

# OPS105 Lab 02 - Exploring The Command Line

Complete this lab by the beginning of the next class.

## Objectives

- To become proficient in using the BASH shell
- To learn about shell substitution
- To understand soft and hard links
- To learn how to use file permissions
- To learn how to manage user processes

## Required materials

- Access to Matrix server

# Lab Instructions

1. Create a new directory called **lab2** in your home directory.  
This will be the place where you complete most of this lab.
2. Using a single **touch** command, create the following empty files in the **lab2** directory:  
**a, a1, a2, a3, aa1, aa2, abc, a1txt, a1.txt, b.txt**
3. Using the above files as a base for experiments, come up with shell expansion expressions, which would cover files as indicated in the table below. Fill in the shell expressions in the appropriate cells.

Use the **echo** command to test your answers before writing them down.

| Shell Expression | a | a1 | a2 | a3 | aa1 | aa2 | abc | a1txt | a1.txt | b.txt |
|------------------|---|----|----|----|-----|-----|-----|-------|--------|-------|
| *                | ✓ | ✓  | ✓  | ✓  | ✓   | ✓   | ✓   | ✓     | ✓      | ✓     |
|                  | ✓ |    |    |    |     |     |     |       |        |       |
|                  |   | ✓  | ✓  | ✓  |     |     |     |       |        |       |
|                  |   | ✓  | ✓  | ✓  | ✓   | ✓   |     |       |        |       |
|                  |   |    |    |    |     |     |     | ✓     | ✓      |       |
|                  |   |    |    |    | ✓   | ✓   | ✓   |       | ✓      |       |
|                  |   | ✓  |    |    |     |     |     | ✓     |        |       |
|                  |   | ✓  |    |    |     |     |     |       | ✓      |       |
|                  |   | ✓  | ✓  | ✓  | ✓   | ✓   |     | ✓     | ✓      | ✓     |
|                  |   |    |    |    |     |     |     |       | ✓      | ✓     |

4. Compare your above table with a classmate. Discuss any differences and make corrections as needed.
5. Using the **ls -li** command, find out the inode number for the **abc** file.

Create a hard link in the **lab2** directory to the **abc** file called **abc2**. Check the inode number for the newly created file – it should be the same as the **abc** file. Also, notice how the link counter (output from the **ls -l** command) has increased by one.

Create a symbolic link to the **abc** file called **abc3**. Look at the inode number for the symbolic link and the link counter shown for **abc**. Can you explain the results? Write your conclusions down.

6. Use **vi** to add some content to the **abc** file. Save it and exit the editor. Use the hard and symbolic links to edit the file further. Do you understand that the text content exists on the disk in only one place?

Delete the **abc** file. Try accessing your text using the **abc2** hard link. You should be successful. Look at the output of the **ls -l** command – the link counter for **abc3** is now set to 1. Do you know why?

Edit your text using the **abc3** link. Notice the message given by the editor. Save the file and look at the output of the **ls -li** command. Notice that the **abc** file has reappeared, but it has a different inode. Look at the link counter for the **abc2** file. Can you explain what happened?

7. It's time to practice file system permissions. Have a look at the permissions of your **lab2** directory. You can use the **ls -la** or **ls -ld** command to see permissions for a directory.

Although this is system-dependent, most likely your directory has **rwX** permissions for you. Experiment with permissions for the lab2 directory and fill in the blanks in the following questions:

- (a) In order to be able to see the directory listing I need the \_\_\_\_\_ permission(s)
- (b) I need the \_\_\_\_\_ permission to be able to "cd" into the directory
- (c) I need the \_\_\_\_\_ permission to create and delete files in the directory

Compare your answers with a classmate, discuss any differences and make corrections as needed. Restore the lab2 directory permissions to allow unrestricted access before proceeding.

8. Have a look at the permissions of the files in the **lab2** directory. Notice how permissions for a symbolic link differ from regular files (hard links).

Experiment with file permissions and fill in the blanks in the following questions:

- (a) I need the \_\_\_\_\_ permission(s) in order to view file contents
- (b) I need the \_\_\_\_\_ permission(s) in order to copy a file
- (c) I need the \_\_\_\_\_ permission(s) in order to delete a file
- (d) When I changed a permission for a file, permissions shown for another hard link pointing to the same file \_\_\_\_\_

Again, compare your answers with a classmate, and make corrections as needed.

9. For this part you will need two terminals. You can either open two terminal windows or use the Crtl+Alt+F1 keys (Crtl+Alt+F7 is your GUI) to login again to your Matrix account in text mode.

Start the top command in the first terminal. Examine all statistics at the top of the screen and have a look at all processes running. Take a note of the current CPU usage.

Leave the top command running.

Use vi in the second terminal to create a file called **infin**, with the following content:

```
while [ 1 ]  
do  
    echo "hi"  
done
```

The above is a very simple shell script, which when run continuously, will keep the CPU very busy. It is important to type the script exactly as shown, including the spaces in the while line.

Give the script execute permissions and start it using the following command: **./infin**

Check the top output of the top command in the first terminal. You should see a marked increase in the CPU load.

Sometimes it may be necessary to kill a resource-hogging process. Identify the **pid** of the script using the top command (you can also use the **ps x** command) and kill the script from the first terminal.