

MANEJO DATOS LINUX

> , GREP , SORT , | , UNIQ , ..

MANEJO DE DATOS EN LINUX

> , >>

pipe |

Suscríbete

cat , more

sort

Head , tail

cut , awk

grep

uniq

Redirección de salida ">" o ">>"

> significa que la salida de un comando o programa la dirige a archivo

Comando > Archivo donde se guarda

>> la salida de un comando o programa reescribe el archivo

Comando >> Archivo que se reescribe

Ver el contenido de archivos

cat file	Muestra el contenido del archivo file
more file	Muestra el contenido del archivo file dividido en pantallas

head y tail

- head file** Muestra las primeras 10 líneas del archivo file
- tail file** Muestra las ultimas 10 líneas del archivo file
- head -n file** Muestra las primeras **n** líneas del archivo file
- tail -n file** Muestra las ultimas **n** líneas del archivo file

grep

- El comando **grep** nos permite buscar, dentro de los archivos, las líneas que concuerdan con un patrón.

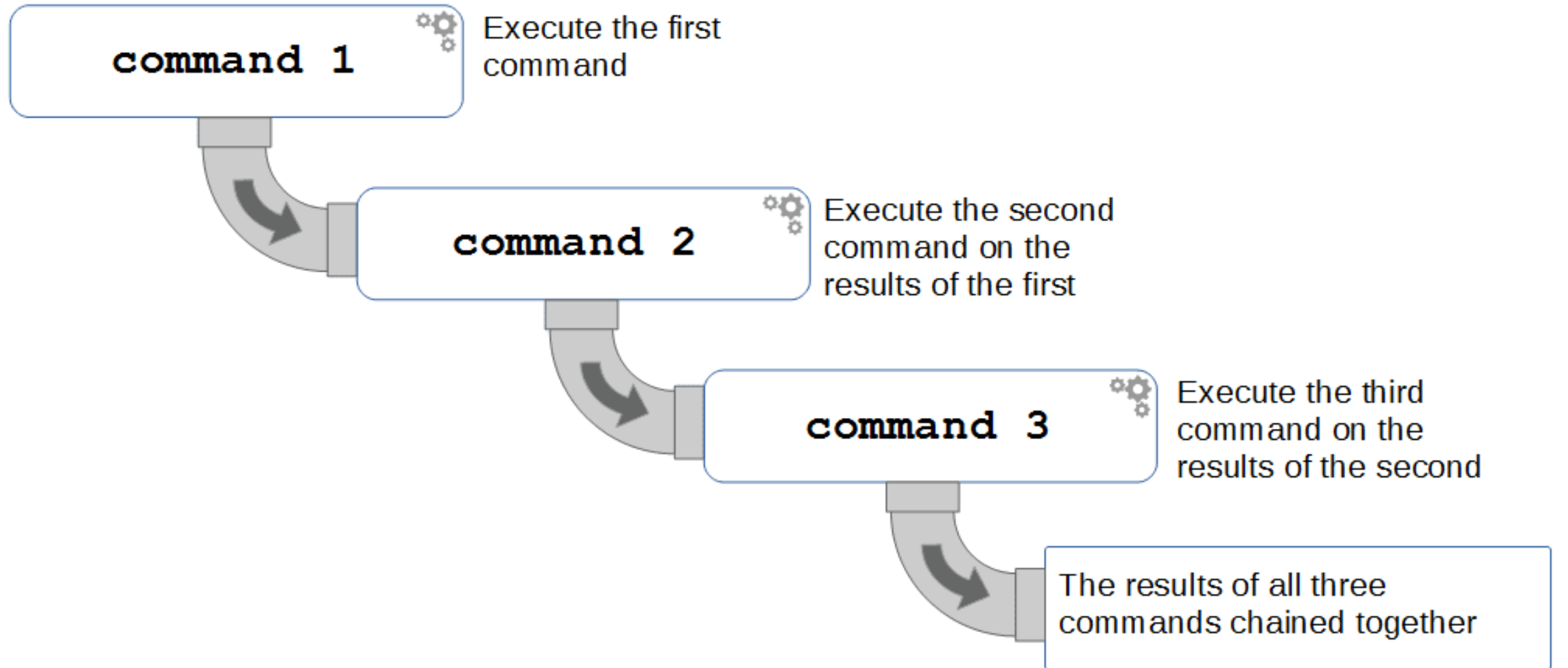
grep "patrón" file Muestra las líneas que
contengan el patrón

grep -v "patrón" file Muestra las líneas que NO
contengan el patrón

Pipe

- Consiste en una cadena de procesos conectados de forma tal que la salida de cada elemento de la cadena es la entrada del próximo
- El pipe es la barra en vertical "|"

Pipe |



WC

wc -l número de líneas

Concatenar

```
cat archivos > archivo_final
```

sort

- Ordenar líneas de los archivos de entrada a partir de criterios de ordenación. Los **espacios en blanco** son tomados por defecto como separadores de campo.

```
sort [opciones] [archivo]
```

- t "SEPARADOR"** Especificar el separador de las columnas
- k número** Especifica la línea de ordenación
- n** Ordena por valor numérico
- r** Invierte el orden

cut

- cut es un comando usado para la extracción de segmentos de las líneas verticales de texto, provenientes de un archivo o bien de la entrada estándar
-
- c INICIO-FINAL** Especifica el carácter inicio y el final
 - f INICIO-FINAL** Especifica la columna o campo(field) inicio y el final
 - d "SEPARADOR"** Especifica el separador de las columnas
 - output-delimiter="NUEVO SEPARADOR"** Especifica el nuevo separador de las columnas

Ejemplos

```
cut -c 4-40 datos_files.txt
```

```
cut -f 1-2 datos_files.txt
```

```
cut -f 1-2 -d " " datos_files.txt
```

```
cut -f 3 -d " " datos_files.txt
```

- Mostrar solo los datos de nombre y apellidos, ordenados alfabéticamente, quitando espacios en blanco y guardando la información en el archivo estudiantes.txt.

awk

- Comando para mejor control de columnas y permite iterar entre ellas

```
awk '{acción}' archivo-entrada
```

```
awk '{condición {acción}}' archivo-entrada
```

awk

```
awk '{acción}' archivo-entrada
```

```
awk '{condición {acción}}' archivo-entrada
```

-F "SEPARADOR" Para especificar un separador de archivos.

\$0 para toda la línea

\$1 para el primer campo

\$2 para el segundo campo

\$n para el campo enésimo campo

print \$10 imprime la columna 10

if(condición) {acción verdadera} {acción falsa}

uniq

- **uniq** es un comando que permite borrar o mostrar las líneas repetidas de un archivo o bien provenientes de la entrada estándar.
- **uniq -c** nos da la frecuencia de las repeticiones