

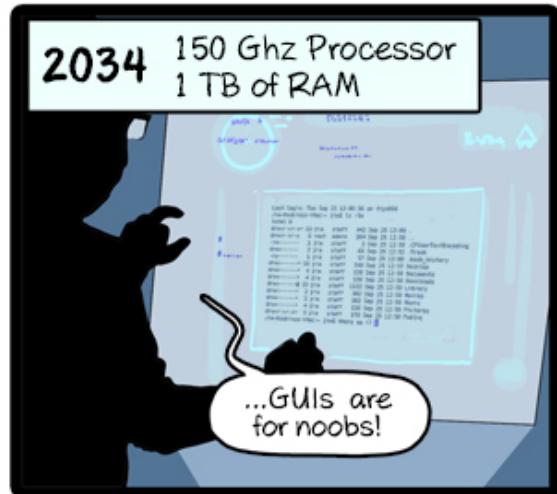
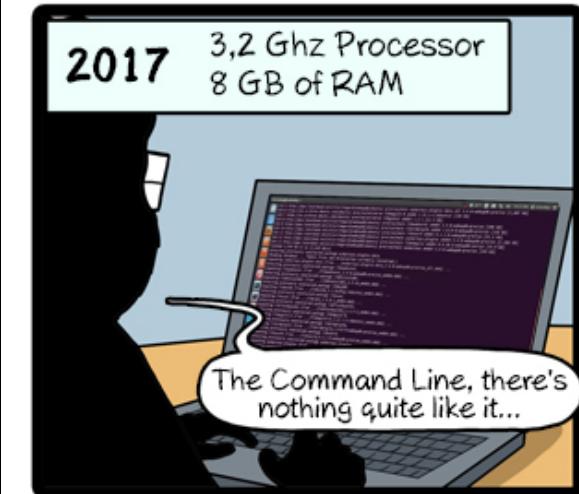
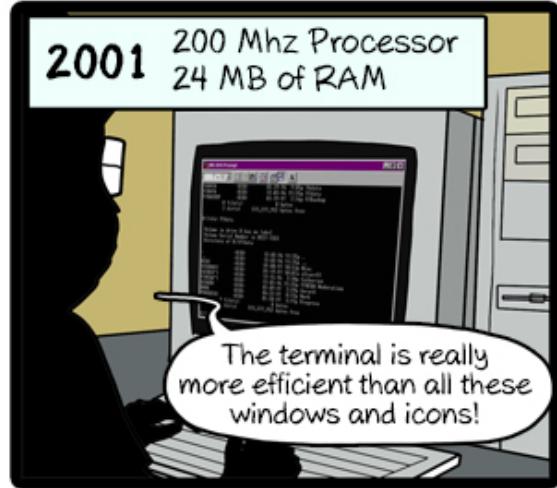
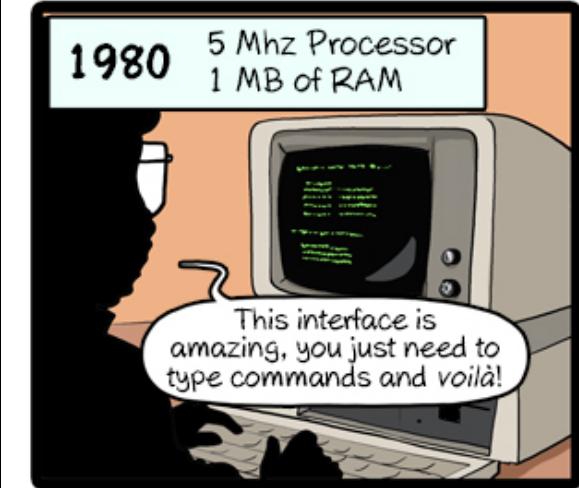
Intro to Computer Science

CS-UH 1001, Spring 2022

Lecture 1 – Command Line Terminal

Recap from last class

- Two kinds of programming languages exists:
 - High level
 - Low level
- Programming languages can have errors:
 - Syntax, Runtime and Semantics Errors
- Installed Python 3.X and it should be running by now!



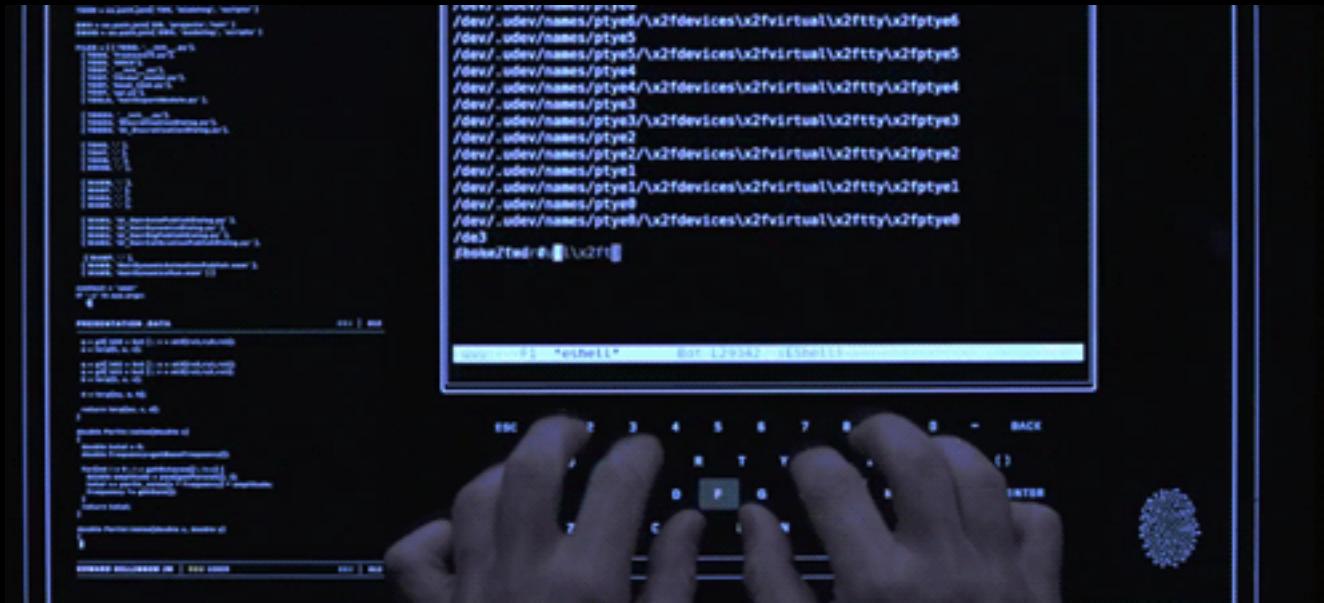
CommitStrip.com

Working With the Terminal

Operating Systems

- An Operating System (OS) is
 - a set of programs that manages a computer's hardware
 - acts as an interface between the user and the hardware
- When computers were invented, there was no graphical user interface
- Linux/Unix is the beginning of the computer OS's
- Windows/MacOS came much later
- Many state-of-the-art OS's are still based on Unix
 - MacOS
 - iOS
 - Android
 - Chrome OS
 - etc

What is the Terminal?



- The command line, or terminal, is a text based interface to the operating system
- You can enter **commands** by typing them on the keyboard and feedback will be given to you similarly as text
- It's is more powerful than you think!

Why you need a Terminal?

