

Linux

Log in under the Linux operating system and open a terminal window.

1 Starting applications

In the linux terminal, you can start applications by typing their name. There is, for example, a text editor called “xed” installed on the computers in the computer pool. You can start it either using the graphical user interface, or in the terminal:

☞ Type `xed` to start the program from the terminal: `xed` will open in a new window. Check the terminal: do you see the command line prompt at the beginning of a new line? Can you still use the terminal and type anything (e.g., `pwd`) in it? (No, you can't.) Close the `xed` window with the mouse to get the prompt back. Now you can enter `pwd`.

☞ Type `xed &` to start the editor again. Check the terminal: this time, you can use it for additional commands. By adding the ampersand ‘&’ after any command, the process is sent to the background, you get the command prompt back immediately, and you can keep using the terminal.

☞ Within the program `xed`, create a new text document. Write some notes about working with the terminal. Save the document under an appropriate name and close `xed`. **Note:** In Linux, spaces and Umlauts (ä, ö, ü) in names of files and directories are generally a bad idea. Trust me on this.

You probably already have the Firefox browser open. But if you didn't, you could also open it from within the terminal, simply by typing its name.

2 Basic linux commands

System utilities are little programs and were discussed in class. They can also be started in the terminal by typing their names. First try out the following linux commands that help you to explore and navigate the file hierarchy:

☞ `pwd` prints the current working directory. What is the name of your current working directory?

☞ `ls` lists the files and directories in the current working directory. You should see the text document you just created.

☞ `less` allows you to view the file contents of this file: type `less file.txt` to view a document named `file.txt`, for example. Note that `less` is for viewing only, not for editing. To exit `less`, type ‘q’ for quit.

🖱️ `mkdir` creates a new directory. `mkdir exercise03` will create a new directory named 'exercise03'. Afterwards, the command `ls` should indicate that you successfully created the new directory. (Of course, you can also create a new directory without the terminal, using the mouse, and it is absolutely ok to do that.)

🖱️ `mv` moves (renames) a file. `mv file.txt exercise03/`, for example, will move the file `file.txt` into the directory `exercise03/`. **Note:** if you forget to end the slash (/) after the name of the directory, you will simply *rename* the file from 'file.txt' to 'exercise03'.

🖱️ Verify that the moving and/or renaming worked: `ls` and `ls exercise03/` (Again, you can also move files and directories without the terminal, using the mouse, and it is absolutely ok to do that.)

`cd` allows you to change into another working directory: when you type the name of a directory after the command `cd`, you will change into it. When you just type the command `cd` on its own, you will change into your own home directory.

- 🖱️ Type `cd exercise03/` to change into the directory `exercise03`.
- 🖱️ Type `cd` to change to your home directory.
- 🖱️ Note: the file browser and the terminal can work together: Type `cd` in the terminal, followed by a space. Then use the mouse to drag a directory from the graphical file browser into the terminal behind the `cd` command: the path to the directory will be copied to the terminal this way.

3 Tab completion

In Unix/Linux systems, you can type only the first few letters of a command or file or directory and then the Tab key: The command will be completed for you if it is unambiguous. Otherwise you will be presented with a number of valid options from which you can choose one. Try this out next:

🖱️ Type `cd` to change into your home directory. If you now type `ls`, the folder `exercise03` should be listed.

🖱️ Type `ls e` and then press the tab key. What happens? Explain.

🖱️ Type `cd e` and then press the tab key. What happens? Explain.

Tab completion is extremely useful: it saves time because you don't have to type in long or cryptic file names by hand. Also, typos are avoided by using the tab completion feature. Get into the habit of using it.

4 Working with file content

During the lecture, I discussed how linux system utilities can be used to efficiently search and filter text within files. Download the file "species.txt" from Moodle and save it in your new directory "exercise03". The file is compressed, so first you need to extract it: right-click the file and select "extract here" from the contextual menu.

☞ View the file content with the command `less`. It contains the species names of every entry in the UniProt SwissProt division, one line per entry. For each sequence, only the species from which the sequence was isolated is given, not the sequence ID or the sequence itself.

☞ How many sequences are in UniProt SwissProt? – To answer this question, simply count the number of lines in this file. You can do this with the utility word count, using the option `-l` to not count words and bytes, but only lines: `wc -l`

☞ How many species are represented in this file? Which species have the most entries? Do this with the utilities `cat`, `sort`, `uniq` and in steps that are connected by pipes, as discussed in class.

5 Optional: The Linux history

☞ Type the word `history` in the terminal window, then hit the Enter key. What happens?

☞ The `history` command prints to the screen the last (few hundred) terminal commands that were issued. If you want to save these for future reference, you can redirect the command history into a file:

`history > history_ex03.txt` for example, would save the current command line history into a file named `history_ex03.txt`. Verify that you created the file using the command `ls`, and view its content by typing `less history_ex03.txt`.

6 Closing the terminal session

☞ To cleanly close the terminal window on the local machine, type `exit`.

7 Optional: Getting help

Here are some options for getting help on commands. Try them out, but don't spend too much time reading through the help pages. Simply remember to come back to them if you need help with a particular command.

- `help pwd`: returns a brief usage summary for the command `pwd`
- `man pwd`: returns a more detailed help (a manual) for the command `pwd`. Note: manual pages are displayed one page at a time. Advance to the next page by pressing the Space Bar. When you're done, press 'q', for quit.

Use the manual pages (`man`) too look up information about the `ls` command. Specifically, use it to find out how you can use `ls` to view the files of a directory sorted by date.