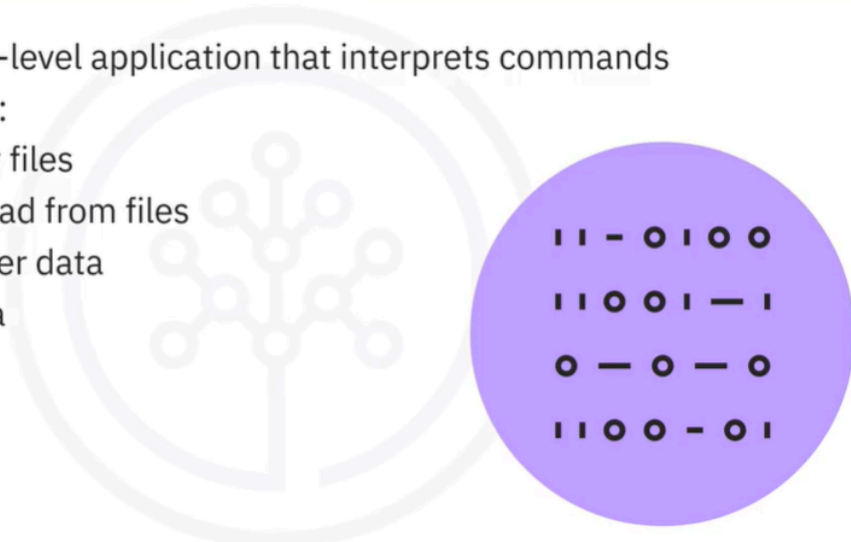


# Linux Shell and Terminal

## Overview of the Linux shell

---

- The **shell** is an OS-level application that interprets commands
- Use commands to:
  - Move and copy files
  - Write to and read from files
  - Extract and filter data
  - Search for data
- Shells:
  - Bash
  - Zsh



- The Linux shell is an OS-level application that interprets commands. In early versions of Unix and Linux, the shell was the only way to interact with the operating system. Today, you can also use graphical user interfaces, but the shell remains a popular and flexible choice and an easy way to run script files. You can use shell commands to perform tasks such as moving and copying files, writing to and reading from files, extracting and filtering data, and searching for data. There are many shell versions, but the base functionality of most is the same. Some popular examples include: Bash And Zsh. You interact with the Linux shell through a Linux terminal.

=⇒ The **Linux shell** is an application that allows you to interact with the Linux operating system by typing commands. Here are some key points about the Linux shell:

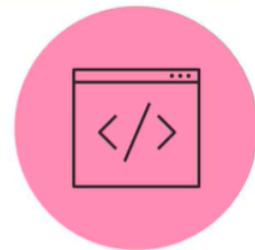
- **Command Interpreter:** It interprets the commands you enter and translates them into actions that the operating system can perform.

- **Text-Based Interface:** Unlike graphical user interfaces (GUIs), the shell is text-based, meaning you type commands instead of clicking on icons.
- **Popular Versions:** Some common types of shells include **Bash** (Bourne Again SHell) and **Zsh** (Z Shell).
- **Task Execution:** You can use the shell to perform various tasks, such as moving files, copying data, and running programs.
- **Scripting:** The shell also allows you to write scripts, which are sequences of commands that automate tasks.

In summary, the Linux shell is a powerful tool for controlling and managing your Linux system through command-line input.

## Overview of a Linux terminal

- The **terminal** is an application you use to interact with the shell
- Enter commands and receive output from them

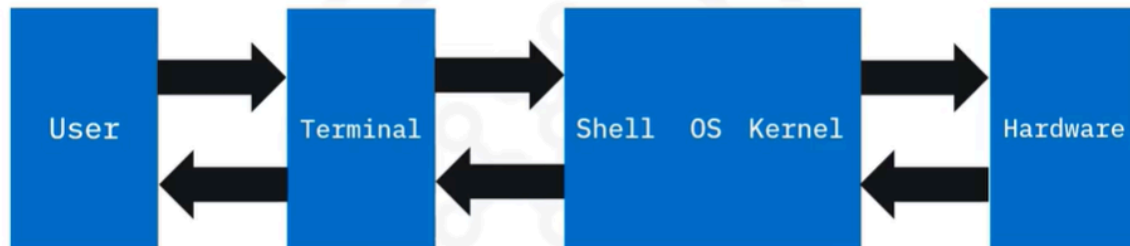


```
/home/me/ $ python myprogram.py  
Hello, World!
```

- A terminal is an application, or user interface, where you enter the commands you want to run and receive any output from those commands. For example, to start the Python application and run a program called "myprogram.py", type `python space myprogram.py`. When you press Enter, the shell runs the command. This program prints the words "Hello, World!" to the terminal.

