UNIX and the Command Line

Childhood Cancer Data Lab

Why learn UNIX and use the command line?

- The command line is, in part, the computational researcher's “lab notebook”
- You can capture small data manipulation steps that are normally not recorded to make research reproducible*
    - Manual manipulation of data files is challenging to troubleshoot, review, or improve*
    - *Don’t take the “small steps” for granted! They matter a lot.*
- With UNIX, you can record all your precise steps - you have typed them rather than "point-and-clicked" them
    - Even better, you can run these steps as a *script*!

*https://swcarpentry.github.io/shell-novice/guide/

Why learn UNIX and use the command line?

- Allows you to automate repetitive tasks*
  - Much less risk of human error, and *much easier* to re-run
  - You will thank yourself for putting in the time up front to write scripts that automate!

- Most bioinformatics/NGS tools are command-line only
- Take your science back into your own hands!
  - Running these tools yourself removes the mystery of what your core/bioinformatics collaborator is doing

*https://swcarpentry.github.io/shell-novice/guide/

Terminology

- UNIX is an *operating system* which features a hierarchical file system and provides commands (little computer programs) to help interact with it
  - We are using the term “UNIX” to refer to all UNIX-like systems like Linux and macOS

- The shell allows us to interact with UNIX and run those commands
  - We access the shell through the terminal (also known as command line)
  - There are many different shells out there, including BASH and zsh (pronounced "zee shell")*

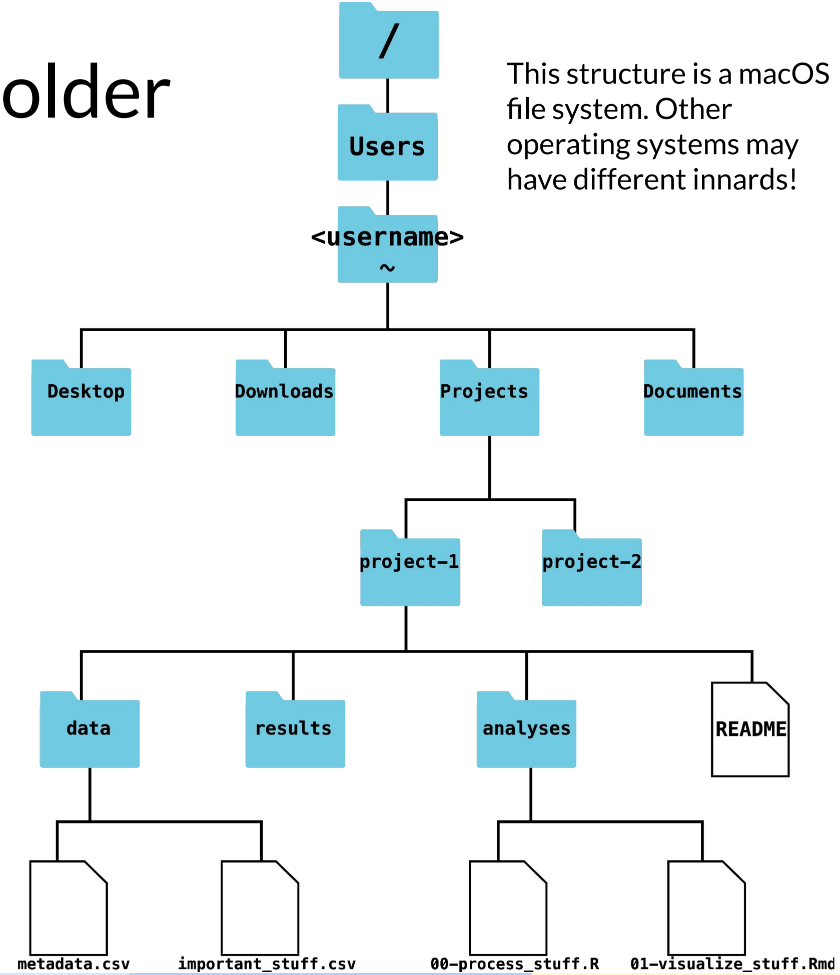
- We use shell scripting languages to write code that the shell can interpret and execute
  - Extremely creatively called, for example, BASH scripting or zsh scripting

Directory is a fancy word for folder

- Files and directories in most operating systems (including UNIX-like!) are organized hierarchically

- Every file/directory has a specific address, or **path**, in the hierarchy

- The top-level directory is known as **root** and denoted **/**
- Your home directory can be referred to with **~**



This structure is a macOS file system. Other operating systems may have different innards!

