Operating System, Shell, Bash, and CLI Fundamentals

.NET

A command-line interface (CLI) is an operating system* Shell*, like Bash*, that uses alphanumeric characters typed on a keyboard to interactively provide instructions and data to the operating system.*

HTTPS://EN.WIKIPEDIA.ORG/WIKI/SHELL_(COMPUTING)

*Keep Reading

What is an Operating System?

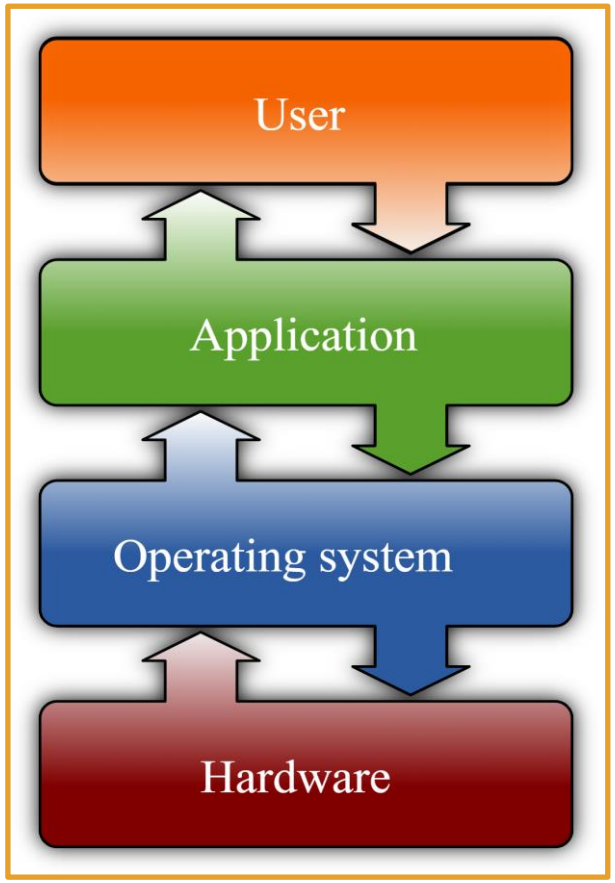
https://en.wikipedia.org/wiki/Operating_system

An operating system (OS) is software that manages computer hardware and software resources. It also provides common services for computer programs that run on the OS such as input, output, and memory allocation.

An OS is a program that acts as an intermediary between programs running on the computer and the computer hardware.

For example, the OS is used for hardware functions such as input, output, and memory allocation.

The application code is usually executed directly by the computer hardware. It frequently makes system calls to the OS or is interrupted by it to perform functions.



OS variety and distribution

https://en.wikipedia.org/wiki/Usage_share_of_operating_systems#:~:text=As%20of%20April%202022%2C%20Android,also%20using%20the%20Linux%20kernel.

Desktop Operating System	% of Market
Microsoft Windows	76.31%
macOS (OS X)	14.66%
Other	4.97%
Linux	2.49%

Mobile Operating System	% of Market
Android	71.59%
Apple iOS	27.68%
Samsung	.39%

Operating System History

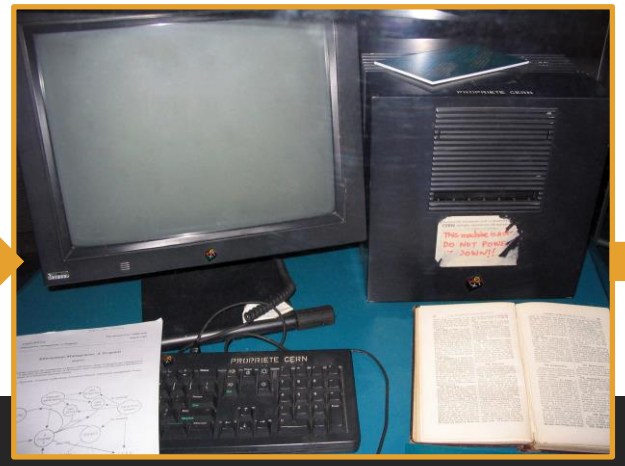
https://en.wikipedia.org/wiki/SHARE_Operating_System

The earliest electronic digital systems had no operating systems. Computers could only run one program at a time.

The earliest modern operating system was created in 1959. The SHARE Operating System (SOS) was an operating system introduced in 1959 by the SHARE user group. SHARE's main objective was to improve the sharing of programs. It enabled the use of runtime libraries, interrupts, and parallel processing.

Operating Systems continued improving. Notably, in the mid 1980's, Apple released its Mac OS and Microsoft released Windows.