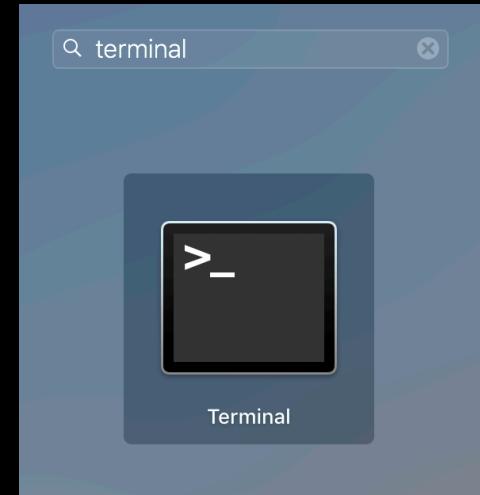
- A terminal is a powerful interface between the system and the user
 - It provides an efficient way to access the true power of the OS better than any graphical interface
- There are just a few standard terminals and commands, but GUI interfaces are countless
 - Some devices don't even have a GUI
- Things done in a terminal are easily repeatable
- Scripting/automating is nearly as easy as entering commands
- Most importantly: You are not a real computer scientist if you don't know how to use the terminal! ☺

Terminal on Windows

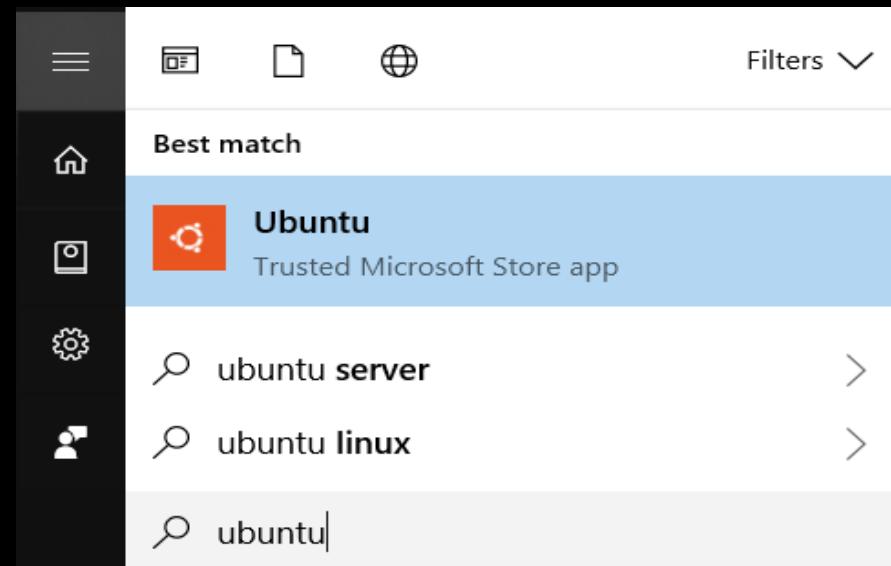
- The Windows terminals (cmd, PowerShell) and the Unix terminal operate on the same principals, but use different commands
- Windows user will use the Windows Subsystem for Linux
 - It is a full Linux/Unix system inside of Windows
 - We use Ubuntu as the OS

Finding the Command Line

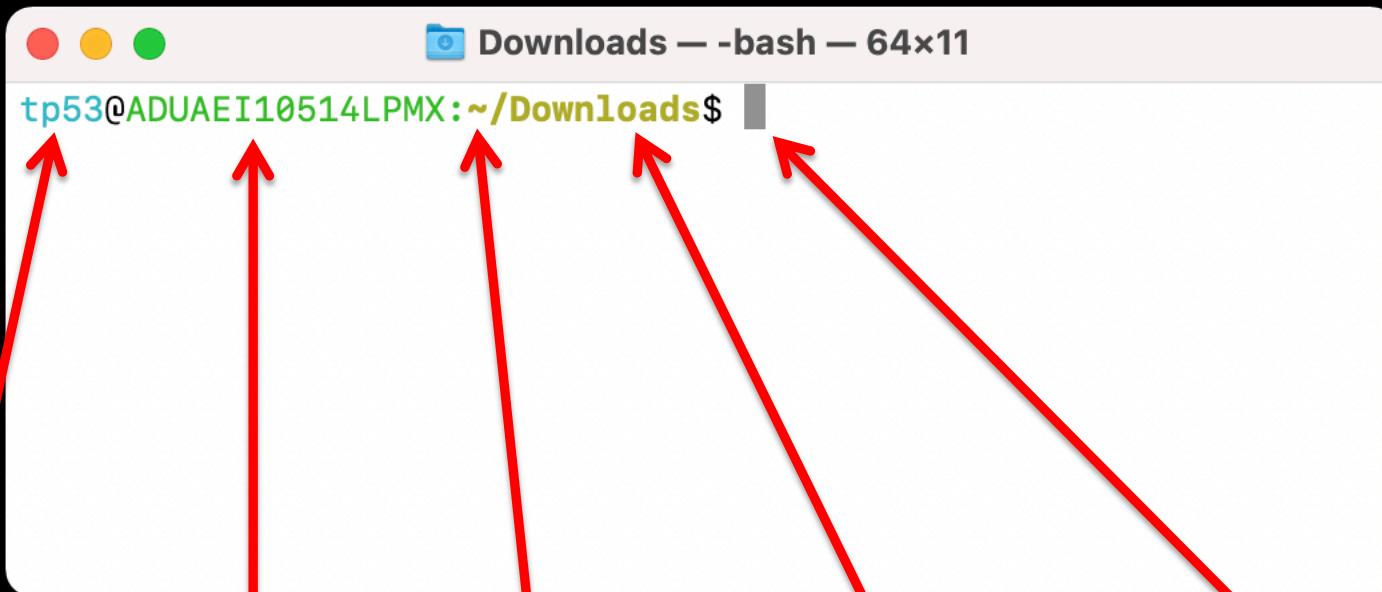
Mac OS



Windows



The Command Line



User name

**Abbreviation for
your Home folder**
(where all your personal files
are stored)

Computer name

**Current working
directory (folder)**

Blinking cursor
(where you can execute
commands)

Terminal Commands

- Terminal commands are programs that are installed on your system
- The components of a **command** are:
 - the **command**
 - any **options** required by the command, preceded by a hyphen/dash (optional)
 - the command's **arguments** (if required)

