

Notes 2: The Linux fs

Each of the commands used for navigating the file system

CD

Definition:

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.

Usage

`cd+destination absolute path or relative path`

Examples:

- cd will take you home`
 - `cd`
- `will expand to the absolute path of the user's home directory
 - `cd ~`
- using the \$HOME environment variable
 - `cd $HOME`

LS

Definition:

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.

Usage

`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 including hidden files in current directory
 - `ls -A`

PWD

Definition:

Displays the absolute path of the current working directory.

Usage

`pwd`

Examples:

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

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**
 - The YOUR HOME directory is your personal directory where all your files are located. You have total ownership of your home directory but outside of the home directory only the root user can make changes. While 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`. Noticed 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. See image for visual reference.
- **Bash special characters**

- `.` (single period): represents the current directory.
- `~` (tilde character): expands the current users home directory. It is like a variable that the shell uses to store the absolute path of the user's home directory.
- `/` (one forward slash): as mentioned earlier, this is the root directory and the shortest path in the system. This is the beginning of the directory tree. There is nothing before it and everything after it.
- `-` (hyphen-minus): is used to move to the previous current working directory.
- `#` (hash or number sign): This is used for single line comments in shell scripting.
- **environment variables**
 - 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 writhing commands that you want to use regales of which user is using the computer. To see a list of your environment variables type `env`. To use the value stored in an environment variable you must prepend the variable name with a `$`
- **user defined variables**
 - `$USER` = stores the current's user username
- **Why do we need use \$ for referencing the value of a variable (using a variable) in a shell script**
 - 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 writhing commands that you want to use regales of which user is using the computer. To see a list of your environment variables type `env`. To use the value stored in an environment variable you must prepend the variable name with a `$`.