssoo-100:~$ id marinapg 2> errores.txt
uid=1002(marinapg) gid=1003(marinapg) grupos=1003(marinapg),1005(educatica)
```

Podemos redirigir ambas salidas, la de error y la estándar, a distintos ficheros. Después de escribir el comando cuyas salidas queremos redirigir, utilizamos la redirección de errores y la estándar, sin importar el orden.

```
javier@ssoo-100:~$ id marinapg 2> errores.txt >> usuarios.txt
javier@ssoo-100:~$ tail -1 usuarios.txt
uid=1002(marinapg) gid=1003(marinapg) grupos=1003(marinapg),1005(educatica)
javier@ssoo-100:~$ cat errores.txt
```

Lo importante es seguir la sintaxis, detrás del signo de redirección hay que especificar la ruta del fichero sobre el que queremos llevar a cabo la redirección.

15. Añade al fichero usuarios.txt información de identificación del usuario jesusrp, si hubiera algún error redirige el error al fichero errores.txt.

El proceso es el mismo que en el caso anterior, solo que en esta ocasión, tenemos que añadir al fichero errores.txt, así que en la redirección de errores también utilizamos una doble redirección para que no se pierda la información que había antes.

```
javier@ssoo-100:~$ id jesusrp 2>> errores.txt >> usuarios.txt
javier@ssoo-100:~$ tail -2 usuarios.txt
uid=1000(javier) gid=1001(javier) grupos=1001(javier),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),119(lpadmin),1000(sambashare)
uid=1002(marinapg) gid=1003(marinapg) grupos=1003(marinapg),1005(educatica)
javier@ssoo-100:~$ cat errores.txt
id: «jesusrp»: no existe ese usuario
```

Mostramos las dos últimas líneas del fichero usuarios.txt para comprobar que se ha guardado información en este fichero y el contenido del fichero errores.txt.

16. Almacena en un fichero llamado root.txt el contenido del directorio /root (no utilices sudo) y la salida de error redirigela a errores.txt

En esta ocasión, vamos a utilizar el comando ls seguido de la ruta del directorio que queremos mostrar. También vamos a cambiar el orden de las redirecciones para comprobar que el resultado es el mismo, se almacenará la salida estándar en el fichero root.txt y la salida de errores en errores.txt

```
javier@ssoo-100:~$ ls /root > root.txt 2>> errores.txt
javier@ssoo-100:~$ cat root.txt
javier@ssoo-100:~$ cat errores.txt
id: «jesusrp»: no existe ese usuario
ls: no se puede abrir el directorio '/root': Permiso denegado
```

Se almacena en el fichero de errores el error generado por el comando ls.

17. Utilizando el comando sudo, añade al fichero root.txt el contenido del directorio /root y redirige la salida de error al fichero errores.txt

Ahora, al utilizar el comando sudo, no se producirá ningún error al mostrar el contenido del directorio /root. Se ha utilizado la opción -l en el comando ls, puesto que el directorio está vacío y así el comando mostrará, al menos, que no hay ficheros.

```
javier@ssoo-100:~$ sudo ls -l /root > root.txt 2>> errores.txt
javier@ssoo-100:~$ cat root.txt
total 0
javier@ssoo-100:~$ cat errores.txt
id: «jesusrp»: no existe ese usuario
ls: no se puede abrir el directorio '/root': Permiso denegado
```

Si usáramos el comando ls sin opciones y el directorio /root estuviera vacío, como es el caso, no mostraría nada y, por tanto, no se almacenaría nada en el fichero root.txt.

18. Añade al fichero allErrors.txt el nombre del usuario actual.

Utilizamos echo para mostrar una cadena y la redirigimos al fichero allErrors.txt.

```
javier@ssoo-100:~$ echo $USER >> allErrors.txt
```

19. Añade al fichero allErrors.txt la fecha y hora actuales

Utilizamos el comando date para mostrar la fecha y hora actual y la redirigimos al fichero allErrors.txt.

```
javier@ssoo-100:~$ date >> allErrors.txt
```

20. Añade al fichero allErrors.txt el contenido del fichero errores.txt

Mostramos el contenido del fichero errores.txt y redirigimos la salida estándar al fichero allErrors.txt con doble redirección para que se añada al final del fichero.

```
javier@ssoo-100:~$ cat errores.txt >> allErrors.txt
```

21. Borra el fichero errores.txt

```
javier@ssoo-100:~$ rm errores.txt
```

Simplemente, borramos el fichero errores.txt

22. Crea un directorio llamado como el usuario actual, dentro del directorio backup dentro del directorio raíz del sistema. Si se produce algún error, que se redirija en el fichero errores.txt

La ruta del directorio sería /backup/\$USER puesto que la variable USER almacena el nombre del usuario actual. Vamos a tratar de crear el directorio con mkdir y redirigimos la salida de errores al fichero errores.txt, con doble redirección para que se añada al final del fichero.

```
javier@ssoo-100:~$ sudo mkdir /backup/$USER 2>> errores.txt
javier@ssoo-100:~$ cat errores.txt
mkdir: no se puede crear el directorio «/backup/javier»: No existe el archivo o el directorio
```

Como podemos observar, se produce un error, en este sistema, puesto que no existe el directorio /backup y no hemos utilizado la opción -p de mkdir para que cree los directorios padres a medida que sean necesarios.

```
javier@ssoo-100:~$ sudo mkdir -p /backup/$USER 2>> errores.txt
javier@ssoo-100:~$ ls -l /backup/
total 4
drwxr-xr-x 2 root root 4096 feb  4 10:53 javier
```

Ahora, utilizando la opción -p, se crea el directorio y no se almacenará ningún mensaje de error nuevo en el fichero errores.txt

23. Copia el directorio /home y todo su contenido en el directorio /backup. Redirige la salida estándar al fichero backup.log y redirige la salida de error al fichero errores.txt

Usamos el comando cp -R para copiar directorios, con el comando sudo puesto que es una tarea de administración y necesitaremos permiso de lectura en /home y todo su contenido, y escritura en el directorio /backup.

```
javier@ssoo-100:~$ sudo cp -R /home /backup >> backup.log 2>>errores.txt
javier@ssoo-100:~$ cat backup.log
javier@ssoo-100:~$ cat errores.txt
mkdir: no se puede crear el directorio «/backup/javier»: No existe el archivo o el directorio
javier@ssoo-100:~$ ls -l /backup/home/
total 16
drwxr-xr-x  3 root root 4096 feb  4 10:55 alfredoff
drwxr-xr-x 15 root root 4096 feb  4 10:55 javier
drwxr-xr-x  3 root root 4096 feb  4 10:55 marinapg
drwxr-xr-x  4 root root 4096 feb  4 10:55 ramonam
```

Como se puede observar, se han utilizado dos redirecciones una de salida estándar al fichero backup.log y otra de errores, 2», al fichero errores.txt.

24. Añade al fichero backup.log el listado del directorio /backup y todo su contenido de forma recursiva con información extendida de permisos propietarioy grupo.

Utilizamos el comando ls con las opciones lhR para mostrar de forma recursiva el contenido del directorio /backup con información extendida y con información de ocupación más comprensible.

Se utiliza doble redirección para añadir la salida estándar al final del fichero backup.log.

```
javier@ssoo-100:~$ ls -lhR /backup >> backup.log
javier@ssoo-100:~$ cat backup.log
/backup:
total 8,0K
drwxr-xr-x 6 root root 4,0K feb  4 10:55 home
drwxr-xr-x 2 root root 4,0K feb  4 10:53 javier
```