

Viewing Directories

pwd prints the "working directory" or your current location in the filesystem.

```
$ pwd
/home/james
```

ls lists the contents of a directory. If no arguments are given, it will list the current directory's contents.

```
$ ls
code/  courses/  hello.txt
```

Navigation

cd allows you to change your current directory. It is typically used with an *argument*, either the directory name to be navigated to, or one of the following:

```
# navigate to {dirname}
$ cd {dirname}

# without an argument, will navigate to home
$ cd
# can also use tilde (~)
$ cd ~

# with a dash (-), go back to prior directory
$ cd -

# two periods (..) means up or parent directory
$ cd ..

# you can use these as part of paths as well
$ cd ../courses/30121
```

Viewing & Editing Files

cat lists the contents of a file directly to the terminal.

```
$ cat file.txt
This is the text within file.txt.
```

code will open the given file or directory in VS Code.

```
$ code hello.py
$ code homework1
```

less will allow you to view longer files that would require scrolling to view.

```
$ less longer-file.txt
```

Making Directories

mkdir creates a new empty directory.

```
$ mkdir newdir
```

If you want to create several nested directories at once, you can pass **-p** before the target name and all intermediate directories will be created as needed.

```
mkdir -p courses/2025/capp30122
```

Copying & Moving Files

cp {source} {destination} copies a file from {source} to {destination}.

```
$ cp hello.txt copy.txt
```

mv {source} {destination} moves a file from {source} to {destination}.

- If {destination} is a filename, the file will be renamed.
- If {destination} is a directory, the file will be moved there.

```
# renames file
$ mv notes.txt old-notes.txt

# moves file into archive directory
$ mv old-notes.txt archive/
```

Removing Files and Directories

rm removes files.

```
$ rm unused.txt
```

rm -r will remove a directory and all contents *recursively*.

```
$ rm -f failed-experiment
```

Remember: there is no undo, so be extremely careful, especially with **rm -r**.

Keyboard Shortcuts

TAB	Complete current word being typed.
Ctrl-P or Up Arrow	Scroll back in command history.
Ctrl-N or Down Arrow	Scroll forward in command history.
Ctrl-A	Move cursor to beginning of line.
Ctrl-E	Move cursor to end of line.
Ctrl-U	Clear prompt to beginning of line.
Ctrl-K	Clear prompt to end of line.
Ctrl-L	Clear prompt of all output.
Ctrl-C	Interrupt/stop currently running command.

Getting Help

man shows a manual page for a given command.

```
$ man ls
```

Within a manual page, you can type **/keyword** to search for a keyword. Type **q** to exit.

tlldr will show a few examples of the command.

```
$ tldr ls
```

Unlike the other commands here, **tlldr** is not installed on every system.