Relative vs. absolute paths

- The path is the address to a file or directory on your computer. There are two ways to formulate paths:
    - The absolute (or full) path represents the path of a given directory or file, *beginning at the root directory*
        - Because they begin at the root, absolute paths always start with /
    - The relative path represents the path of a given directory or file, beginning at (i.e. *relative to*) to the working/current directory

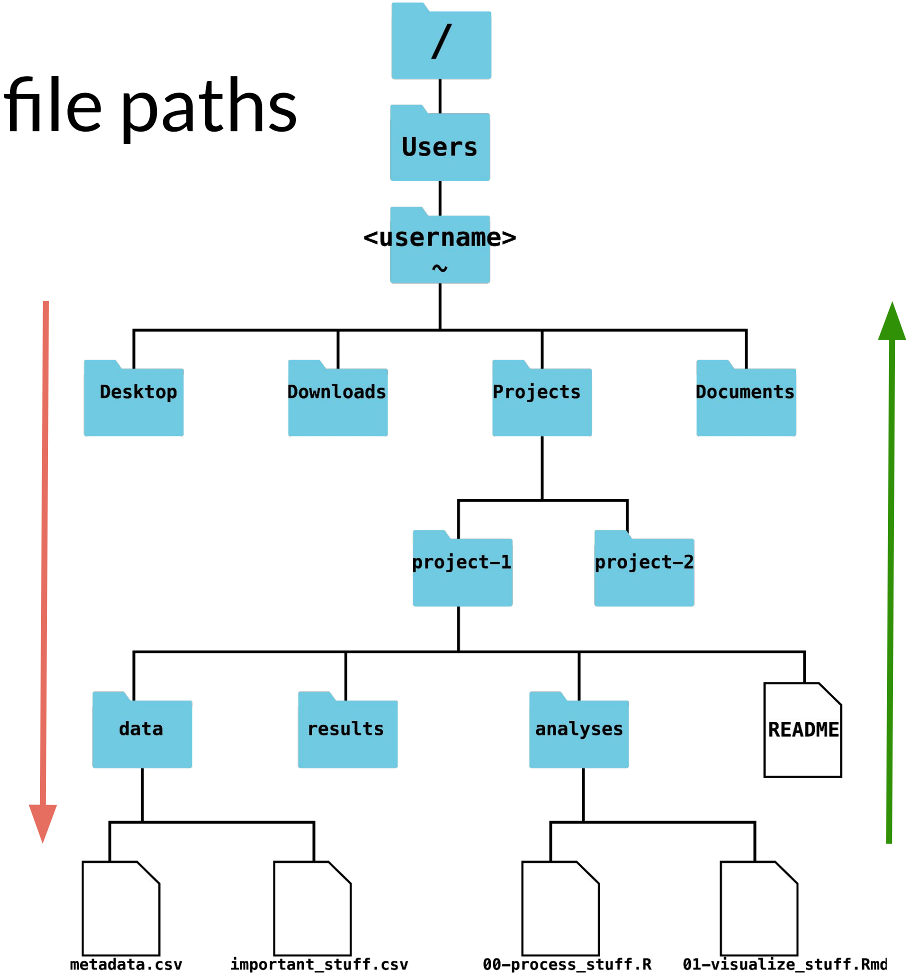
Paths analogy: How do I get to the store?

- The full/absolute path: 15 Main St, Anytown, Anystate, zip code 12345
  - If you have this address, you can find your way to this store from anywhere in the world

- The relative path: Make a left, walk 5 blocks, make a right, and then the store will be on your left.
  - With these directions, you can only find your way to this store from the exact location where you started (i.e, relative to your starting location)

Writing out file paths

We move forwards (down) in the hierarchy with a forward slash /



We move backwards (up) in the hierarchy with two dots . .

What is the the full path to 00-process-stuff.R?

