

MÓDULO:

Sistemas informáticos

**UT4_4_Archivos de
procesamiento por lotes**

ÍNDICE DE CONTENIDO

1	¿Qué son los archivos de procesamiento por lotes?	2
2	Órdenes especiales para los archivos por lotes	2
2.1	Orden Echo	2
2.2	Orden Pause	2
2.3	Orden Rem	3
2.4	Orden call	3
3	Añadiendo parámetros.....	3
4	Estructuras condicionales y repetitivas	3
4.1	Orden If.....	3
4.2	Orden goto.....	4
4.3	Bucles for/do.....	4
4.4	Orden shift	5
5	Uso de variables	5

1 ¿Qué son los archivos de procesamiento por lotes?

Los archivos de procesamiento por lotes permiten agilizar y simplificar el trabajo diario. Para ejecutar estos programas basta solamente con escribir su nombre. También se les conoce como archivos por lotes o archivos **batch**.

En esencia, se trata de un archivo de texto que contiene un listado de órdenes o comandos que se ejecutan uno detrás de otro. Todos los nombres de archivos de procesamiento por lotes deben tener la extensión **.bat**. Para crear un archivo **.bat** se puede emplear cualquier de texto (bloc de notas, Notepad++, etc.) con la única observación de que la extensión tiene que ser **.bat**.

Los archivos de procesamiento por lotes no deben tener el mismo nombre que cualquier otra orden del sistema operativo o de cualquier programa de aplicación que el computador pueda tener. Si se crea de forma accidental un archivo por lotes que tenga el mismo nombre que una orden del sistema operativo, el sistema operativo ejecutará siempre la orden y nunca el archivo por lotes. Todas las órdenes que el archivo por lotes contenga deberán ser escritas en cada línea del archivo, o sea una orden por línea.

La forma más sencilla de cancelar la ejecución de un archivo de procesamiento por lotes, una vez que este ha inicializado su ejecución, es pulsando la combinación de teclas **Ctrl+C**.

2 Órdenes especiales para los archivos por lotes

Dentro de un archivo por lotes se pueden agregar órdenes especiales que le proporcionan un mayor control sobre la forma en que se interpreta o funciona el archivo mismo. Estas órdenes especiales permiten crear archivos de procesamiento por lotes que en realidad llegan a ser casi programas.

2.1 Orden Echo

La orden ECHO tiene dos usos:

- Se utiliza para controlar si el sistema operativo visualiza todas las órdenes por pantalla que contiene el archivo por lotes.
- Mostrar por pantalla algún mensaje.

Echo está activada por defecto. Lo que quiere decir que el sistema operativo visualiza cada orden del archivo a medida que se ejecuta. Si echo se desactiva, no se visualizarán las órdenes de tratamiento por lotes, pero se mostrará cualquier resultado producido por las mismas.

La orden echo tiene la siguiente forma general: `echo on/off/mensaje`

- Para desactivar la orden ECHO, se escribe:

`Echo off`

- Para activarlo, se escribe:

`Echo on`

2.2 Orden Pause

Esta orden sirve para colocar una pausa en la ejecución del archivo de procesamiento por lotes, y adopta la siguiente forma general:

`Pause`

Cuando se encuentra con una orden pause, el sistema operativo visualiza el mensaje:

Presione cualquier tecla para continuar . . .

2.3 Orden Rem

Esta orden sirve para agregar un comentario dentro del archivo por lotes.

2.4 Orden call

Algunas veces necesitará ejecutar otro archivo por lotes desde el interior de un archivo de procesamiento por lotes. La mejor forma de hacerlo es mediante la orden call, cuya forma general es la siguiente:

Call archivoporlot

3 Añadiendo parámetros

A veces se puede crear un archivo de procesamiento por lotes que funcione ligeramente distinto según la forma en que se utilice. Los archivos batch tienen la posibilidad de utilizar un indicador %n, donde 'n' es un número entre 0 y 9 para admitir parámetros reemplazables de entrada.

Se pueden especificar hasta 10 parámetros de entrada (%0 y %9) de los cuales, el parámetro "%0" se sustituye siempre por el nombre del archivo por lotes.

4 Estructuras condicionales y repetitivas

Hasta ahora hemos estado tratando con archivos por lotes secuenciales, en los que cada línea se va ejecutando una tras otra. Pero esto no es lo habitual, puesto que es sabido que la verdadera potencia en un archivo batch reside en la posibilidad de efectuar saltos y bifurcaciones, permitiendo diversas secuencias lógicas.

