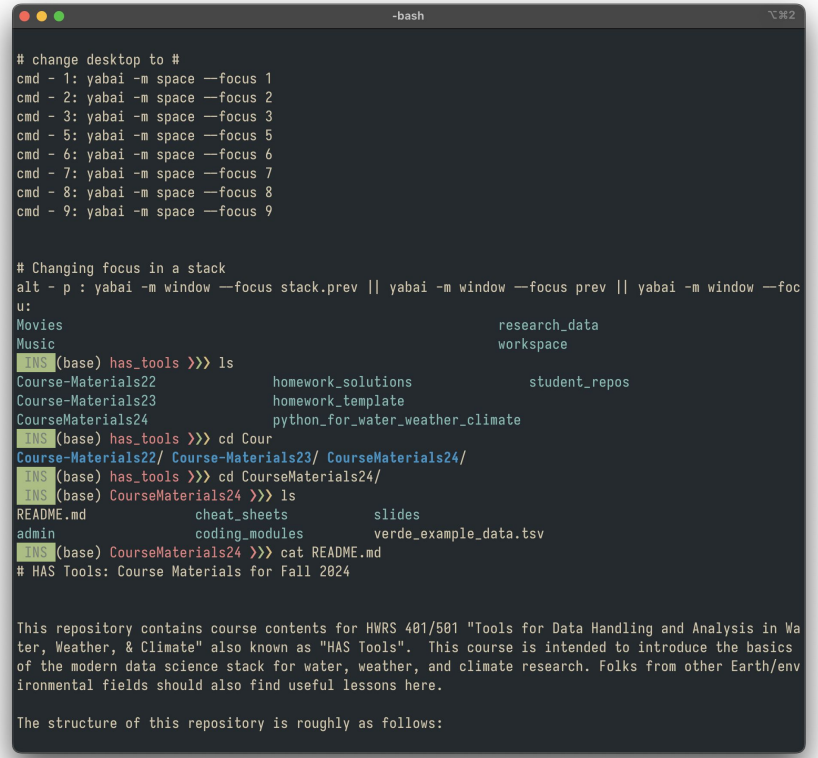
HAS Tools:

The command line

October 30, 2024

What is the command line, and how do we use it?

- The command line is a text interface for typing commands to the computer.
- The terminal is a program that lets you use the command line.
- Our codespaces have a terminal built-in
- Bash is a Unix shell and command language.
- The shell is a program that interprets commands.
- Bash is the default shell in most Linux distributions and macOS.



Why learn how to use the command line?

- Automation: The command line makes it easy to automate repetitive tasks and processes.

Why learn how to use the command line?

- Automation: The command line makes it easy to automate repetitive tasks and processes.
- Portability: Easy to transfer your processes/scripts from one computer to another.

Why learn how to use the command line?

- Automation: The command line makes it easy to automate repetitive tasks and processes.
- Portability: Easy to transfer your processes/scripts from one computer to another.
- Flexibility: The command line comes with many tools and can be customized to suit almost any task.

Why learn how to use the command line?