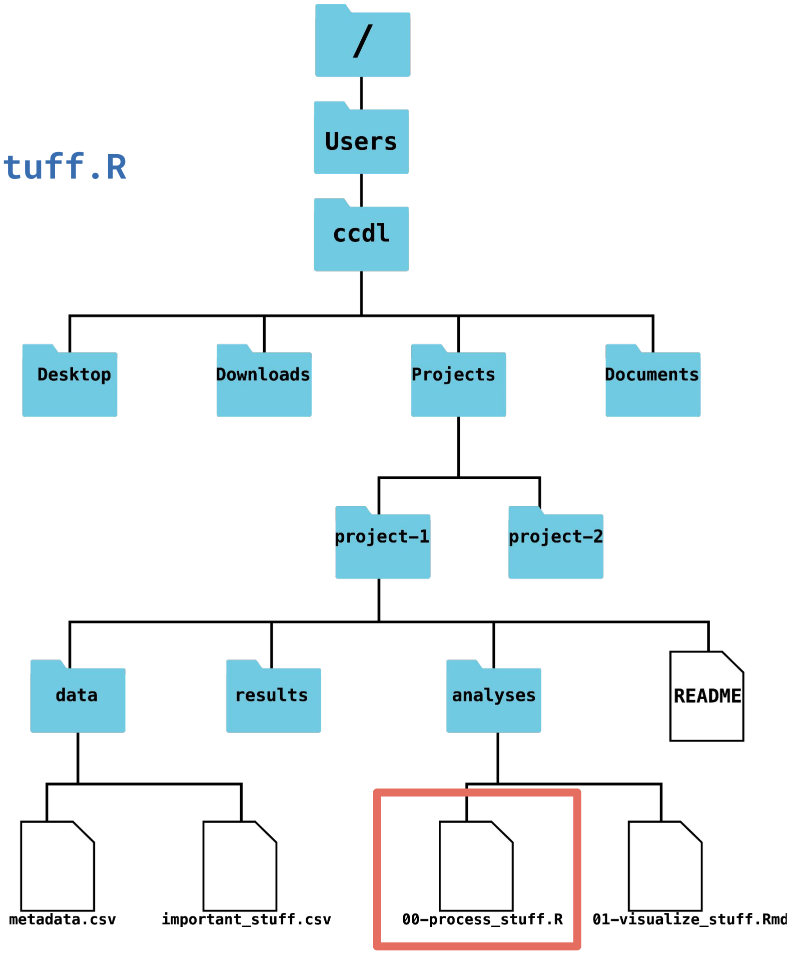
```
/Users/ccdl/Projects/project-1/analyses/00-process-stuff.R
```

What is the the relative path to 00-process-stuff.R, starting from ~/Projects?

```
project-1/analyses/00-process-stuff.R
```

What is the the relative path to 00-process-stuff.R, starting from data?

```
../analyses/00-process-stuff.R
```



Extra practice forming paths

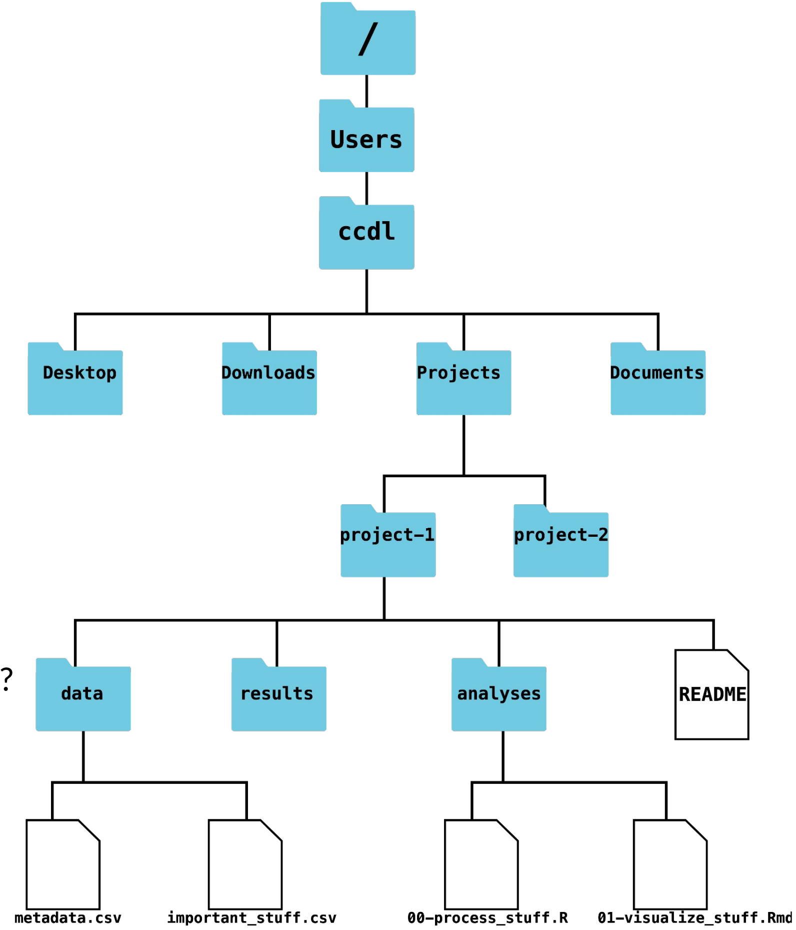
What is the **absolute path** to the project-1 directory?

What is the **relative path** to project-1 from the home directory?

What is the **relative path** to important_stuff.csv from the home directory?

