

GREP

1) Extraer del fichero cliente.txt (dado más abajo) aquellas personas que tengan por nombre Manuel

```
grep -e Manuel cliente.txt
```

2) Extraer del fichero cliente.txt sólo las mujeres. El resultado se almacenará en el fichero PRCgrep2.txt

```
grep -e :M cliente.txt > PRCgrep2.txt
```

3) Contar el número de personas que viven en Sevilla del fichero cliente.txt

```
egrep -c Sevilla cliente.txt
```

4) Mostrar el número de orden que ocupan las personas del fichero cliente.txt que viven en Soria. Debe mostrarse también el nombre del fichero donde se está buscando (cliente.txt en este caso concreto). El resultado se almacenará en el fichero PRCgrep4.txt

```
egrep -inH :Soria: cliente.txt > PRCgrep4.txt
```

5) Extraer del fichero cliente.txt aquellas personas cuyo segundo apellido termine en "ez" junto con las que vivan en Cádiz. El resultado se almacenará en el fichero PRCgrep5.txt

```
grep -e "ez," -e ":Cadiz:" cliente.txt > PRCgrep5.txt
```

5b) Extraer del fichero cliente.txt aquellas personas cuyo segundo apellido termine en "ez"

y que vivan en Sevilla (se tiene que cumplir las 2 condiciones simultáneamente). El resultado

se almacenará en el fichero PRCgrep5b.txt

```
egrep "ez,.*Sevilla:" cliente.txt
```

6) Extraer del fichero cliente.txt todas las personas que son hombres y no viven ni en Sevilla ni en Madrid. El resultado se almacenará en el fichero PRCgrep6.txt

```
grep -e M -v -e Sevilla -e Madrid cliente.txt
```

7) Extraer del fichero cliente.txt todas las personas que en su nombre aparezca Manuel. El resultado se almacenará en el fichero PRCgrep7.txt

```
egrep -w Manuel cliente.txt > PRCgrep7.txt
```

8) Extraer del fichero cliente.txt todas las personas que viven en ciudades cuyo código postal comience por 1 o por 8 y que termine por 4 o por 2. El resultado se almacenará en el fichero PRCgrep8.txt

```
grep -e :1...4: -e :1...2: -e :8...4: -e :8...2: cliente.txt > PRCgrep8.txt
```

Otra forma:

```
grep -e :[18]...[24]cliente.txt > PRCgrep8.txt
```

9) Extraer del fichero cliente.txt todas las personas que viven en ciudades cuyo código postal tenga un 2 como cifra central. El resultado se almacenará en el fichero PRCgrep9.txt

```
grep -e :..2..: cliente.txt > PRCgrep9.txt
```

10) Listar los nombres de los ficheros del directorio /etc que contienen el texto query Solo interesan los nombres de fichero, no las líneas de texto completas ni los mensajes de error de los ficheros a los que no se tenga acceso El resultado se almacenará en el fichero PRCgrep10.txt

```
grep -sl query /etc/* > PRCgrep10.txt
```

SORT

1) Ordenar el fichero `hombres.txt` dado más abajo por apellidos en el fichero `h.txt`

```
sort hombres.txt > h.txt
```

2) Ordenar el fichero `mujeres.txt` dado más abajo por apellidos en el fichero `m.txt`

```
sort mujeres.txt > m.txt
```

3) Mezclar los ficheros `h.txt` y `m.txt` para obtener el fichero `cliente.txt`

```
sort -m h.txt m.txt > cliente.txt
```

4) Ordenar el fichero `cliente.txt` de forma que primero figuren todas las mujeres y luego todos los hombres. El resultado se almacenará en `PRCsort4.txt`

```
sort -t: -k 2 -r cliente.txt > PRCsort4.txt
```

5) Ordenar el fichero `cliente.txt` de forma que primero figuren todos los hombres y luego todas las mujeres pero a su vez deben estar ordenados por apellidos tanto los unos como las otras. El resultado se almacenará en `PRCsort5.txt`

```
sort -t: -k 2,2 cliente.txt > PRCsort5.txt
```

5b) Ordenar el fichero `cliente.txt` de forma que primero figuren todos los hombres y luego todas las mujeres pero a su vez deben estar ordenados por ciudad tanto los unos como las otras. El resultado se almacenará en `PRCsort5b.txt`

```
sort -t: -k 2,2 -k 4 cliente.txt > PRCsort5b.txt
```

6) Ordenar el fichero `cliente.txt` por provincias. Si el orden no fuese el esperado, revisar el modificador `-d`. El resultado se almacenará en `PRCsort6.txt`

```
sort -t: -k 4 cliente.txt > PRCsort6.txt
```

7) Ordenar el fichero `cliente.txt` por provincias y por código postal. El

resultado se almacenará en PRCsort7.txt (Cuidar que los códigos postales de Madrid se ordenen correctamente)

```
sort -t: -k 4,4d -k 5 cliente.txt > PRCsort7.txt
```

8) Ordenar el fichero cliente.txt por provincia. Dentro de cada provincia, por la segunda letra del primer apellido. El resultado se almacenará en PRCsort8.txt

```
sort -t: -k 4,4d -k 1.2n cliente.txt > PRCsort8.txt
```

9) Ordenar el fichero cliente.txt por provincias. Dentro de cada provincia, por la cuarta letra del primer apellido (en orden descendente). El resultado se almacenará en PRCsort9.txt

```
sort -t:-k 4,4d -k 1.4 cliente.txt > PRCsort9.txt
```

10) Ordenar el fichero cliente.txt por provincias (descendente) y por código postal (ascendente) y por la tercera letra de la dirección (descendente). El resultado se almacenará en PRCsort10.txt

```
sort -t: -k 4,4dr -k 5,5n -k 3.3dr cliente.txt > PRCsort10.txt
```

11) ¿ Cómo se podría conocer el número de códigos postales diferentes que existen en el fichero cliente.txt ? (Sugerencia: para hacer este apartado puede ser necesario conocer el funcionamiento de los comandos cut y uniq)

```
cut -f5 -d: cliente.txt | sort | uniq -c
```

He hecho todos los ejemplos y he tenido problemas en la actividad 9 de GREP y en la 10 y 11 de SORT