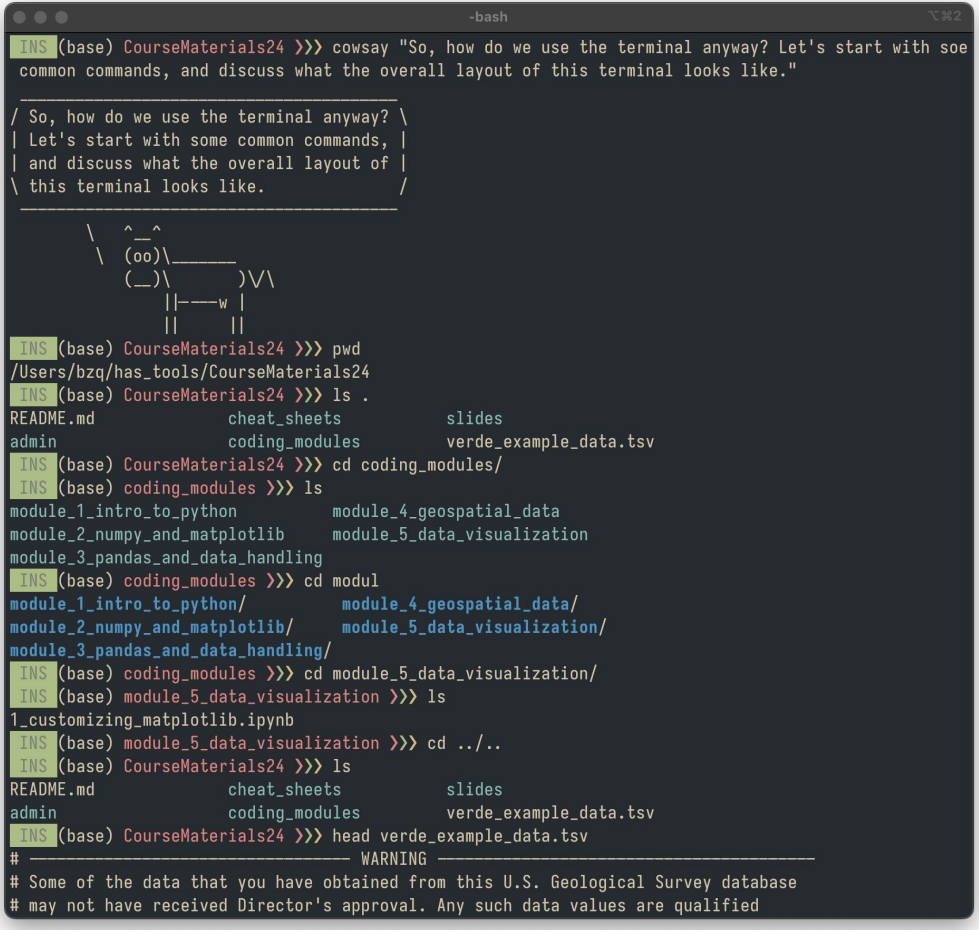
- Automation: The command line makes it easy to automate repetitive tasks and processes.
- Portability: Easy to transfer your processes/scripts from one computer to another.
- Flexibility: The command line comes with many tools and can be customized to suit almost any task.
- Integration: The command line makes it easy to combine multiple tools. Shell scripts are the “glue” of scientific computing!

Why learn how to use the command line?

- Automation: The command line makes it easy to automate repetitive tasks and processes.
- Portability: Easy to transfer your processes/scripts from one computer to another.
- Flexibility: The command line comes with many tools and can be customized to suit almost any task.
- Integration: The command line makes it easy to combine multiple tools. Shell scripts are the “glue” of scientific computing!
- Make you look cool: People will think you are an elite hacker when they see you using the command line.

Quick overview

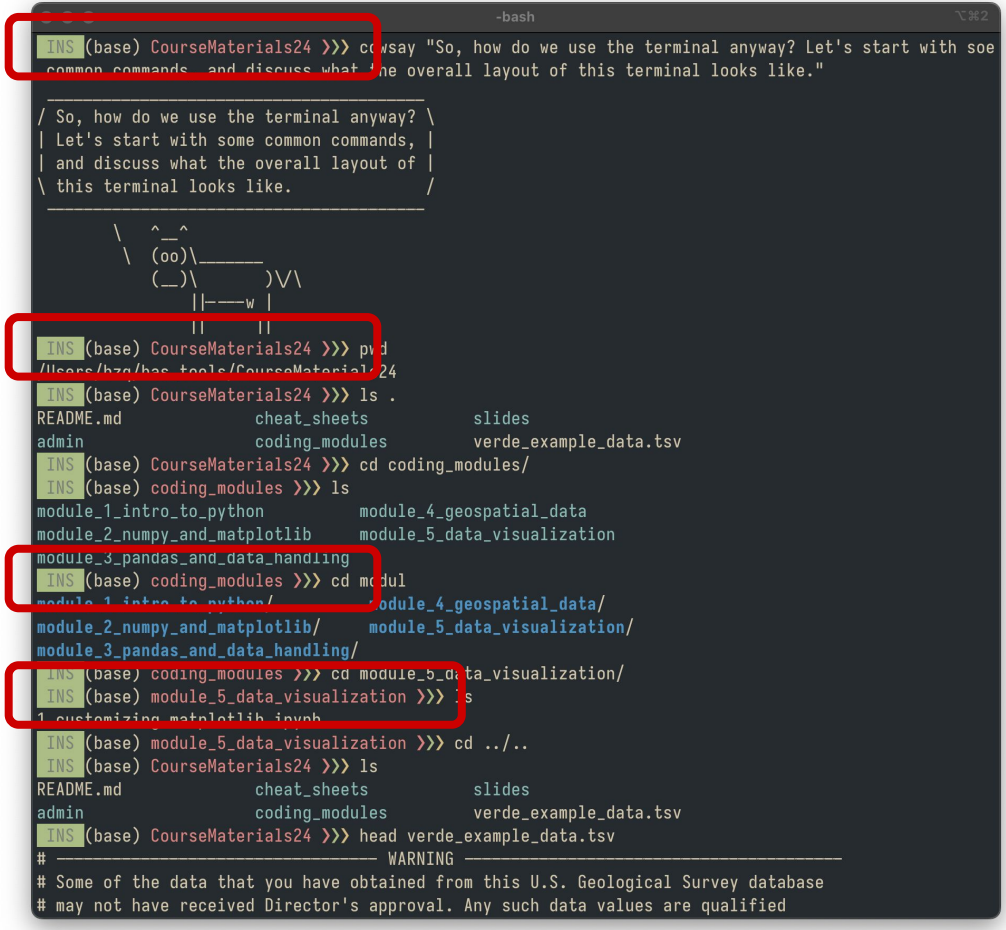
To begin with, let’s just get familiar with the layout and progression of the terminal interface, commonly called at text user interface (TUI)