What is the **relative path** to important_stuff.csv from the **analyses directory**?

What is the **relative path** to important_stuff.csv from the **data directory**?



Extra practice forming paths

What is the **absolute path** to the project-1 directory?

`/Users/ccdl/Projects/project-1`

What is the **relative path** to project-1 from the home directory?

`Projects/project-1`

What is the **relative path** to important_stuff.csv from the home directory?

`Projects/project-1/data/important_stuff.csv`

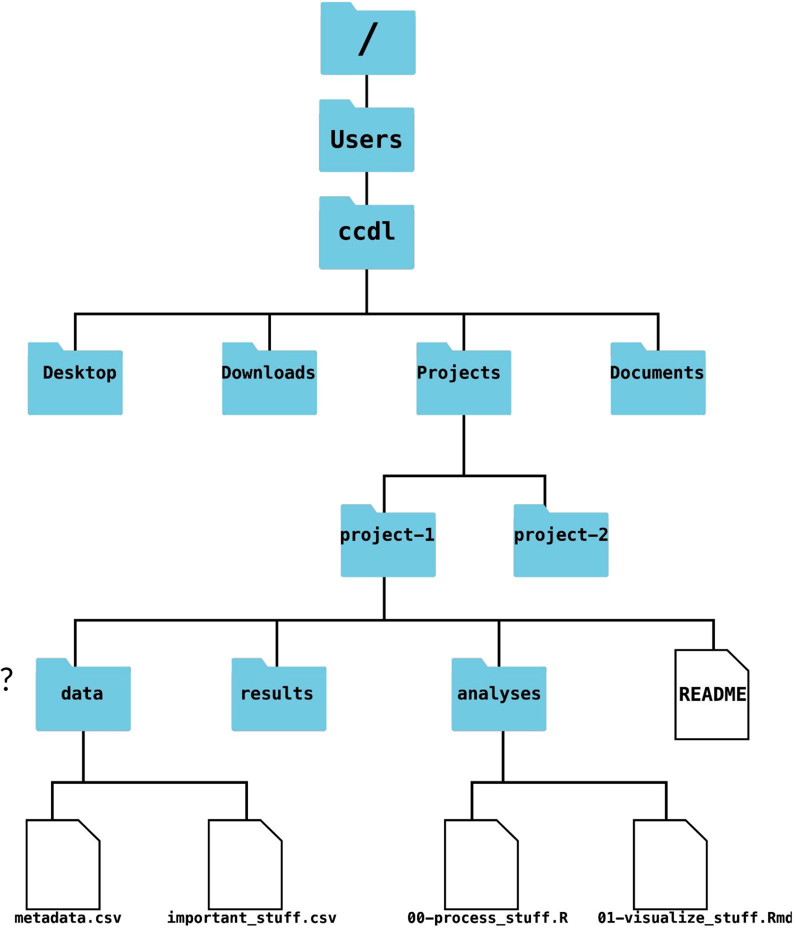
What is the **relative path** to important_stuff.csv from the **analyses directory**?

`../data/important_stuff.csv`

What is the **relative path** to important_stuff.csv from the **data directory**?

