

# Comandos básicos en Linux

Sistemas Operativos  
DC - FCEN - UBA

21 de agosto de 2025

# Pre-requisitos (repasar o aprender en casa)

Supondremos que no deberían tener problemas para:

Conectarse a través de `ssh`.

Moverse entre directorios y buscar archivos.

Operar con archivos y directorios.

Distinguir “allá” de “acá” (camino o path relativo y absoluto).

Editar un archivo de texto.

Escribir un `hello.c`, compilarlo, y ejecutarlo.

Guardar la salida a un archivo.

Distinguir entre la salida normal y la de errores.

Filtrar líneas de texto.

Buscar comandos.

Buscar `syscalls`.

Buscar ayuda.

Buscar `stdlib`.

Buscar en el manual.

Buscar en [Google](#).

Preguntarle a [ChatGPT](#).

Intérprete de comandos

Interfaz de texto.

Ejemplos: `sh`, `csch`, `ksh`, `bash`.

```
$SHELL --version
```

# Manejo de archivos y directorios

## Ver y ejercitar en casa

Directorio absoluto: `/home/usuario`.

Directorio relativo: `../` o `./` o nada.

## Comandos:

`ls` Lista archivos (directorio actual u otro parámetro).

`cd` Cambia el directorio (parámetro o Home).

`cp`, `mv` Copia/Mueve un archivo o directorio.

`rm` Remueve un archivo/directorio.

`mkdir` Crea un directorio.

`rmdir` Elimina un directorio.

## Ver y ejercitar en casa

**cat** Muestra por stdout el contenido de un archivo.

**less** Muestra por stdout el contenido de un archivo (paginado).

**echo** Escribe en stdout lo indicado por parámetro.

**head** Escribe en stdout las primeras líneas de un archivo.

**tail** Escribe en stdout las últimas líneas de un archivo.

# Busqueda, Matching y Procesamiento

## Ver y ejercitar en casa

**find** Busca dentro del arbol de directorio.

```
find /home -name "*.c" -exec ls -al {} \;
```

**grep** Busca coincidencias de cadenas de caracteres dentro de los archivos.

```
grep "hola" archivo.txt
```

**awk** Procesamiento de texto.

```
awk '{ print $1 }' archivo.txt
```

```
ls -l | awk '{ print $1 }'
```

```
ls -l | awk '{total += $5} END {print total}'
```

`man` Muestra las páginas del manual.

`man` Muestra las páginas del manual.

`apropos` Buscador en todas las páginas del manual.



`man` Muestra las páginas del manual.

`apropos` Buscador en todas las páginas del manual.

`pwd` Print Working Directory.

`who` Quién está logueado.

`uptime` Cuánto tiempo lleva prendido el sistema.

`uname -a` Qué kernel de Linux se está ejecutando.

# Manejo de entrada/salida

Archivos especiales:

`stdout` Salida estándar.

Re-direcciones:

> Redirige `stdout` a un archivo.

```
ls > lista_de_archivos.txt
```

# Manejo de entrada/salida

## Archivos especiales:

`stdout` Salida estándar.

`stderr` Salida estándar (errores).

## Re-direcciones:

> Redirige `stdout` a un archivo.

```
ls > lista_de_archivos.txt
```

2> Redirige `stderr` a un archivo.

```
ls 2> lista_de_archivos.txt
```

# Manejo de entrada/salida

## Archivos especiales:

`stdout` Salida estándar.

`stderr` Salida estándar (errores).

## Re-direcciones:

> Redirige `stdout` a un archivo.

```
ls > lista_de_archivos.txt
```

2> Redirige `stderr` a un archivo.

```
ls 2> lista_de_archivos.txt
```

&> Redirige `stdout` y `stderr` a un archivo.

```
ls &> lista_de_archivos.txt
```

# Manejo de entrada/salida

Archivos especiales:

`stdout` Salida estándar.

Re-direcciones:

`>>` Redirige stdout a un archivo en modo `append`.

```
ls >> lista_de_archivos.txt
```

# Manejo de entrada/salida

## Archivos especiales:

`stdout` Salida estándar.

`stderr` Salida estándar (errores).

## Re-direcciones:

`>>` Redirige `stdout` a un archivo en modo `append`.

```
ls >> lista_de_archivos.txt
```

`&>>` Redirige `stdout` y `stderr` a un archivo en modo `append`.

```
ls &>> lista_de_archivos.txt
```

# Manejo de entrada/salida

## Archivos especiales:

`stdout` Salida estándar.

`stderr` Salida estándar (errores).

`stdin` Entrada estándar.

## Re-direcciones:

`>>` Redirige `stdout` a un archivo en modo `append`.

```
ls >> lista_de_archivos.txt
```

`&>>` Redirige `stdout` y `stderr` a un archivo en modo `append`.

```
ls &>> lista_de_archivos.txt
```

`<` Redirige `stdin`.

```
sort < lista_de_archivos.txt
```

# Manejo de entrada/salida

## Archivos especiales:

`stdout` Salida estándar.

`stderr` Salida estándar (errores).

`stdin` Entrada estándar.

## Re-direcciones:

`>>` Redirige `stdout` a un archivo en modo `append`.

```
ls >> lista_de_archivos.txt
```

`&>>` Redirige `stdout` y `stderr` a un archivo en modo `append`.

```
ls &>> lista_de_archivos.txt
```

`<` Redirige `stdin`.

```
sort < lista_de_archivos.txt
```

`|` Pipe. Copia `stdout` a `stdin`.

```
cat lista_de_archivos.txt | sort
```





---

`http://explainshell.com/`

# Permisos

Ejemplo:

```
$ ls -hl
```

```
-rw-rw-r-- 1 user group 445 mar 14 16:12 archivo  
drwxrwxr-x 2 user group 4,0K mar 14 19:31 directorio
```

Permisos:

**r** Read  
**w** Write  
**x** eXecute

Entidad:

**u** User  
**g** Group  
**o** Others

Comandos:

```
chown baader:so archivo.txt  
chmod u+x archivo.txt
```

`vi/vim` Vi/Vim

`nano` Nano

# Momento para preguntas

Hoy vimos un pantallazo sobre comandos de shell y uso de terminal de Linux:

- Manejo de archivos y directorios

- Búsqueda, matching y procesamiento

- Obtención de información

- Manejo de entrada/salida

- Permisos

- Editores

Hoy vimos un pantallazo sobre comandos de shell y uso de terminal de Linux:

- Manejo de archivos y directorios

- Búsqueda, matching y procesamiento

- Obtención de información

- Manejo de entrada/salida

- Permisos

- Editores

La [guía 0](#) contiene ejercitación para [uso de terminal y programación en C](#).

Hoy vimos un pantallazo sobre comandos de shell y uso de terminal de Linux:

- Manejo de archivos y directorios

- Búsqueda, matching y procesamiento

- Obtención de información

- Manejo de entrada/salida

- Permisos

- Editores

La [guía 0](#) contiene ejercitación para [uso de terminal y programación en C](#).

**Estas habilidades serán de gran importancia a lo largo de la cursada, sobre todo en los talleres prácticos.**