Command Line Practice

The command line may seem difficult at first, but once you learn a few basic commands you will have the ability to run a wide variety of bioinformatics software. It's just like learning another (programming) language! If you are already comfortable navigating file directories using `pwd` and `cd`, viewing directory contents using `ls`, moving files using `mv`, and redirecting input and output using pipes, you can skip this tutorial.

If you are brand new to the Command Line, we suggest you start by completing Tutorials 1–4, 8, and 10–14 on this website: https://linuxjourney.com/lesson/the-shell. For more practice, you can also complete the Terminal Basics Tutorial here: https://sandbox.bio/tutorials?id=terminal-basics.

Important notes for following this tutorial:

- Text with a gray background in `monospace font` represents commands to type in.
- Bold text surrounded by **< >** is something you will have to replace with your own username.
- `[file]` and `[dir]` are used as placeholders for the path to the file or directory on which the command is being executed. `[file/dir]` means that the command can be executed on either a file or directory.

General Command Line Tips

- The HOME directory on a Linux machine is `/home/<`**`username`**`>`, where <**`username`**> is the username you use to log into your computer. Instead of writing this out in every file path, it can be useful to replace `/home/<`**`username`**`>` with ~ (the tilde character).
- The only file paths that should start with a forward slash are absolute file paths.
- By default, any commands that name files or folders and that you run in the command line will refer to files in the current working directory, unless you use the absolute path.
- Avoid using spaces in file and folder names.
- File and folder names in file paths are CASE SENSITIVE.
- Be *exceptionally careful* when removing or deleting files, as this action cannot be undone.

Computer Setup Instructions

This tutorial makes use of a number of example files and folders. Before starting the tutorial, follow the steps below to set up the necessary files on your computer:

1. Check if your computer has a folder called *practice* in the HOME directory. Inside this folder there should be three sub-folders: *data*, *fastas*, and *scripts*. If these sub-folders are present on your computer, you can skip steps 2-6 below and proceed directly to the next section (*Command Line Practice*).

2. If you do not yet have the *practice* directory on your computer, download it by downloading *linux-learning.zip* here: https://github.com/CholGen/linux-learning/releases/download/v2/linux-learning.zip.

3. Unzip this file on your computer.

4. Within the *linux-learning* folder you just downloaded, move the *practice* subdirectory into the HOME directory on your computer. You can do this by dragging the *practice* folder from your Downloads folder (or wherever the folder is currently located) into your HOME directory.

5. Confirm that the *practice* folder you downloaded contains three sub-folders: *data*, *fastas*, and *scripts*.

6. Open the command line on your computer. On Mac and Linux computers, this will be the application called *Terminal*. On Windows computers, you may need to open a pre-installed software such as PuTTY or cygwin. If you do not yet have a functional Linux command line on your computer, you will first have to set up the command line following the instructions in Command Line Setup.

Command Line Practice

The instructions listed below are demonstrated in this publicly available lecture video: https://youtu.be/DSathAxq_aA.

---

1. Use the `pwd` command to figure out your current location on the computer.

Type the following into your terminal window and then press **Enter**:

```
pwd
```

You should see something like (may differ slightly on Windows computers):

```
/home/<username>
```

2. Use the `ls` command to list the contents of your current location (the directory in which you currently are).

Type the following into your terminal window and then press **Enter**:

```
ls
```

You should see a list of files and folders. These are all the items that are in your current directory (the HOME directory). This should match the files and folders you see when you open the file explorer application and navigate to the HOME directory.

3. Use the `cd` command to navigate to the *practice* directory (should be stored in your HOME directory).

Type the following into your terminal window and then press **Enter**:

```
cd practice
```

Remember that punctuation matters! And don't hesitate to use the **tab** key to help you complete file paths.

4. Use the `pwd` command to view the absolute file path for your current location within the computer.

Type the following into your terminal window and then press **Enter**:

```
pwd
```

You should see something like (may differ slightly on Windows computers):

```
/home/<username>/practice
```

This indicates that you successfully moved into the *practice* directory.