The rest is history.



OS Components

https://en.wikipedia.org/wiki/Operating_system#:~:text=Plan%209.-,Components,-The%20components%20of
https://data-flair.training/blogs/components-of-operating-system

---

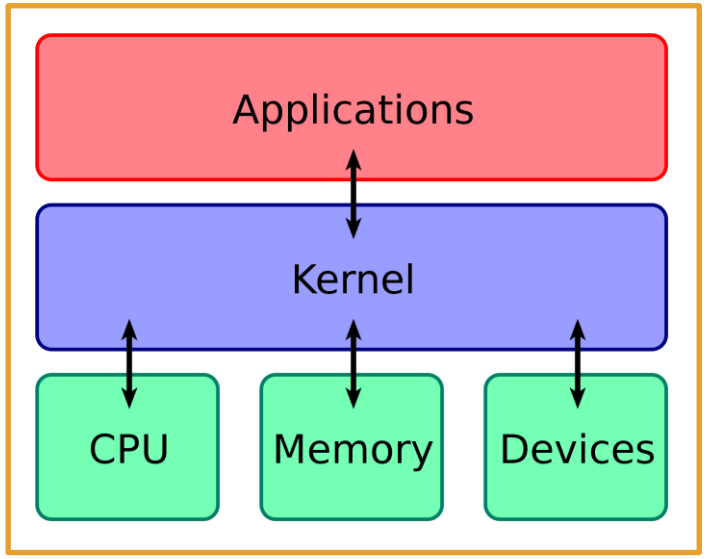
The ***Components*** of an operating system are functionalities that the OS does. They coordinate the work of other parts of the physical architecture of the computer.

The major components of most OS's are:

1. File Management
2. Security Management
3. Networking
4. Process Management
5. Memory Management
6. I/O Management

All software use operating systems to access any of the physical hardware.

Using the mouse or keyboard, a user can give commands to the OS. Applications also use the OS to start other programs and access something online, among other things.

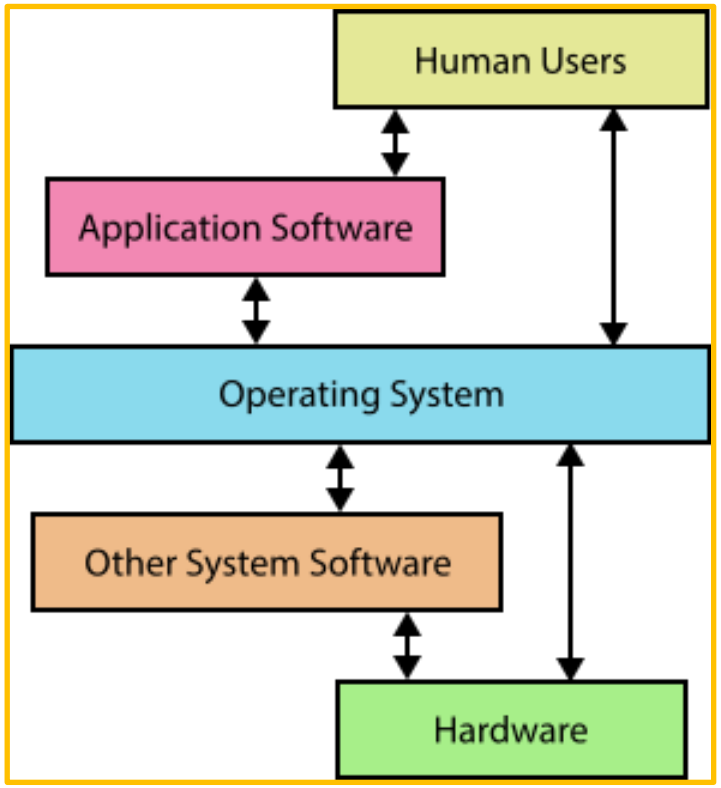


What is a Shell? (1/2)

https://en.wikipedia.org/wiki/Shell_(computing)
https://en.wikipedia.org/wiki/Microsoft_Windows

To understand a ***Shell***, first, you need to understand what an Operating System (OS) is.

- An OS is the foundational program in a computer.
- It is responsible for managing the computers registers, processors, memory, file naming/management, etc.
- You can gain access to the OS and manually manage your computer using a ***Shell*** application.



A ***Shell*** application acts like the shell of a clam. A ***Shell:***

1. prompts users for input,
2. interprets the input, and then
3. handles output from the underlying operating system.

Using correct keywords and commands, a ***Shell*** program gives you limited access to interact with the OS through the ***kernel API***. The ***kernel API*** is also used by applications that run on the computer.

