**UNIX**

[**Operating systems**](http://computer.howstuffworks.com/operating-system.htm) are computer programs. An operating system is the first piece of software that the computer executes when you turn the machine on. The operating system loads itself into [memory](http://computer.howstuffworks.com/computer-memory.htm) and begins managing the resources available on the computer. It then provides those resources to other applications that the user wants to execute. Typical services that an operating system provides include:

* **A task scheduler** - The task scheduler is able to allocate the execution of the [CPU](http://computer.howstuffworks.com/microprocessor.htm) to a number of different tasks. Some of those tasks are the different applications that the user is running, and some of them are operating system tasks. The task scheduler is the part of the operating system that lets you print a document from your word processor in one window while you are downloading a file in another window and recalculating a spreadsheet in a third window.
* **A memory manager** - The memory manager controls the system's [RAM](http://computer.howstuffworks.com/ram.htm) and normally creates a larger [virtual memory](http://computer.howstuffworks.com/virtual-memory.htm) space using a file on the [hard disk](http://computer.howstuffworks.com/hard-disk.htm). (See also [this Question of the Day](http://computer.howstuffworks.com/question175.htm).)
* **A disk manager** - The disk manager creates and maintains the directories and files on the disk. When you request a file, the disk manager brings it in from the disk.
* **A network manager** - The network manager controls all data moving between the computer and the [network](http://computer.howstuffworks.com/home-network.htm).
* **Other I/O services manager** - The OS manages the [keyboard](http://computer.howstuffworks.com/keyboard.htm), [mouse](http://computer.howstuffworks.com/mouse.htm), [video display](http://computer.howstuffworks.com/monitor.htm), [printers](http://computer.howstuffworks.com/inkjet-printer.htm), etc.
* **Security manager** - The OS maintains the security of the information in the computer's files and controls who can access the computer.

Every desktop computer uses an **operating system**. The most popular operating systems in use today are:

* Windows
* Mac OS
* UNIX.

**What is UNIX?**

**Unix** is a family of [multitasking](https://en.wikipedia.org/wiki/Computer_multitasking), [multiuser](https://en.wikipedia.org/wiki/Multiuser) computer [operating systems](https://en.wikipedia.org/wiki/Operating_system) that derive from the original [AT&T](https://en.wikipedia.org/wiki/AT%26T_Corporation) Unix, developed in the 1970s at the [Bell Labs](https://en.wikipedia.org/wiki/Bell_Labs) research center by [Ken Thompson](https://en.wikipedia.org/wiki/Ken_Thompson), [Dennis Ritchie](https://en.wikipedia.org/wiki/Dennis_Ritchie), and others.

**Linux** is an operating system -- very much like **UNIX** -- that has become very popular over the last several years.

**Why testers need to learn UNIX cmds?**

In most of companies application servers are deployed in unix/linux machine- so as a tester we need to access these machines several times

**Tasks**:

Install/uninstall software or application

View log files

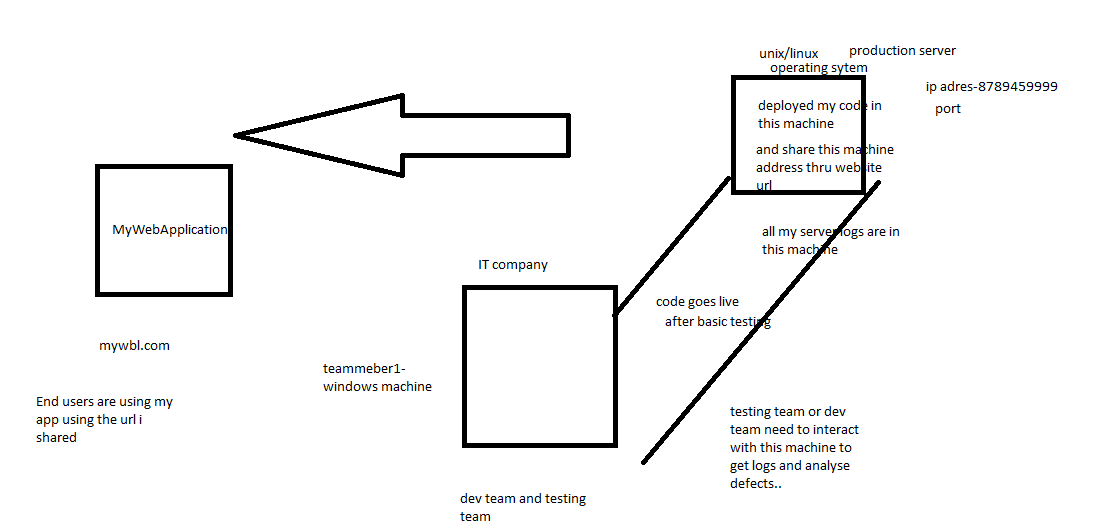
Take backup of log and other files

Cleanup spaces before any installation

Verify memory before any instalation

Verify and kill processes for installation or free memory or process is hanged

Commands used mostly are gunzip , gzip ,gzcat , cat , if else , for , diff , compress ....