# Communicating with Linux system

---



- So how are commands run? First, we have a user who wants to run a command. They enter the command in a terminal, which is then relayed to the shell. The core components of the operating system and kernel translate the command for the hardware to perform. When the hardware completes the command, the kernel reads any changes or results and sends them back via the shell to the terminal for the user's information. The terminal is a powerful way to run applications and interact with your machine.

## The Linux terminal

---

```
/home/me/Documents/ $
```

A screenshot of a Linux terminal window with a dark background. The prompt `/home/me/Documents/ $` is displayed in red text at the top left. The prompt is enclosed in a red rectangular box. The rest of the terminal area is empty.

- Most terminals have a similar user interface for you to enter commands. The area where you enter commands is called the command line. And the vertical line, or cursor, is the command prompt. This indicates where the text that you type will be displayed. In this example, the current working directory is the Documents directory, which is inside the me directory, which is inside the home directory. The current working directory is the location where the shell will look for any commands that you specify to run, for example, the Python program in the previous example. Not all terminals display the full location, or path, of your current directory, so some will just display Documents here.

## Paths in the Linux filesystem

---

- The **filesystem** is the human-readable directory or file location

```
/home/me/Documents/
```

- “**a/b**” indicates the file or directory named *b* inside the directory named *a*
- Special paths:
  - ~ Home directory
  - / Root directory
  - .. Parent of current directory
  - . Current directory



- The path is the human-readable location of a directory or file in the Linux filesystem. The “a slash b” structure indicates that the file or directory named “b” is located inside the directory named “a”. There are also special paths: A single tilde symbol refers to the user’s home directory. A single slash at the beginning of a path refers to the root directory. Two periods refer to the parent of the current directory. And a single period refers to the current directory.

# Changing the current working directory

```
/home $ cd /  
/ $ cd bin  
/bin $ ./ls  
[    cat    cp    dash    dd    echo    expr    kill  
...  
ls mv  ps    rm    sh    stty    tcsh    unlink  
/bin $ cd ~  
/home/me $ ls  
Documents  Pictures  Downloads  Movies  Music  Desktop
```

- So, let's look at how you can use the terminal to change the current working directory. You use the `cd` command that stands for change directory. Enter `cd /` to go to the root directory. And enter `cd bin` to move into the `bin` directory. The `bin` directory, which is inside root, contains programs required by the system. One of these programs, or executable files, is called "`ls`". You can run the `ls` program in the current working directory by entering `./ls`. Use it to display in the terminal window the names of all files and directories within the current directory. Many commands located in the `bin` folder are also built into the shell, so you can run them from other locations too. Navigate to your home directory using `cd ~`. And even though the current working directory, `/home/me`, doesn't contain the `ls` program, you can still run the command successfully.

# Changing the current working directory

---

```
/home $ cd ..  
/ $ cd /media/my-usb-drive  
/media/my-usb-drive $ cd ../../home/me/Documents  
/home/me/Documents $ cd ..  
/home/me $ python ./myprogram.py  
Hello, World!  
/home/me $
```

- Let's look at some more examples. Again, starting in the /home directory Enter "cd .." to change the current working directory to the parent of the existing current working directory. So, in this example the parent of /home is /, or the root directory. To then navigate to a USB drive called my-usb-drive in the media directory, enter cd /media/my-usb-drive.
- You can also navigate up and down the tree in one command. To navigate up to the media directory then up to the root, enter cd ../../ and then to navigate down to the Documents directory in the me directory in the home directory, enter /home/me/Documents. Press Enter to submit the command and move to the Documents folder. Let's move up to the /home/me directory, start the Python application, and run a program in the /home/me directory called myprogram.py.
- This program returns a message to the terminal window.

# Recap

---

In this video, you learned that:

- The shell is an OS-level application for running commands
- You use a terminal to send commands to the shell
- You can use the `cd` command to navigate around your Linux filesystem