



# Terminal and BASH





# /TABLE OF CONTENTS



## /01 Terminal vs GUI

- > What are the diffs?, a little history, when we only have a terminal?

## /02 A little about BASH and installation

- > installation

## /03 Terminal vs gui examples

- > Some examples

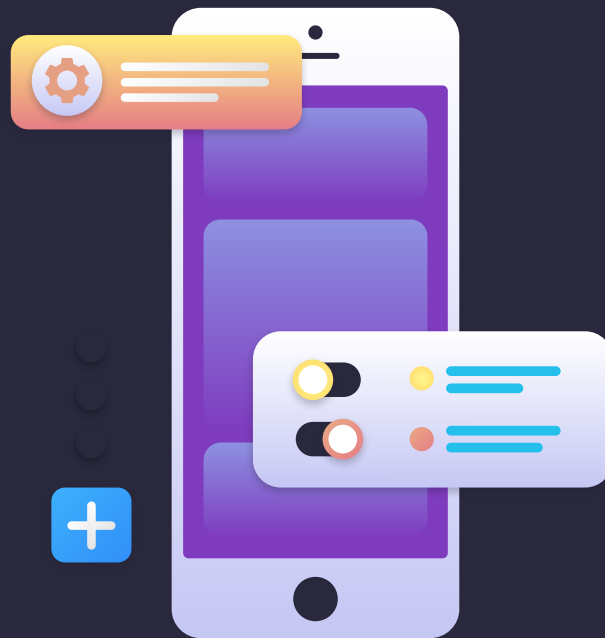
## /04 Some basic bash commands

- > Ls, cd, pwd, cat



# /01

# Terminal vs GUI



# Terminal vs GUI



- In our day to day lives we are working with computers.
- In order to tell computer what to do and to make sure it understands us - we use **interfaces**.
- There are 2 main types: graphical interfaces and text based (command line) interfaces.
- GUI and CLI - for short.






# Terminal vs GUI

- What is GUI?
  - We all know it...



 Direct messages

 Drafts & sent

 Canvases

 Files

 Browse Slack

▼ Channels


# bootcamp10

# general

# learning-materials

# q\_n\_a

# random

 Add channels

# Terminal vs GUI



- In a gui we see sections, buttons, a mouse cursor and text boxes – where we as users can interact with the program.
- Cli is different...

# Terminal vs GUI



- CLI

```
john@ubuntu: ~  
john@ubuntu:~$ ls  
john_directory john_file  
john@ubuntu:~$ ls -l  
total 8  
drwxrwxr-x 2 john john 40 Oct 1 11:10 john_directory  
-rw-rw-r-- 1 john john 5120 Oct 1 11:17 john_file  
john@ubuntu:~$ ls -l -h  
total 8.0K  
drwxrwxr-x 2 john john 40 Oct 1 11:10 john_directory  
-rw-rw-r-- 1 john john 5.0K Oct 1 11:17 john_file  
john@ubuntu:~$ ls -lh john_file  
-rw-rw-r-- 1 john john 5.0K Oct 1 11:17 john_file  
john@ubuntu:~$ ls -l --human-readable john_file  
-rw-rw-r-- 1 john john 5.0K Oct 1 11:17 john_file  
john@ubuntu:~$
```

# Terminal vs GUI

- CLI vs GUI





# Terminal vs GUI

- CLI vs GUI





# Terminal vs GUI



- What are the main diffs?
  - Text vs visual
  - Speed
  - Simplicity (depends)





# Terminal vs GUI



- Which one would you prefer?
- Why not use GUI always?





# /02

## A little about BASH





# A little about BASH



- Every OS has a cli in it.
- mac/linux has bash based terminal, and windows has cmd/powershell
- Each of them can do basically the same things.





## A little about BASH



- In development we usually use linux, so we will use bash.
- BASH = bourne again shell. “Shell” basically means terminal in this context.
- Its a language, similar to a programming language, but here we will use it for commands.



# A little about BASH



- If you have mac or linux you already have BASH built in.
- If you are using windows - you need to install it.
- The easiest way is to install git (which we will need later anyway). It comes with “git bash” terminal.





