

# COMANDOS IMPORTANTES BASH DGIIM

Javier Sáez Maldonado

## 1. Mover entre directorios

**cd** directorio Eso te lleva al directorio que elijas.  
~ Lleva a home y / lleva al directorio raíz del sistema.

## 2. Listar los archivos de un directorio

**ls** [opcion] [archivo]

-a                      Lista los archivos de directorio, incluyendo a los que su nombre empieza por un punto

-l                      los muestra en formato largo

-r                      los lista en orden inverso

-R                      lista los subdirectorios recursivamente adems del directorio actual

-t                      lista segn la fecha de modificacin

## 3. Mostrar rutas completas

**pwd** [opcion]

Esto muestra el directorio completo de trabajo actual

**-P**                      Muestra el directorio fisico sin enlaces simbolicos

## 4. Directorios

**mkdir**[opcion] [nombredirectorio]                      Crea los directorios si no existen  
**rmdir**[opcion] [directorio]                      Elimina los directorios

## 5. Archivos

**touch** [archivo]  
Crea un archivo, y si existia, se pone la fecha y hora actual

**rm** [opcion]  
Borra archivos o directorios

**-r**                      Borra recursivamente

**cp** [opcion] [archivo o directorio]                      Copia el archivo primero al segundo

**-s** crea un enlace en vez de copiar el archivo

**-r** copia recursivamente

**mv** [opcion] [inicio] [final]                      mueve archivos

## 6. Mostrar archivos

**cat** [opcion] [archivo]                      muestra el contenido de un archivo

**-E**                      imprime \$ al final de cada linea

**-b**                      Enumera las lineas no vacas

**-n**                      Enumera cada linea

**-s**                      Suprime las lineas no vacas

**head** [opcion][archivo] Muestra la parte inicial de un archivo, 10 líneas por defecto

**-bytes=[-]K** Imprime las primeras K bytes  
(con **-**imprime todo menos las ultimas k bytes)

**-lines=K** imprime las primeras K líneas (Con **-**imprime todo menos las ultimas k líneas)

**tail** [opcion][archivo] Muestra la parte final de un archivo, por defecto 10 líneas.

**-bytes=K** imprime las ultimas K bytes

**-lines=K** Imprime las ultimas K líneas

**sort** [opcion][archivo] Ordena el contenido de un archivo

**wc** [opcion][archivo] Imprime el numero de líneas, palabras y bytes de un archivo segun el parametro

**-c** Imprime el numero de bytes

**-m** Imprime el numero de caracteres

**-l** Imprime el numero de líneas

**w** Imprime el numero de palabras

**-L** Imprime la longitud de la línea mas larga

### 6.1. Metacaracteres de archivo

**?** Representa cualquier caracter en la posicion indicada

**\*** Representa cualquier caracter

`[]` Representa un rango de caracteres, separados por - si estan separados

`{}` Sustituye un conjunto de palabras separadas por comas que tienen partes comunes

## 7. Permisos

**chmod** [opcion][archivo] Cambia los permisos de acceso a archivos, aadiendo con + y quitando con -

**u,g,o,a** Propietario, Grupo, Resto de usuarios y Todos los usuarios, (en ese orden)

**r,w,x** Lectura, Escritura, Ejecucion

## 8. Imprimir en pantalla

**echo** y **printf**  
Escribiendo algo entre comillas dobles despues de ellos, se imprime lo de dentro de las comillas en la terminal

**Printf** puede realizar mas funciones que echo.  
Esta, imprime los argumentos segun el formato. SI pones un numero entre el % y la letra de formato, se deja una separacion antes del ultimo caracter de tantos caracteres como indique el numero. Si el numero es negativo, se deja la separacion a partir del primer caracter con el siguiente argumento. Para dar formato se usa:

`\"` Comillas dobles

`\'` Comilla simple

`\\` Barra invertida

`\b` Nueva linea

|                      |  |  |
|----------------------|--|--|
|                      | <code>\t</code>                            | Tabulador                                  |
| ASCII de 8 bits      | <code>\0n</code>                           | n=numero octal que representa un caracter  |
| <code>%d</code>      | Un numero con signo                        |  |
| <code>%f</code>      | Un numero decimal sin notacion exponencial |  |
| <code>%q</code>      | Entrecomilla una cadena                    |  |
| <code>%s</code>      | Muestra una cadena sin entrecomillar       |  |
| <code>%x</code>      | Muestra un numero en hexadecimal           |  |
| <code>%o</code>      | Muestra un numero en octal                 |  |
| <b>date</b> [opcion] | Imprime fechas y horas                     |  |
|                      | <code>%a</code>                            | imprime las siglas del dia de la semana    |
|                      | <code>%A</code>                            | imprime el dia de la semana completo       |
|                      | <code>%b</code>                            | Imprime las siglas del mes                 |
|                      | <code>%B</code>                            | Imprime el mes completo                    |
|                      | <code>%c</code>                            | Cambia el orden de salida al orden español |
|                      | <code>%C</code>                            | Imprime los dos primeros digitos del año   |
|                      | <code>%d</code>                            | Imprime el dia del mes                     |
|                      | <code>%D</code>                            | Imprime la fecha                           |
|                      | <code>%F</code>                            | Imprime la fecha completa                  |
|                      | <code>%H</code>                            | Imprime la hora                            |

|                 |           |   |
|-----------------|-----------|---|
|                 | <b>%I</b> | Tambien imprime la hora                     |
|                 | <b>%j</b> | Imprime el dia del año                      |
|                 | <b>%m</b> | Imprime el numero del mes                   |
|                 | <b>%M</b> | Imprime el minuto                           |
|                 | <b>%n</b> | Imprime una nueva linea                     |
|                 | <b>%r</b> | Imprime la hora, tambien se puede hacer co- |
| mo              | <b>%R</b> |   |
|                 | <b>%S</b> | Imprime los segundos                        |
|                 | <b>%t</b> | Imprime una tabulacion                      |
|                 | <b>%u</b> | Imprime el dia de la semana de forma nu-    |
| merica          |           |   |
|                 | <b>%V</b> | Imprime la semana del año. Tambien se       |
| puede hacer con | <b>%W</b> |   |
|                 | <b>%y</b> | Imprime los dos ultimos digitos del año     |
|                 | <b>%Y</b> | Imprime el año.                             |

## 9. Metacaracteres de redireccion

**> nombre** Redirecciona la entrada de una orden para que la obtenga del archivo nombre

**< nombre** Redirige la salida a un archivo de ese nombre

**& nombre** La salida estandar se cambia con la salida de error y se escriben en el archivo nombre

**>> nombre** Es igual que `;` pero aadiendo la salida al final del archivo, sin sobrescribir

**2 > nombre** Redirige la salida de error a un archivo

**| nombre** Crea un cauce entre dos ordenes, la salida de la primera se usa como entrada de la segunda

**|&** Crea un cauce entre dos ordenes usando las salidas estandar y error como entrada de la segunda

### 9.1. Metacaracteres sintacticos

**;** Separa ordenes que se ejecutan secuencialmente

**()** Aisla ordenes separadas

**&&** Ejecuta dos ordenes si la primera tiene exito

**||** Separa ordenes, ejecutando la segunda si la primera falla

## 10. Variables

Para asignar una variable se pone el nombre de la variable, un signo igual y el valor que queremos asignar, que puede ser una constante u otra variable. A cada lado del signo igual no debe haber ningun espacio en blanco. SI delante o detras del igual dejamos un espacio en blanco obtendremos un error, porque lo tomar como si fuera una orden y sus argumentos, no como una variable. Ademas el nombre de una variable puede contener pero no empezar por un digito.