Quick overview

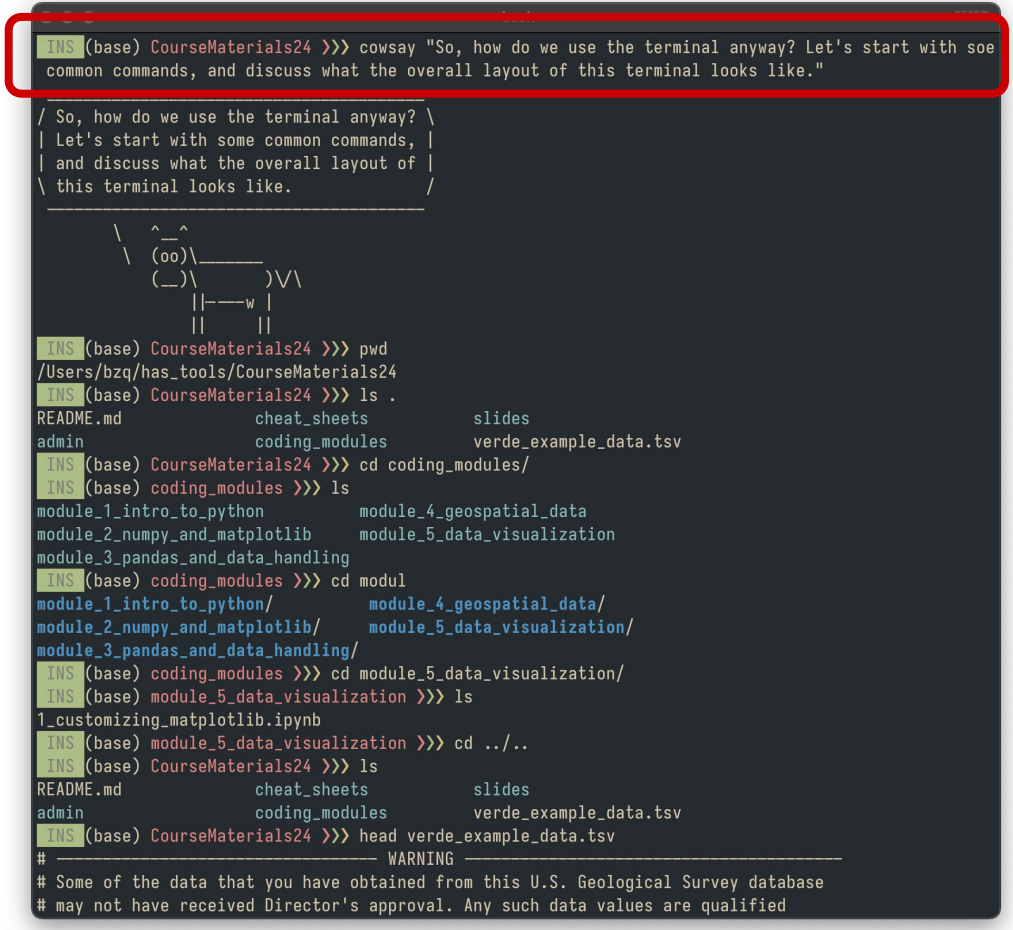
This repeated pattern is the “prompt”. It contains a couple of pieces, and mine is customized here.

