Using Windows at the Command Prompt – with a focus on Java

Overview

- Introduction to the Windows 7 command prompt interface
  - Continuing from introduction in CSCU9A1
  - Background and motivation
  - Launching and using a command prompt window
  - Basic essential commands
  - Executing programs
- Java
  - Compiling and executing Java programs
  - javac and java options

Background …

After the early days of punched cards and paper tape, interactive computing was through a line-by-line typed text command based interface – it was all that was possible:

No computer – just a terminal!

IBM PC, about 1980
Image: Wikipedia

MS-DOS started here…

… and motivation

Graphical User Interfaces (GUIs) require a lot of processing

- Require sufficient CPU power and information transfer speed
- Work poorly if you want to use a remote computer with a GUI
- Restrict use of powerful command line features

Using a command prompt window in a GUI context has benefits

- Fast local working
- Fast remote working
- Access to powerful command line features

And gives insight into how the operating system actually works

Most OS use command line type functions to support GUI operation

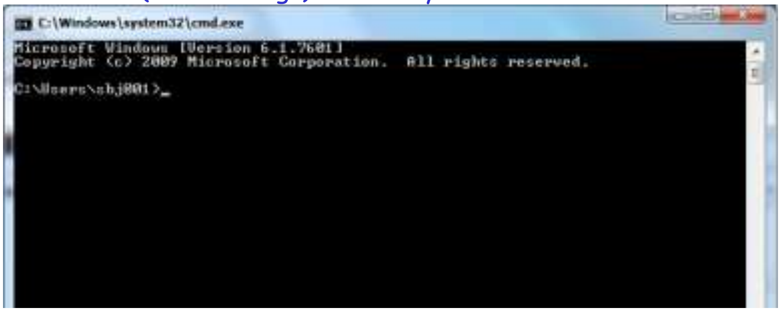
Of course, GUIs *are* useful, so balance needed!

Launching and using a command prompt window

Two ways to launch a Windows command prompt window:

1) Start menu, type **cmd** into the search box, finds **cmd.exe**, Enter

Opens command window with the user's "profile folder" as the current (or "working") directory:

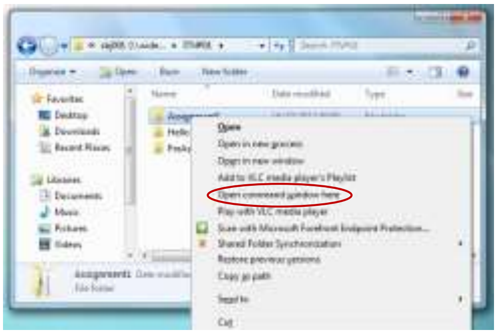


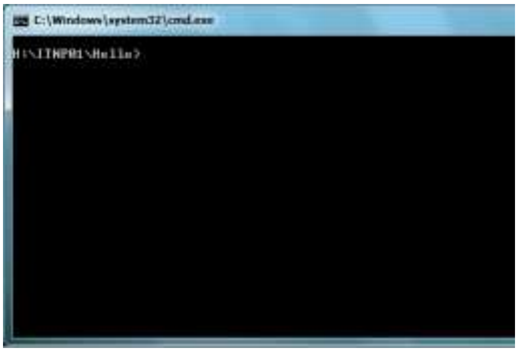
```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\sbj001>_
```

On University lab PCs, this is *not* a useful location!
Need to navigate to somewhere useful (home file store, H:)

2) Shift-right-click the mouse on a folder icon, choose "Open command window here"

Opens command window with *that folder* as the current (or working) directory:





```
C:\Windows\system32\cmd.exe
H:\ITNP01\Hello>
```

On University lab PCs, this *is* a useful location!

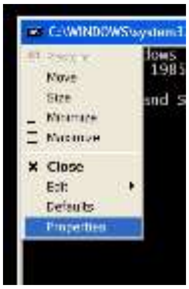
The "command prompt string" shows a home file store directory

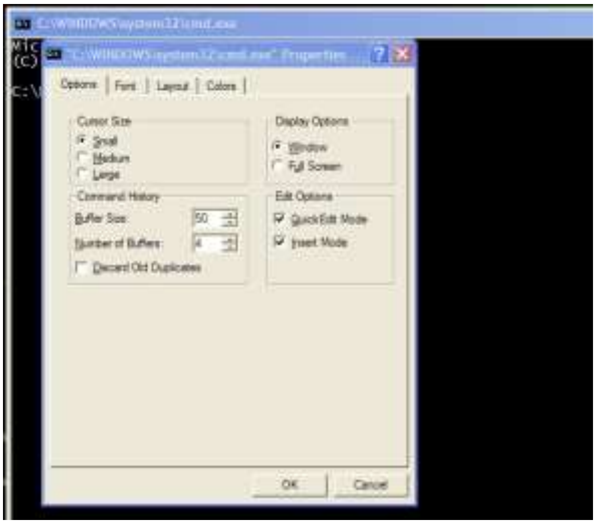
Useful tips and tricks

Customizing: Click the window icon at top left, drop down options appear:

Select Properties to set cursor, font, colours, etc

Under Options, selecting Quick Edit Mode is useful




C:\WINDOWS\system32\cmd.exe
"C:\WINDOWS\system32\cmd.exe" Properties
Options
Font
Layout
Colors
OK
Cancel

Cursor Size
● Small
○ Medium
○ Large

Display Options
● Window
○ Full Screen

Command History
Buffer Size: 50
Number of Buffers: 4
□ Discard Old Duplicates

Edit Options
☑ QuickEdit Mode
☑ Insert Mode

... continued

Re-using previous commands:

- Use up and down arrow keys to select a previous command line
- Enter to re-execute as it is
- Use left and right arrow keys, delete and more typing to edit, then Enter to execute modified command

Copy and paste: (If Quick Edit Mode is selected)

- Highlight text to be copied, Enter to copy
- Right-click the mouse to paste into the current command line

Press Tab key for filename completion

Windows guesses if there are multiple options!

