

UNIX INTERVIEW QUESTIONS

Q #1) What is the description for Kernel?

Ans) **Kernel** is the master program with UNIX operating system that controls the resources of the computer. The resources allocation to different users and tasks handle by this section. Kernel do not have direct communication with the user and it starts separate interactive program call shell to each user when login to the system.

Q #2) What is a single user system?

Ans) A personal computer with an operating system which was designed to operate by a single user at a given time. Single user system becomes more popular since low-cost hardware and availability of wide range of software to perform difference tasks.

Q #3) What are the main features of UNIX?

- Machine independent
- Portability
- Multi-user operations
- Unix Shells
- Hierarchical file system
- Pipes and filters
- Background processors
- Utilities
- Development tools.

Q #4) What is called Shell?

Ans) The interface between user and system called a shell. Shell accepts commands and set them to execute for user operations.

Q #5) What are the responsibilities of a shell?

- Program Execution
- Input/output redirection
- Filename and variable substitution
- Pipeline hookup
- Environment control
- Integrated programming language

Q #6) What is the general format of UNIX command syntax?

Ans) In general consideration, **UNIX shell** commands follows the following pattern
Command (-argument) (-argument) (-argument) (file name)

Q #7) Describe the usage and functionality of the command "rm -r *" in UNIX?

Ans) The command "rm -r *" is a single line command to erase all files in a directory with its subdirectories.

- "rm" – Is for deleting files.
- "-r" – Is to delete directories and subdirectories with files within.
- "*" – Is indicate all entries.

Q #8) Describe the term directory in UNIX?

Ans) A specialized form of a file that maintains a list of all the files which include in it is called a directory. Each file assigns to a directory.

Q #9) Specify the difference between absolute path and related path?

Ans) Absolute path refers to the exact path as defined from the root directory. Related path refers to the path related to the current locations.

Q #10) What is the UNIX command to list files/folders in alphabetical order?

Ans) The 'ls -l' command is used to list down files and folders in alphabetical order. When you use 'ls -lt' is list down files /folders sorted with modified time.

Q #11) Describe links and symbolic links in UNIX?

Ans) The second name for a file called a Link. It is used to assign more than one name for a file. It is not valid to assign more than one name to a directory or to link filenames on different computers.

General command '- ln filename1 filename2'

A symbolic link is the files that use to contain only the name of other files include in it. Directed to the files pointed by it is the operation of the symbolic link.

General command '- ln -s filename1 filename2'

Q #12) What is the FIFO?

Ans) FIFO (First In First Out) is also called named pipes and it is a special file for data transient. Data is read only in the written order. This is use to inter-process communications, where data write to one end and reads from another end of the pipe.

Q #13) Describe fork() system call?

Ans) The command use to create a new process from an existing process is called fork(). The main process is called parent process and new process id called child process. The parent gets the child process id returned and the child gets 0. The returned values are used to check which process which code executed. The returned values are used to check which process which code executed.

Q #14) Explain the following sentence?

It is not advisable to use root as the default login.

Ans) The root account is very important and it can leads to system damage easily with abusive usage. So that the securities that normally apply to user accounts are not applicable to the root account.

Q #15) What is mean by Super User?

Ans) The user with access to all files and commands within the system is called super user. Generally, super user login is to root and the login is secured with the root password.

Q #16) What is process group?

Ans) A collection of one or more processes is called process group. There is a unique process id for each process group. The function "getpgrp" returns the process group ID for the calling process.

Q #17) What are the different file types available with UNIX?

- Regular files
- Directory files
- Character special files
- Block special files
- FIFO
- Symbolic links
- Socket

Q #18) What is the behavioral difference between "cmp" and "diff" commands?

Ans) Both commands for file comparison.

Cmp – Compare given two files with byte by byte and display the first mismatch.

Diff – Display changes that need to done to make both file identical.

Q #19) What are the duties of following commands?

Ans) chmod, chown, chgrp

- **chmod** – Change the permission set of the file.
- **chown** – Change ownership of the file.
- **chgrp** – Change group of the file.

Q #20) What is the command to find today's date?

Ans) The command "date" use to retrieve current date.

```
bash-4.3$ date
Sat Jan 21 19:18:57 UTC 2017
bash-4.3$
```

Q #21) What is the purpose of the following command?