Anything to the left of the ❱❱❱ is part of the prompt, anything to the right is part of the command to be run



Quick overview

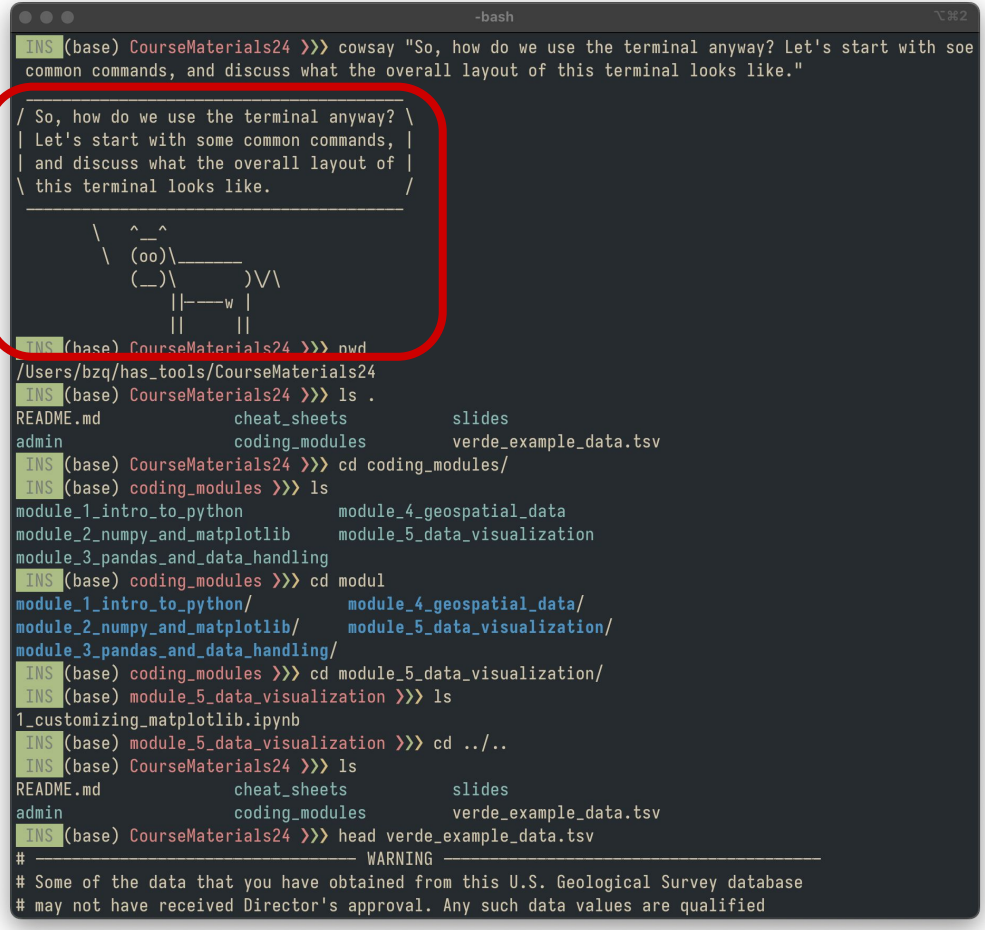
In this example I start with a “cowsay” command, which isn’t installed in the codespaces, but is fun



Quick overview

In this example I start with a “cowsay” command, which isn’t installed in the codespaces, but is fun

When run, it makes the cow say the message I typed!

