

Agenda: Shells

1. Open the terminal. Use appropriate command to determine your login shell.
2. Use `/etc/passwd` file to verify the result of step1.
3. Use the appropriate command to determine your current shell. Is it the same as your login shell?
4. Create a subshell that is not the same shell as your login shell.
5. Check the current shell again. Does it match with your login shell?
6. Create another shell which is same as your login shell.
7. Check the current shell again. Does it match with the previous one?
8. Exit from the subshell and move to your login shell.
9. Check the current shell again. Is it the same as your login shell.
10. Open vi file. Use **who** command and redirect the result to a file called **file1**. Give execute permission to the file and execute it. Use the more command to see the contents of file1
11. Create an alias name file to the execute command of file (`./file`). Now use your alias. Is it working.
12. Open vi file and edit it. Use the date and who command in sequence such that the output of date will display on the screen and the output of who will be redirected to a file called file2. Use the more command to check the contents of the file2. It should be same as file1
13. Use Vi file. Edit the line so that the output of both the commands are redirected to a file called file3. Use the more command to display the result.
14. Use the command-line editor to recall the line you typed before. Edit the line so that the output of the first command (**date**) is redirected to a file called file4. While the result of the **who** command displays on the monitor without changing the relative positions of date and who command. Use the more command to verify the result.
15. Make a duplicate of file3 and call it file3.bak
16. Use the command-line editor to recall the line created before. Edit the line so that the word date will be misspelled as bate. Execute the edited command and append the result to file3. Use more to compare file3 and file3.bak
17. Using command-line editing recall the line created in step15. Edit it so that the errors go to file3 without replacing the file contents. Check the contents of file3 for verification.
18. Using command-line editing, recall the line created in step 16. Edit it so that the output goes to file4 and errors go to file5.
19. Store the word PARENT in a variable called **first**.
20. Print the value of the variable first
21. Create a subshell which is same as your login shell
22. Print the variable first in this subshell. What was printed? If "PARENT" was not printed, explain why?
23. Store the word "SUBSHELL" in a variable called **second**
24. Print the value of the variable **second**
25. Exit from the subshell and move to your login shell
26. Print the value of **first**. What was printed? How do you explain this result?
27. Print the value of **second**. What was printed? How do you explain this result?

- 28.** Check the default value of the shell option noclobber. Create a file with a name which is already existing. What happens to the file?
- 29.** Switch over the noclobber option. Create a file with a name which is already existing. What happens now?