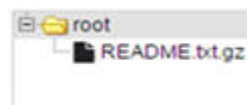
```
sh-4.3$ more README.txt
```

Ans) The command is to display the first part of the file names README.txt which just fit as much as on one screen.

Q #22) Describe the zip/unzip command using gzip?

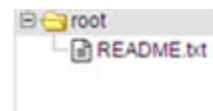
Ans) gzip command creates a zip file using given the filename in the same directory.

```
sh-4.3$ gzip README.txt
sh-4.3$
```



gunzip command unzip the file.

```
sh-4.3$ gunzip README.txt
sh-4.3$
```



Q #23) Explain the method of changing file access permission?

Ans) *There are three sections to consider when creating/changing file access permission.*

- File owner's user ID
- File owner's group ID
- File access mode to define

These three parts arrange as follows.

(User permission) – (Group permission) – (other permission)

Three types of permission can define.

- **r** – Reading permission
- **w** – Writing permission
- **x** – Execution permission

Q #24) How to display the last line of a file?

Ans) This can perform using either "tail" or "sed" commands. The easiest way is to use "tail" command.

```
bash-4.3$ tail -1 README.txt
touch -- Update access and modification time of a file.
bash-4.3$
```

In the above example code, the last line of the README.txt is displayed.

Q #25) What are the various IDs in UNIX processes?

Ans) Process ID is a unique integer that UNIX uses to identify each process. The process executes to initiate other processes is called parent process and its ID is defined as PPID (Parent Process ID).

getppid() – Is to retrieve PPID

Every process is associated with a specific user and is called the owner of the process. The owner has all the privileges over the process. The owner is also the user who executes the process.

Identification for a user is User ID. The process also associated with Effective User ID which determines the access privileges to accessing resources like files.

- **getpid()** – Retrieve process id
- **getuid()** – Retrieve user id
- **geteuid()** – Retrieve effective user id

Q #26) How to Kill a process in UNIX?

Ans) The Kill command accepts process ID (PID) as an in a parameter. This is applicable only for the processes own by the command executor.

Syntax – *kill PID*

Q #27) Explain the advantage of executing processes in the background?

Ans) The general advantage is to execute processes in the background is to get the possibility to execute some other process without waiting for the previous process to get completed. The symbol "&" at the end of the process tells to the shell to execute given a command in the background.

Q #28) What is the command to find maximum memory taking process on the server?

The command top displays the CPU usage, process id, and other details.

Command

```
sh-4.3$ top
```

Output

```
top - 15:23:30 up 47 days, 19:12, 0 users, load average: 1.36, 1.59, 1.32
Tasks: 7 total, 1 running, 6 sleeping, 0 stopped, 0 zombie
%Cpu(s): 3.4 us, 0.1 sy, 0.0 ni, 96.5 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 26386203+total, 20948688+free, 15191116 used, 39184032 buff/cache
KiB Swap: 8388604 total, 8078592 free, 310012 used. 24090606+avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1	14525	20	0	922960	83592	7408	S	0.0	0.0	0:02.39	--CODINGGROUND--
8	14525	20	0	16132	1896	1536	S	0.0	0.0	0:00.00	sh
14	14525	20	0	16128	1624	1360	S	0.0	0.0	0:00.00	bash
15	14525	20	0	16128	1904	1536	S	0.0	0.0	0:00.00	sh
17	14525	20	0	54888	2052	1504	S	0.0	0.0	0:01.33	top
18	14525	20	0	16128	1900	1536	S	0.0	0.0	0:00.00	sh
20	14525	20	0	54888	2048	1504	R	0.0	0.0	0:00.00	top

Q #29) What is the command to find hidden files in the current directory?

Ans) 'ls -lrta' is to display hidden files in current directory.

Command

