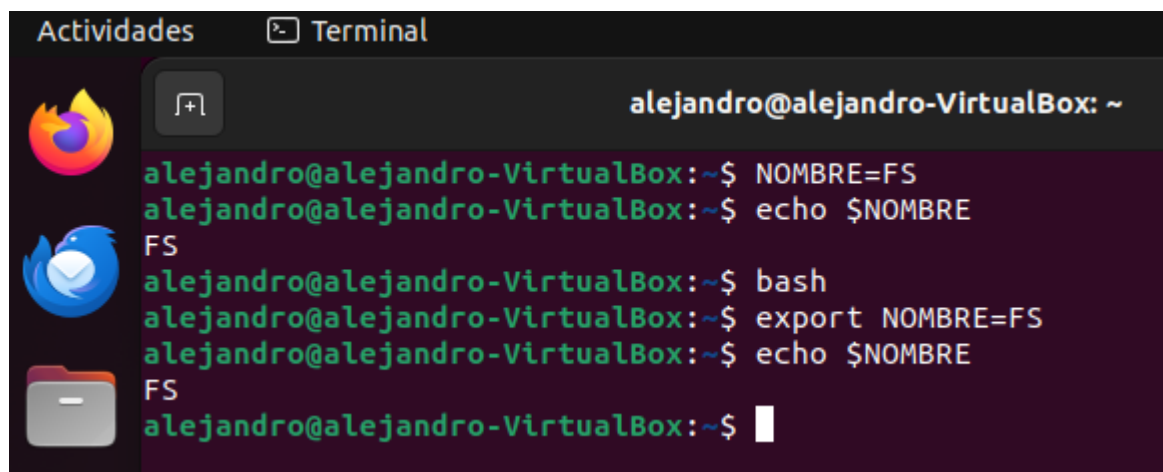


Ejercicio 1: Escriba, al menos, cinco variables de entorno junto con el valor que tienen.

```
SHELL=/bin/bash
LOGNAME=alejandro
PWD=/home/alejandro
HOME=/home/alejandro
LANG=es_ES.UTF-8
```

Ejercicio 2. Ejecute las órdenes del cuadro e indique qué ocurre y cómo puede resolver la situación para que la variable NOMBRE se reconozca en el shell hijo.

La shell hijo no coge las variables definidas en el shell padre, es por ello que debemos exportarlas con el comando export:

A screenshot of a Linux terminal window titled 'Terminal'. The prompt is 'alejandro@alejandro-VirtualBox: ~'. The user enters 'NOMBRE=FS', then 'echo \$NOMBRE', which outputs 'FS'. Then the user enters 'bash' to start a subshell. In the subshell, the prompt is 'alejandro@alejandro-VirtualBox:~\$'. The user enters 'export NOMBRE=FS', then 'echo \$NOMBRE', which outputs 'FS'. Finally, the user enters 'alejandro@alejandro-VirtualBox:~\$' and the prompt returns to the main shell. On the left side of the terminal window, there are icons for 'Actividades', 'Terminal', and a file manager.

```
alejandro@alejandro-VirtualBox: ~
alejandro@alejandro-VirtualBox:~$ NOMBRE=FS
alejandro@alejandro-VirtualBox:~$ echo $NOMBRE
FS
alejandro@alejandro-VirtualBox:~$ bash
alejandro@alejandro-VirtualBox:~$ export NOMBRE=FS
alejandro@alejandro-VirtualBox:~$ echo $NOMBRE
FS
alejandro@alejandro-VirtualBox:~$
```

Ejercicio 3: Compruebe qué ocurre en las expresiones del ejemplo anterior si se quitan las comillas dobles del final y se ponen después de los dos puntos. ¿Qué sucede si se sustituyen las comillas dobles por comillas simples?

