



Digital POWRR How-to Tech Tutorial

Linux Command Line Cheat Sheet

1) Finding where you are located in the system via command line

In Linux, once you have logged into the system and are situated at the command prompt, any time that you need to determine which directory/folder you are currently working within, just type the print-working-directory command:

- **pwd.**
- Issuing the above command will give you the current working directory or folder path where you are located within the system.
 - Why is this important? All commands that you type into a command prompt (unless you provide a specific directory pathway) will apply only to the files and folders that are located inside of the current directory that you are located.
 - For example: if you are located at the /home/jsmith folder and you want to rename files located directly in that folder, then you can just type the “mv” command (see item number 6 below) without the need to specify an additional folder path.

2) Looking for HELP information on a particular command line “command”

You can access help files for a particular command by typing “man” and the actual command that you want to look up information for:

- **man ls** (that is M, A, N, <space>, L, S)
 - This will return information on the ls command found within the Linux manual pages. “ls” is the command used to list files and folders.
- If you have finished reading the particular manual page, then you can just hit the q key to end the man page interface and return to the command line prompt.

3) Creating a new file via the command line in the current directory

The “touch” command can be used to create a new file (that is empty) in the current directory

- Type: **touch FILE1.txt** will create a new file in the current directory called FILE1.txt.
- Type: **touch FILE2.txt** will create a new file in the current directory called FILE2.txt.
 - Note that the files will be empty.

4) How to create a new directory or folder in the current directory you are located:

The “mkdir” command can be used to create a new directory or folder in the current directory that you are located.

- Type: **mkdir ANewDirectoryName** (that is M,K,D,I,R <space>,<name of a new directory>).
 - In this example, this will create a new directory in the current directory called “ANewDirectoryName”.
- Type: **mkdir Docs**
 - In this example, this will create a new directory in the current directory called “Docs”.
- After creating the directory, you can confirm that it is there by typing:
 - **ls -l** to confirm that the new directory is present (see item number 5 below).
- Type: **mkdir Docs/Pictures**
 - In this example, this will create a new directory in the Docs sub-directory called “Pictures” (we will confirm that it was created successfully at a later time).

5) Listing the contents of a folder or directory

When you are located in the directory that you want to see the contents of, use the following command:

- **ls -l** (that is L, S, <space>, <minus>, L)
- The -l option of ls allows you to see all of the detailed information for the files such as the time/date stamps as well as the respective file sizes and security permissions.
- We can use this command to see the file that we just moved in Item number 4 above.

6) How to Move Files or Rename them.

- The “mv” or move command can be used to both move files and rename them.
 - You can type: **mv FILE1.txt FILE3.txt** which will result in FILE1.txt being renamed to FILE3.txt
 - Type: **ls -l** to confirm the rename of the file to FILE3.txt
 - If you add a folder path before the FILE3.txt portion of the previous command, then it will also move the file into that new location while renaming it.
 - For example: **mv FILE3.txt ANewDirectoryName/FILE4.txt** will move FILE3.txt into a previously created folder called ANewDirectoryName located within the current folder that you are working. It will be also renamed to FILE4.txt.

- It is important to note that the new folder you are moving the file into must have been created already. The mv command was not designed to create new directories on-the-fly.
- If you wish to see if FILE4.txt is located inside of the directory ANewDirectoryName, please see item number 8 for instructions on how to change directories into the folder ANewDirectoryName.

7) Copying a file in the command line:

If you want to copy a file from one directory to another directory, use the “cp” command:

- We are already located in the /home/jsmith directory.
- There is another directory beneath that jsmith directory called “Docs”.
- The file FILE2.txt is already in the jsmith directory. We want to copy FILE2.txt from jsmith into Docs.
- The structure of “cp” is as follows (please replace <space> with an actual space):
 - cp <space> OriginalFileName <space> DestinationFilePathAndName
- There are three options (**only choose one**):
 - cp FILE2.txt /home/jsmith/Docs (if you do not explicitly specify the filename after the new destination path, it will assume that you want the same original file name)
 - cp FILE2.txt /home/jsmith/Docs/FILE2.txt (you can specify the exact same destination filename as the original file)
 - cp FILE2.txt /home/jsmith/Docs/FILE5.txt (you can specify a different destination filename as the original file)

8) Navigation

If you want to change directories (browse folders) within the Linux command line, use the “cd” command.

- This command can be used to navigate in all directions: to directories above/outside the current directory, or to directories below/within the current directory.
- From any directory that you may be currently located, if you want to return to the login directory where you started initially, simply type: **cd ~**
- If you are within the directory /home/jsmith and you want to navigate to the folder /home/jsmith/Docs
 - Type: **cd Docs** (you don’t need to type the entire folder path because you are only one folder above the Docs folder)
- From any directory that you are located, if you want to just go up one directory from your current path (we are currently located at /home/jsmith/Docs, so we would be returning to /home/jsmith)
 - Type: **cd ..**

- If you wanted to navigate to two folders within jsmith (and you are located currently at /home/jsmith) to the folder /home/jsmith/Docs/Pictures
 - Type: **cd Docs/Pictures** (because you are currently in the jsmith folder already)
 - OR (CHOOSE ONLY ONE OF THESE TWO COMMANDS)
 - Type: **cd /home/jsmith/Docs/Pictures**
 - This is an example of an absolute path; this tells Linux to start at the top of the directory tree (/) and change to that directory listed. If the first character is a forward slash then it is an absolute path, otherwise it is a relative path (relative to the current directory you are located in).

9) Editing of a configuration file or other type of textual file in the command line

There may be a configuration or text file found on a server that you need to make changes to.

- There is a built in text editor in Linux called “vi”.
 - If you wish to edit a textual file in “vi”, then issue the following command:
 - **vi TextFileName.ext** (assuming that the file TextFileName.ext exists)
 - You should be located in the directory containing the file that you wish to edit. Begin by typing “vi” followed by the full name of the file including any extension.
 - The “vi” editor will open up on the command line screen and then you can begin editing the file.
 - To insert text, type the letter “i”.
 - You can use the arrow keys to move around the text editor.
 - When you are finished editing and want to save your changes, hit the “ESC” key and type the following “:wq” and hit “ENTER” (<colon><w><q>), whereby w stands for write and q stands for quit.
 - If you do not want to save any changes while quitting “vi”, hit the “ESC” key and type the following sequence “:q!” (<colon><q><exclamation point>), whereby q stands for quit.