**UNIX Command**

Unix commands, which are the instructions or programs used to interact with a Unix-based operating system, including systems like Linux and macOS, through the command line interface.

**The kernel**

The kernel of UNIX is the hub of the operating system: it allocates time and memory to programs and handles the filestore and communications in response to system calls.

As an illustration of the way that the shell and the kernel work together, suppose a user types **rm myfile** (which has the effect of removing the file **myfile**). The shell searches the filestore for the file containing the program **rm**, and then requests the kernel, through system calls, to execute the program **rm** on **myfile**. When the process **rm myfile** has finished running, the shell then returns the UNIX prompt % to the user, indicating that it is waiting for further commands.

**The shell**

The shell acts as an interface between the user and the kernel. When a user logs in, the login program checks the username and password, and then starts another program called the shell. The shell is a command line interpreter (CLI). It interprets the commands the user types in and arranges for them to be carried out. The commands are themselves programs: when they terminate, the shell gives the user another prompt (% on our systems).

The adept user can customise his/her own shell, and users can use different shells on the same machine. Most accounts on our clusters have the **bash shell** by default. The accounts on hpc-class use shell specified at https://asw.iastate.edu/cgi-bin/acropolis/user/shell .

The bash and tcsh shells have certain features to help the user inputting commands.

Filename Completion - By typing part of the name of a command, filename or directory and pressing the [**Tab**] key, the shell will complete the rest of the name automatically. If the shell finds more than one name beginning with those letters you have typed, it will beep, prompting you to type a few more letters before pressing the tab key again.

History - The shell keeps a list of the commands you have typed in. If you need to repeat a command, use the cursor keys to scroll up and down the list or type history for a list of previous commands.

These commands allow users to perform tasks such as managing files, directories, processes, and more. Here’s a brief overview of some commonly used Unix commands:

1. **mkdir -** Creates a new directory.

Example: mkdirnew\_folder creates a new directory called new\_folder

****

1. **ls -** Lists the contents of a directory.

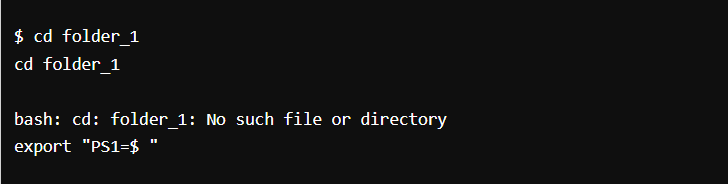
Example: ls lists all files and directories in the current directory.

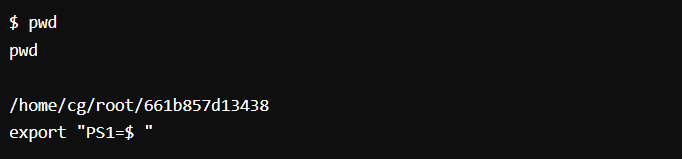
****Example: ls -l lists in long format with detailed information.

1. **cd -** Changes the current directory.

Example: cd Documents changes the current directory to the Documents folder.

Example: cd .. moves up to the parent directory.



1. **pwd** - Prints the current working directory.
2. **Touch** - is used primarily to change the file timestamps—specifically, the access and modification times of a file or directory.



1. **Echo** - is a fundamental utility used to display lines of text or string values to the standard output, which is typically the terminal screen.
2. **cat** - Concatenates and displays files.

Example: cat file.txt displays the contents of file.txt.

1. **rm** - Removes files or directories.

Example: rm file.txt deletes the file.txt.



1. **cp** - Copies files or directories.

Example: cp source.txt destination.txt copies source.txt to destination.txt.

1. **grep** - Searches for patterns in text.

Example: grep "hello" file.txt searches for the word "hello" in file.txt.