Shell scripting tutorials

Tutorial-0

Unix: UNIX is a popular multi-user, multitasking operating system (OS) developed at Bell Labs in the early 1970s.

SHELL

Shell is a UNIX term for the interactive user interface with an operating system. The shell is the layer of programming that understands and executes the commands a user enters. In some systems, the shell is called a command interpreter. A shell usually implies an interface with a command syntax (think of the DOS operating system and its "C:>" prompts and user commands such as "dir" and "edit").

Shell Types

In Unix, there are two major types of shells −

- Bourne shell − If you are using a Bourne-type shell, the $ character is the default prompt.
- C shell − If you are using a C-type shell, the % character is the default prompt.

SHELL SCRIPT

Usually shells are interactive that mean, they accept command as input from users and execute them. However, some time we want to execute a bunch of commands routinely, so we have type in all commands each time in terminal.
As shell can also take commands as input from file, we can write these commands in a file and can execute them in shell to avoid this repetitive work. These files are called Shell Scripts or Shell Programs. Shell scripts are like the batch file in MS-DOS. Each shell script is saved with .sh file extension eg. myscript.sh

1) Function Call

```
#!/bin/bash

sum ()
{
  x=$1
  y=$2
  k=0
  k=$(( $x + $y ))
  return $k
}
sum 7 5
echo "Sum of two no is:$?"
exit 0
```

2) Command line argument

```
#!/bin/bash
x=$1
y=$2
k=$(( $x + $y ))
echo "sum is:$k"
exit 0
```

3) If-else(Conditional statement)

```
#!/bin/bash
big_num()
{
 x=$1
 y=$2
 z=$3

   if [ $x -gt $y ] ; then
             if [ $x -gt $z ] ; then
                         echo "Biggest no is:$x"
            else
               echo "Biggest no is:$z"
              fi
   else
             if [ $y -gt $z ] ; then
              echo "Biggest no is:$y"
             else
                echo "Biggest no is:$z"
             fi
 fi
}
            big_num 2 3 4
            exit 0
```

4) while loop

```
#!/bin/bash
x=5

while [ $x -ge 1 ] ; do
  echo $x
  x=$(( $x - 1 ))
  done
exit 0
```

5) For loop

```
#!/bin/bash
printf "enter the no which table u want "
```

```
read x
y=1
k=0
echo "Multiplication table of $x is as below"

 for ((y=1;y<=10;y++)) do
   k=$(( $x * $y ))
  echo "$x X $y=$k"
 done
exit 0
```

6) Switch statement

```
#!/bin/bash
x=$1
    case $x in
      1)
         echo "this is level 1"
        ;;
      2)
  echo "this is level 2"
  ;;
      3)
  echo "this is level 3"
 ;;
      *)
echo "this is my default level"
;;
             esac
exit 0
```

7) Mathematical computation

```
#!/bin/bash
a=5.66
b=8.67
c=`echo $a + $b|bc`
echo $c
exit 0
```

8) List the content of current directory

```
#!/bin/bash

for x in `ls` ; do
echo $x
done
exit 0
```