command <options> <arguments>

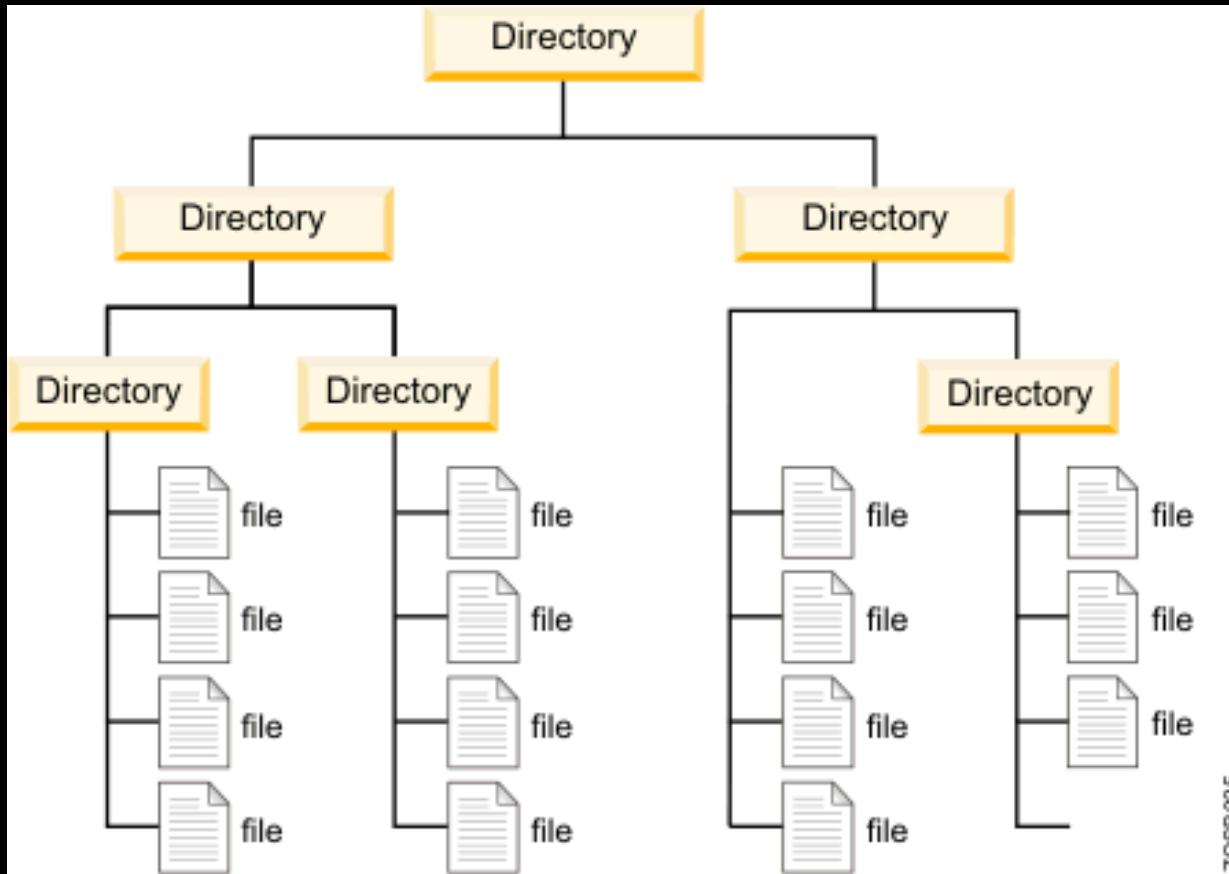
- To run a command, type the name of the command, along with options and/or arguments, and press Enter
- Commands usually operate on the folder you are currently in

File structure

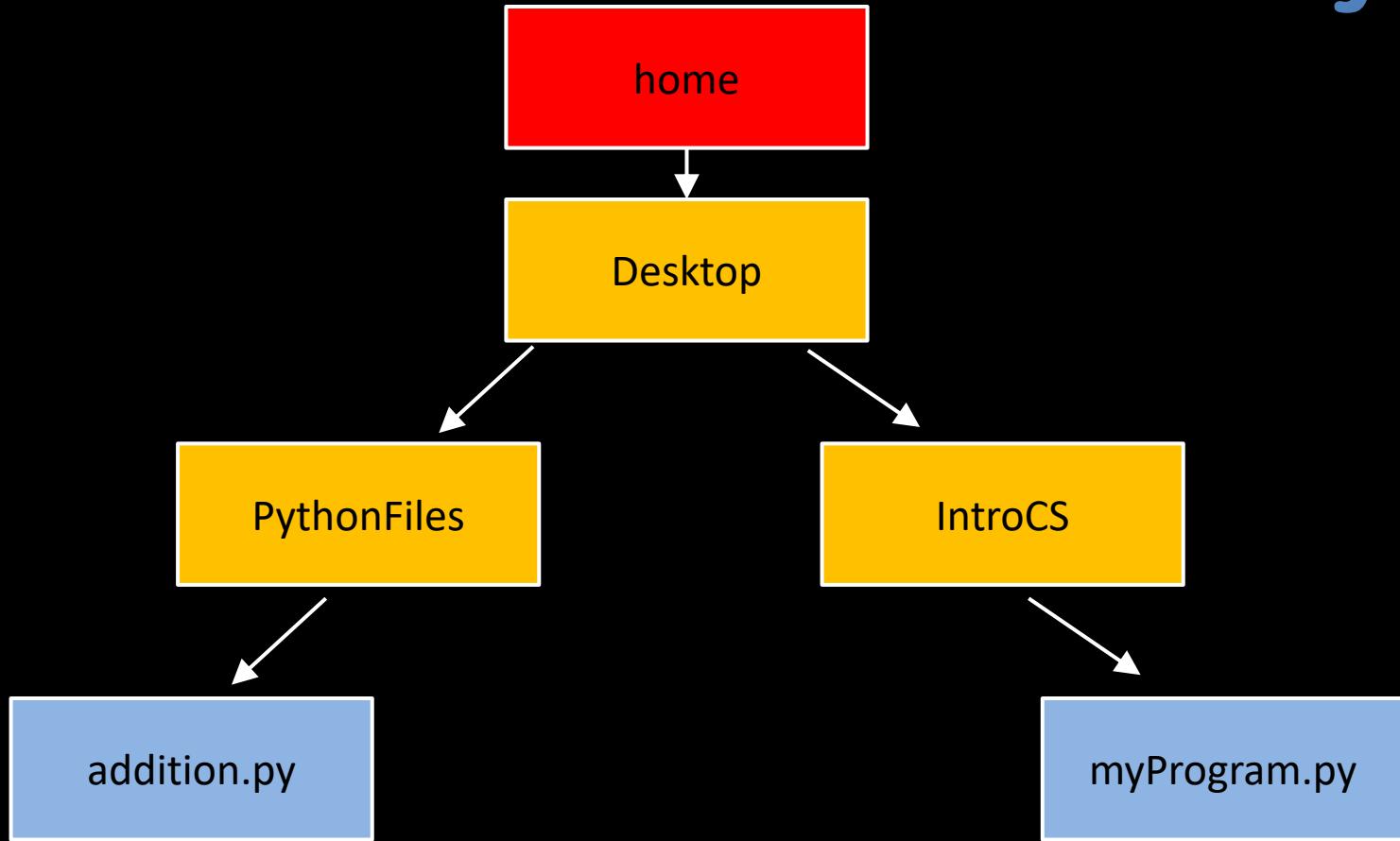
Unix File System

- The Unix file system is organized in a tree structure
- The file tree can be arbitrarily deep
- Every folder is separated by a /
- You have access to the same files and folders of your computer, e.g. Desktop, Documents, Downloads, etc.

