Learning the basics commands of Command Prompt

Cyprus University of Techology

Milto Miltiadou

March 8th, 2017

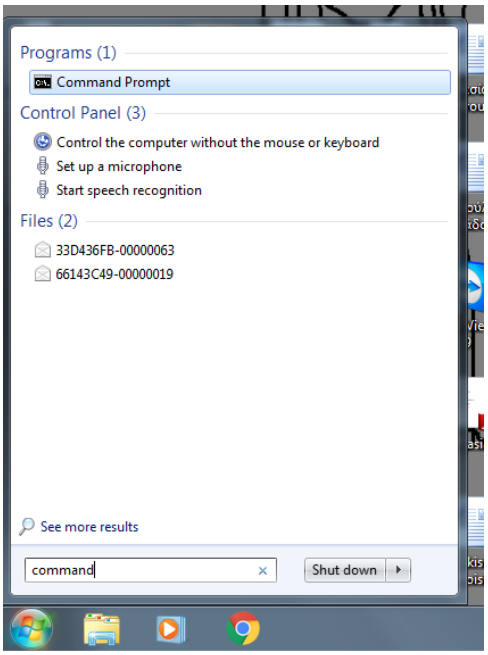
Introduction

This tutorial gives you an overview of the basic line commands that can be executed using the Microsoft Windows Command Prompt available on windows machines. By following the procedures of this tutorial, you will learn to:

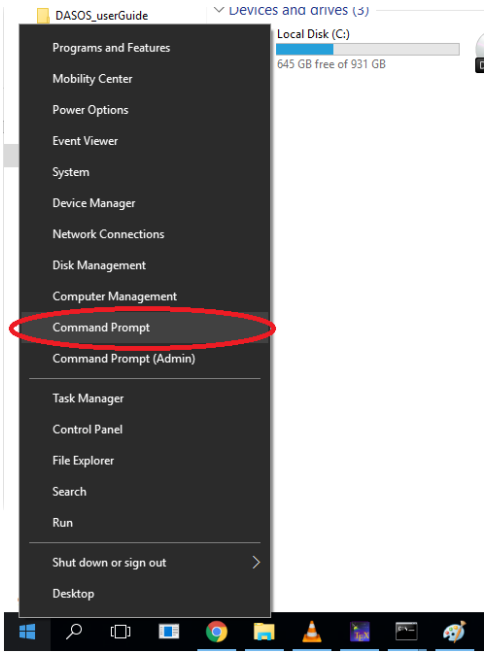
1. Open Command Prompt on Windows
2. View the contents of a directory
3. Change directory
4. More useful commands
5. Executing multiple commands using .bat files

1 Opening the Command Prompt window

The Command Prompt is the command-line interpreter on Windows machines. For example, it allows you to execute line commands for opening or modifying files. For windows 7 and lower, the Command Prompt is found by searching "Command Prompt" on the Star Menu Search bar (Figure 1a). For windows 10, you will find it by pressing right click on the start icon (Figure 1b). Inside the black window that appears, we can write and execute commands. Figure 2 shows a Command Prompt window. The follow sections give an overview of a few basic commands.



(a) Open Command Prompt on Windows 7 and older versions.



(b) Open Command Prompt on Windows 10.

Figure 1: How to open the Command Prompt.