5. Use the `ls` command to list the contents of your current location (the directory in which you currently are).

   Type the following into your terminal window and then press **Enter**:

   ```
   ls
   ```

   You should see a list of files and folders. These are all the items that are in your current directory (the *practice* directory). This should match the files and folders you see when you open the file explorer application and navigate to the *practice* directory.

6. Use the `cd` command to navigate to the *fastas* directory, which is inside the *practice* directory.

   Type the following into your terminal window and then press **Enter**:

   ```
   cd fastas
   ```

   Remember that punctuation matters! And don’t hesitate to use the **tab** key to help you complete file paths.

7. Use the `pwd` command to view the absolute file path for your current location within the computer.

   Type the following into your terminal window and then press **Enter**:

   ```
   pwd
   ```

   You should see something like:

   ```
   /home/<username>/practice/fastas
   ```

   This indicates that you are currently inside the *fastas* directory, which is inside the *practice* directory, which is inside the HOME directory on your computer.

8. Use the `cd` command to navigate back to the HOME directory.

Type the following into your terminal window and then press **Enter**:

```
cd ~
```

The tilde character (~) is an abbreviation for the file path of the HOME directory. This command will always take you back to the HOME directory, no matter where you currently are.

9. Use the `cd` command to navigate to the *fastas* directory, using a relative file path.

Type the following into your terminal window and then press **Enter**:

```
cd fastas
```

You should see an error like this:

```
cd: no such file or directory: fastas
```

This error tells you that there is no *fastas* directory inside your current directory (the HOME directory). We get this error because we used the relative file path for the *fastas* directory with the `cd` command above. If you don't provide the full path, the computer assumes it should look inside the current directory. Therefore, we can only use the relative file path for the *fastas* directory when we are inside the *practice* directory, because that is the directory in which the *fastas* directory is located.

10. Use the `cd` command to navigate to the *fastas* directory, this time using the absolute file path.

Type the following into your terminal window and then press **Enter**:

```
cd ~/practice/fastas
```

We can use the absolute file path for any file or folder and it will always work, no matter where we are currently located on the computer.

11. Go up one directory.

Sometimes you might want to go "up" one directory, i.e., to move to the folder that contains the folder you are currently in. There is a handy shortcut for this. Type the following into your terminal and then press **Enter**:

```
cd ..
```

Take a moment to use `ls` and `pwd` to better understand your current location.

12. Use the `cd` command to navigate to the *fastas* directory.

Since you are currently in the *practice* directory, you can use the relative file path here. Type the following into your terminal window and then press **Enter**:

```
cd Fastas
```

Remember that the command line is case-sensitive, and our directory is called *fastas* not *Fastas*. As a result, you should see an error like this:

```
cd: no such file or directory: Fastas
```

13. Use the `cd` command to navigate to the *fastas* directory using the relative file path.

Type the following into your terminal window and then press **Enter**:

```
cd fastas
```

This command should work, since *fastas* is inside your current directory (assuming you are in the *practice* directory, if you followed the instructions above) and it is now spelled correctly and not capitalized.

14. Use the `ls` command to view the contents of the *fastas* directory.

Type the following into your terminal window and then press **Enter**:

```
ls
```

You should see 4 files with the *.fasta* extension and a *help.txt* file.

15. Use the `less` command to view the contents of the *seq1.fasta* file.

Type the following into your terminal window and then press **Enter**:

```
less seq1.fasta
```

This will open a preview window of the *seq1.fasta* file. Use the arrows or scroll to view more of the file. Press the `q` key on your keyboard at any point in order to quit this preview.

16. Use the `cat` command to view the contents of the *help.txt* file.

Type the following into your terminal window and then press **Enter**:

```
cat help.txt
```

This will print the contents of the *help.txt* file to your screen.

17. CAREFULLY, use the `rm` command to delete the *seq1_old.fasta* file.

    Type the following into your terminal window and then press **Enter**:

    ```
    rm seq1_old.fasta
    ```

    This will permanently delete the *seq1_old.fasta* file. Be extremely careful when using the `rm` command. There is no way to undo the deletion if you make a mistake, so only use it if absolutely necessary.