```
sh-4.3$ /bin/bash
bash-4.3$ ls -lrta
```

Output

```
sh-4.3$ /bin/bash
bash-4.3$ ls -lrta
total 8
drwxrwxrwx 1 cg cg 17 Dec 4 20:14 ..
-rw-r--r-- 1 14682 14682 207 Jan 21 15:29 .cg_conf
drwxrwxrwx 1 cg cg 38 Jan 21 15:29 .
-rw-r--r-- 1 14682 14682 978 Jan 21 15:29 README.txt
bash-4.3$
```

Q #30) What is the command to find the currently running process in Unix Server?

Ans) "ps -ef" command is used to find current running process. Also "grep" with a pipe can use to find specific process.

Command

```
bash-4.3$
bash-4.3$ ps -ef
```

Output

```
bash-4.3$ ps -ef
UID          PID    PPID  C STIME TTY          TIME CMD
14682         1        0  0 15:29 ?           00:00:01 --CODINGGROUND--
14682         8        1  0 15:29 pts/0       00:00:00 sh
14682        14        1  0 15:29 pts/1       00:00:00 sh
14682        16       14  0 15:29 pts/1       00:00:00 /bin/bash
14682        19       16  0 15:48 pts/1       00:00:00 ps -ef
bash-4.3$
```

Q #31) What is the command to find remaining disk space in UNIX server?

Ans) The command "df -kl" use to get a detail description on disk space usage.

Command –

```
bash-4.3$  
bash-4.3$ df -kl
```

Output –

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
overlay	662761284	18710744	644050540	3%	/
tmpfs	131931012	0	131931012	0%	/dev
tmpfs	131931012	0	131931012	0%	/sys/fs/cgroup
/dev/mapper/vg--docker-data	662761284	18710744	644050540	3%	/etc/hosts
shm	65536	0	65536	0%	/dev/shm
tmpfs	131931012	0	131931012	0%	/sys/firmware

```
bash-4.3$
```

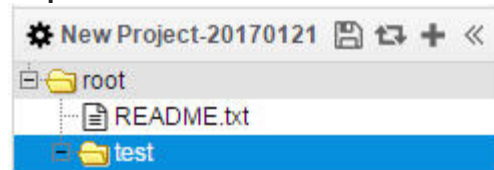
Q #32) What is the UNIX command to make a new directory?

Ans) "mkdir directory_name" is used to create a new directory.

Command –

```
bash-4.3$ mkdir test  
bash-4.3$
```

Output –



Q #33) What is the UNIX command to confirm a remote host is alive or not?

Ans) Either "ping" or "telnet" command can use to confirm a remote host alive or not.

Q #34) What is the method to see command line history?

Ans) The "history" command display all the command used previously within the session.

Command –

```
bash-4.3$ history
```

Output –

```
bash-4.3$ history  
1  ls -lrta  
2  ps -ef  
3  ps -ef  
4  df -kl  
5  df -kl  
6  mkdir test  
7  ping 10.18.243.25  
8  telnet 10.18.243.25  
9  grep  
10 ping 10.18.243.25  
11 history  
12 9  
13 history  
bash-4.3$
```

Q #35) Discuss the difference between swapping and paging?

Ans) Swapping – Complete process is moved to main memory for execution. To provide the memory requirement, process size must be less than the available main memory capacity. The implementation is easy but is an overhead to the system. Memory handling is not more flexible with swapping systems.

Paging – Only the required memory pages are moved to the main memory for execution. The size of the process does not a matter of execution and it no needs to be less than available memory size. Allow a number of processes to load to main memory simultaneously.

Q #36) What is the command to find weather system is 32 bit or 64 bit?

Ans) "arch" or "uname -a" can use for this process.

Command and outcome

(Note: Click on the image for enlarged view)

```
bash-4.3$ arch
x86_64
bash-4.3$ uname -a
Linux 586d9be4f77e 3.10.0-327.36.3.el7.x86_64 #1 SMP Mon Oct 24 16:09:20 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux
bash-4.3$
```

Q #37) Explain 'nohup' in UNIX?

Ans) "nohup" is a special command that is available to run a process in the background. The process started with 'nohup' command is not terminating even the user started to log off from the system.

Q #38) What is the UNIX command to find how many days the server is up?

Ans) "uptime" command returns the number of dates that the server is up.

