KG

← Pre-Course Work

7 / 28

Paths and Directories

You've likely used a vending machine — the kind where you push a combination of buttons to designate your selection and then your chosen treat drops down the chute.

Those vending machine snacks are similar to the files on your computer. Every file or folder can be read, written, or deleted simply by referencing its **position** within your file system. You punch in the coordinates by referencing the file's **path**, just like pushing A7 or D9 on a vending machine.

When we want to interact with our files via the CLI, there are two types of paths we can choose: **absolute** or **relative**.

- **Absolute path:** A path that starts at your home directory (also called the **root directory**) and lists all of the folders and subfolders along the way.
    - For example, let's say we want to find our resume. An absolute path might show something like: `~/Documents/Resumes/My_Current_CV/CV.doc`. The `~` at the beginning of this path tells the computer to build it starting at your home directory. Think of the `~` symbol as shorthand for your home directory, which contains all of your files and documents.
- **Relative path:** A path that starts from your current position in the folder structure, referencing a given file or folder based on this starting position.
    - For example, if you're currently in the `Documents` directory, then the relative path to our resume would be: `Resumes/My_Current_CV/CV.doc`. This path does *not* start in your home directory (`~`). It starts at your current position in the file system, which is `Documents` in this example. (We will cover more on navigating through your file system shortly.)

Pro tip: When you're navigating the command line, `./` is shorthand for "start from the current working directory," whereas `~/` is shorthand for "start from the home directory." If you type `cd ~` in the terminal, it will take you to your home directory!