Linux Commands

1. pwd : print name of current working directory.
   1. Synopsis: pwd [option]
   2. -L, --logical : use PWD from environment, even if contains symlinks.
   3. -P , --physical: avoid all symlinks.
   4. Example
      1. INPUT: pwd
      2. OUTPUT: /home/mehakjotsingh
   5. Windows : cd
2. cd : change the working directory.
   1. Synopsis: cd [-L|-P] [Directory]
   2. cd .. : return to parent directory
   3. -L : force symbolic links to be followed\
   4. -P: use the physical directory structure without using the symlinks
   5. Example:
      1. INPUT: $ cd Desktop
      2. OUTPUT: /Desktop$
   6. cd
3. cp : copy SOURCE to DEST or multiple sources to DIRECTORY
   1. Synopsis:
      1. cp [OPTION]... SOURCE DEST
      2. CP [OPTION]... SOURCE.. DIRECTORY
   2. Options:
      1. --attributes-only: don’t copy the file data just the attributes
      2. -f : if an existing destination file cannot be opened, remove it and try again
      3. -s :make symbolic links instead of copying
   3. Example:
      1. INPUT: cp hello.txt bie.txt
      2. OUTPUT: (no output, a new file name is created named bie.txt, with same content as hello.txt, there was no bie.txt earlier)
   4. Windows: copy
4. ls: list directory contents
   1. Synopsis: ls [OPTION]... [FILE]...
   2. Options:
      1. -a : Do not ignore entries starting with .
      2. -l : use long listing format
      3. -i: print the index number of each file
      4. -m: fill width with a comma separated list of entries
   3. Example:
      1. INPUT: ls -m
      2. OUTPUT: bie, hello
   4. Windows: dir
5. mkdir: create DIRECTORY(ies) if they do not exist
   1. Synopsis: mkdir [OPTION]... DIRECTORY…
   2. Options:
      1. -p: make parent directories as needed\
      2. -m : =MODE (rwx), set file mode
   3. Example:
      1. INPUT: mkdir -p -v a/b
      2. OUTPUT: mkdir: created directory 'a'

mkdir: created directory 'a/b

* 1. Windows: mkdir

1. rm: remove files or directories (directories are not removed by default)
   1. Synopsis: rm [OPTION]... [FILE]...
   2. Options:
      1. -d : remove empty directories
      2. -f, --force: ignore non existence files and arguments, do not prompt
      3. -i : prompt before every removal
      4. -r,-R : to remove each listed directory with all of its content (recursively)
   3. Examples:
      1. INPUT: rm -v -r a
      2. OUTPUT: removed directory ‘a/b’

removed directory ‘a’

* 1. Windows: del

1. chmod: change file mode bits
   1. Synopsis:
      1. chmod [OPTION]... MODE[,MODE]... FILE...
      2. chmod [OPTION]... OCTAL-MODE... FILE…
   2. Options:
      1. -f,-silent : suppress most error messages
      2. -R : change files and directories recursively
      3. -c, --changes: reports when a change is made
   3. Examples:
      1. INPUT: chmod -c u=r hello.txt
      2. OUTPUT:mode of 'hello.txt' changed from 0644 (rw-r--r--) to 0444 (r--r--r--)
   4. Windows: attrib
2. gzip: compress or expand files
   1. Synopsis: gzip [OPTIONS]... [name]..
   2. Options:
      1. -d, --decompress, --uncompress: Decompress
      2. -k : keep input files during decompression or compression
      3. -l, --list: for each compressed file display the compressed size, uncompressed size, ratio, uncompressed name.
   3. Example:
      1. INPUT: gzip -v hello.txt
      2. OUTPUT: hello: 0.0% -- replaced with hello.gz
   4. Windows: ~no default command
3. find : search for files in a directory hierarchy
   1. Synopsis: find [OPTIONS]... [starting-point]... [expression]
   2. Options:
      1. -P: never follow symbolic links(default)
      2. -L: always follow symbolic links
      3. -H: Do not follow symbolic links, except while processing the command line arguments.
      4. -D : debugoptions, Print diagnostic information
   3. Example:
      1. INPUT: find abc
      2. OUTPUT: abc