```
bash-4.3$ uptime
17:07:16 up 47 days, 20:56, 0 users, load average: 0.28, 0.35, 0.51
bash-4.3$
```

Q #39) What is the mode that fault handler executes?

Ans) At the Kernel mode.

Q #40) What is the purpose of "echo" command?

Ans) "echo" command is similar to "ls" command and it displays all the files in current directory.

Q #41) What is the explanation for protection fault?

Ans) When the process access a page, which do not have access permission is refers as protection fault. Also when a process attempt to write on a page whose copy on write bit was set during the fork() system call is incurred for protection fault.

Q #42) What is the method to edit a large file without opening it in UNIX?

Ans) The "sed" command is available for this process 'sed' stands for a stream editor.

Example –

```
bash-4.3$ sed 's/a/aaa/' README.txt
```

Above code will replace from the README.txt file.

```
Following is aaa list of few important Linux/Unix commands.
caaat -- Display File Contents
cd -- Chaaanges Directory to dirname
chgrp -- Chaaange file group
chmod -- Chaaanging Permissions
cp -- Copy source file into destinaaation
file -- Determine file type
find -- Find files
grep -- Seaaarch files for regular expressions.
heaaad -- Display first few lines of a file
ln -- Creaaate softlink on oldname
ls -- Displaay information about file type.
mkdir -- Creaaate a new directory dirname
more -- Displaay data in paginated form.
mv -- Move (Renaaame) a oldname to newname.
pwd -- Print current working directory.
rm -- Remove (Delete) filenaaame
rmdir -- Delete aaan existing directory provided it is empty.
taaaail -- Prints last few lines in a file.
touch -- Updaaate access and modification time of a file.
bash-4.3$ sed 's/aaa/a/' README.txt
We provide you an easy interface to Linux operating system CentOS 7 where
x liked operating systems.
```

Q #43) Describe the concept "Region"?

Ans) Continuous area of processes address space (text, data, and stack) is identifying as a region. Regions are shareable among the processes.

Q #44) What is mean by user area (u-area, u-block)?

Ans) The area is only manipulated by the kernel and it contains the private data. This is unique to the process and each process allocated to u-area.

Q #45) What is called piping?

Ans) "piping" is used to combine two or more commands together. The output of the first command work as the input of the second command, and so on. Pipe character (|) is represent piping.

Q #46) What is the process to count the number of characters and line in a file?

Ans) "wc -c filename" command can use to retrieve the number of characters in a file and "wc -l filename" command can use to retrieve the number of lines in a file.

```
bash-4.3$ wc -c README.txt
978 README.txt
```

Above command returns the number of characters in README.txt file.

