

CS 250: The Linux Operating System  
Assignment 4  
*Working Interactively with BASH (Part 2)*

## Introduction

During this hands-on project, you will login to a Linux workstation, experiment with a few more commands, and get more comfortable using the Bash shell interactively. The purpose of this lab is experimental: try the commands, try to figure out what they do, and try looking them up in the textbook, the videos provided on Blackboard, or online. This assignment will be much easier if you read Chapter 3 from the textbook first!

## Interactive Use of the Bash Shell

1. Log in to your Linux account and open a terminal window.
2. Use the script utility to capture the screen output and save into a file named `assignment-4.txt`.
3. Change your working directory to `CS-250`.
4. Change your working directory to `Examples`.
5. Open file `test1` for editing using the vi editor. Run the vi command in the background.
6. Open file `test2` for editing using the vi editor. Run the vi command in the background.
7. Open file `test3` for editing using the vi editor. Run the vi command in the foreground.
8. Add a line to file `test3` that reads “=====” and suspend command `vi test3`.
9. List all your jobs.
10. Bring command `vi test2` to the foreground and add some text of your choice to the end of the file.
11. Suspend command `vi test2`.
12. Bring command `vi test3` to the foreground and save the file.
13. Bring command `vi test2` to the foreground and save the file.
14. Stop all jobs.
15. Verify the changes made to files `test2` and `test3`.
16. Use the shell filename expansion mechanism to list all the files that begin with `test`.
17. Use the shell history expansion mechanism to recall the command you issues in Step 9.
18. Issue a command to locate the command `srot`. Note the shell’s reaction since there is no such command (`srot`). We will correct the spelling in the next step.
19. Use the shell history expansion mechanism to recall the last command replacing the string `ro` with the string `or`.
20. Use the shell history expansion mechanism to view the manual pages for command `sort`. Hint: use the last argument (`sort`) of the most recent command (`which sort`).
21. Issue a Linux command to include the special directory symbol (`.`) on the command search path (`$PATH`).
22. Create an alias named `back` that, when issued, takes a user from any directory in the filesystem back to the user’s own home directory.

23. Create an alias named `rm` that, when issued (`rm file`), prompts the user for whether to remove the `file`. If the response is not affirmative, the `file` is skipped.
24. Issue a command to append the file `test2` to the end of the file `test1`.
25. Issue a command to send a line-numbered listing of file `test2` to the printer.
26. Write a Bash function `rm` to move its argument files to the `~/Trash` folder.
27. Test your function `rm` using all the files that begin with `test`.
28. Find the Linux version running on your computer.
29. Stop the script utility.
30. Clean up your script output file. This means remove all commands that result in error messages, all extra commands that are not part of this lab, all typing mistakes, and all control characters (except `^M`). This step could be viewed as torture, but it is supposed to give you practice and motivation to learn `vi` well. You must turn in a clean script output file. Points will be deducted if you turn in the “raw” script output file.

### Submission Instructions

Submit your `assignment-4.txt` file on Blackboard.