

Nombre y apellido:

DNI:

1. Escriba un comando para borrar archivos y directorios vacíos en la carpeta actual y sus subcarpetas.

Solución. `rm -r $(find . -empty)`

2. Escriba un comando para mostrar en pantalla *todas* las líneas duplicadas de un archivo.

Solución. `sort < archivo | uniq -D`

3. Escriba un comando para encontrar líneas dentro de un archivo que contengan un DNI. Considere como DNI válido a un número de 8 dígitos, que no comienza con ceros e incluye un guión («-») como separador de miles.

Solución. `grep -Ew "[1-9][0-9]-[0-9]{3}-[0-9]{3}" archivo`

4. Escriba un comando para *quitar* permisos de ejecución a todos los usuarios para cada uno de los archivos del directorio actual que terminen en «.py».

Solución. `chmod a-x *.py`

5. Considere el siguiente script:

</> Código

```
[ $# -ne 1 ] && exit 1

for RUTA in $(echo $PATH | tr ":" " "); do
    [ -x "$RUTA/$1" ] && echo "$RUTA/$1" && exit 0
done

exit 2
```

a) ¿Cuál es el propósito del programa?

Solución. Encontrar la ubicación de un ejecutable.

b) ¿Qué programa cumple con la misma función?

Solución. which.

c) ¿Qué representan los diferentes códigos de salida?

Solución. El código 1 significa que no hay argumentos suficientes, el código 2 significa que el archivo no se encuentra, el código 0 significa que se encontró el ejecutable.

6. El siguiente programa debería realizar una división entre dos números, pero tiene errores. Corríjalo.

</> Código

```
if [ $# != 2 || $2 == 0 ]; then
    echo "No se puede realizar la division."
    exit 1
else
    echo $(( $1 / $2 ))
    exit 0
fi
```

Solución.

```
if [[ $# -ne 2 || $2 -eq 0 ]]; then
    echo "No se puede realizar la division."
    exit 1
else
    echo $(( $1 / $2 ))
    exit 0
fi
```