```
bash-4.3$ wc -l README.txt
23 README.txt
```

Above command returns the number of characters in README.txt file.

UPDATE: Added more commonly asked Unix questions.

Q #47) What do you understand by UNIX shell?

UNIX shell serves as an environment to run commands, programs, and shell scripts and also acts as an interface between the user and the Unix operating system. Shell issues "\$" as the command prompt, which reads input and determines the command to execute.

For example: \$date

This command will display current date and time.

Some of the most famous shells available with most of Unix variants are Bourne Shell, Korn shell, C Shell.

Q #48) Explain the term filter.

A *Filter* is described as a program, which takes input from the standard input, and displays results to the standard output by performing some actions on it.

Standard Input could be text typed on the keyboard, input from other file or output of other file serving as input. Standard output is y default the display screen.

The most popular example of Unix filter id grep command. This program look for a certain pattern in a file or list of files and only those lines are displayed on the output screen which contains the given pattern.

Syntax: \$grep pattern file(s)

Some of the options that are used along with grep command are enlisted below:

- -v: prints line that does not match the pattern.
- -n: print matched line and line number.
- -l: print file names with matching lines.
- -c: prints only the count of matching lines.
- -i: matches either uppercase or lowercase.

Q #49) Write a command to erase all files in the current directory including all its subdirectories.

"rm -r*" is the command used to erase all files in the current directory including all its subdirectories.

- rm: this command is used for deleting files.
- -r : this option will erase all files in directories and sub-directories.
- '*': this represents all entries.

Q #50) What do understand by Kernel?

Unix operating system is basically divided into three parts, namely, the kernel, the shell and the commands and utilities. *Kernel* serves as the heart of the Unix operating system which does not deal directly with the user but rather act as a separate interactive program for users logged in.

It performs the following functions:

- a) Interacts with the hardware
- b) Perform tasks like memory management, file management and task scheduling.
- c) Control computer resources
- d) Help allotting resources to different tasks and users.

Q #51) Describe key features of Bourne shell.

Bourne shell is referred to as the standard shell. The default prompt here is '\$' character.

The key features of Bourne shell includes:

- a) Input/ Output redirection.
- b) Use of Metacharacters for file name abbreviations.
- c) Using shell variables for the customizing environment.
- d) Creation of programs using built-in the command set.

Q #52) Enlist the key features of Korn Shell.

The *Korn shell* is the most advanced as well as an extension to the Bourne Shell which is backward- compatible.

Some of the features of Korn shell are listed below:

- a) Perform command line editing.
- b) Maintains command history so that the user can check the last command executed if required.
- c) Additional flow control structures.
- d) Debugging primitives who help programmers debug their shell code.
- e) Support for arrays and arithmetic expressions.
- f) Ability to use aliases which is defined as the shorthand names for command.

Q #53) What do you understand by shell variables?

A variable is defined as a character string to which a value is assigned, where values could be the number, text, filename, etc. The shell maintains the set of internal variables as well as enables deletion, assignment, and the creation of variables.

Thus the *shell variables* are a combination of identifiers and assigned values that exist within the shell. These variables are local to the shell in which they are defined as well as they work in a particular way. They may have default value or values can be assigned manually by using appropriate assignment command.

To define a shell variable, 'set' command is used.

To delete a shell variable, 'unset' command is used.

Q #54) Describe responsibilities of Shell in brief.

Apart from analyzing the input line as well as initiating the execution of the program entered by the user, Shell also serves various responsibilities.

Find below a brief description:

a) The shell is responsible for the execution of all the programs by analyzing the line and determining the steps to be performed and then initiate the execution of the selected program.

b) Shell allows you to assign values to the variables when specified on the command line. It also performs Filename substitution.

c) To take care of input and output redirection.

d) Performs pipeline hook-up by connecting the standard output from the command preceding the '|' to the standard input of the one following '|'.
'|'

e) Provides certain commands to customize and control environment.

f) Has its own built-in integrated programming language which is typically easier to debug and modify.

Q #55) Explain file system in UNIX.

A *Filesystem in Unix* is referred to as a functional unit or a logical collection of files, where the disk is set aside to store files and inode entries.

This file system consists of the files that are organized into a multi-level hierarchy called a directory tree.

In other words, the file system is a collection of files and directories and has few below-mentioned features:

a) The very top of the file system is defined as the single directory called 'root' that contains other files and directories and is represented by slash (/).

b) These are self-independent and have no dependencies on other file systems.

c) Every file and directory are uniquely identified by:

- Name
- Directory in which it resides
- A unique identifier

d) All files are organized into a multi-level directory known as 'Directory tree'.

Q #56) What do you understand by command substitution?

Command substitution is the method which is performed every time the commands that are enclosed in backquotes are processed by the shell. This process replaces the standard output and displays on the command line.

Command substitution can perform the following tasks:

- Invoke subshell
- Result in word splitting
- Remove trailing new lines
- By using 'redirection' and 'cat' command, allows setting a variable to the content of the file.
- Allows setting a variable to the output of the loop

Q #57) Define inode.

Whenever a file is created inside a directory, it accesses the two attributes, namely, file name and Inode number. The file name is first mapped with Inode number stored in the table and then this Inode number serves as a medium to access Inode. Thus Inode can be defined as an entry created and set aside on a section of the disk for a file system.