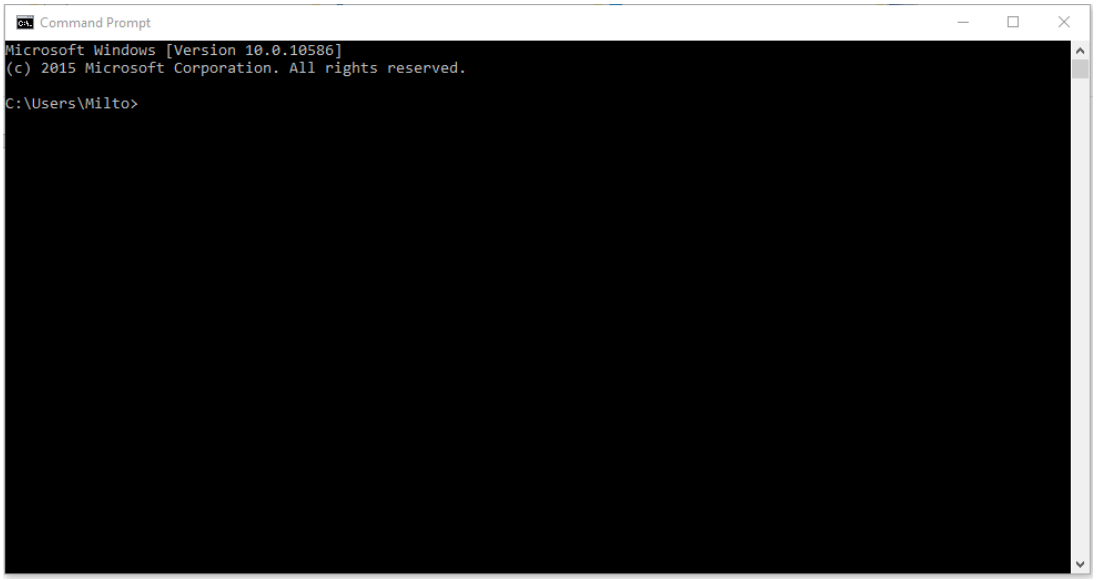


Figure 2: The Command Prompt Window

2 Viewing the contents of a directory - <dir>

In order to view the content of the current directory we use the command **dir**. The name of the command **dir** is derived from the word "directory".

Type the following at the command prompt and press enter:

$: **dir**

A list similar to the following appears:

24/02/2017	**11:17**	**<DIR>**		**.**
24/02/2017	**11:17**	**<DIR>**		**..**
28/07/2016	**21:45**	**<DIR>**		**.cache**
01/03/2017	**11:42**	**<DIR>**		**.qgis2**
01/12/2016	**13:09**		**2,032**	**.recently-used.xbel**
02/03/2017	**11:13**	**<DIR>**		**Desktop**
27/02/2017	**16:58**	**<DIR>**		**Documents**
01/03/2017	**15:44**	**<DIR>**		**Downloads**
15/11/2016	**01:44**	**<DIR>**		**Links**
15/11/2016	**01:44**	**<DIR>**		**Music**
02/03/2017	**15:26**	**<DIR>**		**Videos**

This is the list of all the files and subfiles of the working directory, which is the current directory. The working directory is indicated within Command Prompt. As shown Figure 2 the working directory is **C:\Users\Milto** and the above list contains all the files and subdirectories of that directory.

The <DIR> label indicates that the listed item is a directory itself. If the working directory is not a drive (e.g. **C:**), the first two directories (**.** and **..**) are always listed. The **.** directory is the current working directory (**C:\Users\Milto**) and the **..** directory is the directory of the folder that contains the working directory (**C:\Users**).

When a directory contains many items, the tag **/p** is very useful. For example, type the following command:

$: **dir /p**

This will print a page of the directory list and the next page appears once the button **Enter** is pressed.

3 Changing directory - <cd>

Moving from one directory to another is essential and the command **cd** (named after "change directory") is responsible for that. From the working directory (**C:\Users\Milto**), you can move to the subdirectory **Documents** by typing the following:

$: **cd Documents**

Or you may include the entire path of the directory of interest. For example:

$: **cd C:\Users\Milto\Documents**

By the way, when typing a directory you may use the **tab** button to quickly fill the name's of directories and files. Try typing:

$: **cd C:\Us**

and then press tab. It will automatically fill it to **cd C:\Users**. If more than one options apply, we may looping through them by pressing the button **tab** multiple times.

Additionally, it is essential to be able to move from a working directory to the folder containing that working directory. This is done by the following command:

$: **cd ..**

As mentioned before, the directory **..** is the directory of the folder that contains the working directory. Therefore, by using the command **cd** following with **..** , we can move backwards one folder.

Similarly, the following command does nothing because it brings you to the current directory, which is the **.** :

$: **cd .**

The final command related to changing directory for this tutorial is the following:

$: **cd **

Type the above command in Command Prompt. This command brings you to the root directory, which should by **C:**. Therefore the Command Prompt should now show:

C:\>

Please note that for the final command a baskshlash (\) is used and not a forward slash (/);

4 Quick Overview of some useful commands

There are numerous commands that can be executed from Command Prompt. Here a number of them are listed.

To create a directory, use the command **md** (make directory) as follow. Please replace the <directoryName> with the name of the new directory to be created.

$: **md <directoryName>**

With the command **rd** (remove directory), an empty directory can be deleted:

$: **rd <directoryName>**

For non-empty dirctories the tag \s should be added at the end as follow:

$: **rd <directoryName> \s**

For renaming a file:

$: **ren <oldName> <newName>**

For deleting a file:

$: **del <filename>**

For copying a file:

$: **copy <file> <destination>**

For example, the following command will copy the **file.txt** from the directory **C:Users** inside the folder **C:\HelloWords**:

$: **copy c:\Users\file.txt c:\HelloWorld**

Similarly the command for moving a file is the following:

$: **move <file> <destination>**

The parameter **NUL** represents te empty/null. Here, there is a hack for quickly generating an empty text file using the copy command:

$: **copy NUL emptyFile.txt**

5 Executing multiple commands using batch (.bat) files

A batch file is a script file in Microsoft Windows. It consists of a series of commands, which are stored in a plain text file (most commonly with extension .bat). These commands can be executed by the command-line interpreter (Command Prompt).

