

Introduction to the UNIX command line

Introduction to Linux for Bioinformatics Workshop

Moi University Bioinformatics Hub

Collins Kigen

25th March 2023

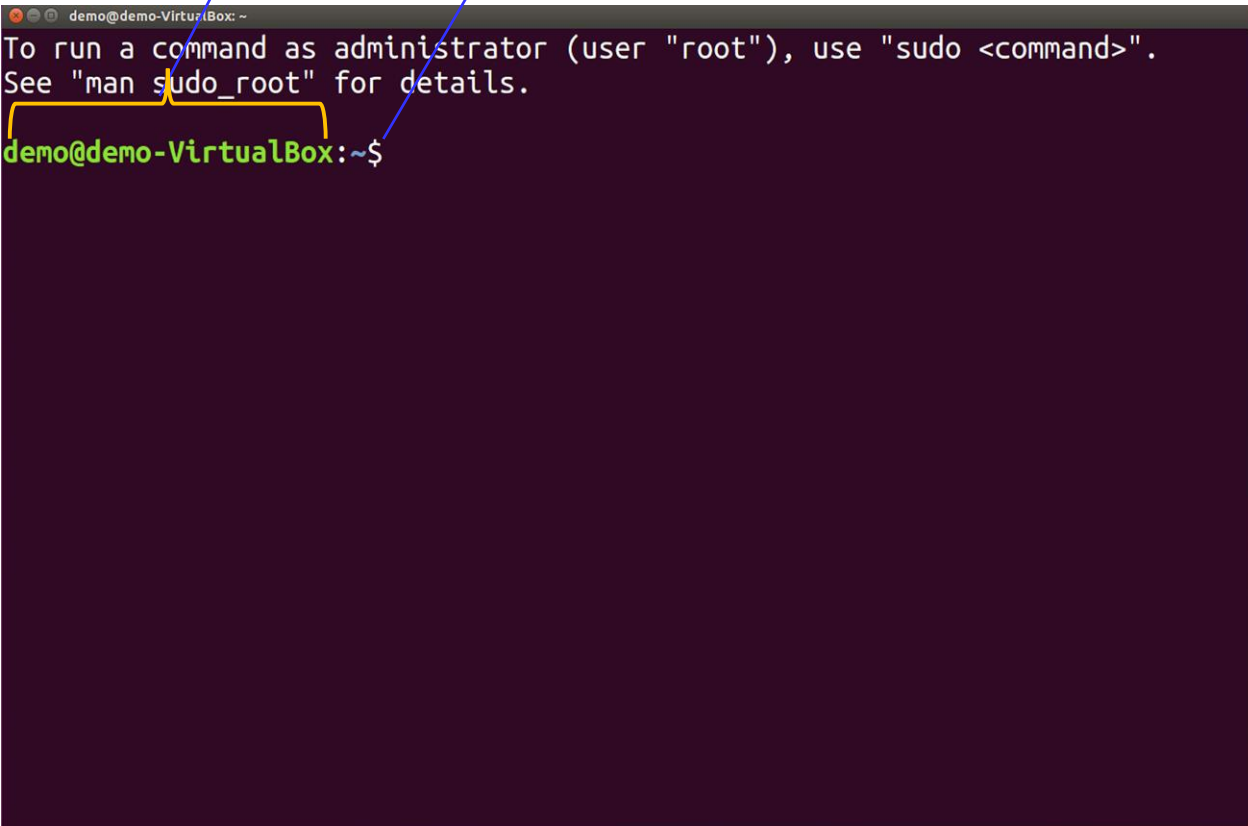
What is the shell?

A shell is a computer program that presents a command line interface which allows you to control your computer using commands entered with a keyboard instead of controlling graphical user interfaces (GUIs) with a mouse/keyboard/touchscreen combination.

The Shell Terminal

Prompt

Dollar sign



- ❖ The \$ dollar sign: the symbol used to signify where you can begin typing in command.
- ❖ Prompt: Shows username, the current folder computer's name etc.



The most popular Unix shell is **Bash** (the **B**ourne **A**gain **S**hell — so-called because it's derived from a shell written by Stephen Bourne).

Shell vs GUI

Graphical User Interface (GUI)



- ❑ Give instructions by clicking a mouse and using menu-driven interactions
- ❑ The visual aid makes it intuitive to learn

Shell

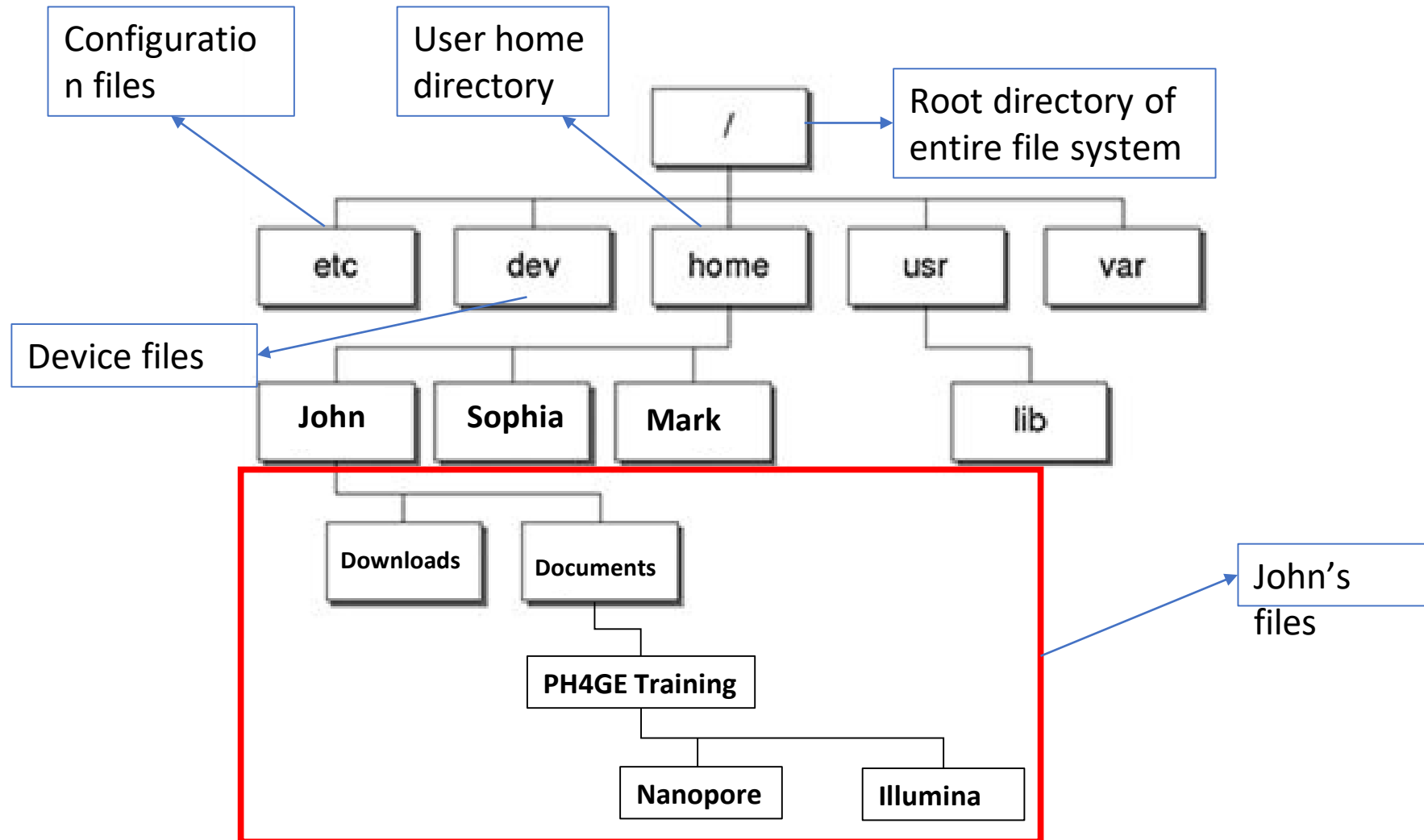
```
caine@caine:~$ ls
Desktop  Downloads  Pictures  Templates
Documents Music      Public   Videos
caine@caine:~$ cd Desktop
caine@caine:~/Desktop$ ls
autopsy.desktop      'Disk Image Mounter.desktop'  Network.desktop
BlockOn-Off.desktop  Guymager.desktop             ubiquity.desktop
Caja-Root.desktop    'Keyboard changer.desktop'
caine@caine:~/Desktop$ touch ok.txt
caine@caine:~/Desktop$ ls
autopsy.desktop      'Disk Image Mounter.desktop'  Network.desktop
BlockOn-Off.desktop  Guymager.desktop             ok.txt
Caja-Root.desktop    'Keyboard changer.desktop'    ubiquity.desktop
caine@caine:~/Desktop$ mkdir practice
caine@caine:~/Desktop$ ls
autopsy.desktop      'Keyboard changer.desktop'
BlockOn-Off.desktop  Network.desktop
Caja-Root.desktop    ok.txt
'Disk Image Mounter.desktop'  practice
Guymager.desktop      ubiquity.desktop
caine@caine:~/Desktop$
```