File and Folder Hierarchy

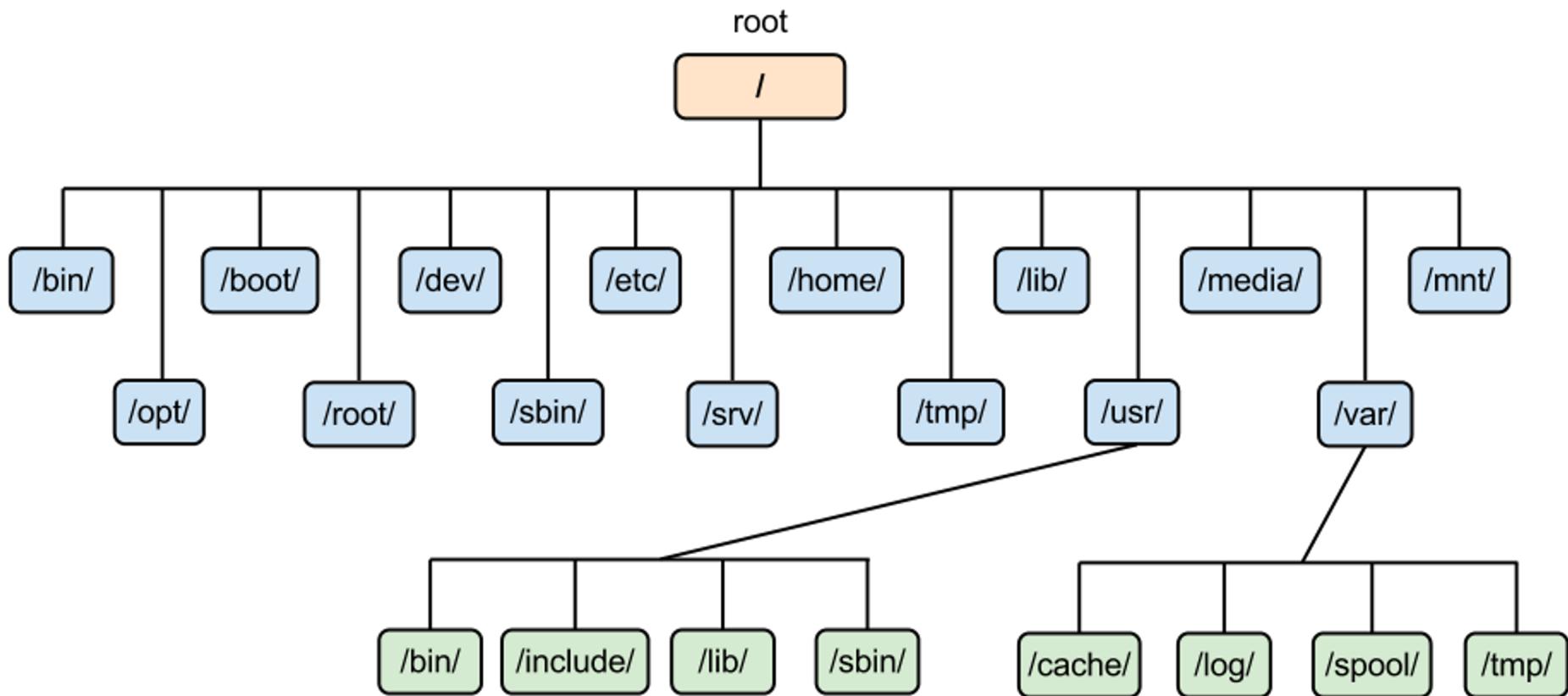


File and Folder Hierarchy



- In a Unix file system, all your personal files are stored in the **home** folder of your OS

Unix File System

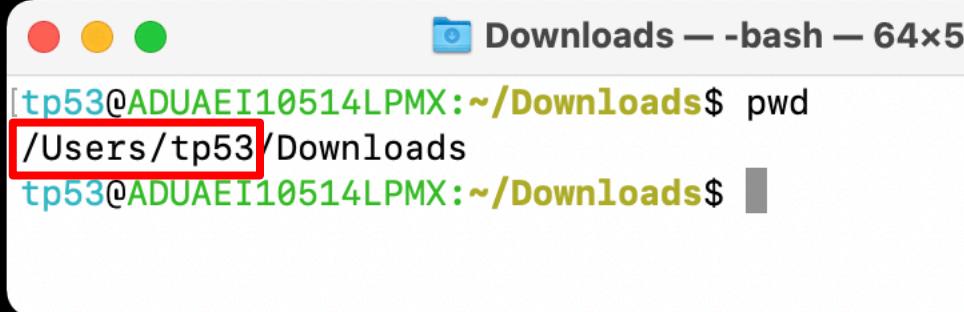


- You can use the terminal to navigate through your file system

Current Working Directory

- The command **pwd** shows your current working directory (the **full path from root**)

MacOS:

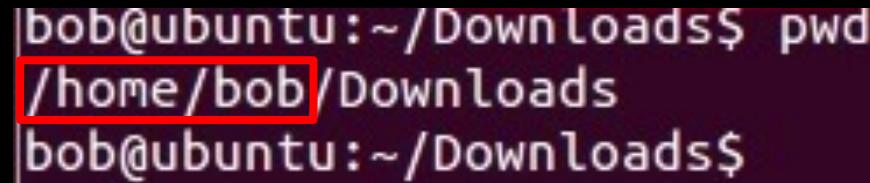


```
[tp53@ADAEI10514LPMX:~/Downloads$ pwd  
/Users/tp53/Downloads  
tp53@ADAEI10514LPMX:~/Downloads$ ]
```

A screenshot of a MacOS terminal window titled "Downloads — bash — 64x5". The window shows the command "pwd" being run, which outputs the full path "/Users/tp53/Downloads". The path is highlighted with a red box.

Home directory

Linux:



```
bob@ubuntu:~/Downloads$ pwd  
/home/bob/Downloads  
bob@ubuntu:~/Downloads$ ]
```

A screenshot of a Linux terminal window showing the command "pwd" being run, which outputs the full path "/home/bob/Downloads". The path is highlighted with a red box.

Home Directory

- All your files (Desktop, Documents, etc.) can be found here:

- On MacOS you are already in your home directory when you open the Terminal

`/Users/YOURUSERNAME/`

- On Windows (Linux Subsystem - Ubuntu) your personal Windows files are here:

`/mnt/c/Users/YOURUSERNAME/`



Windows users, please
keep this path in mind!

Files

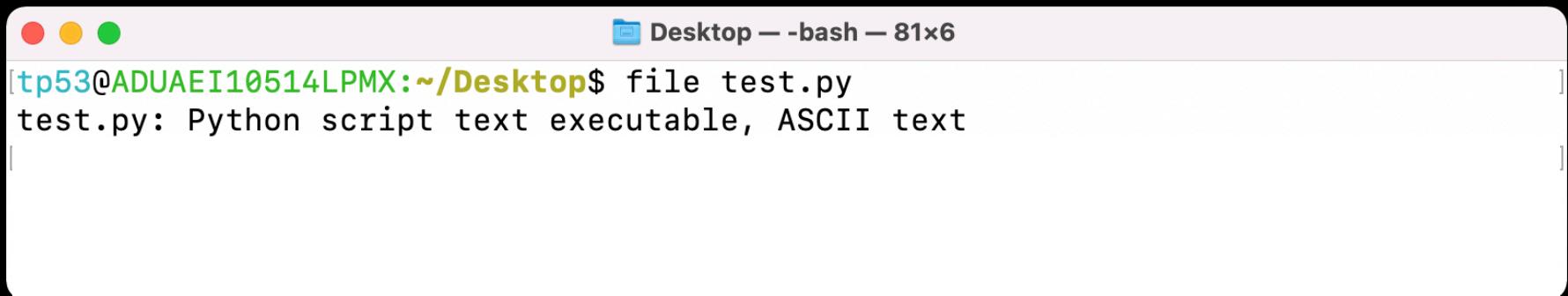
- Generally, two types of files:
 - binary: images, audio, videos, MS Word, executables, etc.
 - text: files that contain characters
- Most files have a file extensions to indicate the type of the file, e.g.: **image.jpg**
 - File name
 - File extension
- Operating systems usually do not show the extensions, but use them to know what program should open it

Common File Extensions

- A JPEG uses **.jpg** or **.jpeg**
- A Word document uses **.docx**, or **.doc** for older versions
- A PowerPoint slide uses **.pptx**, or **.ppt** for older versions
- An MP3 audio file uses **.mp3**
- A text file uses **.txt**
- A Python file uses **.py**

Determine File Types

- The command **file** can be used to determine the type of a file



A screenshot of a macOS terminal window titled "Desktop — -bash — 81x6". The window has three colored window control buttons (red, yellow, green) at the top left. The terminal output shows the command "file test.py" being run, followed by the result "test.py: Python script text executable, ASCII text".

```
[tp53@ADUAEI10514LPMX:~/Desktop$ file test.py
test.py: Python script text executable, ASCII text]
```

