Top 95 Shell Scripting Interview Questions & Answers

By **Great Learning Team** - Apr 27, 2021

Introduction to Shell Scripting

The computer only knows binary language. Earlier, programs were written in binary language, and not in a human-readable format. It was very difficult to read and write code. People started developing kernels and similar things such as compiler, interpreter, editor, which were the core part of the operating system. A compiler used to understand the human-readable format of a specific program and then convert it into machine code. After that, many programming languages came into existence. Now, according to Wikipedia, there are more than 600 programming languages available in the world.

The first shell developed by Ken Thompson for UNIX in 1971. In 1977, Bourne shell was created by Stephen Bourne at bell labs for UNIX. It is a useful shell today. In some Linux systems, it is a default shell. Tenex C shell Korn shell, the Bourne-again shell, exotic shell are the different types of shell. Through this blog, we will learn the top shell scripting interview questions and answers.

Career growth

There are a lot of opportunities from many reputed companies in the world. According to research, Linux Shell Scripting has a market share of about 17. Since 2018, Linux has started to expand its market rapidly. According to PayScale, the average salary for shell scripting skill is $81,951. If you want career growth in DevOps, System administrator, learning shell scripting interview questions would be a great start.

Why Linux Shell Scripting?

Shell scripting makes work easy and reduces time. You can do almost anything by writing a script and automating your task. You can manage multiple systems and servers at a time using shell scripting. There are a lot of advantages of shell scripting, like automating your daily tasks by writing several words or scripts. It combines multiple commands in a single command, and it is easy to write when compared to other programming languages. Before we move on to the shell scripting interview questions, let us look at some examples.

Daily use Examples of Shell scripting by System Admins

- Monitoring your Linux system.
- Data backup and creating snapshots.
- Dumping Oracle or MySQL database for backup.
- Creating email based alert systems.
- Find out what processes are eating up your system resources.
- Find out available and free memory.
- Find out all logged in users and what they are doing.
- Find out if all necessary network services are running or not. For example if the web server failed then send an alert to the system administrator via a pager or an email.
- Find out all failed login attempts, if login attempts are continued repeatedly from the same network IP automatically block all those IPs accessing your network/service via firewall.
- User administration as per your own security policies.
- Find out information about local or remote servers.
- Server configuration.

Shell Scripting Interview Questions

1. What is Linux?

It is a family of **open-source** Unix operating systems based on the Linux kernel.

2. Difference between Linux and Unix?

Linux is a clone of **Unix**. **Linux** default shell is BASH (Bourne Again Shell) while the **Unix** shell is Bourne Shell.

3. What is a kernel?

The kernel is a computer program at the core of a computer's operating system that manages operations of computer and hardware.

4. What is an interpreter?

Interpreter translates the program line by line into machine code.

5. What is a compiler ?

Compiler scans the whole program and converts it into machine code.

6. What is a shell?

Shell is a program and command line interpreter. It is an interface between user and kernel.

7. What is CLI and GUI?

CLI is a command line interface. This user interface enables the user to give commands to interact with the device.

GUI is a graphical user interface. This user interface enables users to interact with devices with the help of graphical icons and visual indicators.

8. Why would we use CLI over GUI?

- CLI gives better control to the user.
- CLI is a best option for professionals who work on more programming languages.
- It required less memory as compared to GUI.
- The speed of the CLI is faster than GUI.

9. What is shell scripting?

It is a text file which contains list or series of command or statements to be executed

10. What is the default login shell? How to change it?

/bin/bash is the default login shell. Using the command "chsh" we can change the default shell.

11. What is the importance of shell scripting?

- If you need to perform the same task repeatedly, you should use shell scripting
- By using shell scripting, you can make your own tools
- It is very useful for a system admin to automate daily tasks

12. What are the various stages of the Linux process?

- Waiting: The process waits for resources
- Running: The process is currently being executed
- Stop: The linux process stop after execution
- Zombie: The process has stopped but it is still active in process table

13. What is the main difference between BASH and DOS?

BASH commands are case sensitive but DOS commands are not case sensitive.

14. What are the components of Linux ?

Kernels, shells, GUI, system utilities and application programs are components of Linux.

15. What is a root user?

It is an admin user that allows you full control of your system.

16. What are the environmental variables?

Environmental variable control shell function as well as other Linux programs.