`important_stuff.csv`

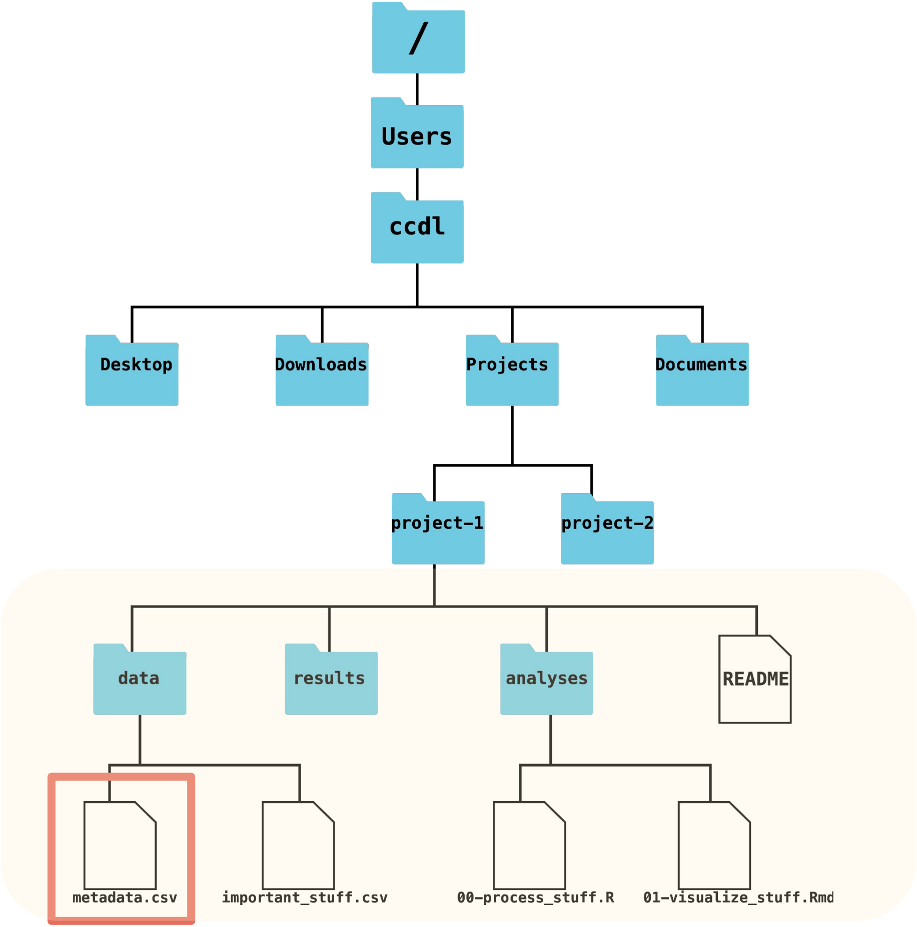
`./important_stuff.csv`

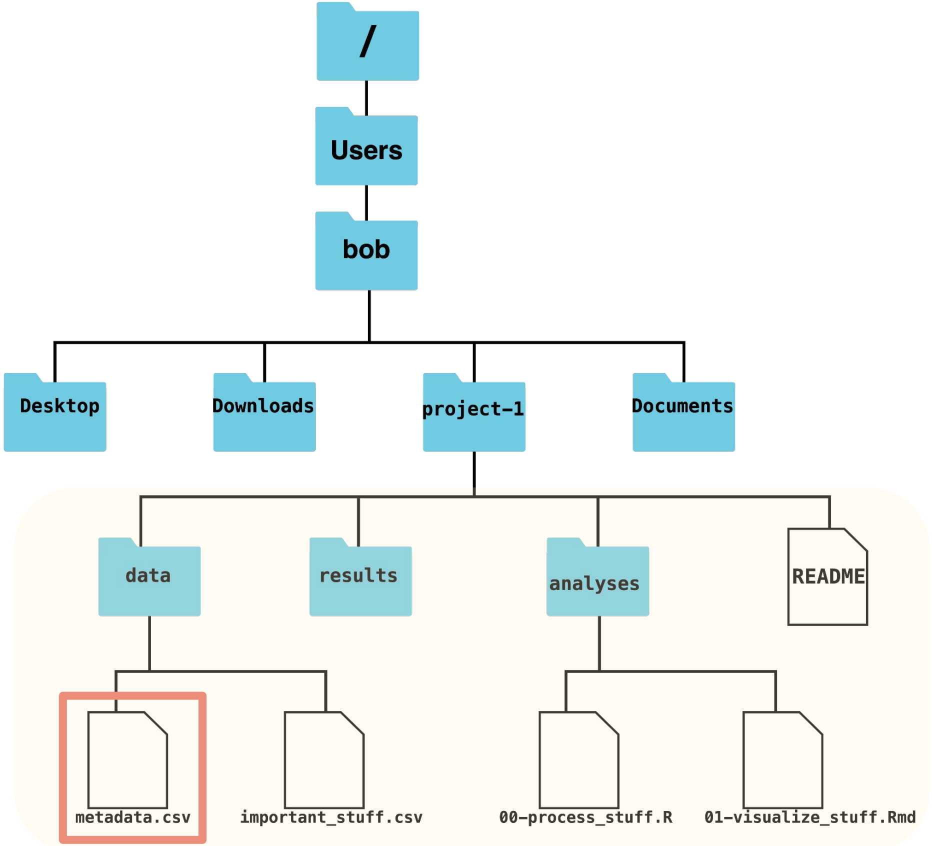


We prefer relative paths! Why do you think?

Consider each computer below.

What is the **absolute path** to `metadata.csv`?
What is the **relative path** to `metadata.csv` from `analyses`?





Let's try out some UNIX commands

- All UNIX commands are actually little computer programs
  - The behavior of UNIX commands can also be modified with *flags*

Command	**What it means**	**Why use it**
`pwd`	**P**rint **w**orking **d**irectory	Figure out where you are on your computer?
`cd`	**C**hange **d**irectory	Move from directory A → B in your computer
`ls`	**L**i**s**t	List contents of a directory

- Note that UNIX is *case sensitive!* CD is not `cd`