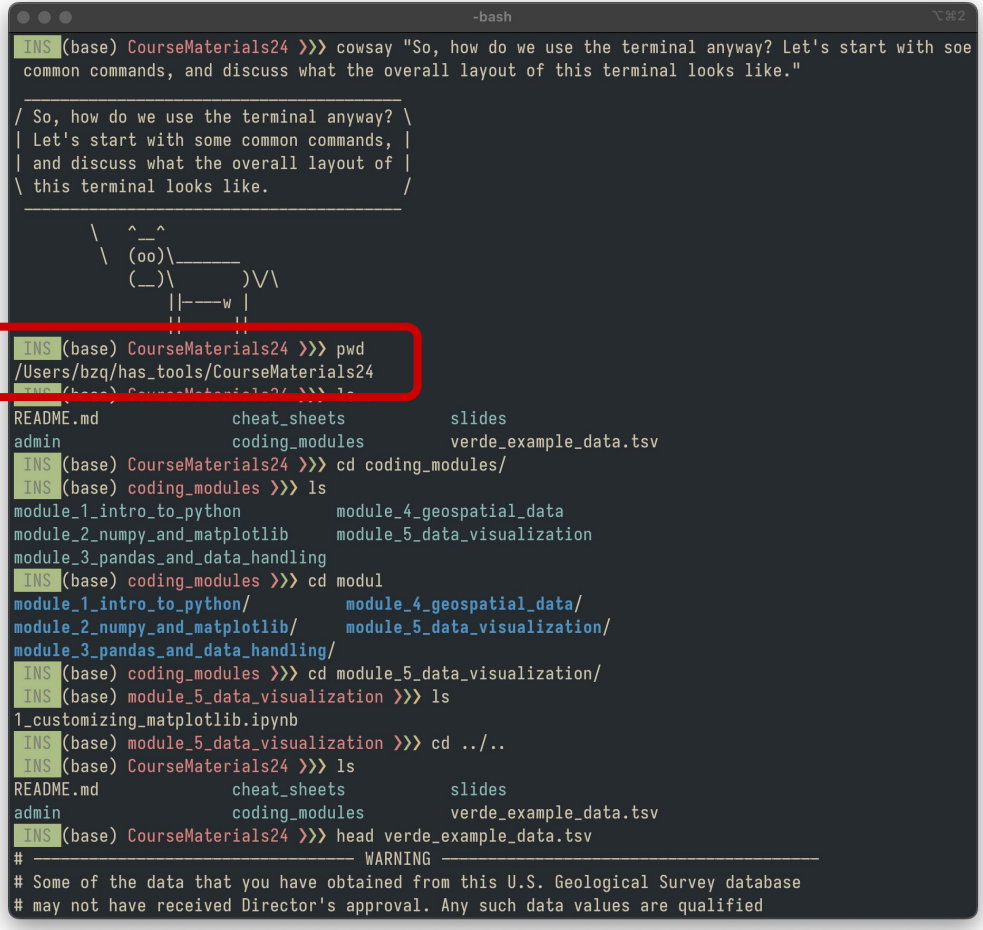


Quick overview

The first real command we will use is `pwd`, which stands for “print working directory”

It just shows you where on the system you currently are.

I am on my laptop, in the CourseMaterials24 repo

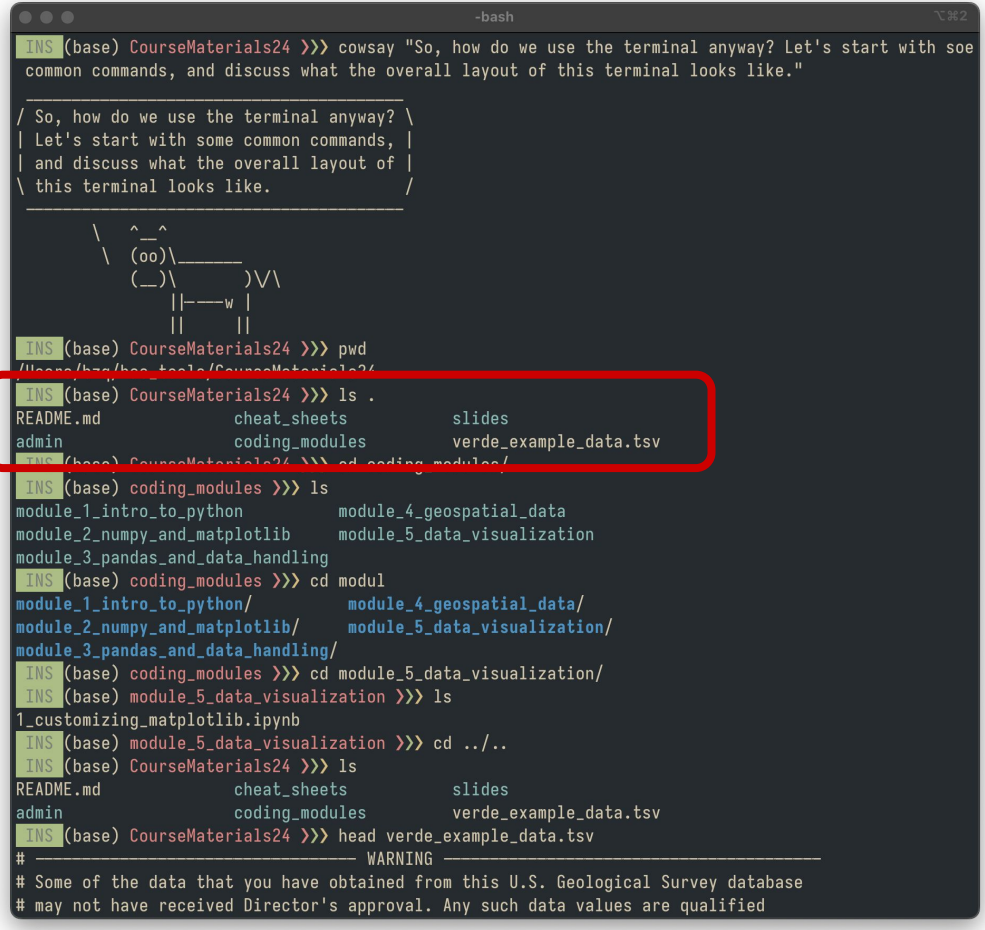


Quick overview

If I want to know what's in my current directory/folder, I can use the `ls` command.

