

1. cd (Change Directory)

- **Description:** Changes the current working directory.
- **Usage/Formula:** `cd [directory_path]`
- **Examples:**
 - `cd /home/user`: Changes the current directory to `/home/user`.
 - `cd ..`: Moves up one directory level (to the parent directory).
 - `cd`: Returns to the user's home directory.
 - `cd -`: Returns to the previous directory.

2. pwd (Print Working Directory)

- **Description:** Displays the absolute pathname of the current working directory.
- **Usage/Formula:** `pwd`
- **Examples:**
 - `pwd`: Outputs the full path of the present working directory, such as `/home/user/documents`.

3. ls (List Directory Contents)

- **Description:** Lists the files and directories within a specified directory (or the current directory if none is specified).
- **Usage/Formula:** `ls [options] [directory_path]`
- **Examples:**
 - `ls`: Lists the contents of the current directory.
 - `ls -l`: Lists the contents in long format (detailed information).
 - `ls -a`: Lists all files, including hidden files (those starting with a dot).
 - `ls -lh`: lists the contents in long format, and human readable file sizes.
 - `ls /etc`: lists the contents of the `/etc` directory.

Definitions

- **File System:**
 - A hierarchical structure used by an operating system to organize and manage files and directories on a storage device. It defines how data is stored, retrieved, and updated.
- **Pathname:**
 - A string of characters that specifies the location of a file or directory within a file system.
- **Absolute Path:**
 - A pathname that specifies the location of a file or directory starting from the root directory (`/`). It always begins with `/`.
 - Example: `/home/user/documents/file.txt`
- **Relative Path:**
 - A pathname that specifies the location of a file or directory relative to the current working directory. It does not begin with `/`.
 - Example: `documents/file.txt` (if `documents` is a subdirectory of the current directory).
- **Your Home Directory vs. The Home Directory:**
 - "The home directory" can refer to the general concept of a user's personal directory. "Your home directory" refers to the specific home directory of the current user. They are the same

thing, but the usage differs depending on context.

- **Parent Directory:**

- The directory that contains another directory. It is the directory one level higher in the file system hierarchy.
- Example: if `/home/user/documents` is the child directory, then `/home/user` is the parent directory.

- **Child Directory or Subdirectory:**

- A directory that is contained within another directory. It is a directory one level lower in the file system hierarchy.
- Example: if `/home/user` is the parent directory, then `/home/user/documents` is the child directory.

- **Bash Special Characters:**

- Characters that have special meanings in the Bash shell, used for various operations like pattern matching, redirection, and command substitution.
- Examples: `*`, `?`, `>`, `<`, `|`, `$`, `;`, `&`, `~`.

- **Environment Variables:**

- Dynamic named values that can affect the way running processes will behave on a computer. They are set by the operating system or the user, and are accessible to all processes.
- Examples: `PATH`, `HOME`, `USER`, `PWD`.

- **User-Defined Variables:**

- Variables created by the user within a Bash script or shell session to store and manipulate data.
- Example: `my_variable="Hello, world!"`.

- **Why use \$ with variables in Bash shell scripting?**

- The `$` symbol is used to access the value of a variable. Without the `$`, Bash treats the variable name as a literal string. When you want to use the *value* stored in a variable, you must precede it with `$`.
- Example:
 - `my_variable="Hello"`
 - `echo $my_variable` (outputs "Hello")
 - `echo my_variable` (outputs "my_variable")