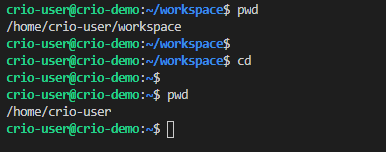
# Pointers to Curious Cats

## Milestone 1

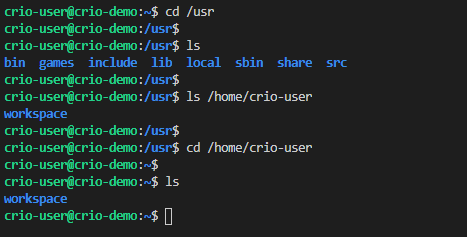
### What happens if we execute the cd command without any directory name? Does the working directory remain the same? (Hint: Use pwd to check where you were taken to)

**cd** command when executed without any arguments (directory name) changes directory to the home directory of the current user



### Can you try the ls command with a directory name as well?

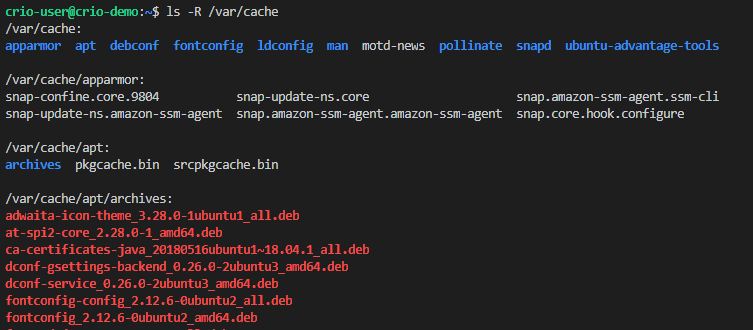
**ls** when executed with an argument (directory name) lists content of that directory instead of the current working directory



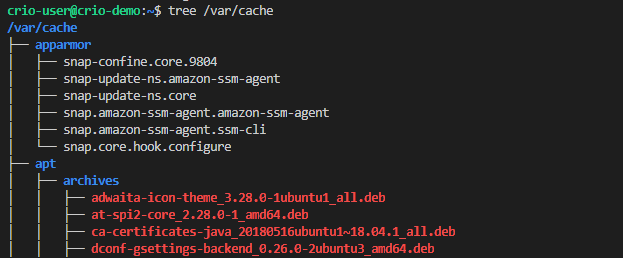
## Milestone 2

### Find a command that prints out all sub-directories and files recursively

* Use **ls -R <directory>**

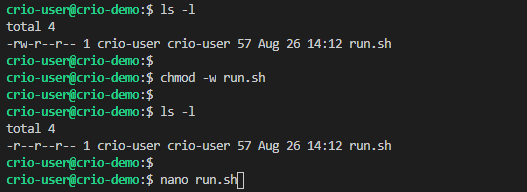


* Use **tree <directory>**

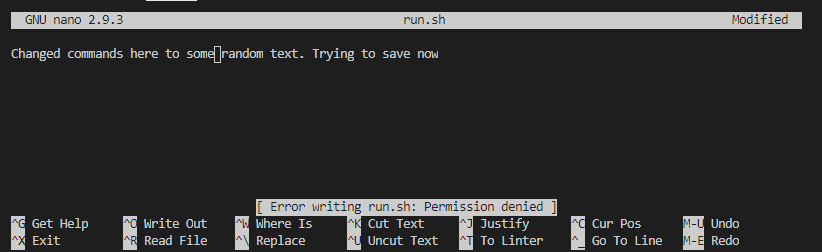


### Try to remove the write permission for the run.sh file. Will you be able to edit the file after this?

Use **chmod -w run.sh** in the directory where the file is to remove write permission. Open the file for editing using a text editor. Eg: **nano run.sh**. Make some changes and hit **Ctrl-X** to try saving & closing the changes. Enter **Y** when asked to save modified buffer (basically changes made) and then hit **Enter** to overwrite the current file.

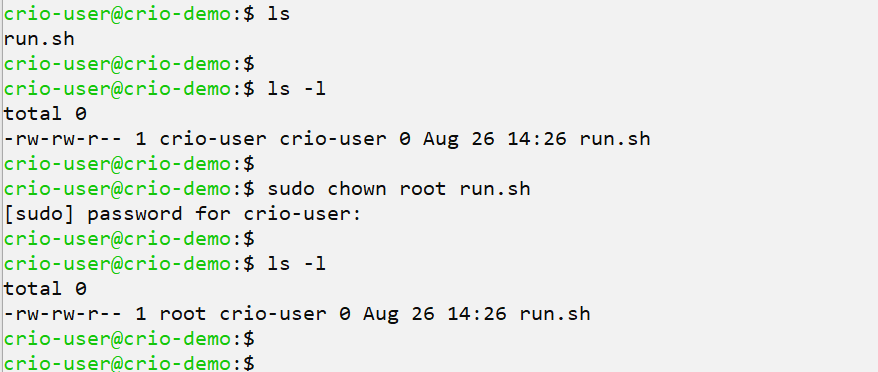


The editor won’t allow to save any changes made



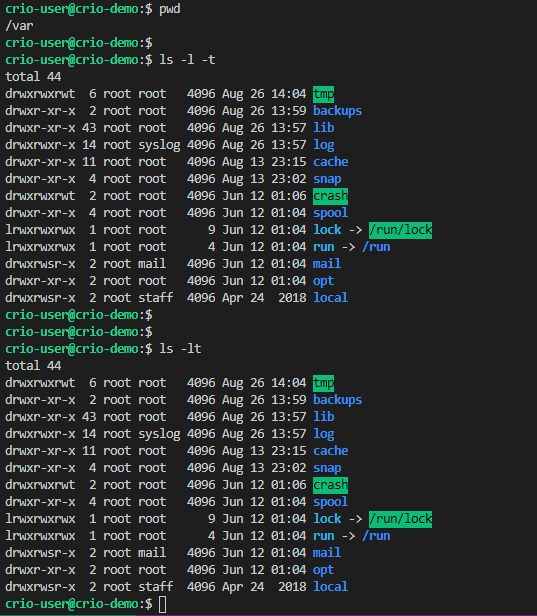
### We looked at how to change the access mode for a file using chmod. Can you find out how we can change the owner of a file?