It just lists out all of the files and folders inside.

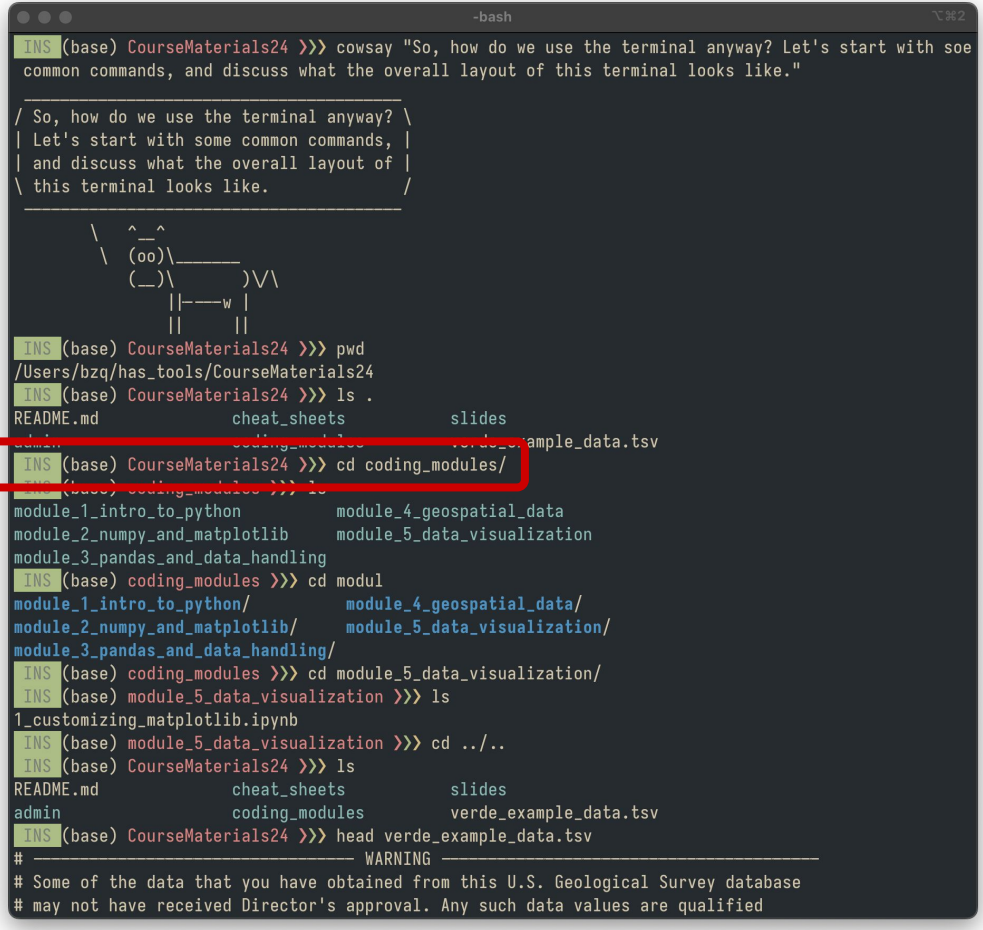
Note folders here are highlighted in blue and regular files are just white



Quick overview

If you want to move to a new location you can use the `cd` command to “change directory”