4.1 Orden If

Con frecuencia resulta útil crear un archivo por lotes que haga cosas distintas dadas unas determinadas condiciones. Para llevar a cabo esto, el sistema operativo nos proporciona la orden if destinada para los archivos de tratamiento por lotes, que adopta la siguiente forma general:

IF condición orden1 ELSE orden2

Si la condición resulta ser verdadera, se ejecuta la orden1 que sigue a la condición. *Orden1* puede ir seguido de la palabra clave *else* y, a continuación, una *orden2* que se ejecutará si la condición especificada es falsa.

También se pueden realizar distintos tipos de comparaciones usando el siguiente formato:

IF [/I] cadena1 comparación cadena2 orden

Donde comparación puede ser:

EQU – igual (equal)

NEQ - no igual

LSS - menor que

LEQ - menor que o igual

GTR - mayor que

GEQ - mayor que o igual

Comprobando si existe un archivo. Se puede comprobar si existe un archivo o un grupo de archivos utilizando la condición **exist** de la orden if, que tiene la siguiente forma general:

```
IF EXIST nombredearchivo orden
```

En el que *nombredearchivo* es el nombre del archivo que se está buscando. El *nombredearchivo* puede incluir tanto un especificador de unidad como un nombre de camino.

El uso de not. Se puede colocar delante de la condición if la palabra **not**, la cual, entonces, cambiará completamente el resultado de la condición.

4.2 Orden goto

La orden **goto** se emplea para indicarle al sistema operativo que ejecute las órdenes de un archivo por lotes en un orden no secuencial. La forma general de goto es:

```
GOTO etiqueta
```

En el que etiqueta es una etiqueta definida en alguna otra parte del mismo archivo por lotes. Cuando se ejecuta goto, el sistema operativo salta a la etiqueta especificada y comienza a ejecutar las órdenes a partir de ese punto. Usando goto, se puede hacer que la ejecución del archivo batch salte hacia delante o hacia atrás en sus líneas de órdenes.

Todas las etiquetas deben comenzar con dos puntos. Cuando se emplea la orden goto, en las etiquetas que contienen órdenes hay que colocar dentro de ellas otra orden goto que salte a otro bloque de órdenes, porque si no, se ejecutarán las ordenes goto que siguen del siguiente bloque.

4.3 Bucles for/do

Se puede repetir una serie de órdenes que usen distintos argumentos utilizando la orden for, que adopta la siguiente forma general:

```
FOR %%var IN (listas argumentos) DO orden
```

```
FOR %%var IN (lista argumentos) DO (
```

```
comando
```

```
comando
```

```
...
```

```
)
```

Observar que la variable "var" va precedida por **dos símbolos de "%"**. Además, **el nombre de la variable debe ser UNA SOLA LETRA** (p.ej: %%n, %%i, %%j ,etc) que irá tomando los valores de la lista de argumentos. Los argumentos deben estar separados por espacios en blanco. For repetirá la orden tantas veces como argumentos existen. Cada vez que for repita, var será sustituido por argumento desplazándose de izquierda a derecha en la lista.

Señalar también que una orden for se puede usar para ejecutar otra orden for, es decir, se puede usar como objeto de do, con lo es posible anidar bucles.

4.4 Orden shift

Como ya sabemos, tan solo hay 10 parámetros sustituibles, del %0 al %9. Se puede emplear la orden shift para conseguir el acceso a más de 10 argumentos de la línea de órdenes.

Cada vez que se ejecuta shift, el contenido de los parámetros reemplazables es desplazado hacia la izquierda una posición. Lo que había en %0 se pierde y el nuevo argumento, si existe, entra en %9, ocupando en %0 lo que había en %1, el %1 tomaría lo que había en %2, etc.

Lo más importante que hay que recordar en torno a los argumentos que se usen en un archivo por lotes, es que deben estar separados por espacios en blanco, ya que el sistema operativo no reconoce ningún otro carácter como separador.

5 Uso de variables

En los archivos batch se pueden utilizar variables para luego poder recuperar su valor introduciendo éstas entre signos de porcentajes. Una vez creada la variable, se invoca con **%variable%**.

`%NombreVariable%`

Para establecer una variable, se utiliza el comando **Set**.

Parámetros:

Set /a → Da valor a una variable utilizando operaciones aritméticas.

Set /p → Permite la entrada de datos por parte del usuario. El valor introducido por el mismo define la variable.

Para saber más sobre los archivos de procesamiento por lotes, se puede consultar el siguiente [enlace](#).