Figure 3 shows an example of a batch file that uses commands explained in this tutorial. Each line that starts with **::** is a comment, which means that they are ignored at execution time and they do not influence the interpretation of the commands.

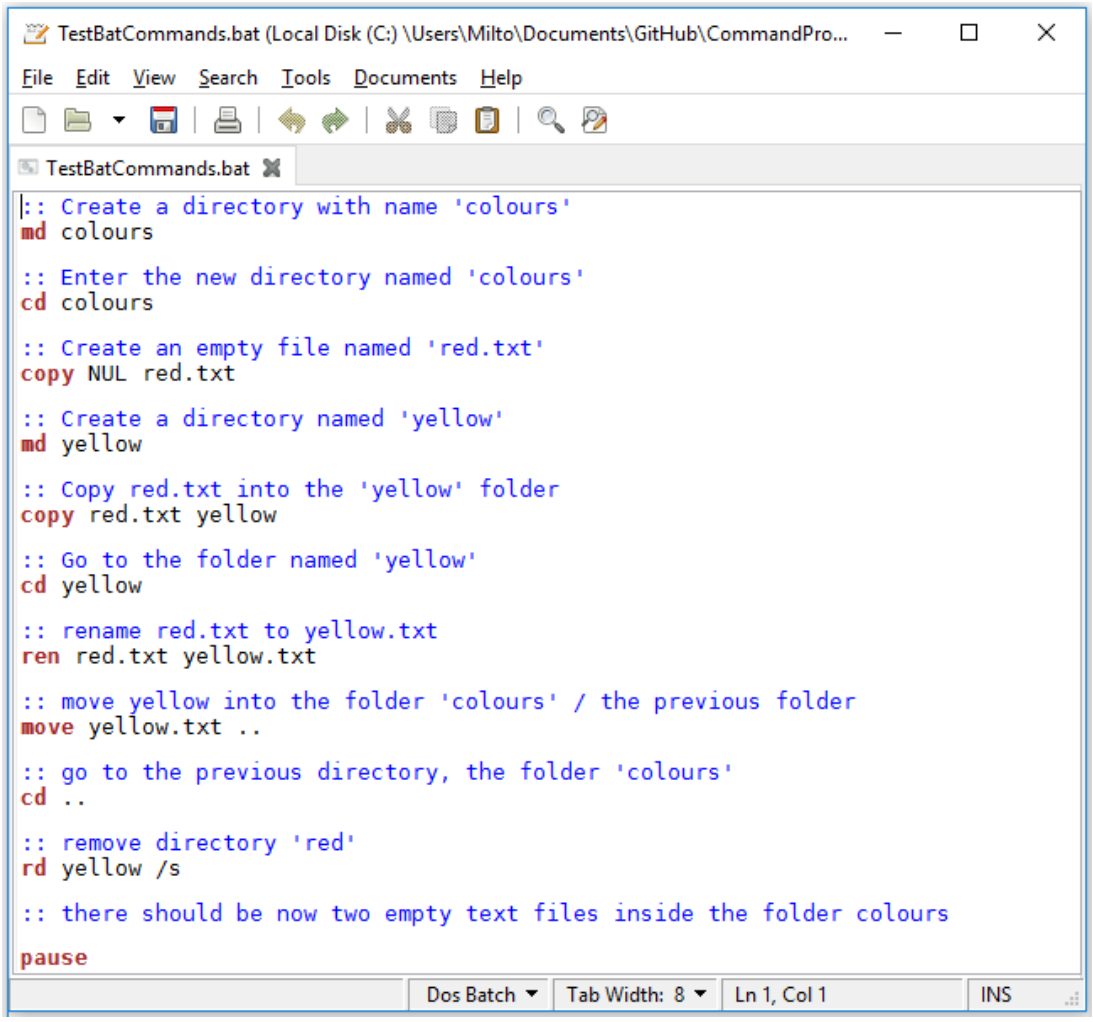


Figure 3: The Command Prompt Window

As mentioned before, batch files are plain text files and they can therefore been edited using a text editor. A good option for editing batch files is the gedit application, which also colours the commands. If gedit is not available, then WordPad will also work fine, but preferably avoid using notepad because notepad fails to identify new lines and editing with it is confusing.

To run a batch script file, just double click on the file and all the commands will be executed.