- ❑ Give instructions by issuing commands. Examples
`ls` , `rm` , `pwd` , `cd` , `cp` , `mv` , `mkdir`
- ❑ Requires expertise

Why the shell?

- ① Many bioinformatics tools can only be used through a command line interface.
- ② With the shell, you can automate repetitive tasks
- ③ The shell makes your work more reproducible.
- ④ The shell makes your work less error-prone
- ⑤ To access remote computers or cloud computing with large amounts of computing power

The Linux File System



Looking at the Contents of any Directory

- `pwd` for **P**rinting the current **W**orking **D**irectory.
- `ls` to view content of current directory
- `ls /` to view the content of other directories (root)
- `ls /home` to view content of /home

Navigating the Shell

- ❖ The built-in command which we use to *change directory* is `cd`

```
cd /  
cd home  
cd collins
```

- ❖ To move back to previous directory

```
cd ..
```

Important to Note

The string `..` is the convention for one directory above, whilst a single dot `.` represents the current directory.

Relative Vs Absolute Paths

Absolute Path

- ❑ Specify a location (file or directory) in the system all the way from the topmost folder (root directory) and thus always start with /

- ❑ To move to *collins* from /home:

```
cd /home/collins
```

Relative Path

- ❑ Specify a location in the system relative the current working directory and will never start with the slash.

- ❑ To move to *collins* from /home:

```
cd collins
```

Working with Files and Directories

- `mkdir folder1` to create a directory
- `touch file1` to create file
- `rm -r folder1` to remove a directory
- `rm file1` to remove a file



Warning!

Command `rm -r` is potentially VERY dangerous.

What would happen if you execute `rm -r /`

Working with Files and Directories

- `mkdir folder1` to create a directory
- `touch file1` to create file
- `rm -r folder1` to remove a directory
- `rm file1` to remove a file

Working with Files and Directories

- `cp -r folder2 newfolder1`
- `cp file1 newfolder1`
- `mv -r folder1` to move a directory
- `mv file1` to move/rename a file

Creating & Viewing a File

- `nano notes.txt` to create/edit a file
- `cat seq.fasta` to view the whole content of a file
- `less seq.fasta` to view file content one page at a time
- `head -n 5 seq.fasta` to view first 5 lines of a file
- `tail -n 5 seq.fasta` to view last 5 lines of a file

Pipe

A pipe (|) is used to connect multiple commands. It takes the output from the previous command and “pipes” it into the input of the following command.

`wc -l example.txt` to count the number of lines in a file

`ls` to list the content of a directory

`ls | wc -l` to count the number of lines in a file

Redirection

Another important character is the greater than sign `>`. This tells the command line to redirect the output to a file, rather than just printing it to the screen (STDOUT)

`ls` to count the number of lines in a file

`ls > content.txt` to count the number of lines in a file

Append

Append means simply **add the data to the file without erasing existing data.**

`echo "Hello" > greeting.txt` to save first line to a file

`echo "How are you" >> greeting.txt` to add more data to existing file

Thank You