Al usar comillas simples la expresión `$(...)` se desactiva.

```
alejandro@alejandro-VirtualBox:~$ echo "Los archivos que hay en el directorio son: $(ls -l)"
Los archivos que hay en el directorio son: total 40
drwxr-xr-x 2 alejandro alejandro 4096 oct 31 16:46 Descargas
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Documentos
drwxr-xr-x 2 alejandro alejandro 4096 oct 31 16:47 Escritorio
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Imágenes
drwxrwxr-x 4 alejandro alejandro 4096 oct 31 16:47 linux-sesion-2-AlexMrtnzz
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Música
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Plantillas
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Público
drwx----- 6 alejandro alejandro 4096 oct 30 19:29 snap
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Vídeos
alejandro@alejandro-VirtualBox:~$ echo "Los archivos que hay en el directorio son: `ls -l`"
Los archivos que hay en el directorio son: total 40
drwxr-xr-x 2 alejandro alejandro 4096 oct 31 16:46 Descargas
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Documentos
drwxr-xr-x 2 alejandro alejandro 4096 oct 31 16:47 Escritorio
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Imágenes
drwxrwxr-x 4 alejandro alejandro 4096 oct 31 16:47 linux-sesion-2-AlexMrtnzz
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Música
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Plantillas
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Público
drwx----- 6 alejandro alejandro 4096 oct 30 19:29 snap
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Vídeos
alejandro@alejandro-VirtualBox:~$ echo Los archivos que hay en el directorio son: "$(ls -l)"
Los archivos que hay en el directorio son: total 40
drwxr-xr-x 2 alejandro alejandro 4096 oct 31 16:46 Descargas
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Documentos
drwxr-xr-x 2 alejandro alejandro 4096 oct 31 16:47 Escritorio
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Imágenes
drwxrwxr-x 4 alejandro alejandro 4096 oct 31 16:47 linux-sesion-2-AlexMrtnzz
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Música
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Plantillas
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Público
drwx----- 6 alejandro alejandro 4096 oct 30 19:29 snap
drwxr-xr-x 2 alejandro alejandro 4096 oct 30 17:02 Vídeos
alejandro@alejandro-VirtualBox:~$ echo 'Los archivos que hay en el directorio son: $(ls -l)'
Los archivos que hay en el directorio son: $(ls -l)
alejandro@alejandro-VirtualBox:~$
```

Ejercicio 4: Pruebe la siguiente asignación:

Se necesita la impresión expr.

```
alejandro@alejandro-VirtualBox:~$ numero=$numero+1
alejandro@alejandro-VirtualBox:~$ $numero
+1: orden no encontrada
alejandro@alejandro-VirtualBox:~$
```

Ejercicio 5. Construya un guión que acepte como argumento una cadena de texto (por ejemplo, su nombre) y que visualice en pantalla la frase Hola y el nombre dado como argumento.

```
alejandro@alejandro-VirtualBox:~$ nano
alejandro@alejandro-VirtualBox:~$ (`chmod +x saludo<<<<-script.sh`)
bash: command substitution: línea 17: error sintáctico cerca del elemento inesperado `<'
bash: command substitution: línea 17: `chmod +x saludo<<<<-script.sh'
alejandro@alejandro-VirtualBox:~$ (`chmod +x saludo_script.sh`)
alejandro@alejandro-VirtualBox:~$ ./saludo_script.sh Alex
Hola, Alex!
alejandro@alejandro-VirtualBox:~$
```

Ejercicio 6. Varíe el guión anterior para que admita una lista de nombres.

```
GNU nano 6.2
#!/bin/bash

echo "Hola, $@"

alejandro@alejandro-VirtualBox:~$ ./saludo_script.sh Alex Juan Pedro
Hola, Alex Juan Pedro
alejandro@alejandro-VirtualBox:~$ echo saludo_script.sh Alex Juan Pedro
saludo_script.sh Alex Juan Pedro
```

Ejercicio 7. Cree tres variables llamadas VAR1, VAR2 y VAR3 con los siguientes valores respectivamente "hola", "adios" y "14". a) Imprima los valores de las tres variables en tres columnas con 15 caracteres de ancho.

a)

```
#!/bin/bash

VAR1="hola"
VAR2="adios"
VAR3="14"

printf "%-15s %-15s %.15s\n" $VAR1 $VAR2 $VAR3

alejandro@alejandro-VirtualBox:~/Escritorio$ nano
alejandro@alejandro-VirtualBox:~/Escritorio$ nano ejers.sh
alejandro@alejandro-VirtualBox:~/Escritorio$ ./ejers.sh
hola          adios          14
alejandro@alejandro-VirtualBox:~/Escritorio$
```

uns

b) ¿Son variables locales o globales?

Son variables locales

c) Borre la variable VAR2.

```
alejandro@alejandro-VirtualBox:~/Escritorio$ ./ejers.sh
hola          14
alejandro@alejandro-VirtualBox:~/Escritorio$
```

d) Abra otra ventana de tipo terminal, ¿puede visualizar las dos variables restantes?

No

e) Cree una variable de tipo vector con los valores iniciales de las tres variables.

