|  |  |
| --- | --- |
| **unix Command Line Interface (CLI)** | |
| man [command] | Open Manual of the Command |
| man ls | What does list command do, and what options are there etc. |
|  |  |
| Part 1: Navigation |  |
| **ls** | List of all files and directories in working directory |
| **pwd** | Print Working Directory |
| **cd** | Change Directory |
| cd / | Goto top of file system hierarchy |
| cd /bin/folder | Goto folder |
| cd ~ | Goto to default user directory |
| cd .. | Go up |
| cd ../../ | Go up two steps |
|  |  |
| Part 2: File Manipulation |  |
| mv | Move / Rename |
| mv [filename] [directory] | Move file to directory |
| mv [filename] [new name] | Rename file |
| rm | Remove |
| mkdir | Create new directory (folder) |
| rmdir | Delete directory if it is empty |
| rmdir -r [directory name] | Loop through contents of directory, if they are empty directories, delete them, then delete this directory as well |
| rmdir -ir [directory name] | Interactively do as above, but ask before deleting each file or folder |
| cp | Copy |
| unzip | Unzip |
| touch | Create new file |
| touch file.abc |  |
| touch folderX/file.abc | Create file.abc in folderX |
| cat | Concatenate (View File Contents) |
| less | See and look up and down file content |
| [software] [filename] | Open file with software |
| vim file.txt | To open a text file |
| matlab file.m | To open file.m with MATLAB |
|  |  |
| Part 3: Finding Information | |
| 2>/dev/null | Put this after a command (specially ‘find’) to send all error messages to null (oblivion) |
| find | Search in current directory tree |
| find / -name [file name] | Exact search |
| find / -name [\*file name\*] | Includes this word |
| grep | Search for a word within a file |
| Grep name /use/bin/filename.txt | Search ‘name’ in file filename.txt |
| !ls | Show history of all commands typed yet |
| Use *regular expressions* (regexp) to refine search | |

|  |  |
| --- | --- |
| Part 4: Change File Permissions | |
| **User Categories:** | **Permissions:** |
| u (you)  g (your group)  o (others) | r (read)  w (write)  x (execute) |
|  |  |
| chmode | Change mode |
| chmode u = rwx [file or directory name] | You can read, write, and execute this file |
| chmode g = rw [file or directory name] | Your user-group members can read and write this file |
| chmode o = r [file or directory name] | Others can only read this file |
| chmode g -w [file or directory name] | - will remove that permission |
| chmode g +e [file or directory name] | + will add that permission |
|  |  |
| Part 5: Alias(es) | |
| Give nick names to long commands | |
| alias [full commands] alias [short alias] | This alias will work till the system is turned off / restart.  \*Save it in .bash\_aliases file in user directory, then restart the system to make it a permanent alias. |
| alias 35fsokfmwl4krm4fm4 alias PingKamra | Avoid typing long commands or Make easy to understand/recall commands for non-technical people |
|  |  |
| Part 6: Shell Scripting | |
| To execute multiple lines of commands in a procedural and/or conditional manner, put them in a .sh or .bash file and set it to run when desired. Scripts can be written to schedule and automate repetitive tasks like:   * backing up data * update software * install and setup everything you need after installing a fresh OS * synchronizing project files with main server * instantly turn on back-up sensor if main sensor goes down * turn off system after a process finishes or at a specified time * push and pull an exam to and from remote at specific timings e.g. 0900-1200 * show greetings and remote systems’ progress reports on start-up * send message to main if someone tries to ssh into a secure server * email me when my father’s phone connects to home’s wifi router | |
|  |  |
|  |  |