abc/a

* 1. Windows: find

1. less: less command does not read the entire input, like to read contents of text file one page(one screen) per time.
   1. Synopsis: less filename
   2. Example:
      1. Input: less hello.txt
      2. Output: (output screen printing contents of hello.txt)
   3. Windows: more
2. tail: output the last part of files, print the last 10 lines of each file to standard output.
   1. Synopsis: tail [OPTION]... [FILE]...
   2. Options:
      1. -c,--bytes=NUM: output the last NUM bytes;
      2. -n,--lines=NUM: output the last NUM lines, instead of last 10;
      3. -q: never output headers giving file names
   3. Example:
      1. Input: tail hello.txt
      2. Output: (last 10 lines of hello.txt)
   4. Windows: tail
3. top: display linux process, system summary operations
   1. Example:
      1. Input: top t
      2. Output: top- 19:06:52 up 6 days, 2:03, 1 user, load average: 0.40, 0.78, 0.74

Tasks: 344 total, 1 running, 273 sleeping, 0 stopped, 0 zombie

%Cpu(s): 6.2 us, 1.5 sy, 0.0 ni, 91.6 id, 0.4 wa, 0.0 hi, 0.2 si, 0.0 st

KiB Mem : 8062752 total, 230488 free, 6137412 used, 1694852 buff/cache

KiB Swap: 2097148 total, 1993140 free, 104008 used. 657032 avail Mem

* 1. Windows: mem

1. wc: print newline, word, and byte counts for each file
   1. Synopsis: wc [OPTION]... [FILE]...
   2. Option:
      1. -c: print the byte counts
      2. -m: print the character counts
      3. -l: print the newline counts
   3. Example:
      1. Input: wc hello.txt
      2. Output: 12 11 25 hello.txt
   4. Windows: fc
2. diff: compare the files line by line
   1. Synopsis: diff [OPtiON]... FILES
   2. Option:
      1. -q: report only when files differ
      2. -y: output in 2 columns
      3. -s:report when 2 files are same
   3. Example:
      1. Input: diff hello.txt hello2.txt
      2. Output: Files hello.txt and hello2.txt are identical
   4. Windows: fc
3. kill: send a signal to a process, default is TERM.
   1. Synopsis: kill [options] <pid> [...]
   2. Options:
      1. -l : to list the available signals
      2. -L: list signal names in a nice table
   3. Example:
      1. Input: kill -9 -1
      2. Output: (kill all processes you can kill)
   4. Windows: task kill
4. Ifconfig: configure network interface
   1. Synopsis: ifconfig [-v] [-a][-s] [interface]
   2. Option:
      1. -a: display all interfaces which are currently available, even if down
      2. -s: display a a short list
      3. -down: causes the driver for this interface to be shut down.
   3. Example:
      1. Input: ifconfig
      2. Output: enp1s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500

ether 50:9a:4c:d1:43:e4 txqueuelen 1000 (Ethernet)

RX packets 0 bytes 0 (0.0 B)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 0 bytes 0 (0.0 B)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