Las variables pueden ser:

- a) Cadenas, cuyo valor es una secuencia de caracteres
- b) Numeros, para usarlos en operaciones aritmeticas
- c) Constantes
- d) Vectores o arrays, conjunto de elementos a los que se accede mediante un indice, que suele ser un numero entero(contando el 0). Para definir estas variables, ponemos los elementos del vector separados por espacios, y para

acceder a uno de ellos escribimos `$(variable(numeroelemento))`

**env/printenv**[opcion]                      Permite visualizar las variables de entorno o globales comunes a todos los shells. Para estas, se usan letras mayusculas

**set**                      Permite visualizar las variables locales

**\$?**                      Esta variable contiene el coigo de retorno de la ultima orden ejecutada, bien sea una instruccion o un guion

**unset** [variable]                      Borra la variable y sus atributos

**declare** [-iarx] [-p][variable[valor]]                      Crea variables con ciertos atributos

**-i**                      Indica que la variable es numerica

**-p**                      Permite visualizar los atributos de la variable

**-a**                      Indica que es una matriz

**-r**                      indica que es de solo lectura

**-x**                      Indica que es exportable

**export** [-fn] [variable[valor]]                      **export -p**

Exporta las variables locales para poder usarlas fuera del shell actual.Si se le da un valor antes, este se le asigna antes de exportarla.

**-f**                      Se refiere a funciones del shell

**-n**                      Borra la propiedad de exportacion para las variables

**-p**                      Muestra una lista de las variables y funciones exportadas



## 10.1. Expresiones con variables

El shell bas ofrece dos posibles sintaxis para manejar expresiones aritmeticas haciendo uso de lo que se denomina expresion aritmetica o sustitucion aritmetica, que evalua una expresion aritmetica y sustituye el resultado de la expresion en el lugar donde se usa

$\$(...)$                        $\$[...]$

Hay que tener en cuenta que las variables que se usen en una expresion aritmetica no necesitan ir precedidas del simbolo \$ para ser sustituidas por su valor, aunque si lo llevan no sera causa de error, y que cualquier expresion aritmetica puede contener otras expresiones aritmeticas ya que se pueden anidar

## 11. Uso de comillas

**Dobles:** Su acotacion es debil. Protegen cadenas desactivando el significado de los caracteres especiales que haya entre ellas.(Excepto los caracteres !,\$ y ‘)

**Simples:** Su acotacion es mas fuerte. Protegen las cadenas desactivando el significado de los caracteres especiales que haya entre ellas, menos el caracter !.

**Invertidas:** Ejecuta las ordenes que se encuentren encerradas entre ellas e incluye en el mismo sitio el resultado que den estas ordenes. Se pueden sustituir por \$(argumentos)

### 11.1. Expresiones

**expr** expresion                      Imprime en pantalla el valor de la expresion.  
Entre los argumentos y el operador debe haber un espacio.

### 11.2. Alias

**alias** [-p][nombre[='valor']]  
Define o muestra alias. Dentro de un alias y entre comillas podemos poner

varias ordenes separadas por ";" de tal forma que se ejecutaran cada una de ellas secuencialmente

**-p** Muestra todos los alias definidos en formato reusable

## 12. Find

**find** [-H][-L][-P][directorios][expresion]

Este comando explora una rama de directorios buscando archivos que cumplan unos criterios. Por defecto, visualiza todos los archivos y directorios del directorio local y subdirectorios incluso los ocultos

**-H** Solo sigue enlaces simbolicos al procesar los argumentos de la linea de comandos

**-L** Sigue los enlaces simbolicos

**-P** Nunca sigue los enlaces simbolicos

Se pueden poner tambien algunas expresiones para limitar la busqueda

**-namenombre** Busca por nombre de archivo

**-atime n** Busca por el ultimo acceso, siendo n el numero sin signo si quieres numero de dias exacto, con un + si quieres un numero de dias mayor al numero y con un - si quieres un numero de dias menor al numero

