



Command-Line Basics

Please go over this section carefully to review/practice the basics of the command line.



Underneath every operating system, there is a shell that allows the user to type commands to control the computer. This shell is called the **command-line (or command-prompt)**, or the **terminal** window. These terms are often used interchangeably.

Why do we need to learn the command-line?

1. we can run the software and repetitive tasks more efficiently
2. we can have better control over system functions and configurations
3. we can run software that is GUI-independent, like "git" (for version control)

For Windows, the underlying shell is based on legacy software called DOS (disk operating system). For macOS and Linux computers, the underlying shell is based on the UNIX operating system. There are different variations of this UNIX shell, for example, bash, or z-shell.

While many commands are the same for the Windows and Mac shells, there are some differences as well.



To open the shell window:

1. Windows command-line window. Type "**cmd**" in the start menu.
2. MacOSX terminal window (either "bash" or "z" shell). Type "**terminal**" at the search bar.

1. Navigate around directories (folders)

Your files and folders (also called directories) are organized in a tree structure. It's important to know "where" you are, "what" is in your current folder, and how to move around from folder to folder.

Let's assume you already have a folder called comp1800 in your Documents folder. Let's now make a working folder for today's exercise with the `mkdir` command.

When you need to specify the name of a file, or files...

- you can use the file name. Example: `dir hello.txt`
- you can use a full directory path of a file. Example: `dir ..\hello.txt` or `ls ../hello.txt` (one directory level up)
- you can use wildcard characters * or ? to specify a set of files. Example `dir .txt` shows all files that end with .txt. Example `dir d*.txt` will list all text files that start with the letter d. you can read more about it here: <https://docs.microsoft.com/en-us/previous-versions/windows/desktop/indexsrv/ms-dos-and-windows-wildcard-characters>



Windows command-line (by default) is NOT case-sensitive.

Traditionally, the backslash separates folders in file path names. Some shell variations recognize the forward slash also.

Mac/Unix/Linux terminal IS case sensitive. The forward-slash separates folders into file names.

NOTE: In Windows 10 there is an option to enable case-sensitivity <https://www.howtogeek.com/354220/how-to-enable-case-sensitive-folders-on-windows-10/>

You can do something like this to create a new folder, and navigate to that location:

```
On Windows: cd c:\users\username\Documents\comp1800 mkdir
demo01 cd demo01 On Mac: cd Documents/comp1800 pwd mkdir demo01
cd demo01
```

2. Create some new text files



No access

You don't have access to this synced block

Request access

Practice example:

1. Create an empty file and call it "file.txt"

```
type nul > file.txt (Windows) dir touch file.txt (Mac) ls
```

2. Write "This is the first line" into this file.

```
echo "This is the first line" > file.txt
```

3. Write "This is the second line" into this file

```
echo "This is the second line" >> file.txt
```

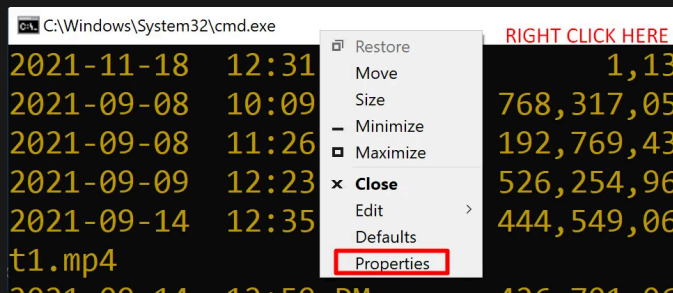
4. Show the contents of the text file.

```
type file.txt (for Windows) more file.txt (for Mac)
```

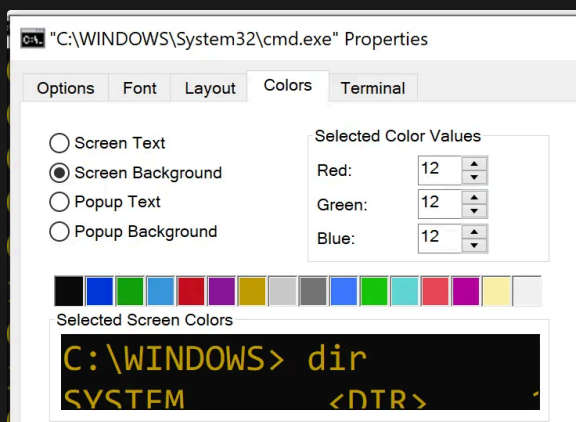
▼ Appendix: Some Miscellaneous Tricks (How to open command line window in a folder)

Question: How do I customize my window?

In Windows command-line, click on the top window bar, go to "Properties" → "Colors".



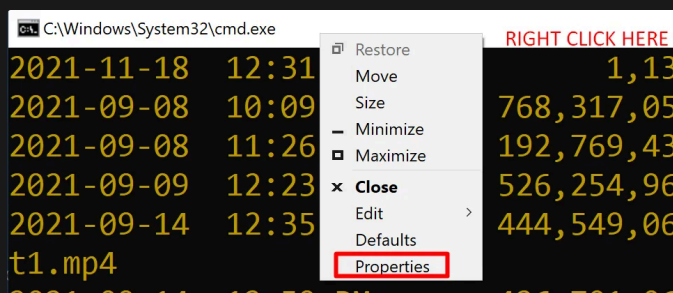
Adjust colour and text here:



On your Mac, go to menu "Terminal"—>"Preferences" to customize.

Question: How do I customize my window?

In Windows command-line, click on the top window bar, go to "Properties" → "Colors".



Adjust colour and text here:

