Notes 2: The Linux FS

- Presentation
- article

Each of the commands used for navigating the file system

> Here is an example from the previous notes! Use the same format for the commands to navigate the file system!

Echo

Definition:

Display a line of text

Usage

`echo` + `option` + `string`

Examples:

- Display a line of text without the new line
  - `echo -n "hello world"`
- Display a line of text that includes a horizontal tab
  - `echo -e "\thello world"`
- Display 2 lines of text in a single echo command
  - `echo -e "Line 1\nLine2"`
- Display 2 lines of text in a single echo command, with the second line starting with a tab
  - `echo -e "Line 1\n\tLine 2"`
- Display 2 lines of text in a single echo command that starts with a tab
  - `echo -e "\tLine 1\tLine 2"`

Definitions of the following terms:

- **File system**
  - The way files are stored and organized.
- **pathname**
  - Part of a file in a filesystem that operates like an adress. It indicates the location of a file in the filesystem.
- **Absolute path**
  - The location of a file starting at the root of the filesystem.
- **Relative path**
  - The location of a file starting from the current working directory or a directory that is located inside the current directory.
- **The difference between YOUR HOME directory and THE HOME directory**

    - Your home directory is refers to the user's directory and the home directory is located in the root.
- **parent directory**
    - A directory containing one or more directories and files.
- **child directory or subdirectory**
    - A better name for this is a subdirectory or subfolder. This is a directory inside another directory.
- **Bash special characters**
    - single period
    - 2 consecutive periods
    - tilde character
    - one forward slash
    - hyphen-minus
    - hash or number sign
    - single exclamation mark
    - 2 consecutive exclamaton marks
- **environment variables**
    - $USER
    - $HOME
    - $PWD
    - $OLDPWD
- **user defined variables**
    - Created by the user and exist only in the script and subshell that runs the script. They allow you to temporarily store data and use it throughout the script.
- **Why do we need use $ for referencing the value of a variable (using a variable) in a shell script**
    - The dollar sign is needed so that the shell can distinguish between regular text and variable names. When the shell encounters the dollar sign, it knows that it is for a variable and needs to be replaced with a value.