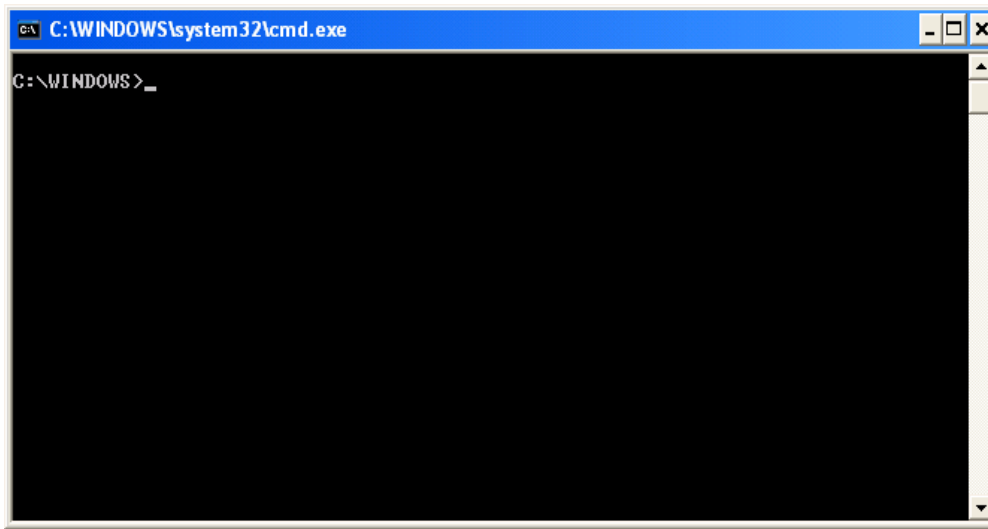# IT Scripting and Automation

Review:

Operating Systems Fundamentals
Command Line Interface - DOS

Lecturer: Art Ó Coileáin

The Command Prompt Window

- The command prompt is run from its own window by invoking the Windows *command interpreter*.

    - A command interpreter is the part of a computer's operating system executes commands entered either interactively by a human user being or from a program.

    - To open the command prompt window type "cmd" into Start-Run, or open the command prompt through Start-All Programs-Accessories.

```
C:\WINDOWS\system32\cmd.exe

C:\WINDOWS>_
```

The command prompt is a window that by default displays the current directory (or "folder").

It has a blinking cursor ready to accept typed in commands.

Some Basic MS-Dos Commands

Help

- How to get help: **help command-name** or **command-name /?**

Navigation

- The Command '**cd**'   *Displays the name of or changes the current directory*
  - **cd **   or   **cd ..**   or   **cd .**   or   **cd NameOfFolder**
- Changing drives: **F:\> E:**

Directory/folder Listing:

- The '**dir**' command
- Wildcard Symbol   *****

Useful Commands

  - Copy: **copy * \windows**
  - Move: **move test.txt \windows**
  - Delete: **del *.pdf**

- Directory Commands: **mkdir** vs **rmdir**   (or **md** vs **rd**)

MS-Dos - Batch Files

- Batch files or scripts are small text files that carry out a series of commands.
- They can be a relatively simple series of commands, or you can put in simple conditional "if" statements or loops.
- Batch files have the extension BAT or CMD.
- We use Notepad to create a simple batch file.

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

  - ```
    ECHO Hello %1
    ```

- More Batch Commands

  - ECHO   :LABEL   CALL   EXIT   GOTO   PAUSE   REM

Windows / Unix

MS-DOS / Windows	UNIX / Linux	
dir	ls	List a directory of files or get information about files
copy	cp	Copy a file from one place to another
move	mv	Move a file from one place to another
del or erase	rm	Delete (remove) a file
type	cat	Type a file out to the screen (or redirected to a printer)
mkdir	mkdir	Attach a new subdirectory to the tree at this tree junction
rmdir	rmdir	Delete a subdirectory

Redirection Operators: '>' and '>>'

- Redirection (>) and append (>>) operators.

What does this MS-dos batch file do?

@echo off

echo My directory Listing > myFile.txt

Dir >> myFile.txt

Echo end of Listing >> myFile.txt

Type myFile.txt

Parameters

- 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

IF Statement

- The IF command allows you to make decisions within a Batch File.

  - *IF* **Condition Command**

- Two strings are equivalent.

  IF String1 == String2 Command.

  This comparison is often used with parameters to check for a particular entry

  A file exists in the current directory. The logical test checking for a file has the format:

  IF EXIST filename

  IF NOT EXIST filename

GOTO & Label

- GOTO and LABELS allow you to control the flow and operations that are performed in a batch file.

`GOTO <label>`

- `<label>` Specifies a text string that is used as a label in the batch program.

- Within a batch program/script, GOTO directs command processing to a line that is identified by a label. When the label is found, processing continues starting with the commands that begin on the next line.
- The destination is written as :*label*

IF EXIST

In this next batch file you will use an IF command of the form:

```
IF EXIST filename COMMAND
```

You will also use the **CALL** command which <u>calls another batch file</u>.

What are these batch files doing?

```
REM todaysMenu.bat
@ECHO OFF
ECHO TODAY'S MENU
ECHO Steak and Chips
ECHO Stir Fried Veg
ECHO Seafood Pasta
ECHO Pizza
```

```
REM menu.bat
@ECHO OFF
IF EXIST h:\todaysMenu.bat GOTO
MENU
ECHO No Menu today
GOTO END
:MENU
CALL h:\todaysMenu
:END
```

Environment Variables

- They help describe the environment in which programs are running

- Programs can query these variables to find out things about the system
  - The name of the computer
  - The name of the user running the program
  - The location it is installed
  - The location for storing temporary files
  - …

- Use the SET command to see a list of all the environment variables available.

Environment Variables (2)

- User Variables vs System Variables.
- Create your own environment variables with the SET command:
  - `e.g. SET MyVAR=Hello World`
- Display the value stored in a variable as follows:
  - `ECHO %MyVar%`
  - This will then display the string "Hello World" on the screen
- Environment variables can also be used within a batch file to **prompt** the user to type something in
- The syntax is:
  - `SET /P variable=[promptString]`
    - `e.g. SET /P choice=What would you like to do?:`
- NOTE: These variables will be lost when you close the command prompt window. To create variables which persist (i.e. will still be there when you restart) you need to use the windows environment management tool.