****

**How UNIX is different from other OS’s?**

|  |
| --- |
| UNIX vs. Windows Hosting |

In the world of web site hosting there are two main types of operating system platforms on which you may host your web site, namely: **UNIX** and **Windows**. Each has its own set of unique features, advantages and disadvantages.  
  
The language which your site is programmed in is what primarily dictates the type of hosting you need.  
  
**Note**: The operating system that you use on your desktop computer (the vast majority of people use some flavor of Windows) has absolutely nothing to do with the one that your host needs to serve your web site. Most personal sites are created with MS FrontPage and even although that is a Microsoft product, it can be hosted perfectly on a UNIX web server with FrontPage Extensions installed.  
  
  
**Stability:**  
UNIX systems (we actually use Linux but for comparison purposes they are identical) are hands-down the winner in this category. There are many factors here but to name just a couple big ones: in our experience UNIX handles high server loads better than Windows and UNIX machines seldom require reboots while Windows is constantly needing them. Servers running on UNIX enjoy extremely high up-time and high availability/reliability.  
  
**Performance:**  
While there is some debate about which operating system performs better, in our experience both perform comparably in low-stress conditions however UNIX servers under high load (which is what is important) are superior to Windows.  
  
**Scalability:**  
Web sites usually change over time. They start off small and grow as the needs of the person or organization running them grow. While both platforms can often adapt to your growing needs, Windows hosting is more easily made compatible with UNIX-based programming features like PHP and MySQL. UNIX-based web software is not always 100% compatible with Microsoft technologies like .NET and VB development. Therefore if you wish to use these, you should choose Windows web hosting.  
  
**Compatibility:**  
Web sites designed and programmed to be served under a UNIX-based web server can easily be hosted on a Windows server, whereas the reverse is not always true. This makes programming for UNIX the better choice.  
  
**Price:**  
Servers hosting your web site require operating systems and licenses just like everyone else. Windows 2003 and other related applications like SQL Server each cost a significant amount of money; on the other hand, Linux is a free operating system to download, install and operate. Windows hosting results in being a more expensive platform.  
  
**Conclusion:**  
To sum it up, UNIX-based hosting is more stable, performs faster and more compatible than Windows-based hosting. You only need Windows hosting if you are going to developing in .NET or Visual Basic, or some other application that limits your choices.

**What is Kernel and shell?**

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

• 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)

• Everything in UNIX is either a file or a process.

• A process is an executing program identified by a unique PID (process identifier).

• A file is a collection of data. They are created by users using text editors, running compilers etc.

**What are different shells un unix?**

Unix shell scripting - KSH, CSH, SH, BASH etc..

Most shells used in modern UNIX environments are meant to conform to the POSIX sh specification. POSIX sh is derived from the original Korn shell (ksh88), which is in turn derived from the earlier Bourne shell, but POSIX sh only specifies a small subset of even ksh88's functionality. A shell that only implements the minimum requirement is missing many features required for writing all but the most trivial of scripts in a safe and reasonable manner. For example, local variables and arrays are non-standard extras.

Therefore, the first reason is to extend the shell with extra features. Different shells choose to focus on different things. For example, **Zsh focuses on advanced interactive features** while **ksh93 (the current "original" korn shell) focuses on powerful programming features** and performance. Even very minimal shells like Dash add at least a few non-standard extras like local variables.

**Does unix have GUI?Then why cmds?**

### Command Line Power

$ **ls -l | grep "Aug"**

If you don't use Unix command lines, that line probably looks like gibberish. What does it do? It gives you a list of all files that were last modified in August. Maybe you were writing the final files for a year-long project then. Your directory has lots of files; you need to find a couple of them but can't remember their names. The command line runs two programs connected by a *pipe* (which is a lot like a water pipe: it lets data flow between programs). The first program, **ls**, lists the filenames in your current directory; its **-l** ("long") option adds details. The second program, **grep**, filters the output of the first program to show only lines that contain *Aug*, the abbreviation that **ls** uses for August.

