

grep

uso

es una utilidad de línea de comando para buscar expresiones regulares dentro de cadenas.

- `$ grep`
 - `-n` : muestra número de línea
 - `-v` : inverso, muestra las líneas que NO tienen la cadena
 - `-w` : palabras completas
 - `-x` : líneas completas, no fragmentos
 - `-P` : uso del formato PERL de expresiones regulares
 - `-color` : marca en color las coincidencias
 - `-i` : `-ignore-case`
 - `-c` : `count`
 - `-r` : búsqueda recursiva

expresiones regulares

ejemplos

- buscar una cadena en una serie de ficheros:

```
find . -name "*.java" -print0 | xargs -0 grep -i ".*Legacy.*xml"
```

- usar un fichero de apoyo de ficheros para realizar acciones sobre los mismos
 - extraído de dokuwiki
 - permite comentarios en el fichero TXT

```
grep -Ev "^(|$|#)" files2remove.txt | xargs -n 1 rm -fd
```

[files2remove.txt](#)

```
# removed in 2011-05-25
conf/words.aspell.dist
lib/styles/style.css

# removed in 2010-11-07
inc/lang/ar/subscribermail.txt
inc/lang/az/subscribermail.txt
inc/lang/bg/subscribermail.txt
inc/lang/ca/subscribermail.txt
inc/lang/ca-valencia/subscribermail.txt
inc/lang/cs/subscribermail.txt
```

- buscar cadenas no coincidentes entre 2 ficheros:

```
grep -vf fichero1 fichero2
```

- filtrado en vivo:

```
tail -f some_log_file.log | grep --line-buffered the_thing_i_want
```

- búsqueda de 2 palabras:

```
grep -e 'palabra1' -e 'palabra2' fichero
grep 'palabra1\|palabra2\|palabra3' fichero
grep -E 'palabra1|palabra2' fichero
egrep -w 'palabra1|palabra2' fichero
```

- mostrar líneas sin comentarios:

```
grep -v ^# <fichero> | grep .
```

- primera parte descarta toda aquella línea que empiece por #
- segunda parte solo deja pasar líneas no vacías

From:

<https://miguelangel.torresegea.es/wiki/> - miguel angel torres egea

Permanent link:

<https://miguelangel.torresegea.es/wiki/linux:bash:grep?rev=1533633222>

Last update: **07/08/2018 02:13**