Here I go into the `coding_modules` directory

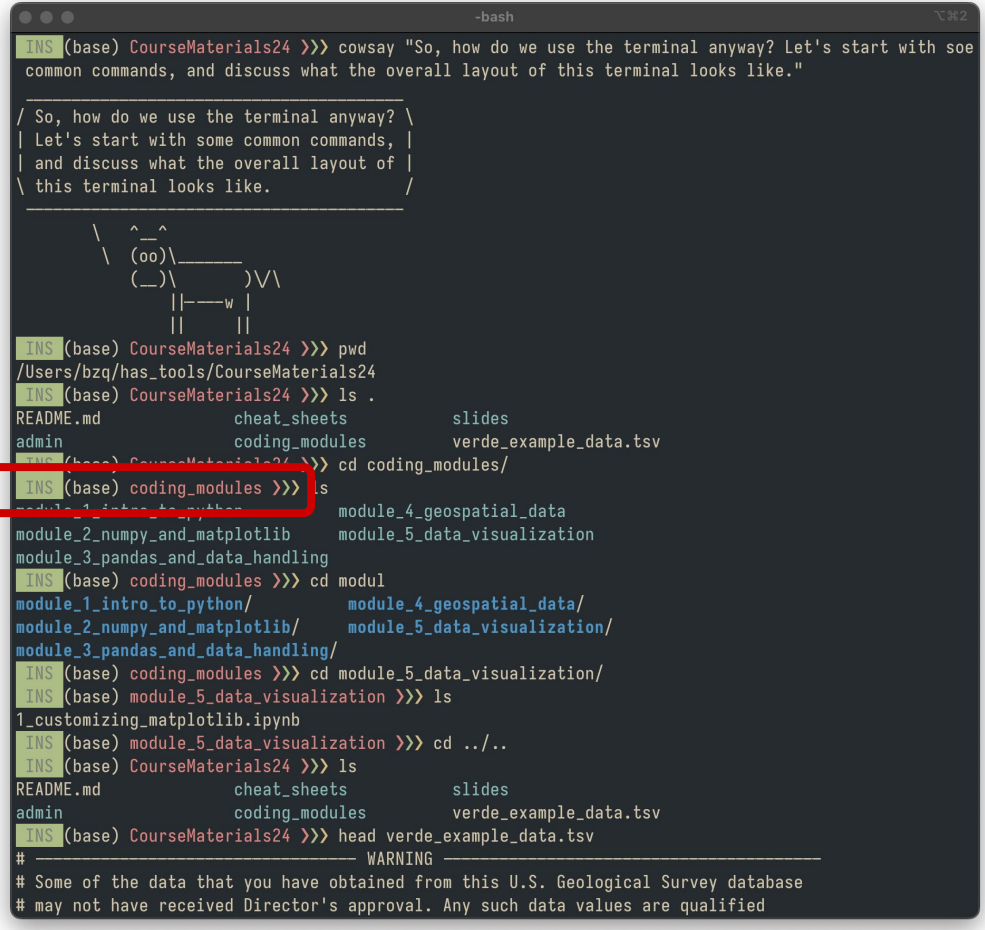


Quick overview

If you want to move to a new location you can use the `cd` command to “change directory”

