1. shell: shell is a program which takes input and gives output

3. Terminal: Terminal is a tool to execute the shell commands, is also called as CLI (Command line interface).

4. / (Forward slash) root directory

5. ~ (Tild) home directory

6. pwd - present working directory

7. cd - change directory

8. cd .. - one step back

9. cd ../.. two steps back

10. ls - list

11. ls -l list long

12. ls -a list of files+hidden files

13. ls -la hidden files+ long list

14. ls -lS list of files in sorting format (Higher volumes to lower)

15. ls -d \*/ list all the directories only

16. I/O redirection:

17. echo "hello world" > test.txt - store the value in text.txt (> it will override the file)

18. echo "hello world" >> test.txt - append the file

19. permissions

20. d(directory) rwx (owner/user) rwx (group) rwx (others)

21. -(file) rwxrwxrwx

22. r-read=4

23. w-write=2

24. x-execute=1

25. cat fileame.txt - To Read content of file

26. cat filename1.txt filename2.txt - To read content of two files

27. cat -b filename.txt - View content with line number and it will ignore empty lines.

28. cat -n filename.txt - view content with line number and it will not ignore empty lines.

29. mkdir sabair - create a directory

30. mkdir sabair/hello - create a directory hello in sabair

31. mkdir -p sabair/hello - create a directory sabair and hello under sabair

32. rm file\_name - remove/delete a file

33. rm -rf - remove/delete a directory

34. cp filename /destination path - copy a file to destination folder

35. mv file1 file2 - rename a file from file1 to file2

36. mv file1 /destination - move file from one location to another

37. wc file\_name - wordcount of file (display number of lines in files)

38. head -3 file\_name - top 3 lies of file

39. tail -2 file\_name - last two lies of file

40. tail -F file\_name - continuously monitor filename

41. less file\_name - show less part of file (as much as terminal allows)

42. touch file\_name - create a zero-byte file

43. sudo -(superuserdo)

44. It allows you to get some extra privileges as the admin does.

45. sudo su -switch to root user

46. Sudo -s - switch to root user (it will ask for Sudo password)

47. Top - Top command provides dynamic real-time view of your system.

48. Kill -kill is used to kill a processes

49. kill -9 PID - kill that particular process and -9 flag is used to kill it forcefully.

50. kill -l - it will list all flags which can be used to kll processes.

51. chmod - change permissions for a file.

52. drwx (owner/user) rwx (group)rwx(other)

53. r= read - 4

54. w= write - 2

55. x= execute -1

56. chown - change owner of file or directory

57. which bash - this is used to find the location of any command or file

58. what is ls - this is used to return the short description of command or any application.

59. useradd sabair - To create a user sabair

60. useradd sabair -m -s /bin/bash (-m is used to create a home directory for a user and -s is shell)

61. passwd sabair - To assign a password to user sabair

62. userdel sabair - Delete user sabair. (This will delete the user sabair and not the home directory)

63. userdel -r sabair - Delete the user sabair and its home directory.

64. groups - it is used to check the list of groups associated with particular user.

65. groupadd java - create a group named java

66. cat /etc/group - To view list of groups

67. groupdel java - To delete a group

68. gpasswd -a user\_name group name (-a to add user to group, -d to remove user from group)

69. df - Is used to display the amount of free space in your filesystem.

70. df -h - It will make more human readable.

71. du - Is used to display the amount of memory taken by files.

72. du -h - It will make more human readable.

73. free - find the free memory of the system.

74. free -m - Human readable to megabytes.

75. wc filename.txt - Is used to see the word count of any file (lines, words, characters)

76. bash script:

77. ===========

78. .sh will be the extension for scripts.

79. It’s not mandatory that we should keep .sh as extension, without .sh as well we can execute our script and it will work fine.

80. Sample script

81. #!/bin/bash (#! = shebang) (/bin/bash location of our bash)

82. ls

83. echo "hello world"

84. #!/bin/bash

85. STRING="hello world"

86. echo $STRING