Use the **chown** command to change file ownership. **chown root run.sh** tells to change ownership of the run.sh file to **root** user. Using **sudo** might ask you to enter a password. You may not be able to try this out on Crio workspace due to **sudo** command usage.



### The -l flag used with ls lists the contents in long form whereas -t flag sorts the files according to their modification time. What if we need to do both?

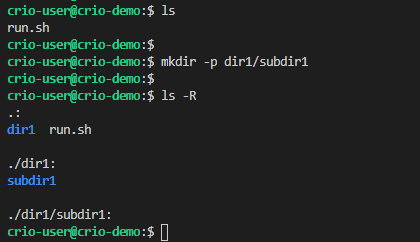
We can use multiple flags along with the commands at a time. The usage could be like below



## Milestone 3

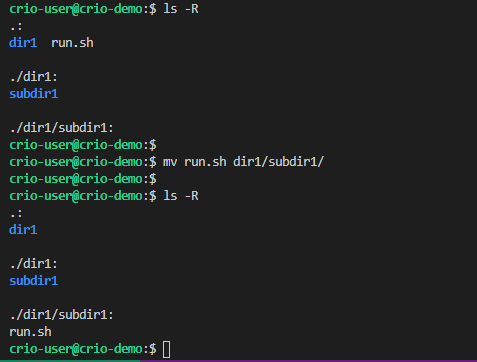
### To create the dir1/subdir1 directory, we can use **mkdir dir1** and then cd to **dir1** and then execute **mkdir dir1/subdir1**. Can you find out if it’s possible to create the **dir1/subdir1** directories using a single **mkdir** call?

We can use the **-p** flag with the **mkdir** call to create nested directories.



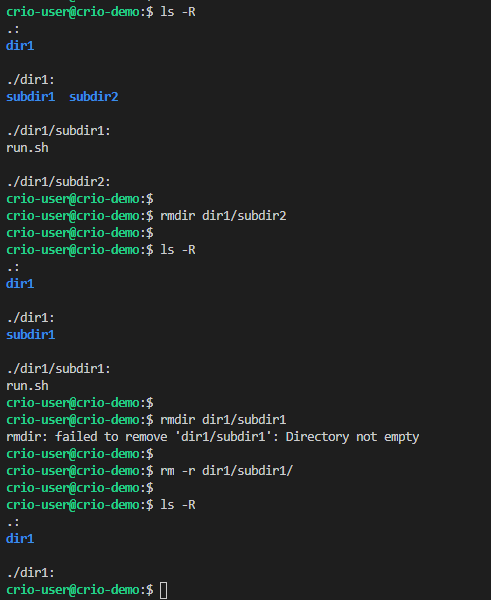
### Does Linux provide some command to just move files from one directory to another rather than having us to do a copy first & then remove the original?

Use the **mv** command to move files from one directory to another



### On second thoughts, we decide to remove the subdir2 directory. How would you delete a directory?

Use the **rmdir** command to delete an empty directory. If the file has contents, **rm -rf** command will have to be used. **-r** recursively deletes files in the directory.



### 

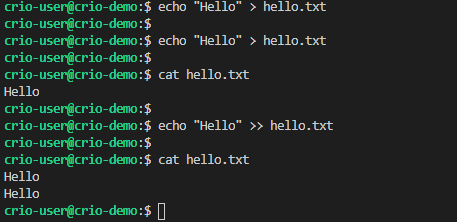
### 

### 

## Milestone 4

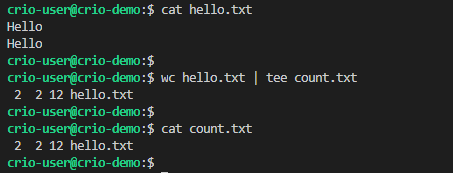
### We came across a couple of ways to redirect the output of a command. > operator writes output of a command to a file. Try running echo "Hello" > hello.txt command two times and cat content of hello.txt. Does it contain one Hello or two? How can we append data rather than overwrite it here when using redirection?

The **>** operator overwrites data in the file and hence only one “Hello” will be saved. Use **>>** to append data.



### The | operator sends output of a command as input to another. What if we needed to print the output of the first command to the terminal as well as redirect it to a file?

The **tee** command is used to redirect output both to the terminal as well. Just like the shape of alphabet T, it redirects to the file as well as to the terminal.



### What if I need to see the first 20 lines of the /var/log/syslog file? Is there some command that lets us fetch some lines from the starting of the file?

Use the **head** command to view some lines of a file from the starting. See [here](https://shapeshed.com/unix-head/)

### What if you want to see contents of a file getting printed to screen as and when the file is being written to?

Use the **tail** command to watch changes made to a file. See [here](https://shapeshed.com/unix-tail/)