# A little about BASH



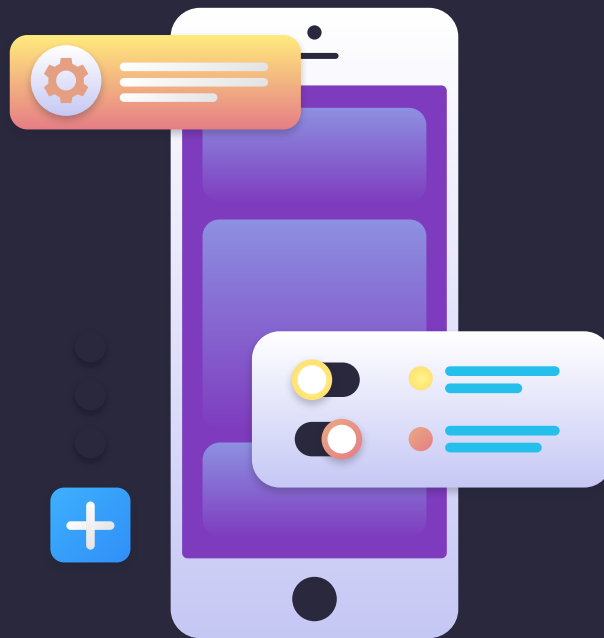
- Install here: <https://git-scm.com/downloads>
- You can use this video if you need:  
[https://www.youtube.com/watch?v=7B0rUHFu44A&ab\\_channel=HowTo](https://www.youtube.com/watch?v=7B0rUHFu44A&ab_channel=HowTo)





# /03

## Terminal vs gui examples





# Cli vs gui examples



task	gui	cli
Create a file	Right click -> create a file	touch <file name>
Read a file contents	Double click on a file	cat <file name>
Move a file to a new location	Open one folder, drag the file to the folder	mv <current file location> <new file location>



## Cli vs gui examples



- This doesn't seem like using cli is easier than using a GUI!
- But consider this: let's say you have a folder with many files, and you want to see how much memory all the image (jpg/png) files that have `<text>` in their name take.
- How would you do it with GUI?

## Cli vs gui examples



- With CLI its one line:

```
find /path/to/folder -type f -iname "*<text>*"
\(-iname "*.jpg" -o -iname "*.png"\) -exec
du -ch {} + | grep total$
```

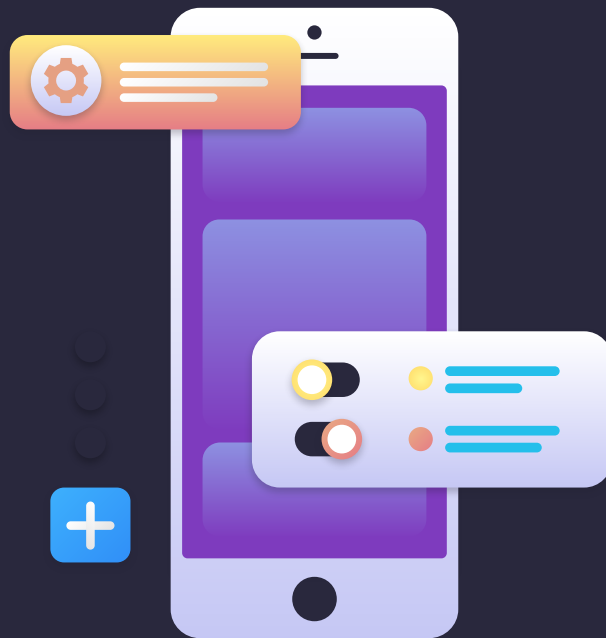


example:

```
find /home/username/Photos -type f -iname
"*vacation*" \(-iname "*.jpg" -o -iname
 "*.png"\) -exec du -ch {} + | grep total$
```

# /04

## Some basic bash commands



# Basic bash commands



- We will go over some basic bash commands:
  - `cd` = change directory. Move from one directory to the other.

Important: relative paths:

`"./"` = in this directory.

`"../"` = in the containing directory.

# Basic bash commands



- We will go over some basic bash commands:
  - `pwd` = the location where I am.
  - `cat` = read contents of a file.
  - `touch` = create a file.
  - `rm` = remove a file.
  - `ls` = list the content of a directory.
  - `grep` = search for text/text patterns (regex).

# Basic bash commands



- We can pass the output of one command to the input of another command, using “|” (pipe) - shift + left of enter.

For example:

```
ls | grep d
```

= will list the contents of the current directory, and will show only the results that include the letter “d”.





# Thanks, and good luck!