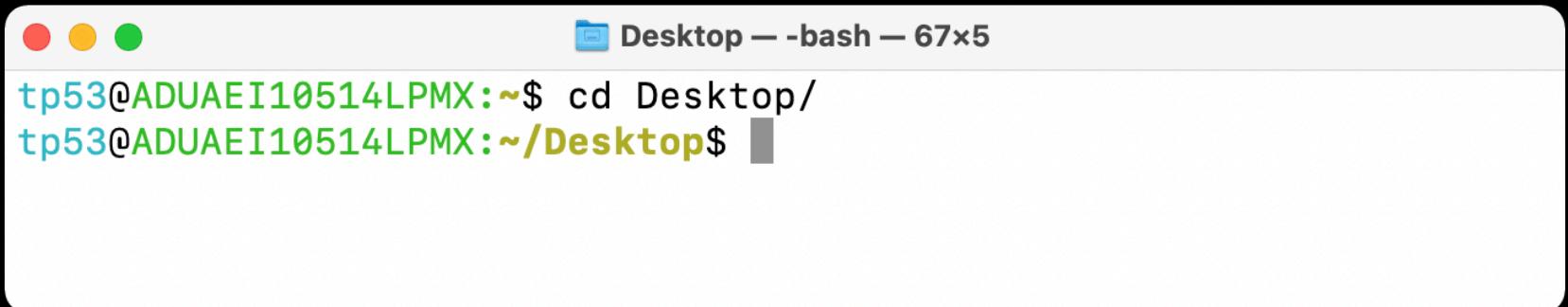
Directory operations

Listing directories/files

- **ls** can be used to list all folders and files in the current working directory
- Useful options:
 - **ls -l** → show all in long format
 - **ls -r** → show in reverse order
 - **ls -1** → show one file per line
 - **ls -S** → sort output by file size

Changing Directories

- **cd** (change directory) is used to change the current working directory



```
tp53@ADUAEI10514LPMX:~$ cd Desktop/
tp53@ADUAEI10514LPMX:~/Desktop$ █
```

- Similar to double clicking on a folder in Explorer (Windows) or Finder (Mac)

Change Directory

- **cd** is also used to move one folder up
 - **cd ..** (.. means one folder up)
 - **cd ../../** (../../ means two folders up)
 - Similar to the up icon/arrow in explorer or finder
- You can go back to your home directory using (MacOS and Linux)
 - **cd ~**
- Windows – Linux Subsystem only:
 - **cd /mnt/c/Users/USERNAME**
 - Keep this line in mind as you have to execute it whenever you open Ubuntu terminal

Change directory cont.

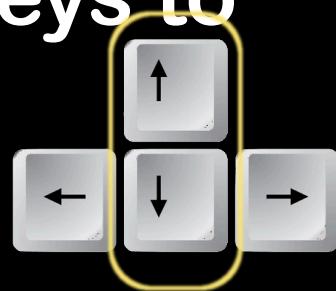
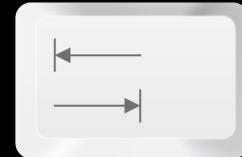
- Folders with spaces in their names, must be ‘escaped’ by \ or using “ ”
 - `cd subfolder\ 1` or `cd “subfolder 1”`

Getting Help with Commands

- The command line/terminal has a built-in documentation for all commands
- The **man** command shows a manual for the command
man command
- Exit the manual page by pressing q
- Fun fact: The **man** command has also a manual

Useful Tricks

- Use tab key to autocomplete file, folder names and commands
- You can use up and down arrow keys to get to the last commands used
- Type **clear** (or **Ctrl-l**) to clear the screen

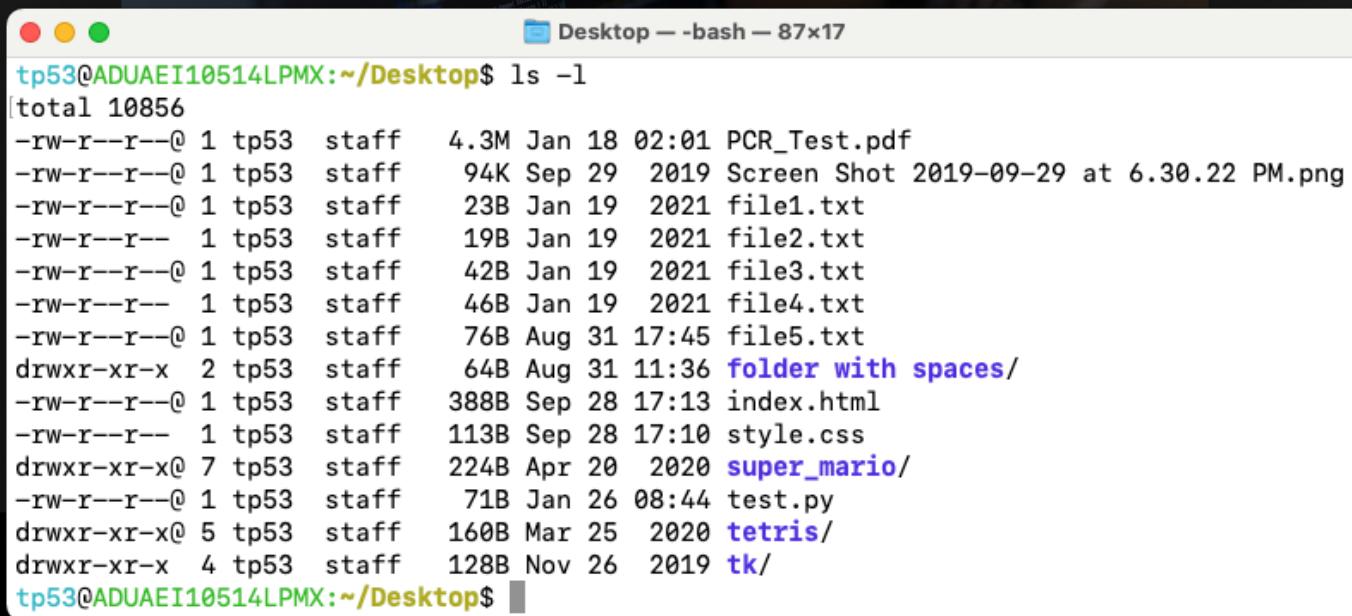


Breakout session I:

Terminal/Command line

Perform the following tasks on the terminal:

1. navigate to your Desktop*
2. use the **ls -l** command to list all files on your Desktop
3. Compare the output with the the files on your Desktop



```
tp53@ADUAEI10514LPMX:~/Desktop$ ls -l
total 10856
-rw-r--r--@ 1 tp53  staff   4.3M Jan 18  02:01 PCR_Test.pdf
-rw-r--r--@ 1 tp53  staff   94K Sep 29  2019 Screen Shot 2019-09-29 at 6.30.22 PM.png
-rw-r--r--@ 1 tp53  staff   23B Jan 19  2021 file1.txt
-rw-r--r--  1 tp53  staff   19B Jan 19  2021 file2.txt
-rw-r--r--@ 1 tp53  staff   42B Jan 19  2021 file3.txt
-rw-r--r--  1 tp53  staff   46B Jan 19  2021 file4.txt
-rw-r--r--@ 1 tp53  staff   76B Aug 31  17:45 file5.txt
drwxr-xr-x  2 tp53  staff   64B Aug 31  11:36 folder with spaces/
-rw-r--r--@ 1 tp53  staff  388B Sep 28  17:13 index.html
-rw-r--r--  1 tp53  staff  113B Sep 28  17:10 style.css
drwxr-xr-x@ 7 tp53  staff  224B Apr 20  2020 super_mario/
-rw-r--r--@ 1 tp53  staff   71B Jan 26  08:44 test.py
drwxr-xr-x@ 5 tp53  staff  160B Mar 25  2020 tetris/
drwxr-xr-x@ 4 tp53  staff  128B Nov 26  2019 tk/
tp53@ADUAEI10514LPMX:~/Desktop$
```

*Windows Subsystem Linux: **cd /mnt/c/Users/USERNAME/Desktop**

Let's take a break !!!!!