Here I go into the `coding_modules` directory

Which is then shown in my status line as my current location

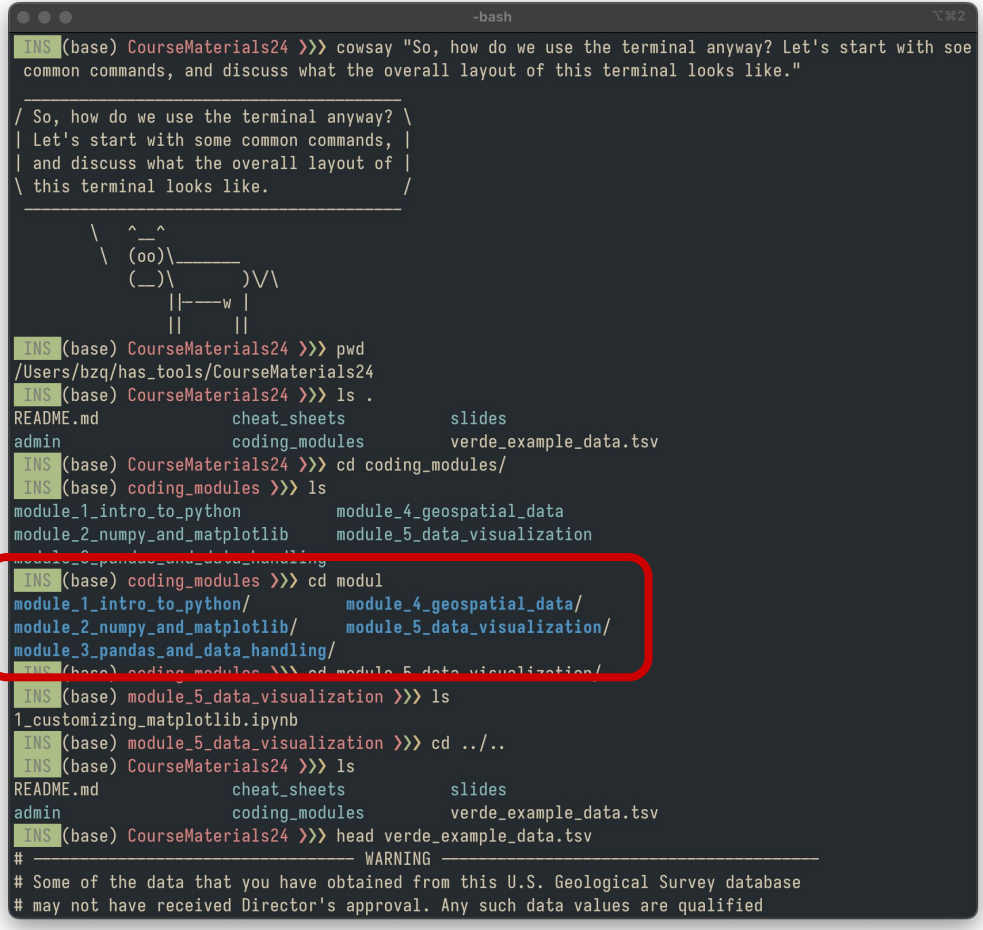


Quick overview

If you aren't sure of the exact names, you can type a partial name and then hit <tab> to get completions

If only one option exists, bash will automatically fill it in when you hit <tab>

Otherwise it will show you a list of options below

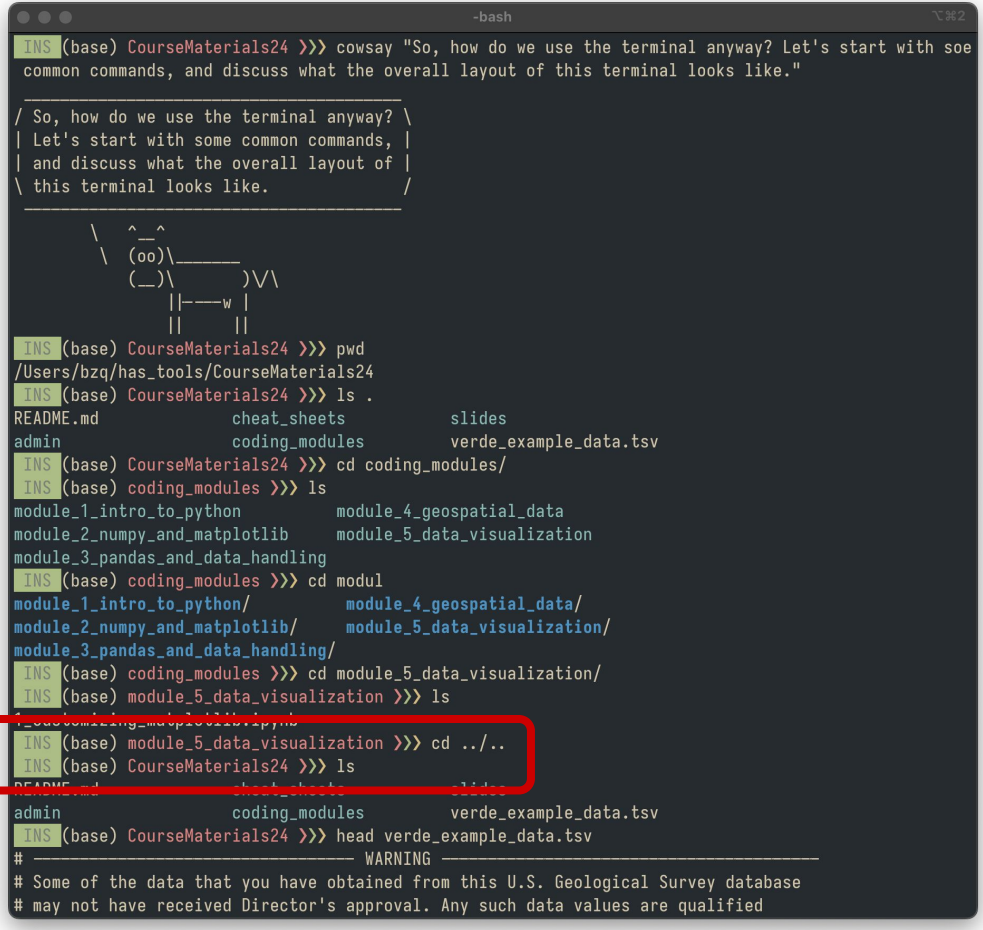


Quick overview

To go up directories (or back) you can use `..` in your `cd` command.

Each `..` goes up one directory.

Since directories are separated by `/` you can do `cd ../..` to go back 2 levels



Quick overview

And of course there are other commands that you can use.

We won't cover all of them here, but here are some additional resources:

https://linuxjourney.com/lesson/the-shell

https://www.kea.nu/files/textbooks/humblesec/thelinuxcommandline.pdf