The Command Line

CST 205

Early days

- 1980s computers only had a so-called **terminal** to control them

- Graphical User Interfaces would come later

Why do we still care?

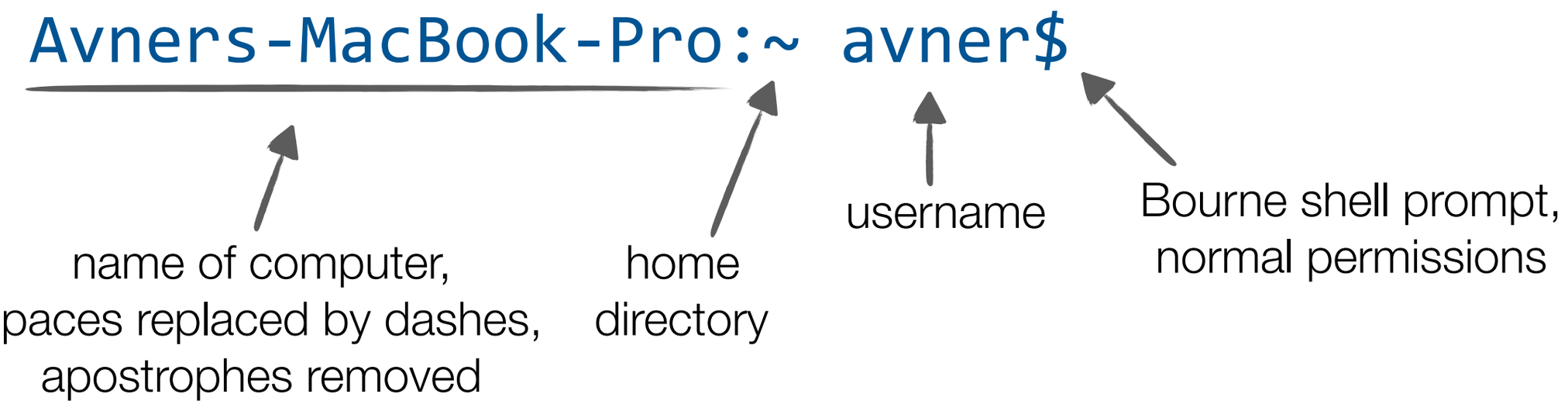
- How most professional developers work
- Speeds up development
- Allows more fine-grained control

A few more details

- Text-based
- Don’t use mouse/trackpad, just keyboard
- Terminal *emulators*
  - MacOS —> Terminal 
  - Windows —> PowerShell 

Open up your OS's terminal emulator

- macOS



- Windows

`cd`

- The `cd` command allows you to **c**hange **d**irectory

  `cd Desktop`

  - You have now navigated to your Desktop

  `cd ..`

  - Takes you *back* (or *up*) a directory

- You can find out where you are with `pwd`

`mkdir`

- The mkdir command will create (*make*) a directory.

```
mkdir 205_practice
```

- **1st note:** `mkdir` does not place you in the newly-created directory.

- **2nd note:** We are creating a new directory *relative* to our current directory.

Create files

- macOS

```
touch my_file.txt
touch ~/Desktop/hello.py
```

- Windows PowerShell

```
New-Item my_file.txt
New-Item ~/Desktop/hello.py
```

Path shortcuts

- The ~ refers to your home directory
- The / refers to your root directory
- Use ~ and / to get absolute paths

- The following do the same thing:

```
cd ~/Desktop/205_practice
```

vs.

```
cd /Users/[username]/Desktop/205_practice/
cd '/Users/[username]/Desktop/205_practice/'
```

macOS

Windows PowerShell

A few more commands

- `mv` allows you to move or rename files
- `rm` allows you to remove files
- `ls` allows you to list (or see) the files and/or folders in a directory
    - On macOS, `ls -a` will show hidden files and folders
- `cp` allows you to copy files

Lastly…

- Speed up your work by using **tab** to autocomplete

- View the contents of files in both PowerShell and Terminal with the `cat` command.

- For example, given a file named `.top_secret.txt`:

```
cat .top_secret.txt
```