What is a Shell? (2/2)

https://en.wikipedia.org/wiki/Shell_(computing)
https://en.wikipedia.org/wiki/Microsoft_Windows

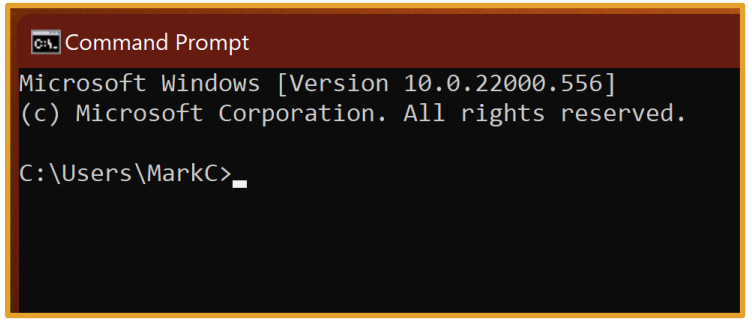
There are only two different variations on a ***Shell***.

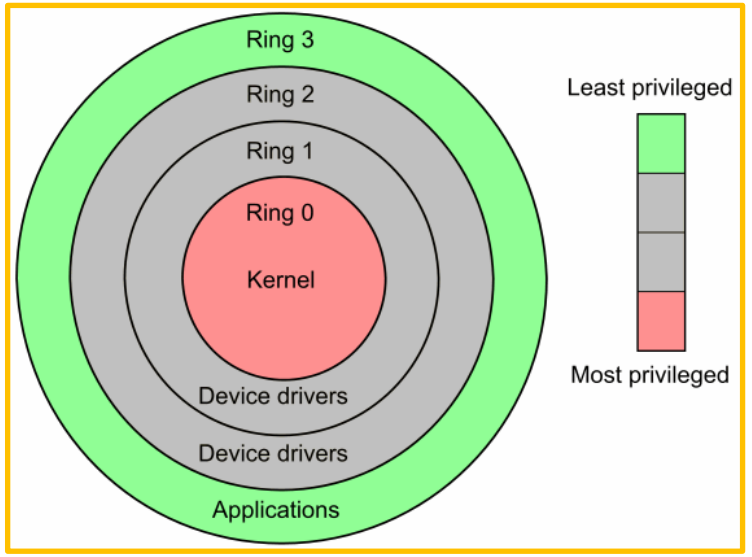
- Command-Line Shell – All interaction is through text with syntax rules.
- Graphical User Interface – Interaction is done with a mouse using images representing files, applications, etc.

We use the ***Graphical User Interface*** version of our computers ***Shell*** when we use our mouse to drag and drop, copy and paste, or select files or applications to open.

The ***Command-Line Shell*** is a text only interface. The ***Command-Line Interface*** is referred to as a ***CLI***. It is very powerful when combined with the many special keywords built into the Operating System that help us manage different aspects of our computers.

A ***Shells*** privileges are somewhat limited when run normally, but most have an Administrator Mode. Command Prompt is the Windows Shell. When opening it, you can select “Run as Administrator”. This will allow you to perform actions that are normally prohibited by the OP.





Different CLI Shells

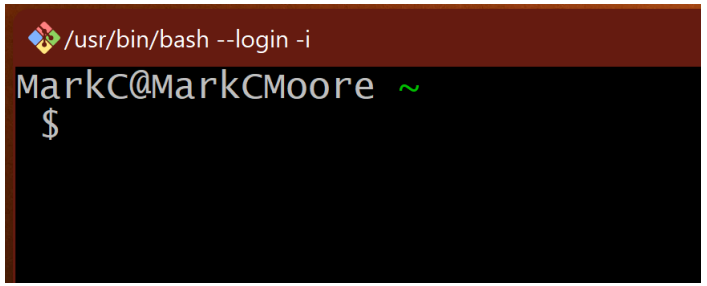
https://opensource.com/resources/what-bash
https://snipcademy.com/linux-command-line-environment
https://devhints.io/bash

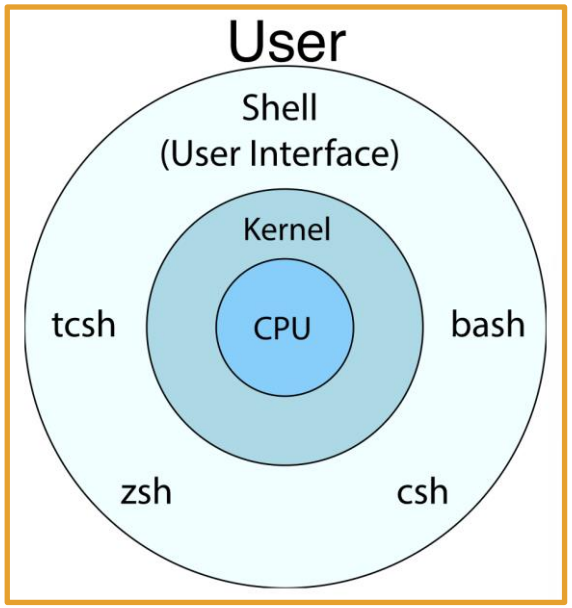
Since a ***Shell*** is just another application, it can be replaced with a different (compatible) ***Shell*** application.