17. What is LILO?

LILO is a boot loader used in Linux .It is used to load the operating system into main memory to start its operation.

18. What are the different types of commonly used shells on a Linux system?

- **Bash/Bourne Again Shell**: This is the most common shell available on all Linux and based systems. It is open source and freeware.
- **CSH or C Shell**: This Shell scripting program uses the C programming's shell syntax and it's almost similar to C.
- **KSH or Korn Shell**: Korn is a Unix based Shell scripting program, initially based on the Bash Shell Scripting. This shell is quite advanced and it's a high level programming language.
- **TCSH**: There is no specific full form of TCSH. It is as it is. TCSH is an advanced version of Berkeley Unix C shell. It again supports C style syntax.

19. Which command is used to execute a shell file?

First Set execute permission on your script using chmod command
chmod +x script-name-here.sh

To run your script:
./script-name-here.sh
Another option to execute shell script:
sh script-name-here.sh

20. Name of Editors which are available in almost all UNIX?

vi/vim

21. What is interactive and non-interactive shell?

/bin/bash and /bin/sh is interactive shell

/sbin/nologin shell is non-interactive shell

22. What is the absolute and relative path?

Absolute path is the full path of the directory. It always starts with "/" .

eg. cd /var/tmp/abrt/

Relative path is necessary from current location to reach particular directory doesn't start with "/".

eg. cd .. , cd –

23. How to create, read and delete files?

touch command is used for creating files. eg. #touch filename

cat command is used for reading files. eg. #cat filename

rm command is used for delete a file eg. #rm –f filename

24. How to create and delete a directory ?

mkdir command is used for creating a directory. Eg. # mkdir filename

rmdir command is used for remove directory eg. #rmdir filename

25. How to create multiple text files and directories?

To create multiple text file touch file name {} command is used

eg. suppose we want create 4 files then we type #touch filename{1..4}

To create multiple directory mkdir filename {} command is used

eg. suppose we want to create 4 directory then we type mkdir filename {1..4}

26. What is the use of head and tell command?

Head command is used for display started 10 lines

Tail command is used for display started 10 lines

27. How to find the current shell which you are using?

$echo $SHELL command is used for find current shell

28. How to find an available shell in your system?

Cat /etc/shells command is used to find available shells in your system.

29. How to create shortcuts in Linux?

To create shortcut "link" command use. There are two types of link: hard link and soft link.

30. Tell me the difference between hard link and soft link?

Deleting the original file does not affect the hard link but Deleting the original file makes the soft link inactive.

31. How will you pass and access arguments to a script in Linux?

For pass arguments in script "scriptname arg1 arg2 arg3 …"

For access arguments in script can be accessed inside the script as "$1 , $2 .. $n"

32. What is the significance of $#?

It represents the total number of arguments passed by string.

33. What is the difference between $* and $@?

$* consider the entire set of positional parameters as a single string but $@ treat each quoted argument as a separate argument.

34. Explain "s" permission bit in a file?

"s" bit also called "set user id"(SUID) bit. "s" on file causes the process to have the privileges of the owner of the file during the instance of the program.

35. What are the different types of variables used in Shell Script?

System defined variable : system defined variable created by os itself. These variables are generally defined in capital letters. It can be viewed by the "set" command.

User defined variable : it created by system users. Value of variable can be view by using "echo $variablename" command.

36. What is the difference between = and ==?

= use for assign value to variable

== use for string comparison

37. What is the use of a pipe operator? How to execute multiple commands in one line?

The pipe operator is used for one by one execution of command but commands should not be dependent on each other.

38. What are the different modes of vi editors?

Command mode : this is a mode where you start

Edit mode : this mode allows you to do next editing.

Ex mode : In this mode you interact with vi with instruction to process

39. What is redirection?

Redirection is the process of direction data from one output to another.

40. How to find the status of the process?

Ps ux command user for find status of process.

41. How to check memory status?

Free command is useful for checking memory status.

42. How to debug a shell script?

To debug a shell script we execute the script with the "-x" or "-nv" option.

43. Which command is used for comparing the string in the shell script?

To compare the string "test" command is used.

44. What is the difference between $! And $$?

$! Shows process id of the process that recently went into background

$$ gives the process id of the currently executing process

45. Which command is used to find all information of the user?

"finger" command shows all information of users.

