

Notes 4

Commands used for navigating the file system.

File System

Definition:

The way files are stored and organized.

Usage:

Used to organize our directory in the linux file system.

Examples:

```
/home/, /root/, /bin/
```

Pathname

Definition:

Indicates the location of the file in the filesystem (like an address).

Usage:

```
/home/+/user/+/pathname/
```

Examples:

```
/home/mauricio/Downloads/
```

Absolute Path

Definition:

The location of a file starting at the root of the file system.

Usage:

```
/home/+/user/+/pathname/+/filename.ext
```

Examples:

```
/home/mauricio/Downloads/song3.mp3
```

Relative Path

Definition:

The location of a file starting from the current working directory or a directory that is located inside the current working directory.

Usage:

`/pathname/+filename.ext`

Examples:

`/Downloads/song3.mp3`

Difference between YOUR HOME directory and THE HOME directory

Definition:

YOUR HOME directory refers as the user's personal directory, meanwhile **THE HOME** directory contains all users' home directory.

Usage:

`/home/+/username/` for **YOUR HOME**. `/~/` you can use the tilde character to navigate **THE HOME** directory.

Examples:

`/home/mauricio /home`

Parent Directory

Definition:

A directory containing one or more directories and files.

Usage:

`/+ /home/+ /user/` (if necessary)

Examples:

`/home/user` (which can be the parent directory of "Downloads")

Child Directory or Subdirectory

Definition:

A child directory or subdirectory is a directory inside another directory.

Usage:

```
/home/+/user/+/pathname/
```

Examples:

```
/home/mauricio/Downloads
```

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:

```
~ ! . . .
```

Examples:

~ represents the current users home directory. ! used for repeating command from the history.

Environment Variables

Definition:

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 users' environment which makes them ideal when using the computer.

Usage:

```
$+Environment Variable
```

Examples:

`$USER` = Stores the current's user username. `$HOME` = Stores the absolute path of the current's user home directory.

User defined variables

Definition:

User-defined variables are created by the user and assigned values using the assignment operator (=).

Usage:

```
name of variable += + any name you want it to display
```

Examples:

```
name=John Doe
```

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

Definition:

Used for accessing the value stored in a variable in Bash.

Usage:

```
$ + User defined variable
```

Examples:

name="John" "\$name" will display the the variable name with its assigned value.