18. Run the `ls` command to view the contents of the *fastas* directory.

    You should now see 3 files with the *.fasta* extension and a *help.txt* file.

19. Use the `mv` command to move the *help.txt* file out of the *fastas* directory and into the *practice* directory.

    The `mv` command takes two input arguments. When we use the `mv` command to move a file or folder, these arguments are (1) the file path for the file/folder you want to move, and (2) the file path for the new location for this file/folder. Type the following into your terminal window and then press **Enter**:

    ```
    mv help.txt ~/practice
    ```

    If you now run the `ls` command to view the contents of the *fastas* directory, you should see 3 files with the *.fasta* extension and no *help.txt* file. If you run `ls ~/practice` to look inside of the practice directory, however, you should see the *help.txt* file there.

20. Use the `mv` command to rename the *seq1.fasta* file as *ref.fasta*.

    The `mv` command takes two input arguments. When we use the `mv` command to rename a file or folder these arguments are (1) the file path for the file/folder you want to rename, and (2) the file path for the new name for this file/folder. Type the following into your terminal window and then press **Enter**:

    ```
    mv seq1.fasta ref.fasta
    ```

    Run the `ls` command to view the contents of the *fastas* directory. You should see the file with the new name.

21. Go up one directory.

Type the following into your terminal and then press **Enter**:

```
cd ..
```

Take a moment to use `ls` and `pwd` to better understand your current location.

22. Use the `mkdir` command to make a new folder in your current location called *my_analysis*.

Type the following into your terminal window and then press **Enter**:

```
mkdir my_analysis
```

Run the `ls` command to view the contents of your current directory. You should see the new folder.

23. Use the `ls` command to view the contents of the *fastas*, *scripts*, and *data* directories, using the relative file paths for each (this works if you are in the *practice* directory that contains these sub-directories).

Type the following 3 commands into your terminal window separately and press **Enter** after each one:

```
ls fastas
ls scripts
ls data
```

24. Use the `ls` command to view the contents of the *data* directory, using the absolute file path.

Type the following commands into your terminal window press **Enter**:

```
ls ~/practice/data
```

This should give the same result as `ls data` above. This directory should contain a file called *my_regions.bed*, which we will investigate below. Files that end in .bed are often used to store primer sequences and their positions along a reference genome.

25. Use the `cd` command to navigate to the *data* directory.

    Type the following into your terminal window and then press **Enter**:

    ```
    cd data
    ```

26. Use the `head` command to peek at the first couple of lines of the *my_regions.bed* file.

    Type the following into your terminal window and then press **Enter**:

    ```
    head my_regions.bed
    ```

27. Use the `tail` command to peek at the last couple of lines of the *my_regions.bed* file.

    Type the following into your terminal window and then press **Enter**:

    ```
    tail my_regions.bed
    ```

28. Use the `less` command to view the full contents of the *my_regions.bed* file.

    Type the following into your terminal window and then press **Enter**:

    ```
    less my_regions.bed
    ```

    Use the arrows or scroll to view more of the file. Press the `q` key on your keyboard to quit this preview.

29. Use the `wc` command to find out the number of lines, words and characters in the *my_regions.bed* file.

    Type the following into your terminal window and then press **Enter**:

    ```
    wc my_regions.bed
    ```

    You should see 3 numbers: (1) the number of lines, (2) the number of words, and (3) the number of characters.

30. Use the `wc` command with the lines option to print out only the number of lines in *my_regions.bed*.

    Type the following into your terminal window and then press **Enter**:

    ```
    wc -l my_regions.bed
    ```

You should see one number, the number of lines in the file.

31. This all seems really simple – why am I using the command line instead of clicking around on my computer?

    Great question. The commands you used today are very basic, but the Linux command line can be used to perform infinitely complex tasks on your computer. You can even write a list of commands (called a script) and execute all of them at once with just one simple command. More importantly, a lot of bioinformatics software can only be run from the command line, so it is a useful skill to be able to execute commands and run programs, even if you are not yet able to write programs yourself.

There are many, many resources online (and in print) to teach you how to use the Linux command line. We encourage you to keep practicing these commands and to look online or ask if you have questions!