**-type d** Busca directorios

**-type f** Busca archivos regulares

**-size n** Busca archivos por tamao de bloques. Funciona igual que el -atime, y se pueden incluir las letras c para buscar en bytes, k para buscar en kilobytes, M para buscar por Megabytes y G para buscar por Gygabyes

**-exec** Permite aadir una orden que se aplicar a los archivos localizados. Se situa a continuacion de la opcion y debe ter-

minarse con un espacio, un caracter \y a continuacion un ;. Se usa {} para representar el nombre de los archivos localizados

**-ok** Es similar a -exec, pero solicita confirmacion en cada archivo encontrado antes de encontrar la orden

## 13. Grep

**grep** [opciones]patron[archivo]

Permite buscar cadenas en archivos usando patrones para especificar esa cadena. Lee de una lista de archivos especificados como argumentos y escribe aquellas lineas que contengan la cadena. Se pueden buscar lineas especiales usando:

**-x** Localiza las lineas que coincidan totalmente con el patron especificado

**-v** Selecciona todas las lineas que no contengan el patron especificado

**-c** Produce solo un recuento de lineas coincidentes

**-i** Ignora las distinciones entre mayusculas y minusculas

**-n** Aade el numero de linea en el archivo fuente a la salida de las coincidencias

**-l** Selecciona solo los nombres de los archivos que coincidan con el patron de busqueda

**-e** Es especial para el uso de multiples patrones o si el patron empieza por el caracter -

**-E** Toma la expresion como una expresion regular extendida (egrep)

**-F** Toma la expresion como una cadena literal  
escapando los simbolos (fgrep)

**-r** Lee todos los subdirectorios recursivamente

Hay tambien algunos caracteres especiales:

**[aeiou]** , que busca una vocal minuscula

**[A-Z0-9]** Una mayuscula o una cifra

**[^0-9]** Busca cualquier caracter que no sea una  
cifra

**\*** Indica que el elemento que le precede debe  
estar 0 o mas veces

**.** Concuerda con un caracter

**\$** SI aparece al final de la expresion significa fi-  
nal de linea

**^** Si aparece al principio de la expresion signifi-  
ca principio de linea

**\** Elimina el significa especial al caracter que le  
sigue

### 13.1. Egrep

**egrep** patron [archivo] Es lo mismo que grep -E

**?** Indica que el elemento que le precede debe  
estar 0 o 1 vez

**+** Indica que el elemento que le precede debe  
estar una o mas veces

**{n}** Indica que el elemento que le precede debe estar exactamente n veces

**{n,m}** INdica que el elemento que le precede debe estar entre n y m veces

**(expr1 |expr2)** Indica que puede aparecer expr1 o expr2

### 13.2. fgrep

**fgrep** patron [archivo] es lo mismo que grep -F

### 13.3. rgrep

**rgrep** patron [archivo] es lo mismo que grep -r

## 14. Guiones o Script

Los conceptos importantes a la hora de crear un guion segun unos argumentos son:

**\$0** Nombre del guion o script llamado, solo se usa dentro del guion

**\$1... \${n}** SOn los argumentos que se le dan a un guion. A partir del numero 9, se dan entre llaves.

**\$\*** Contiene el nombre del guion y todos los argumentos que se le han dado

**\$@** Es equivalente a \$\*

**arg:-val** Si el argumento tiene valor y es no nulo, continua con us valor, y si no se le asigna el valor indicado por val

**arg?:val** Si el argumento tiene valor y es no nulo, sustituye su valor. En caso contrario, imprime el valor de val y sale del guion. SI val es omitida, imprime un mensaje indicando que el argumento es nulo o no esta asignado