I PROMISED MY DOCTOR

ONLY ONE CUP A DAY...

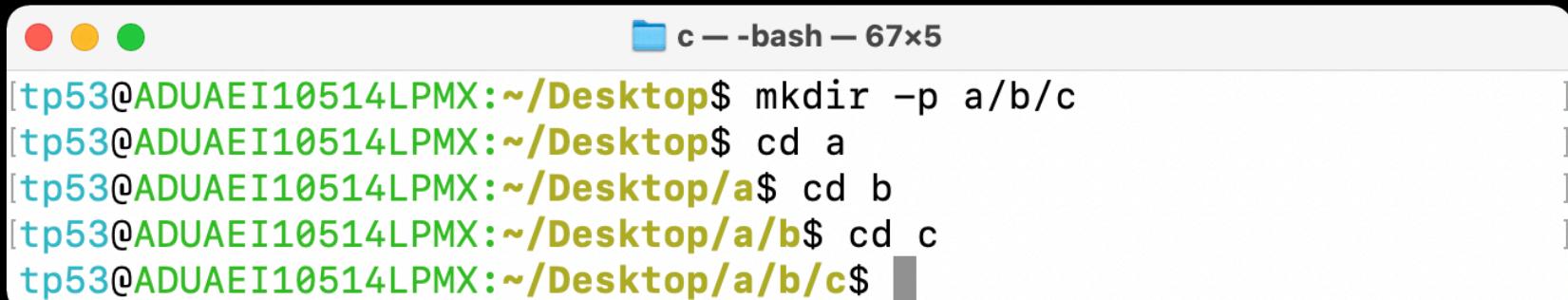
Creating directories

- **mkdir (make directory):**



```
[tp53@ADUAEI10514LPMX:~/Desktop$ mkdir test
[tp53@ADUAEI10514LPMX:~/Desktop$ cd test
tp53@ADUAEI10514LPMX:~/Desktop/test$ ]
```

- Make subdirectories at once using the option **-p**:

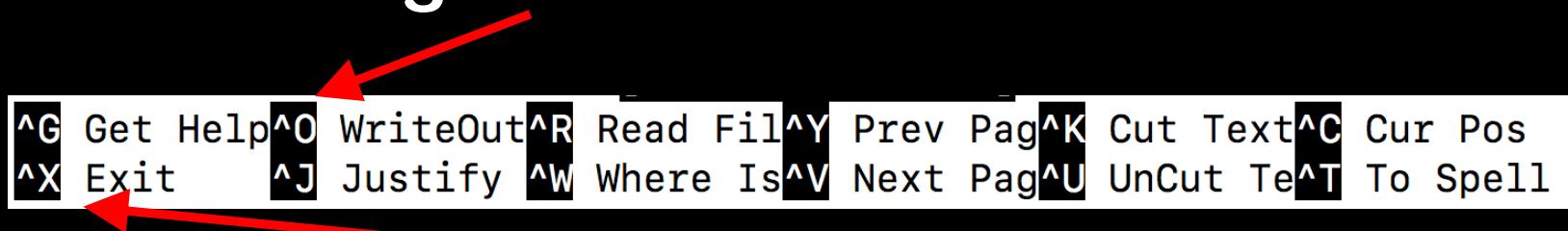


```
[tp53@ADUAEI10514LPMX:~/Desktop$ mkdir -p a/b/c
[tp53@ADUAEI10514LPMX:~/Desktop$ cd a
[tp53@ADUAEI10514LPMX:~/Desktop/a$ cd b
[tp53@ADUAEI10514LPMX:~/Desktop/a/b$ cd c
tp53@ADUAEI10514LPMX:~/Desktop/a/b/c$ ]]
```

File Access

Creating/Editing a file

- You can use **nano** to create or edit a file
 - For example: **nano intro.txt**
 - if the file does not exist, it will be created!
- Save using **ctrl-o** and enter



- You can exit **nano** by pressing **ctrl-x**
 - If you edit the file you need to confirm if you want to save by typing **y** and then **enter**

Viewing a file with **cat**



```
Desktop -- bash -- 56x12
tp53@ADUAEI10514LPMX:~/Desktop$ cat intro.txt
Hi!

My name is Thomas.

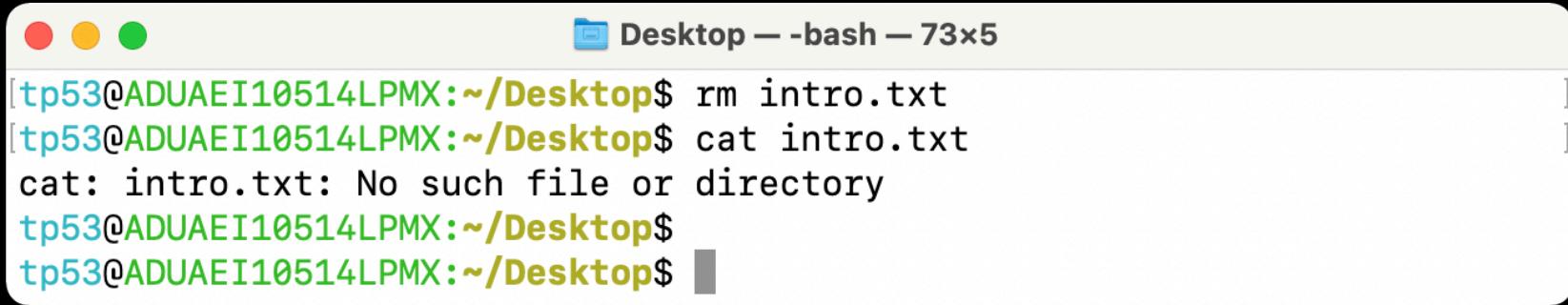
:)

tp53@ADUAEI10514LPMX:~/Desktop$
```

- **cat** gives you a sneak peak on what's inside a file and returns to the command line

Deleting files

- **rm** can be used to delete files



```
[tp53@ADUAEI10514LPMX:~/Desktop$ rm intro.txt
[tp53@ADUAEI10514LPMX:~/Desktop$ cat intro.txt
cat: intro.txt: No such file or directory
tp53@ADUAEI10514LPMX:~/Desktop$ 
tp53@ADUAEI10514LPMX:~/Desktop$ ]]
```

- If you want to delete a folder use, the option **-r**

rm -r foldername

- If you want to delete all files inside a folder, but not the folder, use asterisk (*)

rm foldername/*

Moving Files

- You can use the **mv** command to move files within your file structure

mv filename foldername/

Examples:

Move a file into a folder called myfolder:

mv file.txt foldername/

Move a file one folder up:

mv file.txt ../

Renaming Files

- You can also use **mv** to rename files within the same location

mv filename newfilename

- Example:

mv file.txt file_new.txt

Copying Files

- You can copy a file from one location to another using **cp**

cp filename foldername/

Examples:

Copy file.txt into a folder called myfolder:

cp file.txt myfolder/

Make a copy of the file (within same folder):

cp file.txt file_new.txt

Copying Folders

- You can copy an entire folder using the **-r** option (**-r** stands for recursively):

```
cp -r folder other_folder
```