Every OS has a ***Shell***. You can learn about them here.

There are many alternative ***Shell*** applications.

- The most popular ***Shell*** is ***Bourne Again SHell*** (bash).
- ***Bash*** is natively installed on Linux machines (like Macintosh).
- ***Bash***, like every ***Shell***, has its own keywords and syntax.





Emulators

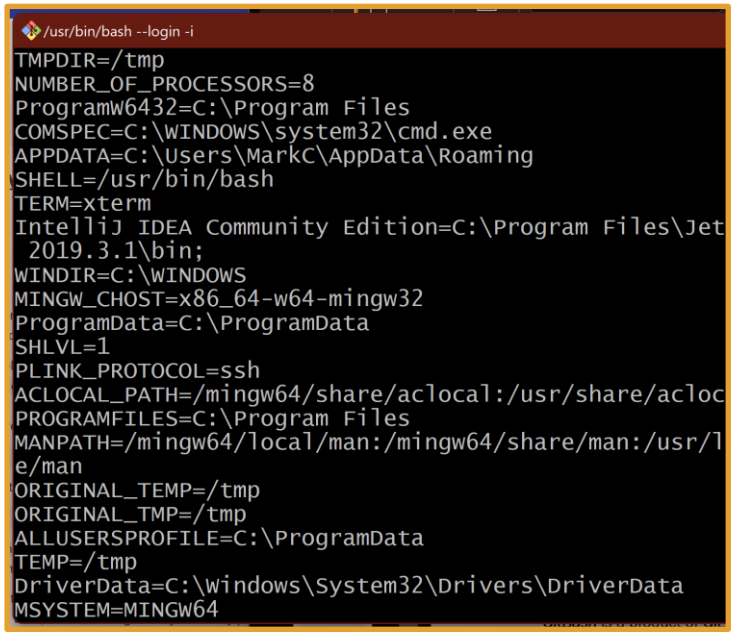
https://en.wikipedia.org/wiki/Emulator
https://snipcademy.com/linux-command-line-environment

"Emulation" is the ability of a computer program to imitate another program. An ***emulator*** enables a host system to run software designed for a guest system.

At Revature, we will use a ***bash*** emulator called ***GitBash*** for the duration of this course.

- ***GitBash*** is a product of ***Git***.
- ***GitBash*** emulates the ***bash CLI.***
- ***GitBash*** allows you to use ***Git*** and ***bash*** commands in the same text-based interface.
- Download ***GitBash*** HERE.

We can use ***GitBash*** on a Windows computer to interact with GitHub.com using ***Git*** commands while also interfacing with the Windows OS ***kernel API*** in the same ***CLI*** window.



Common Bash Commands

https://devhints.io/bash
https://github.com/LeCoupa/awesome-cheatsheets/blob/master/languages/bash.sh

Command	Explanation
ls	Lists the files in the current directory.
cd [~]	Change directory to the Home directory. [] means optional.
cd ..	Changes directory to one level up.
mkdir <dirname>	Creates a new directory with the given name.
touch <filename>	Creates a file with the given name. Updates the last accessed date to the current date.
pwd	Prints the path to the current (working) directory to the CLI.
rm <name>	Deletes the named file.
rmdir <dirname>	Deletes the named directory.
up-arrow	Cycle through all previous commands in reverse order.
cd <dirname>	Move into the named directory.

Common Bash Commands

https://devhints.io/bash
https://github.com/LeCoupa/awesome-cheatsheets/blob/master/languages/bash.sh

Command	Explanation
history	Lists the last 100 commands you've used.
which	Gives which version of
find	
grep	Acronym for "Global search for Regular Expression and Print out". Grep lists the files that match the pattern in the command.
echo	Prints the message or the value of a variable to the console.
cat	Prints the contents of a text file to the console. Cat is short for concatenate. (ex. cat filename.txt)
--help	Will give the options, flags, and format of commands available from this point in the command tree.(ex. dotnet run --help)