Basic essential Windows commands

(often, colloquially, "DOS commands")

The command window allows you to work 'under the hood' of the operating system – to get the same effect as windows & point-click:

- You start off in some part of the file store, and may navigate around
- Working directory/folder always shown by the 'command prompt'
- Listing contents of current/another folder/directory

```
dir <optional folder name>
```

- Changing your current folder

```
cd <folder name>     or     chdir <folder name>
cd ..        will move you "up" a folder
```

- To change 'drives', type the drive name on a command line, e.g. `H:`
- Creating/deleting a folder/directory

```
mkdir <folder name>      or     md <folder name>
rmdir <folder name>      or     rd <folder name>
```

... continued

Getting help:

- To see the standard commands available, type `help`
- For help on a particular command, type `help` followed by the name of the command e.g. `help dir`
- You will notice that the options for `dir` are extensive and they can be used to do some quite powerful tasks
- ***Use `help` to find out how to copy, rename and delete files***

More commands:

```
type <file name>           Display file text contents
title <text>               Change the window's title bar
exit                       Close the command window
... | more                 Paginate long output
... > <file name>          Redirect output to file
start <file or folder name>   Open the doc or folder
```

Running Programs

Run a program by typing its name on a command line:

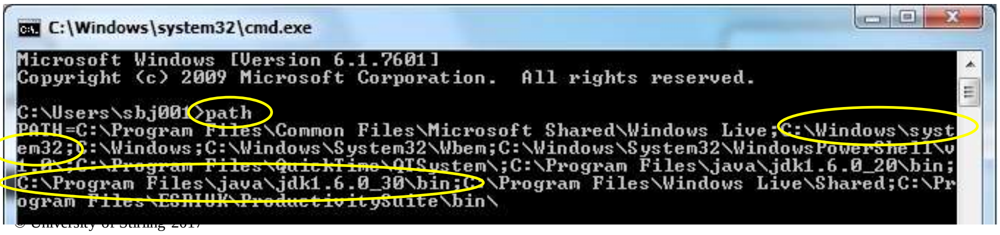
- For example, to start up Notepad:

notepad or **notepad <file name>**

- Or Java: **java -version** or **java HelloWorld**

Windows normally expects an executable file to be called **<something>.exe** and *in the current directory*

- So how does the command "**notepad**" work?
- Windows adds **.exe** then *searches* a list of folders for a file called **notepad.exe**
- The list of folders is called the *"path"* - can see it by typing **path**:



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\sbj001>path
PATH=C:\Program Files\Common Files\Microsoft Shared\Windows Live;C:\Windows\syst
em32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v
1.0\;C:\Program Files\QuickTime\QTSystem\;C:\Program Files\java\jdk1.6.0_20\bin;
C:\Program Files\java\jdk1.6.0_30\bin;C:\Program Files\Windows Live\Shared;C:\Pr
ogram Files\ESRIUK\ProductivitySuite\bin\
```

notepad.exe is in **C:\Windows\System32\notepad.exe**

That is just where Microsoft decided to put it!

What about **java.exe**?

Installed to (typically)

C:\Program Files\Java\jdk1.6.0_30\bin\java.exe

(which is in the path on previous slide)

But it is usually copied to **C:\Windows\System32\** at installation

So**java.exe** is on the path twice!

System admins decide what the path will be!

Can give full "path names" for programs on the command line:

"C:\Program Files\Java\jdk1.6.0_30\bin\java.exe"
(note the " ") **HelloWorld**

Essential for programs not (currently) on the path, eg:

"C:\Program Files\Textpad 5\textpad.exe"

Java

As you have already seen, Java programs can be directly executed via a command line:

- To start a Java program, you type:

```
java MyProgram
```

- `java.exe` is the Java Virtual machine, JVM

The launch process is:

- Windows adds `.exe` to `java` and searches the path
- Runs `java.exe`
- `java.exe` takes the given program name `MyProgram` (technically a "class" name), adds `.class`
- Java searches for `MyProgram.class` in the *current* directory
- Java loads the bytecode from `MyProgram.class` into RAM
- Locates the `main` method, and calls it

Compiling Java Programs

In order for the Java program to run, the file `MyProgram.java` must be translated into *byte code*

- This can also be achieved in the command window by:

```
javac MyProgram.java
```

- This uses a different program called `javac.exe` which is the Java Compiler (hence the 'c' at the end)
- javac finds the file `MyProgram.java` in the current folder and translates it into *bytecode* in `MyProgram.class` - a new or overwritten file
- The Java VM, `java.exe`, understands and can execute the program

`javac.exe` is installed in the same folder as `java.exe`

Java command line options

The `java` and `javac` programs have many possible options.

To find out what they are, just type `java` or `javac` without any other parameters

For example:

- `java -version` will tell you which version of the Java Virtual Machine you are using
- `java -splash:<image file> <class name>` will show a splash screen with the specified image while launching the program
- `javac -d <folder name> <source file>` will place the compiled bytecode file in the named folder
- `javac –verbose <source file>` will output messages about what the compiler is doing

- You will try Java command line options in the lab

End of lecture