43+ Linux terminal shortcuts every power Linux user should know:

The Linux terminal interface may be difficult for a new Linux user to grasp, especially because it relies heavily on the arrow keys to move around.

Furthermore, it can be tiresome to constantly retype the commands, each with a slight variation.

Bash, on the other hand, comes with a plethora of simple keyboard shortcuts and history functions to ensure efficient terminal use.

Having these terminal shortcut keyboard key combinations memorized will save you a lot of time.

To help you make better use of your time, I've compiled a list of the most frequently used Bash keyboard shortcuts ↓

It's worth noting that some of these shortcuts may be different depending on the Shell you're using. Because Bash is the most popular shell, this list is centered on it.

I also ran the majority of them on zsh to make sure they were all compatible.

Cursor Movement

1. Ctrl + A - Go to the beginning of the line you are currently typing on.

2. Ctrl + E - Go to the end of the line you are currently typing on.

3. Ctrl + XX - Move between the beginning of the line and the current position of the cursor.

4. Alt + F - Move cursor forward one word on the current line.

5. Alt + B - Move cursor backward one word on the current line.

6. Ctrl + F - Move cursor forward one character on the current line.

7. Ctrl + B - Move cursor backward one character on the current line.

Text manipulation

8. Ctrl + U - Cut the line from the current position to the beginning of the line, adding it to the clipboard. If you are at the end of the line, it cuts the entire line.

9. Ctrl + K - Cut the line from the current position to the end of the line, adding it to the clipboard. If you are at the beginning of the line, cut the entire line.

10. Ctrl + W - Delete the word before the cursor, adding it to the clipboard.

11. Ctrl + Y - Paste the last thing from the clipboard that you cut recently (undo the last delete at the current cursor position).

Access your History

12. Ctrl + R - Lets you search through previously used commands.

13. Ctrl + G - Leave history searching mode without running a command.

14. Ctrl + J - Lets you copy current matched command to command line without running it, allowing you to make modifications before running the command.

15. Alt + R - Revert any changes to a command you've pulled from your history, if you've edited it.

16. Ctrl + P - Shows last executed command, i.e. walk back through the command history (Similar to up arrow).

17. Ctrl + N - Shows next executed command, i.e. walk forward through the command history (Similar to down arrow).

18. Ctrl + _ - Undo previous command.

19. Ctrl + O - Executes and re-enters the command found via Ctrl + R and Ctrl + S.

20. !! - Expand the previous command.

21. !-n - Expand nth most recent command.

22. !n- Expand nth command in history.

23. !<command>- Expand most recent invocation of command <command>.

24. !<text> - Expand the last command that begins with a specific text.

25. !n:p or !<text>:p - Print the expanded command before executing it.

27. ^string1^string2 - Expand the last command, replacing string1 with string2.

28. !# - Expand the entire command line typed so far.

29. !$ - Expand last parameter of most recent command.

30. !* - Expand all parameters of most recent command.

31. !^ - Expand the first parameter of the most recent command.

Terminal/Process control

32. Ctrl + L - Clears the screen, similar to the clear command.

33. Ctrl + S - Stop all output to the screen. This is useful when running commands with lots of long output. But this doesn't stop the running command.

34. Ctrl + Q - Resume output to the screen after stopping it with Ctrl+S.

35. Ctrl + C - End currently running process and return the prompt.

36. Ctrl + D - Log out of the current shell session, similar to the exit or logout command. In some commands, acts as End of File signal to indicate that a file end has been reached.

37. Ctrl + Z - Suspends (pause) currently running foreground process, which return shell prompt. You can then use bg command allowing that process to run in the background. To again bring that process to foreground, use fg command. To view all background processes, use jobs cmd.

38. Tab - Auto-complete files and directory names.

39. Tab - Tab Shows all possibilities, when typed characters doesn't uniquely match to a file or directory name.

40. Ctrl + S - Gets the next most recent command.

Bash Information

41. ~ - TAB TAB List all Linux users.

42. Ctrl + I - Completes the command like TAB.

43. Alt + ? - Display files/folders in the current path for help.

44. Alt + * - Display files/folders in the current path as parameter.

Close Terminal

45. Ctrl + Shift + W - To close terminal tab.

46. Ctrl + Shift + Q - To close entire terminal.

That concludes today's thread; be sure to experiment with these shortcuts and be amazed.

The shortcuts listed above are only the tip of the iceberg. The more you learn about Linux terminal commands and Bash shortcuts, the more straightforward and appealing Linux becomes.

If you found this thread valuable:

1. Toss us a follow for more daily threads on Linux, sysadmin and devops → @linuxopsys

2. Like and RT the first tweet so other Linux folks can find it too.