VAR123=(hola adios 14)

f) Muestre el segundo elemento del vector creado en el apartado e.

```
#!/bin/bash

VAR1="hola"
VAR3="14"
VAR123="hola adios 14"

printf "%-15s %-15s %.15s\n" $VAR1 $VAR3 $VAR123

echo ${VAR123[1]}
```

Ejercicio 8. Cree un alias que se llame ne (nombrado así para indicar el número de elementos) y que devuelva el número de archivos que existen en el directorio actual.

```
alejandro@alejandro-VirtualBox:~$ ne
Archivos en el directorio actual: 7
alejandro@alejandro-VirtualBox:~$
```

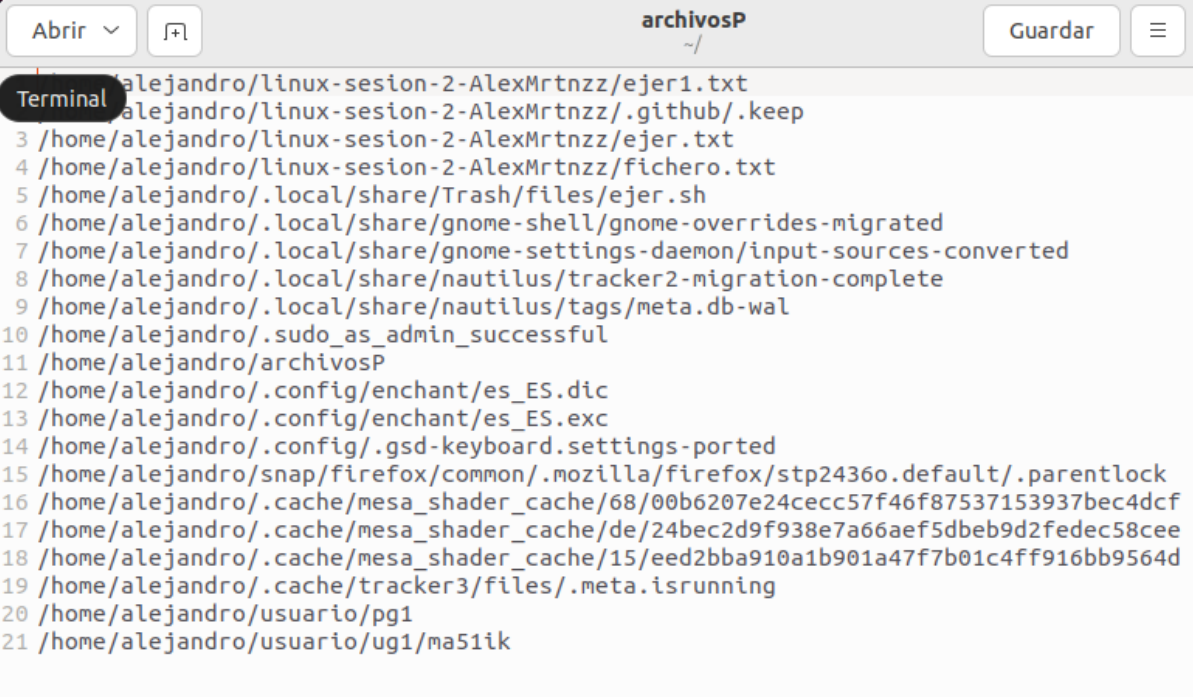
¿Qué cambiaría si queremos que haga lo mismo pero en el directorio home correspondiente al usuario que lo ejecuta?

```
alias ne='cd;ls -A | wc -l'
```

