Notes 4

# File System Navigation Commands (CIS106)

## 1. `pwd` (Print Working Directory)

- Description: Displays the full path of the current working directory.
- Usage: pwd
- Examples: pwd Displays the current working directory, e.g., /home/student

---

## 2. `cd` (Change Directory)

- Description: Changes the current directory to a specified directory.
- Usage: cd [directory_path]
- Examples: cd /home/student/Documents Changes to the Documents folder.
- cd .. Moves to the parent directory.
- cd / Changes to the root directory.

---

## 3. `ls` (List Directory Contents)

- Description: Lists the contents of a directory.

- Usage: ls [options] [directory]

- Examples: ls Lists files in the current directory.

- ls /home/student/ Lists contents of the student's home directory.

- ls -l Lists contents in long format (with detailed file information).

- ls -a Shows hidden files (files starting with (.)

---

## 4. `mkdir` (Make Directory)

- Description: Creates a new directory.

- Usage: mkdir [directory_name]

- Examples: mkdir project Creates a directory named 'project'.

- mkdir -p /home/student/project/subdir, Creates 'subdir' inside 'project', and any parent directories that don’t exist.

---

## 5. `rmdir` (Remove Directory)

- Description: Removes an empty directory.

- Usage: rmdir [directory_name]
- Examples: rmdir old_folder, Removes the 'old_folder' directory if it's empty.
- rmdir /home/student/temp , Removes the empty 'temp' folder.

---

## 6. `rm` (Remove Files or Directories)

- Description: Removes files or directories. Can delete non-empty directories if specified.
- Usage: rm [options] [file_or_directory]
- Examples: rm file.txt Deletes 'file.txt'.
- rm -r folder Deletes a directory named 'folder' and all its contents.
- rm -f file.txt Forcefully deletes 'file.txt' without confirmation.

---

## 8. `mv` (Move or Rename Files and Directories)

- Description: Moves or renames files and directories.
- Usage: mv [source] [destination]
- Examples: mv file.txt new_file.txt Renames 'file.txt' to 'new_file.txt'.
- mv file.txt /home/student/backup/ Moves 'file.txt' to the 'backup' folder.
- mv folder/ /home/student/new_folder/ Moves or renames 'folder' to 'new_folder'.

---

## 9. `touch` (Create or Update File Timestamps)

- Description: Creates a new, empty file or updates the timestamp of an existing file.
- Usage: touch [file_name]
- Examples: touch newfile.txt Creates an empty file named 'newfile.txt'.
- touch file.txt Updates the timestamp of 'file.txt'.
- touch file1.txt file2.txt Creates or updates two files at once.

## Definition of The Following Terms:

#### File System

- A file system is a method and data structure that an operating system uses to manage files on a disk or partition. It controls how data is stored and retrieved, managing file hierarchy, permissions, and directories. Examples of file systems include FAT32, NTFS (Windows), ext4 (Linux), and HFS+ (Mac).

#### 2. Pathname

- A pathname is the address or location of a file or directory in the file system. It specifies the route that must be taken to reach a file or directory, either relative to the current directory or as an absolute path starting from the root directory.

#### 3. Absolute Path

- An absolute path specifies the full path to a file or directory, starting from the root (/) directory, regardless of the current working directory. It always begins with a /.
  - Example: /home/user/Documents/file.txt

#### 4. Relative Path

- A relative path specifies the location of a file or directory relative to the current working directory. It does not begin with a / and depends on where you are in the directory structure.
  - Example (assuming current directory is /home/user): Documents/file.txt refers to the file.txt inside the Documents folder within the current directory.

#### 5. The Difference Between "Your Home Directory" and "The Home Directory"

```
* Your Home Directory: This is a personal directory assigned to a user when
they log into a system. For example, if your username is student, your home
directory might be /home/student.
* The Home Directory: This generally refers to the /home directory where
all user home directories are stored (e.g., /home/student, /home/user2,
etc.). It's the parent directory of individual user home directories.
```

#### 6. Parent Directory

- A parent directory is the directory that contains the current directory (also called the working directory). Every directory, except the root (/), has a parent directory. You can move to the parent directory using cd ...
  - Example: The parent directory of /home/student/Documents is /home/student.

#### 7. Child Directory or Subdirectory

- A child directory (or subdirectory) is a directory that is contained within another directory (its parent directory). It's part of the hierarchical structure of a file system.
  - Example: If /home/student/Documents is the parent directory, then /home/student/Documents/Work is a child directory of Documents.

#### 8. Bash Special Characters

- In Bash, special characters have special meanings and are used to control various aspects of the shell or to perform specific tasks. Some examples include:

  - $ (dollar sign): Used to reference variables. ~(tilde): Refers to the home directory. / (slash): Used to separate directory names in a path. . (dot): Refers to the current directory. .. (double dot): Refers to the parent directory. | (pipe): Redirects the output of one command to the input of another.
  - > (greater than): Redirects output to a file.
  - (asterisk): Wildcard that matches any string of characters.

**9. Environment Variables**

- Environment variables are dynamic variables in the shell that affect how processes behave on the system. They hold system-wide information and settings, such as paths to system directories or configuration options. Examples include PATH, HOME, USER, and SHELL.
  - Example: echo $HOME outputs the path to the current user's home directory.

**10. User-Defined Variables**

- User-defined variables are custom variables that users create during a session or in a script to store data. They can hold strings, numbers, or other types of information. These variables can be used to simplify and automate shell scripting tasks.