Shells

- A program that provides interface between us and operating system.
  - Operating system controls resources in computer
  - We interact with OS through shell
  - It is program like any other
  - Interpret command line & Run scripts
  - **script** - interpreted text program
  - There are multiple shells to choose from - zsh, bash etc.
- **Keyboard shortcuts**
  - **control + a** = beginning of line
  - **control + e** = end of line
  - cursor can be any where in line for command to execute (just press return)
  - **right option + left Arrow / right Arrow** = jump words
  - **control + d** = delete one letter to right
  - **control + c** = kill the program
  - **control + l** = clear history i.e . everything upwards from current line
  - **control + w** = remove one word from left of cursor
  - **control + u** - get rid of everything on particular line
- **Variables**
  - **varName=value**
  - **$var_name** - used to refer to variable (**$** - indicates, we are referring to a variable)
  - we can also use quotes like "$var_name" will treat words as one string and not remove spaces (not an issue with mac zsh)
  - **set** command - we can see all variables
    - search for variable = /var_name
  - to use particular shell (bash | zsh) we can write the keyword and start using that shell.
  - **exit** - exit out of shell
  - **export varName** - we can export variable across shells (making variable as environment variable)
  - **unset varName** - unset env. variable
  - **$PATH** - all the directories to our current position
    - folder are delimited by colon (:)
    - we can append or prepend on our $PATH variable
    - appended directories will be searched after everything else in PATH variable
    - prepended directories will be searched before everything in PATH variable
  - **Child shell** inherits variables from parent shell (shell created inside another shell)
- Commands not built in will exists as file in file-system
- **echo $SHELL** / **echo $0** - know which shell we are in or using
- **pico filename** - we can open a file in pico text editor
- **tail filename** - prints last few lines of file
- There are certain scripts that are called once we login into system, some aren't.
- ZDOTDOR - Home Directory

- For making PATH environment variable change permanently we need to make changes in **.zshrc**
- We can write a command and unix system will try to execute that command
- There are builtin commands that are not in fileSystem.
    - These are commands that are built into shell itself
- **Job Control**
    - **cntrl +z** = get out of program without ending it
    - **jobs** - running jobs (stopped programs in between)
    - **fg number**- specific number of program we want to bring to fore ground and get it running aga
    - [1]**+** - plus sign indicates most recent program we exit out of without ending it
        - if we want to get back to most recent exited program, we can just use '**fg**'
    - **cntrl + c** - will not work in exiting on editors in shell
    - **&** - start a program in background
- **Kill Processes**
    - process id - maintained by entire unix system
    - **ps** - commands / processes running at that terminal
        - TTY (Tell a type) - what terminal we are on
    - **kill -9 process_id** (9 - sigkill)
- **Redirection & Piping**
    - Standard output = screen (default)
    - Redirect - to files (monitor, files)
        - '**>**' - redirect (default behaviour to override file contents) (only redirects STDOUT not STDERR)
            - '**2>**' - redirects both STDOUT and STDERR
            - **2> /dev/null** - sending output to nowhere (for STDERR)
        - '**>>**' - redirect (default behaviour to append to file)
        - '**<**' - redirecting input
        - '**<<** ' - multiline (helps in here document)
    - Piping - output of one command into input of other command
        - ' **|** '
        - we can use multiple pipe commands