

Home
 How To
 Commands
 Shell Scripting
 Linux Interview
 Contact Us
 About Us
 Write For LinuxTechi

SHELL SCRIPTING

1

FOLLOW:

25 Linux Shell Scripting interview Questions & Answers

SUBSCRIBE TO LINUXTECHI

GO

FOLLOW US

BY [PRADEEP KUMAR](#) · FEBRUARY 2, 2015

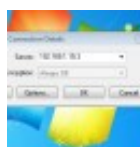
Q:1 What is Shell Script and why it is required ?

Ans: A Shell Script is a text file that contains one or more commands. As a system administrator we often need to

Recent

Comments

Tags



Install & Configure VNC Server in CentOS 7 and RHEL 7

August 18, 2015

10 YaST2 Command Examples in SUSE Linux

issue number of commands to accomplish the task, we can add these all commands together in a text file (Shell Script) to complete daily routine task.

Q:2 What is the default login shell and how to change default login shell for a specific user ?

Ans: In Linux like Operating system “/bin/bash” is the default login shell which is assigned while user creation. We can change default shell using the “chsh” command . Example is shown below :

```
# chsh <username> -s
<new_default_shell>
# chsh linuxtech -s /bin/sh
```

Q:3 What are the different type of variables used in a shell Script ?

Ans: In a shell script we can use two types of variables :

- System defined variables
- User defined variables

System defined variables are defined or created by



August 11, 2015



Installation Steps of VirtualBox on OpenSUSE 13.1 & 13.2

August 4, 2015



How to Install and use Lsyncd on CentOS 7 & RHEL 7

July 26, 2015



How to Install TeamViewer On Ubuntu 15.04

June 20, 2015



Extend Swap Space using Swap file in Linux

June 14, 2015



Command Line Arguments in Linux Shell Scripting

June 2, 2015



Linux Real Time scenarios & Issues with their solutions

May 24, 2015

[Next »](#)

LIKE US ON FACEBOOK

Operating System(Linux) itself.

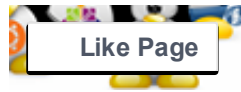
These variables are generally defined in Capital Letters and can be viewed by “**set**” command.

User defined variables are created or defined by system users and the values of variables can be viewed by using the command “**echo \$<Name_of_Variable>**”



Linuxtech

337 likes



guide & Linux interview Questions

Share



Q:4 How to redirect both standard output and standard error to the same location ?

Ans: There two method to redirect std output and std error to the same location:

Method:1 **2>&1** (# ls
/usr/share/doc > out.txt 2>&1)

Method:2 **&>** (# ls
/usr/share/doc &> out.txt)

Q:5 What is the Syntax of “nested if statement” in shell scripting ?

Ans : Basic Syntax is shown below :

if [Condition]
then

```
command1
command2
.....
else
if [ condition ]
then
command1
command2
....
else
command1
command2
.....
fi
fi
```

Q:6 What is the use of “\$?” sign in shell script ?

Ans:While writing a shell script , if you want to check whether previous command is executed successfully or not , then we can use “\$?” with if statement to check the exit status of previous command. Basic example is shown below :

```
root@localhost:~# ls
/usr/bin/shar
/usr/bin/shar
root@localhost:~# echo $?
0
```

If exit status is 0 , then
command is executed
successfully

```
root@localhost:~# ls  
/usr/bin/share
```

```
ls: cannot access  
/usr/bin/share: No such file or  
directory
```

```
root@localhost:~# echo $?  
2
```

If the exit status is other than 0,
then we can say command is
not executed successfully.

Q:7 How to compare numbers in Linux shell Scripting ?

Ans: test command is used to
compare numbers in if-then
statement. Example is shown
below :

```
#!/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

Q:8 What is the use of break command ?

Ans: The break command is a simple way to escape out of a loop in progress. We can use the break command to exit out from any loop, including while and until loops.

Q:9 What is the use of continue command in shell scripting ?

Ans The continue command is identical to break command except it causes the present iteration of the loop to exit, instead of the entire loop. Continue command is useful in some scenarios where error has occurred but we still want to execute the next commands of the loop.

Q:10 Tell me the Syntax of “Case statement” in Linux shell scripting ?

Ans: The basic syntax is shown below :

case word in
value1)

```
command1
command2
.....
last_command
!!
value2)
command1
command2
.....
last_command
;;
esac
```

Q:11 What is the basic syntax of while loop in shell scripting ?

Ans: Like the for loop, the while loop repeats its block of commands a number of times. Unlike the for loop, however, the while loop iterates until its while condition is no longer true. The basic syntax is :