## 15. Operadores aritmeticos

**+, -, \*, /, %** Son suma, resta, multiplicacion, division y resto de la division

**\*\*** Potenciacion

**++** Incremento en una unidad. PUede ir como prefijo o sufijo de una variable. Si se usa como prefijo (++variable) primero se incrementa la variable y luego se hace lo que se desee con ella; si se utiliza como sufijo (variable++) primero se hace lo que se quiera con la variable y luego se incrementa

**-** Decremento de la variable en una unidad. Funciona igual que el incremento.

**(...)** Indica una agrupacion para evaluar conjuntamente, indicando el orden de evaluacion de las subexpresiones

**,** Es un separador de expresiones con evaluacion secuencial

**=** si x=expresion, se asigna a x el resultado de evaluar la expresion. No puede haber huecos en blanco a los lados del igual

**+= -=** x+=y equivale a x=x+y e igual con el menos pero con un menos

**\*= /=** Es igual que el anterior pero con producto y cociente

**%=** x%=y equivale a x=x %y

### 15.1. Operadores relacionales

|                                |                             |
|--------------------------------|-----------------------------|
| <b>A=B / A==B / A -eq B</b>    | A es igual a B              |
| <b>A!= B / A -ne B</b>         | A es distinta de B          |
| <b>A &lt;B / A -lt B</b>       | A es menor que B            |
| <b>A &gt;B / A -gt B</b>       | A es mayor que B            |
| <b>A &lt;= B / A -le B</b>     | A es menor o igual que B    |
| <b>A &gt;= B / A -ge B</b>     | A es mayor o igual que B    |
| <b>!A</b>                      | A es falsa                  |
| <b>A &amp;&amp; B / A -a B</b> | A y B son verdaderas        |
| <b>A   B/ A -o B</b>           | A es verdadera o B es falsa |

## 16. Test

**test** expresion

Esta orden evalua un condicional y da como salida un 0 si la expresion ha dado como resultado verdadera (true) o un 1 si la evaluacion ha resultado falsa (false) o se dio un argumento no valido

Expresiones, indicaremos lo que comprueba:

|            |                   |                                       |
|------------|-------------------|---------------------------------------|
|            | <b>-a</b> archivo | Es un archivo y existe                |
| bloques    | <b>-b</b> archivo | archivo existe y es un dispositivo de |
| caracteres | <b>-c</b> archivo | archivo existe y es un dispositivo de |
|            | <b>-d</b> archivo | archivo existe y es un directorio     |

|  |                                      |  |
|--|--------------------------------------|--|
| o regular  | <b>-f</b> archivo                    | archivo existe y es un archivo plano   |
| mismo grupo del usuario  | <b>-G</b> archivo                    | archivo existe y es propiedad del      |
| lico   | <b>-h</b> archivo                    | archivo existe y es un enlace simbo-   |
| usuario  | <b>-O</b> archivo                    | archivo existe y es propiedad del      |
| miso de lectura sobre el   | <b>-r</b> archivo                    | archivo existe y el usuario tiene per- |
| miso de escritura sobre el   | <b>-w</b> archivo                    | archivo existe y el usuario tiene per- |
| miso de ejecucion sobre el   | <b>-x</b> archivo                    | archivo existe y el usuario tiene per- |
| misio de ejecucion sobre el, o es un directorio y el usuario tiene permiso de busqueda en el       | <b>-x</b> archivo                    | archivo existe y el usuario tiene per- |
|  | <b>-s</b> archivo                    | archivo existe y no es vacio           |
| mas reciente que archivo 2 segun la fecha de modificacion o si archivo1 existe y archivo2 no       | <b>archivo1 -nt archivo2</b> archivo | archivo 1 es                           |
| mas antiguo que archivo 2 segun la fecha de modificacion o si archivo1 existe y archivo2 no        | <b>archivo1 -ot archivo2</b> archivo | archivo 1 es                           |
| enlace duro a archivo 2, es decir, si ambos se refieren a los mismos numero de dispositivo e inode | <b>archivo1 -ef archivo2</b> archivo | archivo 1 es un                        |