* 1. Windows: ipconfig

1. shutdown: halt, power-off or reboot the machine
   1. Synopsis: shutdown [OPTIONS…] [TIME] [WALL...]
   2. Options:
      1. -P: poweroff
      2. -r: reboot
      3. -k : do not halt or power off or reboot , just write wall messages
   3. Example: shutdown -P
   4. Windows: shutdown
2. which: locate a command
   1. Synopsis: which [-a] filename …
   2. Options:
      1. -a: print all matching pathnames of each argument
   3. Example:
      1. Input: which python
      2. Output: /usr/bin/python
   4. Windows: where
3. cat: concatenate files and print on the standard output
   1. Synopsis: cat [OPTION]... [FILE]...
   2. Options:
      1. -b: number non empty output lines
      2. -E: display $ at end of each line
      3. -s
   3. Example:
      1. Input: cat hello.txt hello2.txt
      2. Output: (concatenated text of both files)
   4. Windows: type
4. mv: move files
   1. Synopsis: mv [OPTIONS]... [-T] SOURCE DEST
   2. Options:
      1. -b : make a backup of each existing destination file
      2. -n: do not overwrite existing file
      3. -f: do not prompt before over writting
   3. Example:
      1. Input: mv -v hello.txt a/hello.txt
      2. Output: renamed ‘hello.txt’ -> ‘a/hello.txt’
   4. Windows: move
5. grep: searches for PATTERN in each FILE.
   1. Synopsis: grep [OPTIONS] PATTERN [FILE...]
   2. Options:
      1. -E,--extended-regexp: interpret PATTERN as an extended regular expression
      2. -F, --fixed-strings:Interpret PATTERN as a list of fixed strings
      3. -G: Interpret PATTERN as a basic regular expression
      4. -c: Display count of matching lines for each input file.
   3. Example:
      1. Input: grep -c 123 hello2.txt
      2. Output: 0
   4. Windows: find
6. cut: remove sections from each line of files, print selected part if lines from each FILE to standard output.
   1. Synopsis: cut OPTION… [FILE]...
   2. Options:
      1. -c=LIST : select only these characters
      2. -b=LIST : select only these bytese
      3. -d=DELIM: use DELIM instead of TAB for field delimiter
   3. Example:
      1. Input: cut -c 1 hello2.txt
      2. Output: (prints the first column of characters)
   4. Windows:cut
7. history: used to view previous executed command
   1. Synopsis: history
   2. Options:
      1. [n]: To show the limited number of commands that executed previously
      2. history | grep PATTERN: to find PATTERN in history
   3. Example:
      1. Input: history | grep g++
      2. Output: 1031 g++ sumOfmedians.cpp

