

Practical Linux

Practical 3

Core 2

General questions

Let us assume the directory structure presented in Figure 1 with **student** as your **working directory**.

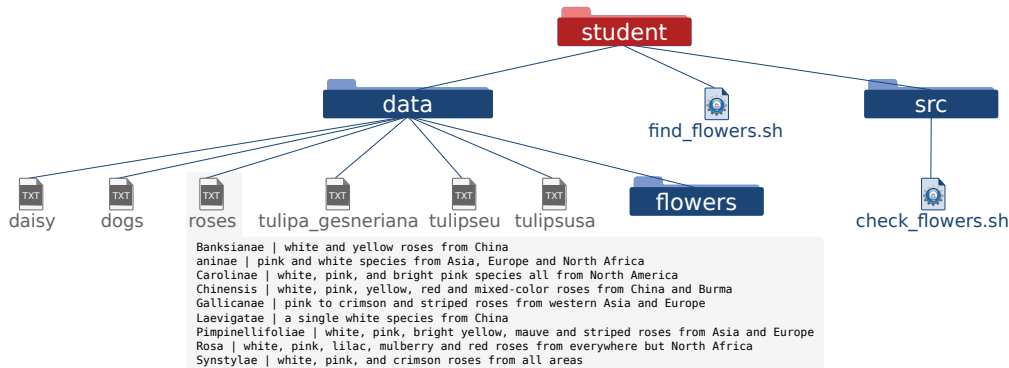


Figure 1: Directory graph for the general questions section.

1. What command could you use to create a directory named **animals** inside the **data** directory?
 - (a) `dir humans`
 - (b) `mkdir humans`
 - (c) `crdir data/humans`
 - (d) `create humans`
 - (e) `mkdir data/animals`
 - (f) this is impossible
2. Assuming that **find_flowers.sh** is an executable script, how would you run it?
 - (a) `find_flowers.sh`
 - (b) `run find_flowers.sh`
 - (c) `/find_flowers.sh`
 - (d) `execute /find_flowers.sh`
 - (e) `./find_flowers.sh`
 - (f) this is impossible
3. What command could you use to move the **find_flowers.sh** file into the **src** directory?
 - (a) `mv /find_flowers.sh /src`
 - (b) `rm find_flowers.sh src`
 - (c) `move find_flowers.sh src`
 - (d) `mv find_flowers.sh src`
 - (e) `cp find_flowers.sh /src`
 - (f) this is impossible
4. Assuming that **check_flowers.sh** is an executable script, how would you run it?
 - (a) `/src/check_flowers.sh`
 - (b) `run check_flowers.sh`
 - (c) `./check_flowers.sh`
 - (d) `src/check_flowers.sh`
 - (e) `execute src/check_flowers.sh`
 - (f) this is impossible
5. What command will display the lines containing **red** from the **roses** file?
 - (a) `show red roses`
 - (b) `locate red roses`
 - (c) `grep red roses`
 - (d) `find red data roses`
 - (e) `grep red data/roses`
 - (f) this is impossible
6. What command will move only the **tulipseu** and **tulipsusa** files into the **flowers** directory?

- | | |
|---|---|
| (a) <code>mv data/flowers tulips</code> | (d) <code>move data/flowers data/tulips*</code> |
| (b) <code>put into data/flowers data/tulips*</code> | (e) <code>mv data/tulips* data/flowers</code> |
| (c) <code>mv data/tulip* data/flowers</code> | (f) <code>this is impossible</code> |

Creating and working with directories and files

Make sure that your **working** directory is your **home** directory. (`pwd`, `cd`)

Run a sequence of commands to make your **home** directory tree look as in Figure 2. This will include:

1. Rename the `project_x` directory to `project_genetics`. (`mv`)
2. Create three **new directories** inside the `project_genetics` directory named: `data`, `src`, `doc`. (`mkdir`)
3. Move the `script.sh` file into the `project_genetics/src` directory. (`mv`)
4. Move all the `.gb` files from `project_genetics` into the `project_genetics/data` directory. (`mv`, `*`)
5. Make a copy of the `project_genetics/README.md` into the `project_genetics/doc` directory. (`cp`)