Ejercicio 9. Indique la línea de orden necesaria para buscar todos los archivos a partir del directorio home que tengan un tamaño menor de un bloque. ¿Cómo la modificaría para que además imprima el resultado en un archivo que se cree dentro del directorio donde nos encontremos y que se llame archivosP?

```
alejandro@alejandro-VirtualBox:~$ find $HOME -size 1
/home/alejandro/saludo_script.sh
/home/alejandro/.gitconfig
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/config
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/HEAD
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/refs/heads/main
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/refs/remotes/origin/HEAD
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/hooks/pre-applypatch.sample
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/hooks/post-update.sample
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/hooks/applypatch-msg.sample
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/hooks/pre-merge-commit.sample
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/objects/b9/24138b75bf95b19255cead8808fbb39d89a6
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/index
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/packed-refs
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/logs/HEAD
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/logs/refs/heads/main
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/logs/refs/remotes/origin/HEAD
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/description
/home/alejandro/linux-sesion-2-AlexMrtnzz/.git/info/exclude
/home/alejandro/linux-sesion-2-AlexMrtnzz/README.md
/home/alejandro/.local/share/Trash/files/VAN2
/home/alejandro/.local/share/Trash/files/VAN3
/home/alejandro/.local/share/Trash/files/VAN1
/home/alejandro/.local/share/Trash/info/VAN2.trashinfo
/home/alejandro/.local/share/Trash/info/VAN3.trashinfo
/home/alejandro/.local/share/Trash/info/ejer.sh.trashinfo
/home/alejandro/.local/share/Trash/info/VAN1.trashinfo
/home/alejandro/.local/share/session_migration-ubuntu
/home/alejandro/.local/share/evolution/tasks/system/tasks.ics
/home/alejandro/.local/share/keyrings/login.keyring
/home/alejandro/.local/share/keyrings/user.keystore
/home/alejandro/.local/share/gvfs-metadata/root
/home/alejandro/Escritorio/ejers.sh
/home/alejandro/Escritorio/ejer.sh
/home/alejandro/.config/pulse/cookie
/home/alejandro/.config/pulse/5873adb1eb704006acd649dc7872870b-default-source
/home/alejandro/.config/pulse/5873adb1eb704006acd649dc7872870b-default-sink
/home/alejandro/.config/ibus/bus/5873adb1eb704006acd649dc7872870b-unix-0
/home/alejandro/.config/ibus/bus/5873adb1eb704006acd649dc7872870b-unix-wayland-0
/home/alejandro/.config/gedit/accel
/home/alejandro/.config/user-dirs.locale
/home/alejandro/.config/gtk-3.0/bookmarks
/home/alejandro/.config/gnome-initial-setup-done
/home/alejandro/.gnupg/pubring.kbx
/home/alejandro/snap/snap-store/959/.local/share/themes
/home/alejandro/snap/snap-store/959/.local/share/glib-2.0/schemas/org.gnome.desktop.search-pro
.xml
/home/alejandro/snap/snap-store/959/.local/share/glib-2.0/schemas/org.gnome.desktop.notificati
```

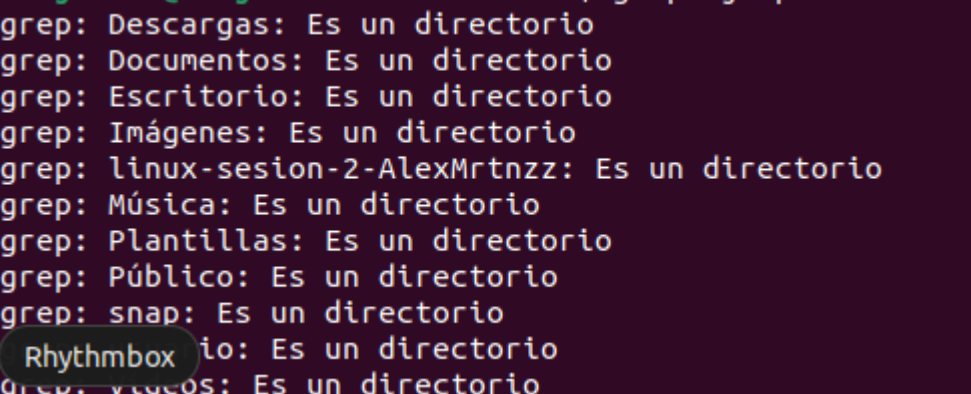
```
alejandro@alejandro-VirtualBox:~$ find $HOME -size -1 > archivosP
```



```
1 /home/alejandro/linux-sesion-2-AlexMrtnzz/ejer1.txt
2 /home/alejandro/linux-sesion-2-AlexMrtnzz/.github/.keep
3 /home/alejandro/linux-sesion-2-AlexMrtnzz/ejer.txt
4 /home/alejandro/linux-sesion-2-AlexMrtnzz/fichero.txt
5 /home/alejandro/.local/share/Trash/files/ejer.sh
6 /home/alejandro/.local/share/gnome-shell/gnome-overrides-migrated
7 /home/alejandro/.local/share/gnome-settings-daemon/input-sources-converted
8 /home/alejandro/.local/share/nautilus/tracker2-migration-complete
9 /home/alejandro/.local/share/nautilus/tags/meta.db-wal
10 /home/alejandro/.sudo_as_admin_successful
11 /home/alejandro/archivosP
12 /home/alejandro/.config/enchant/es_ES.dic
13 /home/alejandro/.config/enchant/es_ES.exc
14 /home/alejandro/.config/gsd-keyboard.settings-ported
15 /home/alejandro/snap/firefox/common/.mozilla/firefox/stp2436o.default/.parentlock
16 /home/alejandro/.cache/mesa_shader_cache/68/00b6207e24cecc57f46f87537153937bec4dcf
17 /home/alejandro/.cache/mesa_shader_cache/de/24bec2d9f938e7a66aef5dbeb9d2fedec58cee
18 /home/alejandro/.cache/mesa_shader_cache/15/eed2bba910a1b901a47f7b01c4ff916bb9564d
19 /home/alejandro/.cache/tracker3/files/.meta.isrunning
20 /home/alejandro/usuario/pg1
21 /home/alejandro/usuario/ug1/ma51ik
```