Searching for Files

- You can use **find** to search for files within a directory

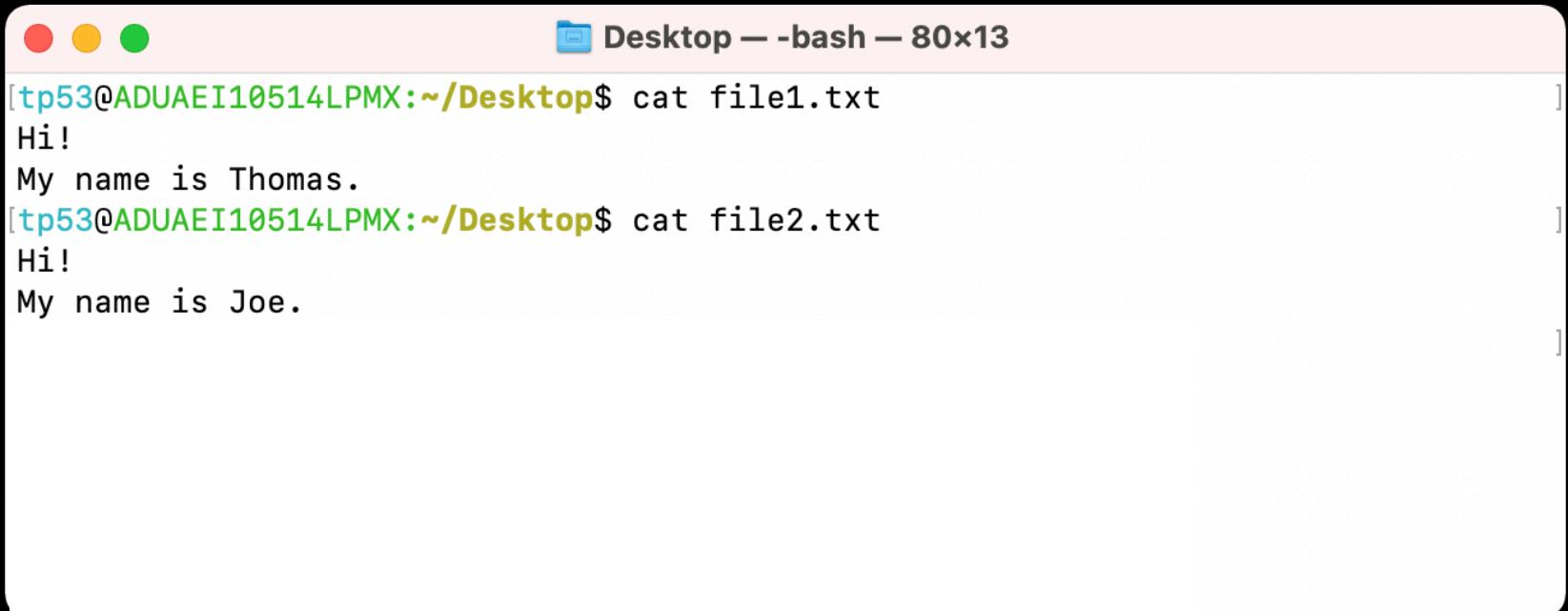
find folder –name file

- Example:
If you are searching for all .py files in your current directory (and below)

find . –name '*.py'

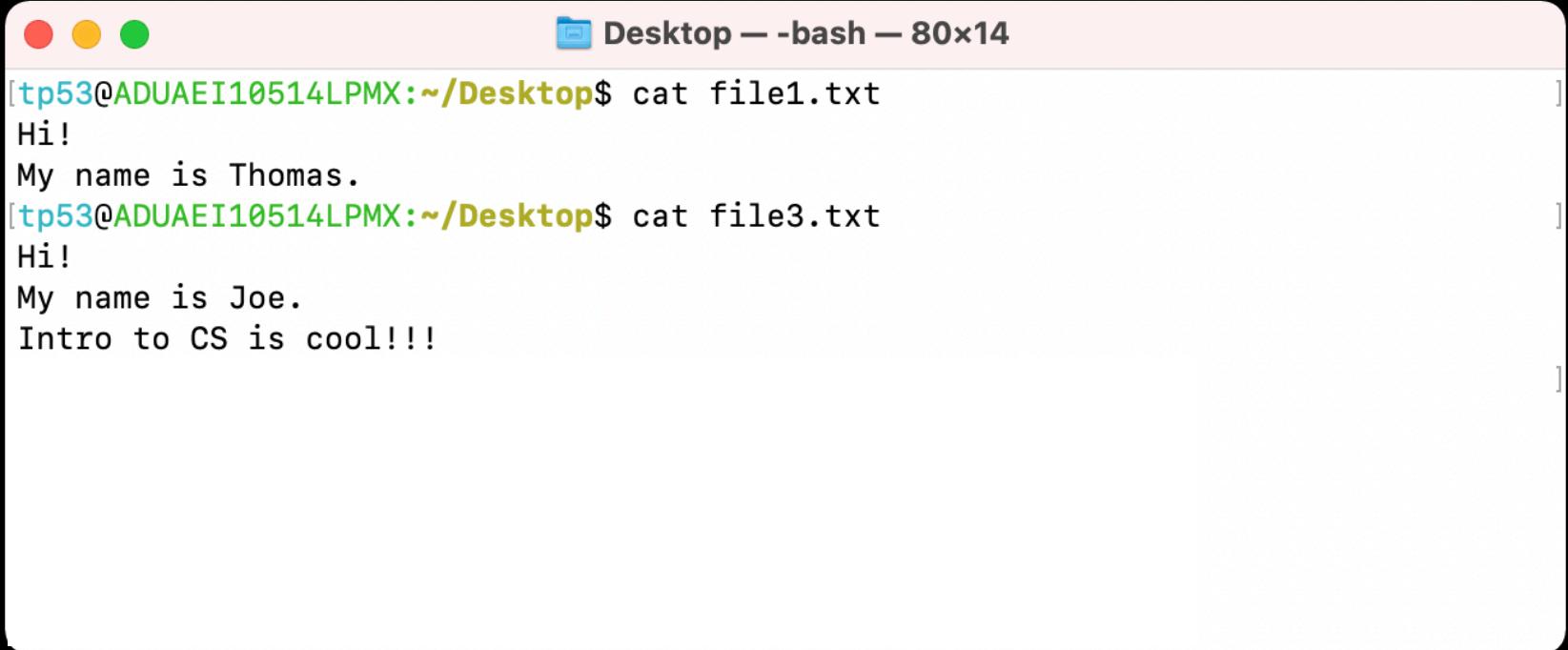
Compare text files

- You can use **diff** to compare text files and show where they differ



```
[tp53@ADUAEI10514LPMX:~/Desktop$ cat file1.txt
Hi!
My name is Thomas.
[tp53@ADUAEI10514LPMX:~/Desktop$ cat file2.txt
Hi!
My name is Joe.
```

