Notes 4: The Linux FS

- [Presentation](#)
- [Article](#)

Each of the commands used for navigating the file system.

PWD

Usage:

- Displays the absolute path of the current working directory.

Formula:

- `pwd`

Examples:

- Print the absolute path of the current working directory.
    - `pwd`

CD

Usage:

- Changes the current working directory. In other words, it moves you from one directory to another. By default, it will always send you to your home directory.

Formula:

- `cd` + `destination absolute path or relative path`

Examples:

- Go (change your current directory) to your home directory (there is more than 1 way of doing this):
    - `cd` (without any arguments, `cd` will take you home)
    - `cd` + `~` (using the `~` special character will expand to the absolute path of the user's home directory)
    - `cd` + `\$HOME` (using the `$HOME` environment variable)
    - `cd /home/$USER/Downloads` (using $USER environment variable in the path)
- Go to a specified directory with absolute path:
    - `cd /usr/share/themes`
- Go to a specified directory with relative path assuming your current working directory is `/home`:
    - `cd maria53/Downloads/`
- Go to the previous working directory. This is useful when you are working with 2 directories located far in the directory tree:
    - `cd -`

- Go to the previous directory in the directory tree. One directory above.:
  - `cd ../`
- Go to 2 directories above the directory tree:
  - `cd ../../`

LS

Usage:

- `ls` is used for listing files and directories. By default it will list the current directory when no directory is specified. Listing means to see what is inside a directory.

Formulas:

`ls` + `option` + `directory to list`

Examples:

- See all the options of the ls command (extracted from the man page):
  - `ls --help`
- List the current directory:
  - `ls`
- List all the files inside a given directory:
  - `ls -A /usr/share/fonts/X11` (absolute path)
  - `ls -A Documents/` (relative path assuming that the $PWD is $HOME)
- Long list a directory:
  - `ls -lA ~/Pictures`
- List a directory recursively:
  - `ls -R Documents/`
- Long list a directory only:
  - `ls -ld Documents/`
- List a directory sorted by last modified:
  - `ls -t Documents/`
- List a directory sorted by file size:
  - `ls -S Documents/`
- Long list a directory excluding group and owner information, with human readable file size and sorted in reverse order:
  - `ls -lhgGr Documents/`

Definitions of the following terms:

- File system:
  - The way files are stored and organized.
- pathname:
  - Also known as file path. This is the location of a given file in your computer. A path name can be absolute path or relative path.
- Absolute path:

- - The location of a file starting at the root of the file system. For example, `/home/maria53/Downloads/list.txt` is the absolute path of the file `list.txt`. The advantage of absolute paths is that they can be used at any point of the file system regardless of your current directory. Any command that is given an absolute path will be able to find the file because it will start at the beginning of the filesystem. The disadvantage is that a command can be long to type if the file path is long.
- Relative path:
  - The location of a file starting from a child directory of the current working directory or from the current directory itself. The advantage of using relative path is that typing commands is faster. The disadvantage of relative paths is that they cannot work from anywhere in the filesystem. In order for a relative path to work, a file must be reachable from the current directory onwards. Another disadvantage of relative paths is that they require a better mental understanding of the linux filesystem in the sense that you must keep a mental image of the directory tree that you are working with. An example of a relative path would be `Downloads/list.txt` assuming that the current working directory is `/home/maria53`.
- The difference between YOUR HOME directory and THE HOME directory:
  - "Your home directory" is your user's personal directory where all your files are located. Every user has it's own home directory just like in a apartment complex they all residents have their own apartment while sharing the common areas. You have total ownership of your home directory but outside of the home directory only the root user can make changes. An example absolute path, assuming that user name is `maria53`, would be /`home/maria53`. "The home directory" is the parent directory of all the home directories. This is where all the users' home directory are. The absolute path of this directory is `/home`. Notice that it starts at 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:
  - Function like commands that tell the shell to perform a specific action without having to type the complete command. These special characters make working on the command line more efficient.
- Environment variables:
  - Store values of a user's environment and can be used in commands in the shell. These values can be unique to the user's environment which makes them ideal when writing commands that you want to use regales of which user is using the computer.
- User defined variables:
  - Are created by the user and exist only in the script and subshell that runs the script.
- Why do we need use `$` with variables in bash shell scripting?
  - In Bash shell scripting, the `$` symbol is used to reference, or expand, variables. Without it, the shell treats the variable name as plain text rather than retrieving its stored value. To use the value stored in an environment variable you must prepend the variable name with a `$`.