

Práctica DOS

EL EDITOR EDIT

El editor *EDIT* se invoca con la instrucción `edit` desde línea de comandos seguido del nombre del archivo que se quiere editar o crear. Si no se teclea ningún nombre, el editor arranca con un fichero sin nombre que se puede guardar con un nombre determinado y donde se quiera al salir del editor.

Practicar: Crear con el editor un fichero nuevo, introduciendo un texto cualquiera.

El editor *EDIT* tiene varios menús:

- ✓ Archivo: Operaciones referentes al manejo del fichero que estemos creando o editando.
- ✓ Edición: Cortar, copiar, pegar, borrar texto. Si se quiere repetir la última búsqueda que se hizo, también se puede hacer con la tecla de acceso rápido F3.
- ✓ Buscar: Buscar, reemplazar texto.
- ✓ Ver: Opciones de la ventana.
- ✓ Opciones: Configuración, colores.
- ✓ Ayuda: Comandos, versión del editor. También se puede obtener ayuda, con la tecla F1.

Practicar: Navegar por los menús y opciones.

Utilizar el menú de ayuda para averiguar el funcionamiento de las diferentes opciones básicas del editor.

A partir de un texto que contenga el fichero practicar las opciones de insertar texto, borrar texto, movimiento del cursor, copiar, cortar, pegar, etc.

Utilizar las opciones de búsqueda de una cadena dentro del fichero, y las opciones del menú de manejo de archivos.

PROCESAMIENTO POR LOTES (.BAT)

Los ficheros de procesamiento por lotes contienen varios comandos que se ejecutan secuencialmente sin intervención del usuario. Su ejecución se interrumpe pulsando `Ctrl + C`.

Practicar: Editar un fichero nuevo llamado `Estado.bat` e introducir las líneas necesarias para que se vea por pantalla la versión del DOS que estamos utilizando, la fecha actual y la hora actual.

Practicar: Realizar un fichero nuevo llamado `VisualizaSubdirectorio.bat` que cambie a un directorio distinto del activo, por ejemplo el directorio de windows, y nos saque un listado pantalla por pantalla de los ficheros y directorios que hay en él.

El carácter `@` situado delante de una instrucción dentro de un fichero batch evita que la instrucción aparezca en la pantalla al ejecutarse el fichero (pero NO evita que aparezca el resultado de la instrucción!!!). La instrucción `echo` es similar, pero se aplica a grupos de instrucciones: `Echo off` evita la aparición de las instrucciones siguientes, hasta

que aparezca echo on o termine el fichero .bat. Por último, la instrucción *rem* permite insertar comentarios en un fichero batch.

Practicar: Realizar un fichero nuevo llamado EchoYRem.bat que tenga como inicio el nombre de la autora y la fecha actual, pero que esté como comentario al programa. Después copiar el fichero a un fichero nuevo sin que salga por pantalla, y borrar el fichero de manera que esto sí aparezca por pantalla.

Las instrucciones de DOS tienen opciones que permiten especificar diferentes funciones para la misma instrucción. Se puede conseguir la misma funcionalidad en los ficheros .bat mediante el uso de parámetros: el texto que se introduzca al lado del nombre del fichero .bat puede ser utilizado dentro del propio proceso batch, al ser almacenado en una especie de variables de la forma %1, %2, %3,

Practicar: Editar un fichero que se llame Leer.bat, que incluya las siguientes líneas:

```
@echo off
type %1 | more
echo Gracias!!!!
```

Guardar el fichero, teclear 'leer Leer.bat' y observar lo que ocurre.

Control del proceso de ejecución

Por defecto la ejecución de los ficheros batch es secuencial, pero existe un conjunto de instrucciones que permiten romper la secuencialidad:

1. **PAUSE:** Detiene temporalmente la ejecución de un fichero por lotes, hasta que se pulsa una tecla.

Practicar: Escribir un fichero .bat que haga un listado de los ficheros del directorio, primero ordenados por nombre y luego por extensión. Entre ambas operaciones meter un PAUSE.

2. **GOTO:** Desvía el proceso de ejecución de un fichero por lotes a otra parte del fichero. Se utiliza escribiendo: *GOTO [:] etiqueta :etiqueta*. Si se diese lugar un bucle infinito al utilizar esta estructura habrá que cortar la ejecución del fichero con Ctrl+C.

Practicar: Escribir un fichero .bat donde la orden *GOTO* desvíe la ejecución secuencial del fichero. Para comprobarlo introducir entre *GOTO* y la etiqueta un mensaje de echo, que no aparecerá si el *GOTO* se ejecuta correctamente.

3. **IF:** Desvía el proceso de ejecución condicionalmente, dependiendo de si es cierta o no una condición. Si la condición es cierta, se ejecuta la instrucción específica de la condición, y después se ejecuta la línea siguiente al if. Si la condición es falsa, salta directamente a ejecutar la instrucción siguiente al if.

If condición instrucciónIf
RestoInstrucciones

Practicar: Escribir un fichero .bat donde se compare la entrada del usuario. Si el usuario elige la opción a, sacar un mensaje advirtiendo de ello y si no sacar un mensaje diciendo que el usuario eligió otra opción.

4. **CALL:** Permite ejecutar otro fichero .bat desde el fichero en el que se hace la llamada con CALL. Cuando termine la ejecución del segundo, el control del proceso regresará a la instrucción siguiente a la línea que tiene el call. Su sintaxis es:

Call fichero

Practicar: Escribir un fichero .bat donde se haga la llamada al fichero Leer.bat, usando esta instrucción. Después de la llamada sacar un mensaje de finalización.