Inode serves as a data structure and nearly stores each information that is required to be known about a file.

This information includes:

- a) File location on the disk
- b) Size of the file
- c) Device Id and Group Id
- d) File mode information
- e) File protection flags
- f) Access privileges for owner, group.
- g) Timestamps for file creation, modifications, etc.

Q #58) Enlist common shells with their indicators.

Enlisted below are the common shells with their indicators:

Shell	Indicators
Bourne Shell	sh
C Shell	csch
Bourne Again shell	Bash
Enhanced C shell	tcsh
Z Shell	zsh
Korn Shell	ksh

Q #59) Enlist some commonly used network commands.

Some commonly used networking commands in Unix are enlisted below:

- a) telnet: it is used for remote login as well as for communication with another hostname.
- b) ping: it is defined as an echo request for checking network connectivity.
- c) su: derived as a user switching command.
- d) hostname: determines the Ip address and domain name.
- e) nslookup: performs DNS query.
- f) traceroute: method to determine the number of hops and response time required to reach the network host.
- g) netstat: it provides a lot of information like ongoing network connection on the local system and ports, routing tables, interfaces statistics, etc.

Q #60) How is cmp command different from diff command?

'*cmp*' command is basically used for byte by byte comparison of two files to determine the first mismatch byte. This command does not use the directory name and displays the first encountered mismatched byte. Whereas, '*diff*' command determines the changes that are to be performed on the files in order to make the two files identical. In this case, directory names can be used.

Q #61) What is the role of the superuser?

There are basically three types of accounts in Unix operating system:

- Root account
- System accounts
- User accounts

'Root account' is basically referred to as a 'Super user'. This user has completely open access or says control on all files and commands on a system. This user can also be assumed as a system administrator and thus has the ability to run any command without any restriction. It is protected by the root password.

Q #62) Define piping.

When two or more commands are required to be used at the same time as well as run them consecutively, 'piping' process is used. Here two commands are connected so that, the output of one program serves as the input for another program. It is denoted by the symbol '|'.

Enlisted below are few commands where piping is used:

- grep command: searches files for certain matching pattern.
- sort command: arranges lines of text alphabetically or numerically.

Q #63) Explain the types of path names that can be used in UNIX.

In a file system for any operating system, there exists the hierarchy of directories, there 'Path' is defined as the unique location to a file/ directory to access it. There are basically two types of the path that are used in Unix. These can be defined as follows:

There are basically two types of the path that are used in Unix. These can be defined as follows:

a) *Absolute Pathname*: It defines a complete path specifying the location of a file/ directory from the very start of the actual file system i.e. from the root directory (/). Absolute pathname addresses system configuration files that do not change location. It defines a complete path specifying the location of a file/ directory from the very start of the actual file system i.e. from the root directory (/). Absolute pathname addresses system configuration files that do not change location.

b) *Relative Pathname*: It defines the path from the current working directory where the user is i.e. the present working directory (pwd). Relative pathname signifies current directory, parent directory as well as also refers to file that are either impossible or inconvenient to access. It defines the path from the current working directory where the user is i.e. the present working directory (pwd). Relative pathname signifies current directory, parent directory as well as also refers to file that are either impossible or inconvenient to access.

Q #64) Explain Superblock in UNIX?

Each logical partitions in Unix are referred to as File system and each file system contains, a 'boot block', a 'super block', 'inodes', and 'data blocks'. Super block is created at the time of creation of file system. It describes the following: *Super block* is created at the time of creation of file system.

It describes the following:

- State of the file system
- Total size of the partition
- Block size
- Magic number
- Inode number of the root directory
- Count of the number of files, etc

There are basically two types of superblocks:

a) *Default superblock*: It has its existence always as a fix offset from the beginning of the system's disk partition.

b) *Redundant superblock*: It is referenced when the default superblock is affected by system crash or some errors.

Q #65) Enlist some filename manipulation commands in UNIX.

Some Filename manipulation commands along with their description is enlisted below in table:

Command	Description
cat filename	Displays contents of the file
cp source destination	Used to copy source file into the destination
mv old name new name	Move/rename and old name to the new name
rm filename	Remove/delete filename
Touch filename	Changing modification time
In [-s] old name new name	Creates soft link on old name
Is -F	Displays information about file type

Q #66) Explain links and symbolic links.

Links are defined as a second name which is used to assign more than one name to a file. Although links are referred to as a pointer to another file it cannot be used to link filenames on different computers.

A *Symbolic link* is also known as the soft link is defined as a special type of file that contains links or references to another file or directory in the form of absolute or relative path. It does not contain the data actually in the target file but the pointer to another entry in the file system. Symbolic links can also be used to create a file system.

The following command is used to create a symbolic link:

- Ln -s target link_name
- Here, path is 'target'
- Name of the link is represented by link_name.

Q #67) Explain alias mechanism.

To avoid typing long commands or to improve efficiency, the alias command is used to assign another name to a command. Basically, it acts as a shortcut to the larger commands which can be typed and run instead.

For creating an alias in Unix, following command format is used:

```
alias name='command you want to run'
```

Here, replace the 'name' with your shortcut command and replace 'command you want to run' with the larger command of which you want to create an alias of.

For example:

```
alias dir 'Is -sFC'
```

here, in the above example 'dir' is the another name for the command 'Is-sFC'. Thus user now simply is required to remember and use the specified alias name and the command will perform the same task as to be performed by the long command.

Q #68) What do you know about wildcard interpretation?

Wildcard characters are some special kind of characters that represent one or more other characters. *Wildcard interpretation* comes into picture when a command line contains these characters. In this case, when the pattern matches the input command, these characters are replaced by sorted list of files.

Asterisk()* and *Question mark(?)* are usually used as wildcard characters to set up a list of files while processing.

Q #69) What do you understand by terms 'system calls' and 'library functions' with respect to UNIX command?

System calls: As the name implies, system calls are defined as an interface which basically is used in the kernel itself. Although, they may not be fully portable but these calls request the operating system to perform tasks on behalf of user programs.

The system calls appear as a normal C function. Whenever a system call is invoked within the operating system, the application program performs context switch from user space to kernel space.

Library functions: The set of common functions that are not part of the kernel but is used by the application programs are known as 'Library functions'. As compared to system calls, library functions are portable and can perform certain tasks only in 'kernel mode'. Also, it takes lesser time for execution as compared to the execution of system calls.

Q #70) Explain pid.

A *pid* is used to denote unique process id. It basically identifies all the processes that run on the Unix system. It does not matter whether the processes are running in the foreground or in the background.

Q #71) What are the possible return values of kill() system call?

Kill() system call is used to send signals to any processes.

This method returns the following return values:

- a) Returns 0 : It implies that process exists with the given pid and system allows sending signals to it.
- b) Return -1 and errno==ESRCH: It implies that there is no existence of the process with specified pid. There may also exist some security reasons which is denying the existence of the pid.
- c) Return -1 and errno==EPERM: It implies that there is no permit available for the process to be killed. The error also detects whether the process is present or not.
- d) EINVAL : it implies invalid signal.

Q #72) Enlist the various commands that are used to know about the user information in UNIX.

The various commands that are used for displaying the user information in Unix are enlisted below:

- a) Id: displays the active user id with login and group.
- b) Last: displays the last login of the user in the system.
- c) Who: determines who is logged onto the system.
- d) groupadd admin: this command is used to add group 'admin'.
- e) Usermod -a: user to add existing user to the group.

Q #73) What do know about tee command and its usage?

'tee' command is basically used in connection with pipes and filters.

This command basically performs two tasks:

- a) Get data from standard input and send it to the standard output.
- b) Redirects a copy of the input data to the specified file.

Q #74) Explain mount and unmount command.

Mount command: As the name suggests, the mount command mounts a storage device or file system onto an existing directory and thus making it accessible to users.

Unmount command: This command unmounts the mounted file system by safely detaching it. It also the task of this command to inform the system to complete any pending read and write operations.

Q #75) What is "chmod" command?

Chmod command is used to change file or directory access permission and is the most frequently used command in Unix. According to mode, chmod command changes the permission of each given file.

The syntax of chmod command is:

Chmod [options] mode filename.

Here in the above format, options could be:

- -R: recursively change the permission of file or directory.
- -v: verbose, i.e. output a diagnostic for every file processed.
- -c: report only when the change is made.
- Etc.

Q #76) Differentiate Swapping and Paging.

The difference between Swapping and Paging can be seen in the below table:

Swapping	Paging
It the procedure of copying the entire process from main memory onto secondary memory.	It is a memory allocation technique where the process is allocated memory wherever available.
For execution, the whole process is moved from swap device to the main memory.	For execution, only the required memory pages are moved from swap device to the main memory.
The than main memory.process size must be equal to or less	The process size does not matter in this case.
It cannot handle the memory flexibly.	It can handle the memory more flexibly.