```
while [ test_condition ]
do
commands...
done
```

Q:12 How to make a shell script executable ?

Ans: Using the chmod command we can make a shell

script executable. Example is shown below :

```
# chmod a+x myscript.sh
```

Q:13 What is the use of “#!/bin/bash” ?

Ans: #!/bin/bash is the first of a shell script , known as shebang , where # symbol is called hash and ‘!’ is called as bang. It shows that command to be executed via /bin/bash.

Q:14 What is the syntax of for loop in shell script ?

Ans: Basic Syntax of for loop is given below :

```
for variables in list_of_items  
do  
command1  
command2  
....  
last_command  
done
```

Q:15 How to debug a shell script ?

Ans: A shell script can be debug if we execute the script with ‘-x’ option (sh -x myscript.sh). Another way to

debug a shell script is by using
'-nv' option (sh -nv
myscript.sh).

Q:16 How compare the strings in shell script ?

Ans: test command is used to compare the text strings. The test command compares text strings by comparing each character in each string.

Q:17 What are the Special Variables set by Bourne shell for command line arguments ?

Ans: The following table lists the special variables set by the Bourne shell for command line arguments .

Special Variables	Holds
\$0	Name of the Script from the command line
\$1	First Command-line argument
\$2	Second Command-line argument

.....
\$9	Ninth Command line argument
\$#	Number of Command line arguments
\$*	All Command-line arguments, separated with spaces

Q:18 How to test files in a shell script ?

Ans: test command is used to perform different test on the files. Basic test are listed below :

Test	Usage
-d file_name	Returns true if the file exists and is a directory
-e file_name	Returns true if the file exists
-f file_name	Returns true if the file exists and is a regular file
	Returns true if the file

-r file_name exists and have read permissions

-s file_name Returns true if the file exists and is not empty

-w file_name Returns true if the file exists and have write permissions

-x file_name Returns true if the file exists and have execute permissions

Q:19 How to put comments in your shell script ?

Ans: Comments are the messages to yourself and for other users that describe what a script is supposed to do and how its works.To put comments in your script, start each comment line with a hash sign (#) . Example is shown below :

```
#!/bin/bash
# This is a command
echo "I am logged in as $USER"
```

Q:20 How to get input from the terminal for shell

script ?

Ans: 'read' command reads in data from the terminal (using keyboard). The read command takes in whatever the user types and places the text into the variable you name. Example is shown below :

```
# vi /tmp/test.sh
```

```
#!/bin/bash
```

```
echo 'Please enter your name'
```

```
read name
```

```
echo "My Name is $name"
```

```
# ./test.sh
```

```
Please enter your name
```

```
LinuxTechi
```

```
My Name is LinuxTechi
```

Q:21 How to unset or de-assign variables ?

Ans: 'unset' command is used to de-assign or unset a variable. Syntax is shown below :

```
# unset <Name_of_Variable>
```

Q:22 How to perform arithmetic operation ?

Ans: There are two ways to

perform arithmetic operations :

1. Using `expr` command (`# expr 5 + 2`)
2. using a dollar sign and square brackets (`[$[operation]`) Example : `test=$((16 + 4)) ; test=$((16 + 4))`

Q:23 Basic Syntax of do-while statement ?

Ans: The do-while statement is similar to the while statement but performs the statements before checking the condition statement. The following is the format for the do-while statement:

```
do
{
statements
} while (condition)
```

Q:24 How to define functions in shell scripting ?

Ans: A function is simply a block of code with a name. When we give a name to a block of code, we can then call that name in our script, and that block will be executed.

Example is shown below :

```
$ diskusage () { df -h ; }
```

Q:25 How to use bc (bash calculator) in a shell script ?

Ans: Use the below Syntax to use bc in shell script.

```
variable=`echo "options; expression" | bc`
```

Tags: shell scripting

👍 YOU MAY ALSO LIKE...



Command Line Arguments in Linux Shell Scripting	Introduction To Shell Scripting Basics	Shell Scripting – Checking Conditions with if
15 MAR, 2014	26 OCT, 2014	
2 JUN, 2015		

1 RESPONSE

Comments 1

→ Pingbacks 0



shell scripting in linux

🕒 July 23, 2015 at 4:00 am

Thanks for the informative post. If you want to learn more about this than you can join training program also. This is the best option.

Reply

LEAVE A REPLY

Name *

Email *

Website

Comment



[Privacy & Terms](#)



Post Comment