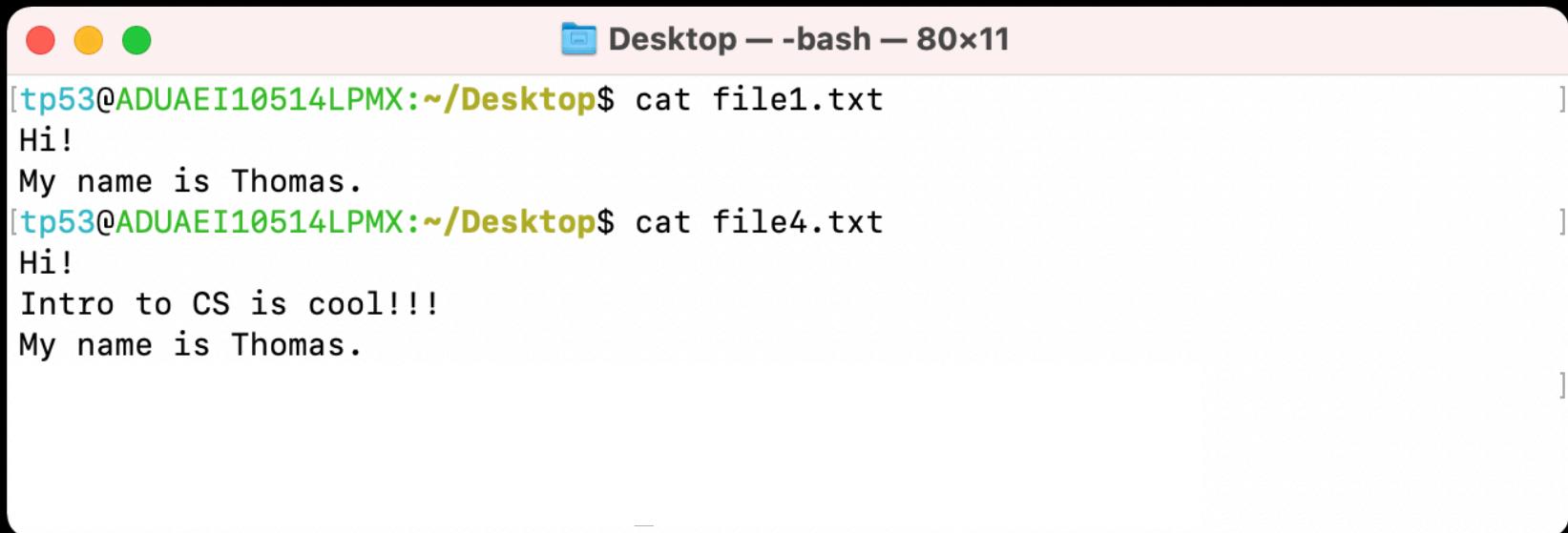
Compare text files



Desktop — bash — 80x14

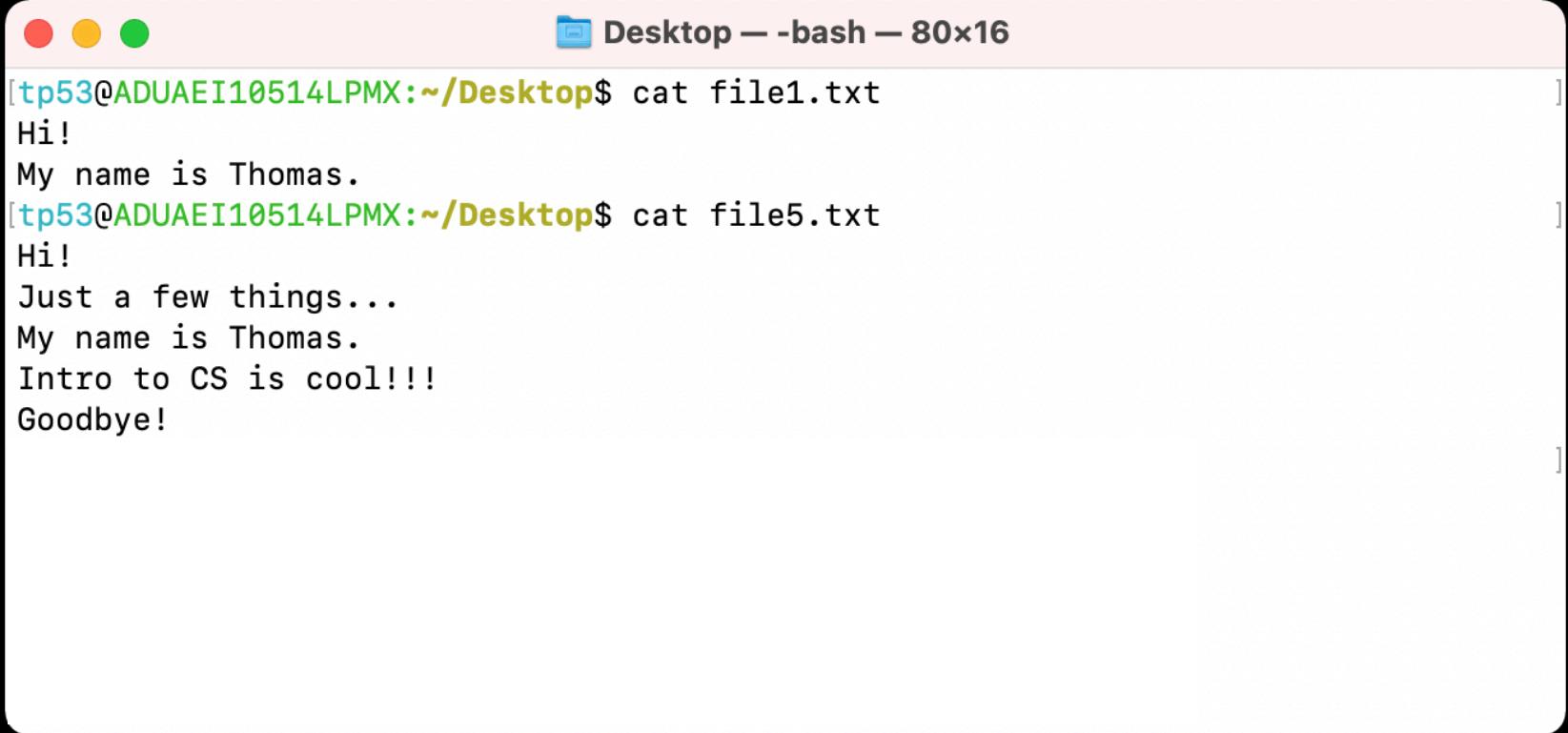
```
[tp53@ADUAEI10514LPMX:~/Desktop$ cat file1.txt
Hi!
My name is Thomas.
[tp53@ADUAEI10514LPMX:~/Desktop$ cat file3.txt
Hi!
My name is Joe.
Intro to CS is cool!!!]
```

Compare text files



```
[tp53@ADUAEI10514LPMX:~/Desktop$ cat file1.txt
Hi!
My name is Thomas.
[tp53@ADUAEI10514LPMX:~/Desktop$ cat file4.txt
Hi!
Intro to CS is cool!!!
My name is Thomas.
```

Compare text files



Desktop — bash — 80x16

```
[tp53@ADUAEI10514LPMX:~/Desktop$ cat file1.txt
Hi!
My name is Thomas.
[tp53@ADUAEI10514LPMX:~/Desktop$ cat file5.txt
Hi!
Just a few things...
My name is Thomas.
Intro to CS is cool!!!
Goodbye!
```

Redirection

- You can redirect the output of a command to a file

ls > file.txt → create a new file

ls >> file.txt → append to a file

Terminating a Running Program

- You can stop any running program in the terminal at any time by pressing

CTRL-C

Breakout session II:

Terminal/Command line

Perform the following tasks on the terminal:

1. navigate to your Desktop*
2. make a directory called **introCS**
3. change directory to the **introCS** folder
4. create a file called **test.txt** and write your name into it
5. rename **test.txt** to **another.txt**
6. make a copy of the **another.txt** file and call it **copy.txt**
7. delete the **another.txt** file
8. create subfolders inside **introCS** and call them **lecture1/ex_1**
9. move the **copy.txt** file in **introCS** into the folder **lecture1/ex_1**
10. navigate back to your Desktop
11. Have a great weekend :)

*Windows Subsystem Linux: **cd /mnt/c/Users/USERNAME/Desktop**