

Introduction to the Windows Command Prompt

cd -change directory

cd .. (parent dir)

**cd ** (root dir of current disk)

cd x: (change disk, note effects on current dir of current drive.)

dir - list directory contents

dir /s, /l, /w, /o, /a (subdirs, lowercase, wide-2col, ordered-alphabetic, all)

dir /d (directories only)

type - print out plain text file

print - send plain text to default printer

prntscrn command-use PrintScreen to put a screen capture in the clip board:

use *alt-prntscrn* to capture the current window. Paste from clipboard to a program to get the image.

help to learn about commands. **help command** to learn about a specific command, e.g.: **help dir**

cls to clear the screen

copy to copy a file

del to erase a file

ren to rename a file

mkdir to create a directory

rmdir to delete a directory

deltree to delete folder and subfolders

Wildcards: Allow you to enter ambiguous file names for some commands.

? for a character

* for any number of characters

echo displays a message on screen. E.g. **echo Who is using this computer?**

@echo off stops a batch file from displaying commands as it does them.

set shows values that have been set (variables)

set followed by a letter shows variables that start with that letter, e.g. **set n** will show n variables.

set name=value creates a new variable and gives it a value

set /a 4+4 The /a parameter tells set to do arithmetic. In this case, it will print 8. Valid operators include *, /, +, -, and % (for remainder). Note: In batch files, use %% for the remainder function.

set /p name=Prompt String will set a variable to the value typed in after printing the Prompt String

The variable **%random%** is a built-in variable that returns a random number between 0 and 32767.

To get a more useful number, use the remainder operator % like this:

set /a myrandom=%random% % 10 + 1

This will set **%myrandom%** to a random number between 1 and 10. Change the 10 to get a different top number, like 100 or 6.

Pipe | allows one command's output to be passed to another command.

sort allows items to be sorted in several ways, default is alphabetic/numeric sort by starting character.

For example '**dir | sort**' will sort by the file date from oldest to newest.

if allows you to only do things under certain conditions, **else** lets you do things if conditions aren't met.

Example: **if %mynum% == 2 echo It's Two.**

if %mynum% == 2 (echo It's Two) else echo It's Not Two.