

Introduction To Batch Files in Windows

Introduction To Batch Files

- Batch files or scripts are small text files that carry out a series of commands. Here are some of the things you can do with a batch file:
 - Customize your system with programs that run at startup time
 - Copy files from multiple directories to create backups
 - Run a series of programs with one command
 - Simplify complex commands by putting them in a file with a simple name
 - Any repetitive series of commands that need to be run in several directories

Introduction To Batch Files

- Batch files provide a way to do many things that can't be done in a graphical user interface (GUI) such as Windows.
- They can be a relatively simple series of commands, or you can put in simple conditional "if" statements or loops!

cmd.exe – Windows Command Interpreter

- Batch files have the extension BAT or CMD.
- Files of this type are recognized and executed through a system file called the ***command interpreter***.
- In Windows the command interpreter is the file **cmd.exe**.
- The large assortment of versatile commands available in Windows makes batch files a powerful tool.

Constructing a batch file

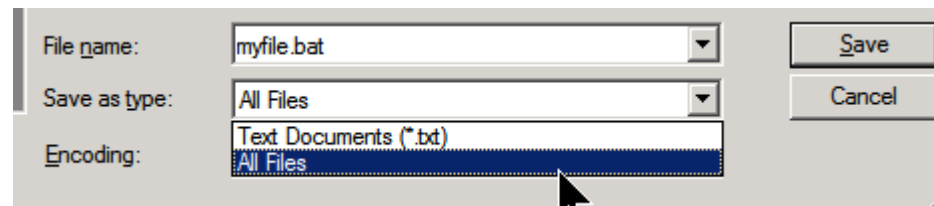
- We will use Notepad to create a simple batch file which will execute as follows:
 - It will print the message *“Hello this is a test batch file”*
 - Execution will then pause, and on entering any key, will display a list of all files on your home drive:

Steps to Creating a Batch file

- Click **Start - Run**
- Open **notepad** and type the lines below in the file

```
@ECHO OFF  
ECHO Hello this is a test batch file  
PAUSE  
DIR C:
```

- Save the file with a “.bat” extension, e.g. **myfile.bat**
- **Be sure that Notepad is saving as "All files" and not as a text file. See the figure below.**



Running the Batch File

- There are 3 ways in which you can run the batch file:
 - Open command prompt and type the filename of file including the full path if necessary.
 - Double-click on the name of the file in the GUI
 - Use Start-Run and type the filename of file including the full path

Batch Commands

- The first line in a batch file usually consists of the command **@ECHO OFF**
 - By default, a batch file will display its commands as it runs.
 - The @ command is used to prevent this display (“echo”) back

Adding Parameters to a Batch File

%1, %2, .. etc

- A **parameter** is a piece of information that you can supply with a command on the command line when running a batch file,
 - which can be used to modify the action in some way.
- The percent followed by a numeric value, (beginning with 1), allows users to add arguments within a batch file.
- Below is an example of what can be used in a batch file.
ECHO Hello %1

Exercise

Create a batch file with the command ECHO Hello %1

- *Run the file through the command line, (or start-run).*
- *Now run the file but add your name on the command line, e.g.*
C:\> myNameFile Martin
(myNameFile is the name of your batch file)
- *What does it display?*
- *What happens if you run it with more than one argument, e.g.*
C:\> myNameFile to everyone
- *i.e. You want to display the string “Hello to everyone”*
- **Hint:** To fix it you can extend the command with %2, %3, etc, with %2 referring to argument 2, etc

More Batch File Commands

- **ECHO** Will echo a message in the batch file. Such as "ECHO Hello World" will print **Hello World** on the screen when executed.
 - Note if you have not typed @ECHO OFF at the beginning of the batch file this will also print "ECHO Hello World" and "Hello World".
 - If you would just like to create a blank line, type **ECHO.** adding the period at the end creates an empty line.
- **PAUSE** Prompt the user to press any key to continue.
- **REM** Allows you to place “remarks” into the batch file without displaying or executing that line when the batch file is run.
- **EXIT** Exits from the console window

How would you check that a file exists?

For example, what line of code in your batch file would check that **myName.txt** exists in a particular location e.g. in the **c:\temp\lab9** folder ?

Answer:

Simple Answer:

```
DIR C:\temp\lab9\myName.txt
```

More Correct Answer:

```
IF EXIST C:\temp\lab9\myName.txt (  
    ECHO file exists  
) ELSE (  
    ECHO file doesn't exist  
)
```

How would you check that a file contains a particular string?

For example, what line of code in your batch file would check that **MyName.txt** contains the string “Joe”?

Answer:

Find “Joe” myName.txt

- Note: although Windows/MS-DOS is not case sensitive, the ‘Find’ command **is** case sensitive
 - “Joe” is not the same as “joe”
- To find out more about the ‘Find’ command, look up the help.