Command Line

Part I

Slides : burtenshaw.xyz/bootcamp/command

Goals

- open a terminal
- navigate the directory tree
- create/ delete files
- create/ delete folders
- use redirection and piping

Tip

- Devhints

Installation Check

Windows users

Ubuntu for windows

Mac and Linux users

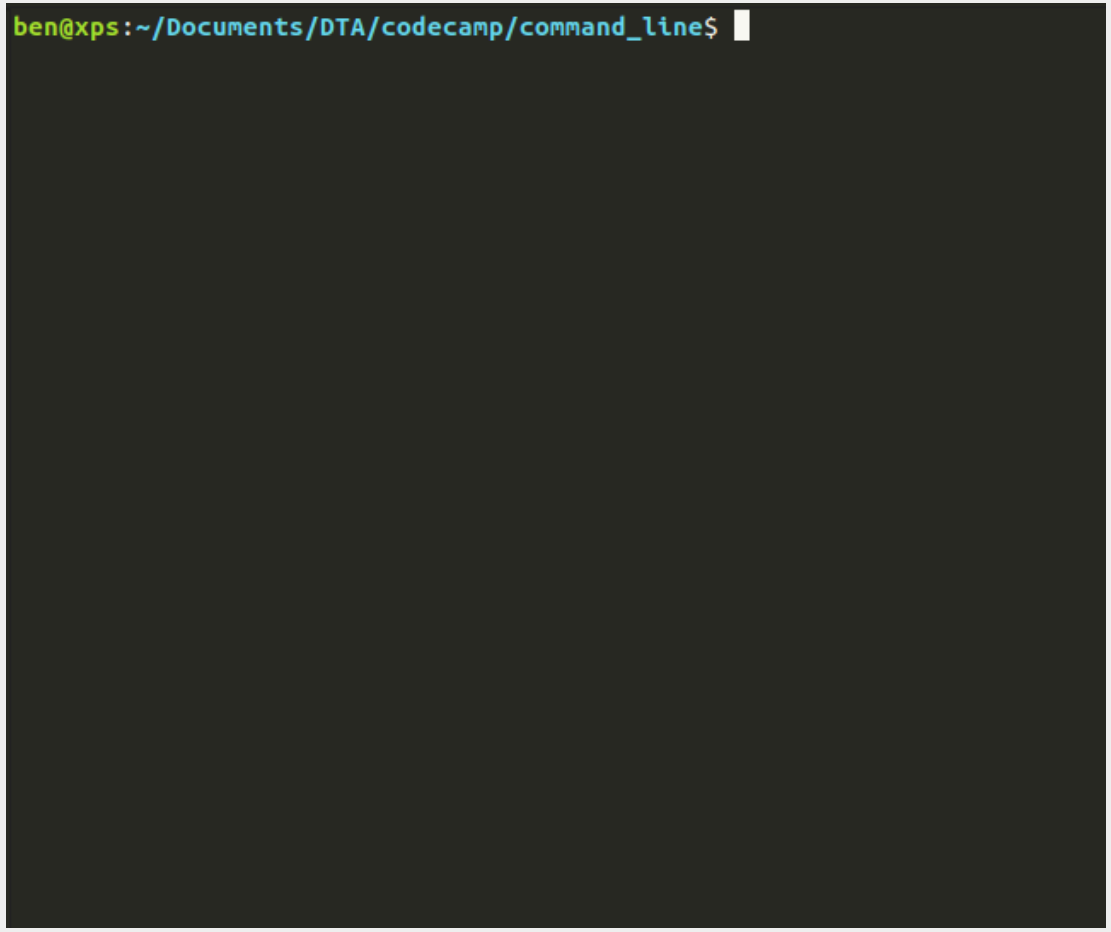
Built-in mac terminal

Last resort

Ubuntu for everyone

What is a terminal?

- A terminal is a text based interface to a program.
- It exists on every of operating system.
- We will be learning `bash`, the most common linux command line interface.



Go to your terminal emulator.

Open a Terminal

Look around

show location

```
pwd
```

list the contents of the directory

```
ls
```

Question Why terminal **emulator**?

Navigation

Move around

change directory

```
cd
```

go up the tree

```
cd ..
```

Question Why two dots?

Creating Folders and Files

Making stuff

Make a directory

```
mkdir
```

Make a file

```
touch
```

Creating Folders and Files

Reading stuff

Read a file

```
less
```

Edit text

```
nano
```

```
vi
```

Moving and manipulating files

Move a file or folder

```
mv
```

Remove a file or folder

```
rm
```

Moving and manipulating files

Copy a file

```
cp
```

Copy a file

```
cp file.txt copied_file.txt
```

Copy a folder

```
cp -r folder copied_folder
```

Try : Make a folder with files in and copy it.

Creating files using redirection

return text

```
echo "Hello World"
```

adding text to a file

```
echo "Hello file" > test_1.txt
```

Try : Experiment with returning text

Reading text from files

read the last line

```
tail
```

read a text file

```
less
```

Try : Make 3 text files with a line in each.

A Possible Solution :

```
echo "This is a test" > test_1.txt
echo "This is a second test" > test_2.txt
echo "This is a third test" > test_3.txt
```

Concatenating files

1 _ concatenate lines

```
cat
```

2 _ concatenate multiple files

```
cat test_1.txt test_2.txt test_3.txt
```

Concatenating files

3 _ add an extra line

```
echo "New line!" >> combined.txt
```

4 _ return the extra line

```
tail combined.txt
```

Redirection

add a string to a file

```
echo "Hello world" > output.txt
```

append a string to a file

```
echo "Hello world" >> output.txt
```

directing command output into a file

```
ls > list_of_files.txt
```

Python on the command line

create a python file

```
echo "print('in a python file')" > python_file.py
```

execute a python file

```
python python_file.py
```

direct the result of a python file to another file

```
python python_file.py > output_file.txt
```

Wildcards

any file

```
ls *
```

a file with a specific extension

```
ls * .txt
```

a file containing a sequence of characters

```
ls *_1*
```

Wildcards

Try : Describe the below command

```
cat t* > combined.txt
```

Word count

count the contents of a file

```
wc combined.txt
wc -w combined.txt # count words
wc -l combined.txt # count lines
```

Try How many lines are in `combined.txt`?

Try Count all the lines in all the files

1 _ list all the files and store it in a temporary file

```
ls ~ > file_list.txt
```

2 _ count all of the lines in that text file

```
wc -l file_list.txt
```

3 _ remove the temporary file

```
rm file_list.txt
```

A bit of plumbing

Piping

More elegantly

```
ls | wc -l
```

Try : Test and explain the above command

A bit of plumbing

Piping

Showing unique lines

```
cat combined.txt | uniq | less
```

Searching for text

```
ls | grep test
```

Searching in files

```
cat *.txt | grep hello
```

A piping exercise

Piping and Redirecting

write a search result to a file.

```
ls | grep test > output.txt
```

search lines in files for 'hello

```
cat *.txt | grep hello > output.txt
```

Try : Experiment with searching

Bonus Exercise

Adapt `frequency.py` so that it creates a character frequency distribution of a text file, and writes it to a file named `output.txt`.

This command should run it.

```
echo text.txt | python frequency.py > output.txt
```

Resources

- A dev hints cheatsheet
- Terminal Tutorial
- Ubuntu for everyone
- Ubuntu for windows