

Navigating the filesystem

If you need help at any time, put your **red** sticky note on the back of your laptop. When you've finished the steps on the *front* of this page, put your **green** sticky note on the back of your laptop. Then, you can turn the page over and try out some of the more advanced tricks on the back while you wait for the rest of the group to be ready.

Commands in this lesson: `pwd`, `ls`, `cd`, `mkdir`

Basics

First, check where you are currently located in the filesystem with the `pwd` (“**p**rint **w**orking **d**irectory”) command:

```
pwd
```

Next, **list** the contents of the directory you are in:

```
ls
```

To create a new directory inside our current directory, run `mkdir` and specify a name for the new directory, like

```
mkdir new
```

You can **change** **d**irectory by running `cd` and specifying the directory you want to change to. For example, to change to the directory you've just created, run

```
cd new
```

and then use

```
pwd
```

again to verify your current working directory.

Relative and absolute paths

You may have noticed that when you run the `pwd` command, it gives you a full path with several directory names separated by a `/` character. This is a *full path*. For example, after running the commands above, I would see the following output for `pwd`:

```
/home/ff524/new
```

When you run commands that involve a file or directory, you can always give a full path, which starts with a `/` and contains the entire directory tree up until the file or directory you are interested in. For example, you can run

```
cd /home/ff524
```

to return to your home directory. Alternatively, you can give a path that is *relative* to the directory you are in. For example, when I am inside my home directory (`/home/ff524` - yours will be different), which has a directory called `new` inside it, I can navigate into the `new` directory with a relative path:

```
cd new
```

or the absolute path:

```
cd /home/ff524/new
```

Shortcuts

- Running `cd` with no argument takes you to your home directory.
- The shorthand `..` refers to “the directory that is one level higher” (can be used with `cd` and with other commands).
- The shorthand `~` refers to the current user’s home directory (can be used with `cd` and with other commands).
- After navigating to a new directory with `cd`, you can then use `cd -` to return to the directory you were in previously.

Try these commands. Before and after each `cd` command, run `pwd` to see where you have started and where you ended up after running the command.

```
cd      # takes you to your home directory
cd ..   # takes you one directory "higher" from where you were before
cd ~    # takes you to your home directory
cd ../.. # takes you two directories "higher" from where you were before
cd -    # takes you to the directory you were in before the last time you ran "cd"
```