If that list is still too long and you remember that you did the work in the middle of August, you could narrow the search. Change the search pattern to match any date between, say, *Aug 10* and *Aug 19*:

http://www.linuxdevcenter.com/pub/a/linux/2001/11/15/learnunixos.html

**What are flavors/varieties of unix?**

|  |  |  |
| --- | --- | --- |
|  | **Linux** | **Unix** |
|  |  |  |
| **Text mode interface** | BASH (Bourne Again SHell) is the Linux default shell. It can support multiple command interpreters. | Originally the Bourne Shell. Now it's compatible with many others including BASH, Korn & C. |
| **GUI** | Linux typically provides two GUIs,[KDE and Gnome](http://www.diffen.com/difference/GNOME_vs_KDE). But there are millions of alternatives such as LXDE, Xfce, Unity, Mate, twm, ect. | Initially Unix was a command based OS, but later a GUI was created called Common Desktop Environment. Most distributions now ship with Gnome. |
| **Operating System Names** | **Distribution**: Redhat Linux, Fedora Linux, Debian Linux, Suse Enterprise Linux, Ubuntu Linux, Mandriva, Slackware, Gentoo linux, PCLinuxOS | HP-UX, IBM AIX, Sun Solaris, Mac OS X, IRIX |

## Common Things between Linux & UNIX

Both share many common applications such as:

1. GUI, file, and windows managers (KDE, Gnome)
2. Shells (ksh, csh, bash)
3. Various office applications such as OpenOffice.org
4. Development tools (perl, php, python, GNU c/c++ compilers)
5. Posix interface(portable os interface)

**Now how to run or where to run cmds to practice them?**

There are many ways you can get/install shell on windows like cygwin, ubuntu on windows or we can do on cloud too..

For mac users- u already have terminal

How to get ubuntu on windows:

If it s windows 10 – directly they gave option to install bash shell

1. Go to run-type settings-go to update settings-for developers- click on developer mode
2. Go to controlpanel- programs-turn on/off windows featutres- click on windows subsystem for linux
3. after restart the machine ..It will ask for uname and pwd..give anything…
4. go to run- type ubuntu—and open the bash shell

if its not windows 10:

you can set up ubuntu virtual machine in windows – by downloading ubntu for windows and installing it – this takes time

cygwin - <http://cygwin.com/> - download 64/32 bit .exe file(choose as per ur configuration) and go to ur downloads folder and run that exe file

vim editor…

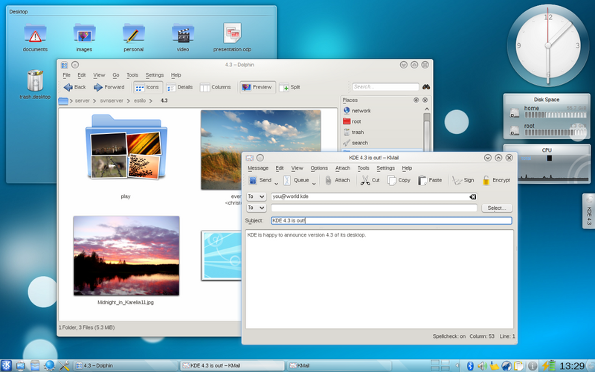
clud shell - https://cloud.google.com/shell/docs/

**In real time they mostly use putty to connect to unix machine from windows**

**A Sample UNIX Desktop Screenshot**

[](http://s0.cyberciti.org/uploads/faq/2006/02/unix-desktop.png)

**A Sample Linux Desktop Screenshot**

[](http://www.cyberciti.biz/faq/what-is-the-difference-between-linux-and-unix/linux-kde-desktop/)

Pwd – present working directory

ls - list of files and directories

ls –l : list of files including the hidden files

mkdir – create directory

cd - change directory

cd .. – go back to previous directory

cd ~ - take you to root dir

cd . – it leaves u in current directory

echo ‘content’ > filename.txt - create a file with content

rm – remove a file

cp – copy file

rmdir – delete a dir but make sure that dir is empty before u use this cmd

mv – cut/move the file and copy to desired locan

grep

head

tail

chmod

man – manual for any command

find

apropos

pipe symbol

input cmd - <

output cmd - >

ps

kill

bg

fg

sleep

who

sort

vi filename.sh

press enter

script editor mode

there to type something – first press i

give all your script cmds

to save and come out:

esc

:

wq

a=10

b=5

a+b-> 10+5