1032 g++ bitonicSearch.cpp

1034 g++ test.cpp4

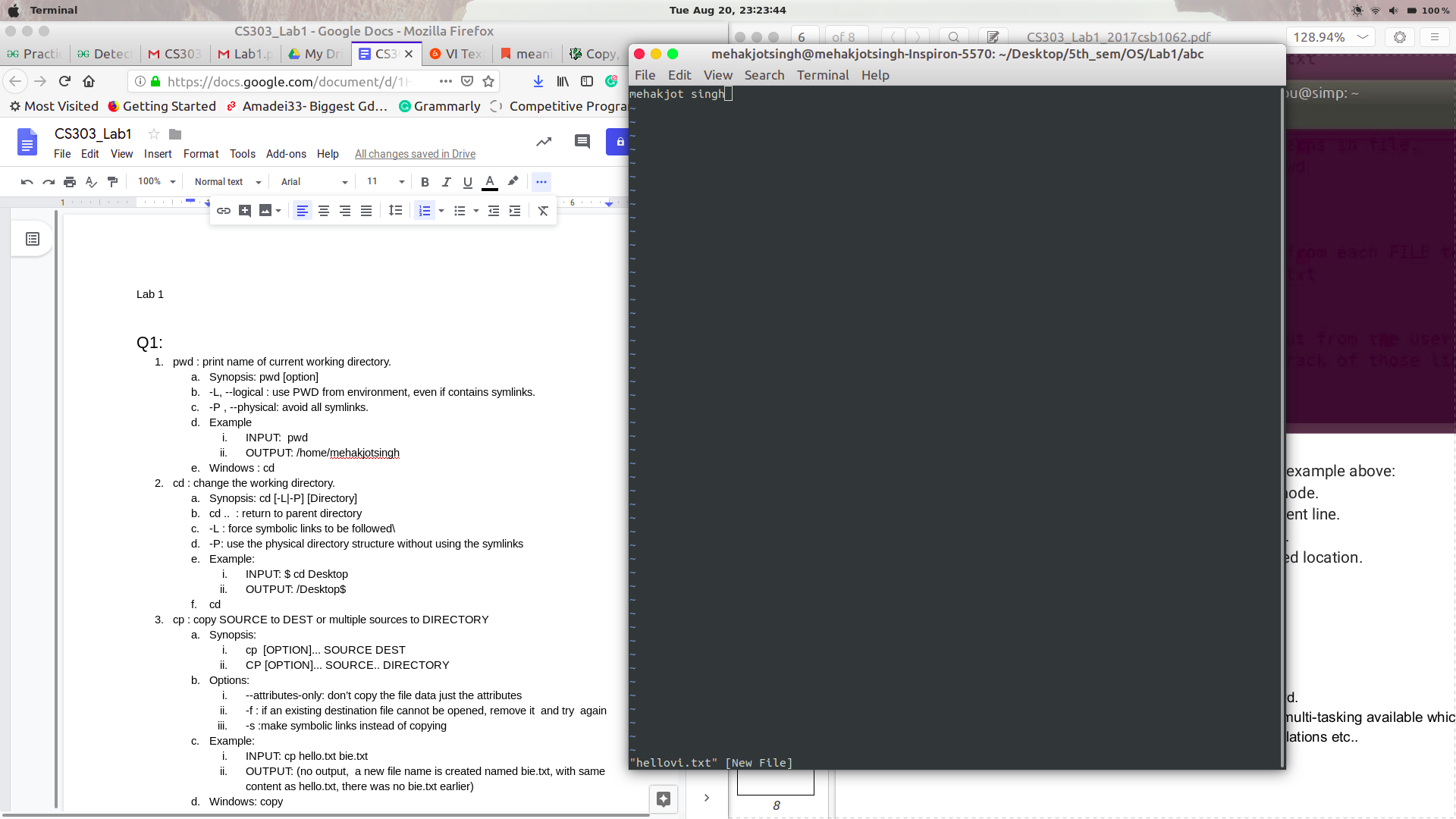
* 1. Windows: F7

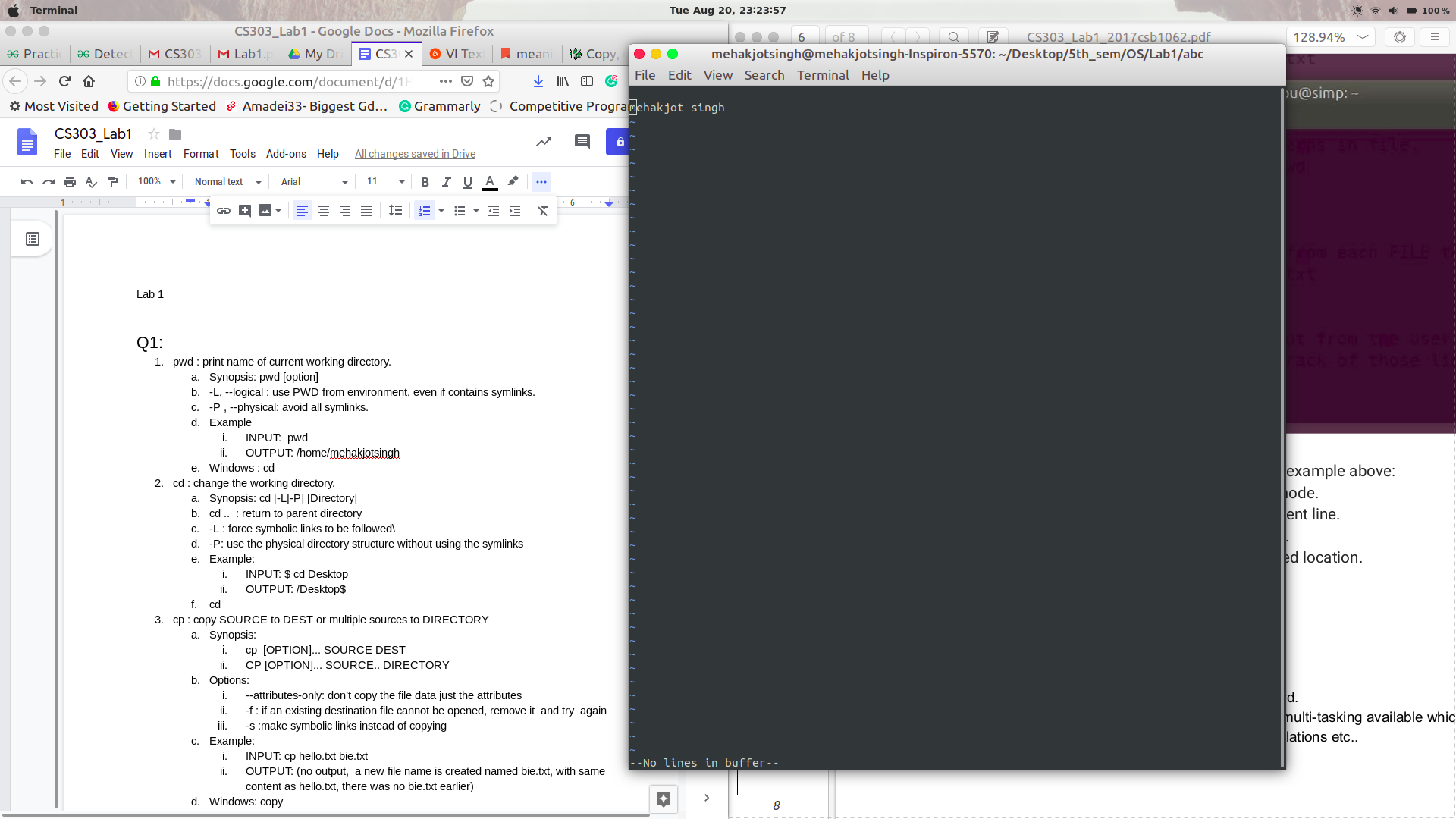
1. echo: display a line of text
   1. Synopsis: echo [SHORT-OPTION].... [STRING]....
   2. Options:
      1. -n: do not output trailing newline
      2. -e: enable interpretation of backslash escapes
   3. Example:
      1. Input: echo hi
      2. Output: hi
   4. Windows: echo

*Source:$ man [command] (linux manual)*

***Vi Editor***

1. Starting Vi Editor: vi filename
2. Command Mode : Understands only commands, allows to move the cursor, copy, cut paste text.
3. Insert Mode : for inserting text, we can enter this mode by pressing ‘i’ in command mode and exit by pressing ‘Esc’. Once in insert mode any key is understood as input for the file.
4. Editing in vi:
   1. i :Insert at cursor **(goes into insert mode)**
   2. a : Write after cursor **(goes into insert mode)**
   3. A:: Write at the end of line **(goes into insert mode)**
   4. ESC :Terminate insert mode
   5. u : Undo last change
   6. U : Undo all changes to the entire line
   7. o: Open a new line **(goes into insert mode)**
   8. dd: Delete line
   9. ndd : Delete n lines
5. Moving cursor
   1. k - Move cursor up
   2. j - Move cursor down
   3. h -Move cursor left
   4. l - Move cursor right
6. Copying ,Cutting, Pasting
   1. y: copies the current character or the selected text
   2. d: copies the current character or the selected text
   3. p: paste the copied text after cursor





Steps Taken

1. Press i to enter insert mode
2. Enter text : “mehakjo singH”
3. Press Escape to exit insert mode
4. Press ctrl V to select line
5. Press d to selected text
6. Press p to paste in next line