1. **INSTALLING GIT FOR WINDOWS:**

Install Git for Windows or Mac depending on the machine.

When installing in windows, use the first option to install

1. BASIC SHELL COMMANDS

Git commands are basically case sensitive

* 1. pwd – prints the current working directory
  2. ls – lists all the directories in the working directory
  3. ls -F = F is for flag which means it will highlight directories with a forward slash
  4. ls –help – script command followed by –help provides the list of commands that are associated with the script command (ls in this case)
  5. ls -l – l is to list all the properties of the files available in the folder, with the permissions and size of the file, last modified information etc.
  6. ls -lh – the addition of h prints the information in human readable format, like printing the size in human readable units
  7. ls -lha – the addition of a lists all the system files as well which were not visible in previous commands
  8. cd “C:/Users/SSOMA/Documents/Projects” – cd command is used to navigate to any given address
  9. cd .. – this is to move one folder up
  10. mkdir Test – mkdir is to create directories in the current working directory. Directory names should not have any spaces. If not, then the directory name should be enclosed within double quotes. If the directory names with spaces are given without spaces, then two directories are created (ex: mkdir test 1 will create a directory test and 1)
  11. touch test\_document.txt – touch is used to create a file
  12. vim test\_document.txt – This is to add content to the text file. VIM is one of the editors to edit file content. (Other editors like nano are also available). For vim, once the editor window is open, press INSERT and start typing the content. Once the content is added, press “ESC” and the :wq to save and quit the editor. (w – write, q – quit). For exiting without saving, just type “:q!” (q is for quitting, ! is to override the default behavior of write. If ! is not provided, it will throw an error)
  13. less test\_document.txt – If the entire document is not to be displayed, and only few of the lines are to be displayed, then less can be used
  14. mv “File Name” “Destination” – this is used to move files around the file structure
  15. rm “file\_name” – Remove the file available in the current folder. Files once removed cannot be recovered
  16. rmdir “dir\_name” – remove the directory name
  17. cp /molecules/methane.pdb . – cp is to copy file in the destination folder, and then . represents to copy the file in the current working directory
  18. mv methane.pdb methane\_molecules.pdb – there is not proper command for renaming a file. So, mv (move command) is used to rename a file
  19. ls e\* - This is a wildcard search. Lists out any files that starts with e. (In windows this does not bring directory names, but in mac it also returns dir names as well)
  20. history – list out all the history of commands
  21. history >> history.txt – Creates a file history and appends all the history in to the file
  22. wc propane.pdb – gives a summary on the number of lines, number of words, and number of letters
  23. wc \*.pdb – gives the word count statistics of all the files with an extension .pdb
  24. wc -l \*.pdb – This provides only the number of line stats
  25. wc -l \*.pdb > lengths.txt – This is to pass the results in to a text file
  26. sort -n lengths.txt – Sort the data using numbers
  27. | - The is a piping command used for passing the results of one command to another, so that we can perform multiple commands at once
  28. Wc -l \*.pdb | sort -n | head -n 3 = This command will generate the summary statistics of all files in .pdb extension, sort based on the numbers, and then return only the top 3 lines