

**Nombre:**

1. Crear un fichero llamado *prueba.txt*

- Cuyo contenido será:

Teruel  
Madrid  
Cuenca  
Salamanca  
Valencia  
Sevilla

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ nano prueba.txt
```

```
GNU nano 7.2
Teruel
Madrid
Cuenca
Salamanca
Valencia
Sevilla
```

2. Mostrar el siguiente contenido del archivo:

- Todas las líneas del fichero con los números de línea
- Las dos primeras líneas del fichero
- La última línea del fichero
- La primera línea y las dos últimas del fichero

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ cat -n prueba.txt
```

```
1 Teruel
2 Madrid
3 Cuenca
4 Salamanca
5 Valencia
6 Sevilla
```

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$
```

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ head -2 prueba.txt
```

```
Teruel
Madrid
```

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$
```

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ tail -1 prueba.txt
Sevilla
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ 
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ head -1 prueba.txt ; tail -2 prueba.txt
Teruel
Valencia
Sevilla
usuario@usuario-VirtualBox:~/Escritorio/Tarea$
```

3. Crear con el mismo comando la estructura de directorios *dir1/dir2/dir3* y *dir1/dir4/dir5*.

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ mkdir -p dir1/dir2/dir3 ; mkdir -p dir1/dir4/dir5
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ ls
dir1  prueba.txt  texto.txt
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ ls dir1/
dir2  dir4
usuario@usuario-VirtualBox:~/Escritorio/Tarea$
```

4. Copiar *prueba.txt* en los directorios *dir2* y *dir4* con un solo comando.

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ cp prueba.txt dir1/dir2 ; cp prueba.txt dir1/dir4
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ ls dir1/dir2/
dir3  prueba.txt
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ ls dir1/dir4/
dir5  prueba.txt
usuario@usuario-VirtualBox:~/Escritorio/Tarea$
```

5. Localiza todos los ficheros *prueba.txt* en la estructura de directorios.

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ find -name prueba.txt
./prueba.txt
./dir1/dir2/prueba.txt
./dir1/dir4/prueba.txt
usuario@usuario-VirtualBox:~/Escritorio/Tarea$
```

6. Crea el directorio *Datos1* y, si éste se crea correctamente, que se cree el directorio *Datos2*.

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ mkdir Datos1 && mkdir Datos2
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ ls
Datos1 Datos2 dir1  prueba.txt  texto.txt
usuario@usuario-VirtualBox:~/Escritorio/Tarea$
```

7. Crea el directorio *Datos1* y, si éste da error, que se cree el directorio *Datos3*.

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ mkdir Datos1 || mkdir Datos3
mkdir: no se puede crear el directorio «Datos1»: El archivo ya existe
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ ls
Datos1 Datos2 Datos3 dir1  prueba.txt  texto.txt
usuario@usuario-VirtualBox:~/Escritorio/Tarea$
```

8. Si los dos anteriores puntos han funcionado correctamente, se habrán creado los directorios *Datos1*, *Datos2* y *Datos3*. Muestra que evidentemente es así.

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ ls
Datos1 Datos2 Datos3 dir1 prueba.txt texto.txt
usuario@usuario-VirtualBox:~/Escritorio/Tarea$
```

9. Cambiar el nombre del fichero *prueba.txt* por *definitivo.txt*.

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ mv prueba.txt definitivo.txt
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ ls
Datos1 Datos2 Datos3 definitivo.txt dir1 texto.txt
usuario@usuario-VirtualBox:~/Escritorio/Tarea$
```

10. Hacer una copia de *dir1* con todo su contenido y llamarlo *dirx*.

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ cp -r dir1 dirx
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ ls
Datos1 Datos2 Datos3 definitivo.txt dir1 dirx texto.txt
usuario@usuario-VirtualBox:~/Escritorio/Tarea$
```

11. Con una sola línea, muestra el contenido del directorio *dir2* y el directorio *dir4*.

```
usuario@usuario-VirtualBox:~/Escritorio/Tarea$ ls dirx/dir2/ ; ls dirx/dir4/
dir3 prueba.txt
dir5 prueba.txt
usuario@usuario-VirtualBox:~/Escritorio/Tarea$
```

12. Crear un grupo llamado *GrupoClase*

```
usuario@usuario-VirtualBox:~$ sudo addgroup grupoclase
info: Seleccionando un GID del rango 1000 a 59999 ...
info: Añadiendo el grupo 'grupoclase' (GID 1004) ...
usuario@usuario-VirtualBox:~$
```

### 13. Crear un usuario llamado *UsuarioClase*

```
usuario@usuario-VirtualBox:~$ sudo adduser usuarioclase
info: Añadiendo el usuario 'usuarioclase' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Añadiendo el nuevo grupo 'usuarioclase' (1005) ...
info: Adding new user 'usuarioclase' (1005) with group 'usuarioclase (1005)' ...
info: Creando el directorio personal '/home/usuarioclase' ...
info: Copiando los ficheros desde '/etc/skel' ...
Nueva contraseña:
CONTRASEÑA INCORRECTA: La contraseña tiene menos de 8 caracteres
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
Cambiando la información de usuario para usuarioclase
Introduzca el nuevo valor, o presione INTRO para el predeterminado
    Nombre completo []:
    Número de habitación []:
    Teléfono del trabajo []:
    Teléfono de casa []:
    Otro []:
¿Es correcta la información? [S/n] s
info: Adding new user 'usuarioclase' to supplemental / extra groups 'users' ...
info: Añadiendo al usuario 'usuarioclase' al grupo 'users' ...
usuario@usuario-VirtualBox:~$
```

### 14. Añadir el usuario *UsuarioClase* al grupo *GrupoClase*

```
usuario@usuario-VirtualBox:~$ sudo adduser usuarioclase grupoclase
info: Añadiendo al usuario 'usuarioclase' al grupo 'grupoclase' ...
usuario@usuario-VirtualBox:~$
```

### 15. Mostrar información del usuario *UsuarioClase* y comprobar que efectivamente pertenece al grupo *GrupoClase*

```
usuario@usuario-VirtualBox:~$ id usuarioclase
uid=1005(usuarioclase) gid=1005(usuarioclase) grupos=1005(usuarioclase),100(users),1004(grupoclase)
usuario@usuario-VirtualBox:~$
```

### 16. Mostrar información del grupo *GrupoClase* donde se vea que *UsuarioClase* pertenece a él

```
usuario@usuario-VirtualBox:~$ groups usuarioclase
usuarioclase : usuarioclase users grupoclase
usuario@usuario-VirtualBox:~$
```

17. Eliminar el grupo *GrupoClase* solo si está vacío, lo que se supone que dará error.

```
usuario@usuario-VirtualBox:~$ sudo delgroup --only-if-empty grupoclase
fatal: ¡El grupo `grupoclase' no está vacío!
usuario@usuario-VirtualBox:~$
```

18. Eliminar el usuario *UsuarioClase*.

```
usuario@usuario-VirtualBox:~$ sudo deluser usuarioclase
info: Eliminando el «crontab» ...
info: Eliminando al usuario `usuarioclase' ...
usuario@usuario-VirtualBox:~$
```

19. Eliminar el grupo *GrupoClase* solo si está vacío.

```
usuario@usuario-VirtualBox:~$ sudo delgroup --only-if-empty grupoclase
info: Eliminando al grupo `grupoclase' ...
usuario@usuario-VirtualBox:~$
```