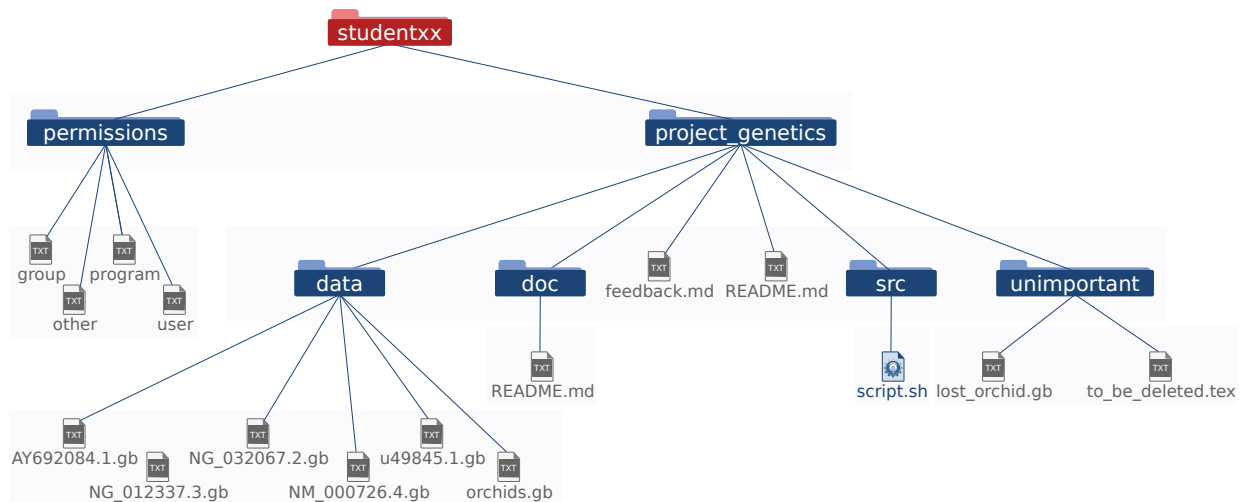


Figure 2: Home directory tree

Remove the `project_genetics/unimportant` directory. (`rm`)

Inspect, edit, and search

1. Print the `project_genetics/README.md` file contents. (`cat`)
2. Open the `project_genetics/README.md` file with `nano` and edit it according to what it is mentioned inside it. (make use of key combinations in `nano`, e.g., `Ctrl-k`, `Ctrl-x`, ...; see cheat sheet)
3. Print again the `project_genetics/README.md` file content to check the changes. (`cat`)
4. Inspect the `project_genetics/data/orchids.gb` file content first with the `cat` command and after that with the `less` command in order to see the differences between the two commands.
5. Search for the appearances of `LOCUS` within the `project_genetics/data/orchids.gb`. (`grep`)

6. Search for the appearances of **ORGANISM** within all the **.gb** files in the **project_genetics/data** directory. (`grep, *`)
7. Open the **manual** page of **grep** and search for the **option** on how to **suppress the prefixing of file names on output**. (`man`; use the `/` key to search for keywords inside the manual page)
8. Make use of the previous **grep** command found **option** in the search for the appearances of **ORGANISM** within the all **.gb** files in **project_genetics/data**.
9. Rerun the previous command and redirect its output to a file named **organisms.txt** inside the **project_genetics/data** directory. (`grep, *, >`)
10. Check the content of the newly **organisms.txt** created file. (`cat` or `less`)
11. Make sure that your working directory is **project_genetics/data**.
Run the **project_genetics/src/script.sh** script. Some error message should appear indicating that the resource was not found.
12. Edit the **project_genetics/src/script.sh** file and update the URL after the `wget` command such that **important** is replaced with **unimportant**. Save the file and close the editor.
13. Make sure that your working directory is **project_genetics/data**.
Run again the **project_genetics/src/script.sh** script. This time the script should have successfully run and a new file named **lost_orchid.gb** should have appeared in your working directory, i.e., **project_genetics/data**.
14. Append the contents of **lost_orchid.gb** to the **orchids.gb** file. (`cat, >>`)
15. Remove the **lost_orchid.gb** file. (`rm`)
16. Print the first three organisms from the **organisms.txt**. (`head`)
17. What is the total number of organisms that appear in the **organisms.txt** file? (`wc`)
18. What is the disk usage of the data folder? (`du`)

Congratulations, you have finished this practical session!