Notes 4

Defintions of the following terms:

File System

- Definition:
  - The way files are stored and organized
- Usage:
  - To organize and manage files on a storage device
- Examples:
  - **NTFS** (New Technology File System)
    - The default file system on Windows OS (Operating Systems)
  - **HFS+** (Hierarchical File System Plus)
    - The standar file system used on Apple devices
  - **ext4**
    - The most common file system used on Linux distributions

---

pathname

- Definition:
  - The location of a given file in your computer. Can be an absolute path or relative path
- Usage:
  - A string containing the path of the URL for the location
- Examples:
  - **Absolute pathname**
    - A path name that starts with a backslash: "\a\b\c"
  - **URL pathname**
    - Path portion of a URL: "https://(insert_link_here).com"
  - **Relative pathname**
    - Pathname that doesn't start with a backslash: "a\b\c"

Absolute path

- Definition:
  - The location of a file starting at the root of the file system

- Usage:
    - Can be used at any point of the file system regardless of your current file directory
- Example:
    - Absolute path of the file "list.txt"
        - "/home/maria53/Downloads/list.txt"

Relative path

- Definition:
    - The location of a file starting from a child directory of the current working directory or from the current directory itself.
- Usage:
    - Used as shortcuts to save time while accessing files and directories
- Examples:
    - Assuming that the current working directory is "/home/maria53"
        - "../Downloads/list.txt"

YOUR HOME directory VS. **THE HOME** directory

- Definition (USER HOME DIRECTORY):
    - This is your user's personal directory where all your files are located.
- Usage:
    - To store personal files, configurations, and data specific to your user account.
- Examples:
    - **Linux**
        - /home/⟨username⟩, /usr/home/⟨username⟩
    - **Windows**
        - \Users\⟨username⟩
    - Check how much space your home directory has
        - `echo $HOME`
- Definition (THE HOME DIRECTORY):
    - This is the parent directory of all the home directories.
- Usage:

    - To store all the users' home directory
- Examples:
    - The absolute path of this directory:
        - /home

---

parent directory

- Definition:
    - A dirtectory containing one or more directories and files.
- Usage:
    - Can be used to contain other folders and files
- Examples:
    - Parent directories:
        - /home/user1, /home
    - Parent directories in a relative path:
        - ../, ../File.txt
    - Used in a bash command
        - `cd ..`

---

child directory/ subdirectory

- Definition:
- A subdirectory or subfolder. This is a directory inside another directory
- Usage:
    - To make files inside other files
- Examples:
    - Subdirectory of the website "example.com/blog/"
        - "/blog/
    - Used in a bash command
        - `cd ../../file.png`

Bash special characters

- Definition:

    - Special characters are function like commands that tell the shell to perform a specific action without having to type the complete command

- Usage:
    - Make working on a command line more efficient
- Examples:
    - `.` **(single period)**
        - Represents the current directory.
    - `..` **(2 consecutive periods)**
        - Represents the parent directory.
    - `~` **(tilde character)**
        - Expands the current users home directory.

---

environment variables

- Definition:
    - Store values of a user's environment and can be used in commands in the shell
- Usage:
    - When writing commands that you want to use regales of which user is using the computer.
- Examples:
    - `$USER`
        - Stores the current's user username
    - `$HOME`
        - Stores the absolute path of current's user home directory
    - `$PWD`
        - Stores the absolute path of the present working directory.

---

user defined variables

- Definition:
    - Variables that a user creates within a shell script to store and manipulate values
- Usage:
    - Can be used to store values that can be referenced and manipulated throughout the script
- Examples:
    - Assign `state` the value `Colorado`
        - `state=Colorado`

    - Storing a file path
        - `my_file_path="/home/user/documents/report.txt" cat $my_file_path`
    - Calculating values in a script
        - `num1=10 num2=5 result=$((num1 + num2)) echo "The sum is: $result"`

Why do we need to use $ with variables when bash shell scripting?

- Usage:
    - When you want to use the value of a variable, you need to precede the variable name with `$`.
    - Without the `$`, the shell will interpret the variable name as a literal string rather than a reference to the variable.
- Examples:
    - Without use of `$`
        - `name="John" echo "Hello, name"` Outputs: Hello, name
    - With the use of `$`
        - `name="John" echo "Hello, John"` Outputs: Hello, John
    - Omit the `$` when using variables within specific contexts
        - `x=5 y=10 echo $((x + y))` Outputs: 15