46. Which four fundamental components of every file system?

Boot block: it contains a small program called MBR which loads the kernel during system boot up.

Super block: super block contains all information about the file system.

Inode block: it contains inode for every file of the file system.

47. What is the Crontab?

Crontab stands for ***cron table*** because it uses the job scheduler ***cron*** to execute tasks. The crontab is a list of commands that you want to run on a regular schedule, and also the name of the command used to manage that list.

48. How many fields are present in a crontab file?

The **five fields** contain information on ***when to execute the command*** .

- minute(0-59)
- hour(0-23)
- day(1-31)
- month(1-12)
- day of the week(0-6, Sunday = 0).

49. What are the two files of crontab command?

cron.allow which decides the users need to be ***permitted*** for using the **crontab command**.

cron.deny which decides the users need to be ***prevented*** from using the **crontab command**

50. What are the different commands available to check the disk usage?

- **df:** It is used to check the free disk space.
- **du:** It is used to check the directory wise disk usage.
- **dfspace:** It is used to check the free disk space in terms of MB.

51. How to open a read-only file in the Shell?

vi –R <Filename>

52. How to find out how long the system has been running?

by using the command "uptime".

53. How to connect to a remote server and execute some commands?

We can use ssh to do this:

ssh username@serverIP -p sshport

Once the above command is executed, enter the password.

54. How to connect to a database server from Linux?

mysql –S serverName –U username –P password

55. How can I set the default permission to all users on every file which is created in the current shell?

umask 777

56. How will I insert a line "ABCD" at every 50th line of a file?

sed '50i\ABCD' filename

57. How to find the total disk space used by a specific user?

du -s /home/username

58. How to print the login names of all users on a system?

/etc/shadow file has all the users listed.

awk –F ‘:’ ‘{print $1}’ /etc/shadow|uniq –u

59. Write a shell script to get current date, time, username and current working directory.

```
#!/bin/sh
echo “Hello, $LOGNAME”
echo “Today’s date is `date`”
echo “Username is `who i am`”
echo “Current directory is `pwd`”
```

60. How to check if a directory exists?

```
#!/bin/sh
if [ -d $mydir ]
then
echo "Directory exists"
fi
```

61. Explain the file permissions.

r – read   4

w- write   2

e- execute   1

62. Given a file, replace all occurrence of word “ABC” with “DEF” from 10th line till end in only those lines that contains word “MNO”

sed –n ’10,$p’ file1|sed ‘/MNO/s/ABC/DEF/’

63. How will you find the 19th line of a file using only tail and head command?

tail +19 file1|head -1

64. How to Use the sed command to replace the content of the file?

```
if cat fille
ABCD
EFGH
Then O/p should be
EFGH
ABCD

sed '1! G; h;$!d' file1
```

65. I want to create a directory such that anyone in the group can create a file and access any person’s file in it but none should be able to delete a file other than the one created by himself.

We can create the directory giving read and execute access to everyone in the group and setting its sticky bit “t” on as follows:

```
mkdir direc1

chmod g+wx direc1

chmod +t direc1
```

66. How to get the 3rd element/column from each line from a file?

```
#!/bin/sh
awk ‘{print $3}’ $1
```

67. Write down the Syntax for all the loops in Shell Scripting.

For Loop:

```
for var in word1 word2 … wordn

do

Statement(s) to be executed for every word.

done
```

While Loop:

```
while command

do

Statement(s) to be executed if command is true

Done
```

Until Loop:

```
until command

do
```

Statement(s) to be executed until command is true

done

68. How to pass arguments to a script?

./script argument used for passed argument to a script.

Example:

```
./script.sh  file.txt
cat script.sh
#!bin/bash
Cat $1
```

69. How to use arguments in the script?

In the script we use first argument as $1 and second argument as $2

Example: to move file one destination($1) to another($2)

```
./move.sh file.txt /text
cat move.sh
#!/bin/bash
mv $1 $2
```

70. How to get 5rd element from each line from the file?

awk '{print $5}'

71. how to find process name from process ID?

"ps –p pid" command used to find the process name.

72. How to create alias command in shell?

Alias name="command whose alias to be created"

73 . What is the c and b permission field of the file?

C specifies whether the file is character special file or block special file.