Ejercicio 10. Indique cómo buscaría todos aquellos archivos del directorio actual que contengan la palabra “ejemplo”.

```
alejandro@alejandro-VirtualBox:~$ grep ejemplo *
```



```
grep: Descargas: Es un directorio
grep: Documentos: Es un directorio
grep: Escritorio: Es un directorio
grep: Imágenes: Es un directorio
grep: linux-sesion-2-AlexMrtnzz: Es un directorio
grep: Música: Es un directorio
grep: Plantillas: Es un directorio
grep: Público: Es un directorio
grep: snap: Es un directorio
Rhythmbox io: Es un directorio
grep: videos: Es un directorio
```


Ejercicio 11. Complete la información de find y grep utilizando para ello la orden man.

```
FIND(1)                                General Commands Manual

NAME
    find - search for files in a directory hierarchy

SYNOPSIS
    find [-H] [-L] [-P] [-D debugopts] [-Olevel] [starting-point...] [expression]

DESCRIPTION
    This manual page documents the GNU version of find. GNU find searches the directory tree
    at each given starting-point by evaluating the given expression from left to right, according
    to the rules of precedence (see section OPERATORS), until the outcome is known (the left hand
    operand is false for and operations, true for or), at which point find moves on to the next file name.
    If no starting-point is specified, '.' is assumed.

    If you are using find in an environment where security is important (for example if you
    are using it to search directories that are writable by other users), you should read the 'Security
    Considerations' chapter of the findutils documentation, which is called Finding Files and
    findutils. That document also includes a lot more detail and discussion than this manual
    page. You may find it a more useful source of information.

OPTIONS
    The -H, -L and -P options control the treatment of symbolic links. Command-line arguments
    following these are taken to be names of files or directories to be examined, up to the first
    that begins with '-', or the argument '(' or '!'. That argument and any following arguments
    are taken to be the expression describing what is to be searched for. If no paths are given,
    the current directory is used. If no expression is given, the expression -print is used (but
    you probably consider using -print0 instead, anyway).
```

```
GREP(1)                                User Commands

NAME
    grep, egrep, fgrep, rgrep - print lines that match patterns

SYNOPSIS
    grep [OPTION...] PATTERNS [FILE...]
    egrep [OPTION...] -e PATTERNS ... [FILE...]
    fgrep [OPTION...] -f PATTERN_FILE ... [FILE...]

DESCRIPTION
    grep searches for PATTERNS in each FILE. PATTERNS is one or more patterns separated by
    characters, and grep prints each line that matches a pattern. Typically PATTERNS should be
    enclosed in double quotes when grep is used in a shell command.

    A FILE of "-" stands for standard input. If no FILE is given, recursive searches examine
    the current working directory, and nonrecursive searches read standard input.

    In addition, the variant programs egrep, fgrep and rgrep are the same as grep -E, grep
    -F, and grep -R, respectively. These variants are deprecated, but are provided for
    compatibility.

OPTIONS
    Generic Program Information
    --help Output a usage message and exit.

    -V, --version
        Output the version number of grep and exit.

    Pattern Syntax
    -E, --extended-regexp
        Interpret PATTERNS as extended regular expressions (EREs, see below).
```

Ejercicio 12. Indique cómo buscaría si un usuario dispone de una cuenta en el sistema.

```
grep usuario /etc/passwd
```

Ejercicio 13. Indique cómo contabilizar el número de ficheros de la propia cuenta de usuario que no tengan permiso de lectura para el resto de usuarios.

```
alejandro@alejandro-VirtualBox:~$ find ~ -maxdepth 1 -type f ! -perm /o=r | wc -l
2
```

Ejercicio 14. Modifique el ejercicio 8 de forma que, en vez de un alias, sea un guión llamado numE el que devuelva el número de archivos que existen en el directorio que se le pase como argumento.

```
alejandro@alejandro-VirtualBox:~$ nano
alejandro@alejandro-VirtualBox:~$ ./ejerss.sh
bash: ./ejerss.sh: Permiso denegado
alejandro@alejandro-VirtualBox:~$ (`chmod +x ejerss.sh`)
alejandro@alejandro-VirtualBox:~$ ./ejerss.sh
26
alejandro@alejandro-VirtualBox:~$ █
```