74. How to get 5rd element from each line from the file?

awk '{print $5}'

75. How to find process name from process id?

"ps –p pid" command used to find the process name.

76. What is the use of a shebang line?

Shebang line at the top of each script determines the location of the engine which is to be used in order to execute the script.

77. How to add two strings?

```
S1=”hello”
S2=”world”
Let s3=$s1+$s2
Echo $s3
```

78. How to add two integers?

```
Int1=3
Int2=2
Int3=$int1+$int2
Echo $int3
```

79. Write a script to check if a file exists on the system?

```
If [-f /var/www/html]
Then
Echo “file exists”
Fi
```

80. Why #!/bin/sh or #!/bin/bash at the beginning of every script?

That line tells which shell to use.

81. From given file name find the count of lines containing word “ABC”

grep –c “ABC” filename

82. What would be the output of command: echo ${new:-variable}

variable

83. how to print all array indexes?

echo ${!array[@]} used to print all array indexes.

84. How to remove elements from an array with id 3?

Unset array[3]

85. Write a script to compare numbers?

```
#!/bin/bash
X=10
Y=20
If [ $x -gt $y ]
Then
```

```
Echo “ x is greater than y”
Else
Echo “y is greater than x”
Fi
```

86. What are the types of permission at the file level in Shell?

Owner permissions: The permissions granted to a user with all the rights at the root level.

Group permissions: The permissions granted to a user with all the rights in a particular group.

Other permissions: The permissions granted to a user with all the rights globally to a file.

87. How to use comments in shell scripting?

```
#! /bin/bash
# addition
((sum=30+20))
#print
Echo$sum
```

88. How to use multi-line comments in shell scripting?

```
#! /bin/bash
: ‘
This script calculates squares of 8.
‘
((square=8*8))
echo $area
```

89. Write a script that receive input from user.

```
#! /bin/bash
Echo  -n  “enter input:”
read input
echo “ you entered: $input”
```

90. Write a script using the AND operator.

```
#! /bin/bash
Echo -n “enter number:”
read number
if  [[ ($number  -lt 10 )  && ($number%2  -eq  0)]]
echo  “even number”
else
echo “odd number”
fi
```

91. Write an example of OR operator.

```
#! /bin/bash
Echo -n “enter any number : “
read n
```

```
if  [[ ( $n –eq 10 || $n  -eq 45) ]]
then
echo “ you win”

else  “ you lost!”
fi
example of elif
#! /bin/bash
Echo –n “enter number :”
read number
if  [[ $number  -gt 10]]
then
echo “number  is greater than 10”
elif  [[ $num –eq 10 ]]
then
echo ‘number is equal to 10”
else
echo “ number  is less than 10”
fi
```

92. Write a script for adding multiple values.

```
#! /bin/bash
Sum=0
For (( counter=1 ; counter<5 ;counter++))
Do
Echo –n “enter your number”
read n
(( sum+=n))
#echo –n “$counter”
Done
Printf   “\n”
echo  “result is : $ sum”
```

93. Write an example of a function.

```
#! /bin/bash
Function Add()
{
echo  –n “enter number:”
read a
echo  –n  “enter 2nd number :”
read b
echo “addition  is: $((  a+b ))”
}
Add
```

94. How to send mail using shell script?

```
#! /bin/bash
Recipientadmin@example.com
Subject=”script”
Message=”this mail send by script”
‘mail –s $subject $recipient <<< $message’
```

95. Write script to print current date and time.

```
#! /bin/bash
year= ‘date +%Y’
month=’date  +%m’
day=’date +%d’
hour=’date +% H’
minute=’date +%M’
second=’date +%S’
echo ‘date’
echo  “current date is :$day-$month=$year”
echo  “current time is :$hour:$minute:$second”
```

This brings us to the end of the blog on shell scripting interview questions. We hope that you were able to benefit from this blog on shell scripting interview questions and are now better-equipped to attend your upcoming interviews.

If you wish to upskill in this domain and learn more concepts, check out the various Free Online Courses offered by Great Learning Academy.

Great Learning Team

Great Learning's Blog covers the latest developments and innovations in technology that can be leveraged to build rewarding careers. You'll find career guides, tech tutorials and industry news to keep yourself updated with the fast-changing world of tech and business.