Addis Ababa University
Department of Computer Science
CoSc2043 – Network and System Administration

Lab Manual- Part II

Content

Experiment	Practical's Name
1	Shell Scripting

Shell Scripting

Shell Scripting is an open-source computer program designed to be run by the Unix/Linux shell. Shell Scripting is a program to write a series of commands for the shell to execute. It can combine lengthy and repetitive sequences of commands into a single and simple script that can be stored and executed anytime which, reduces programming efforts.

What is Shell?

Shell is a UNIX term for an interface between a user and an operating system service. Shell provides users with an interface and accepts human-readable commands into the system and executes those commands which can run automatically and give the program's output in a shell script.

How to Write Shell Script in Linux/Unix

Shell Scripts are written using text editors. On your Linux system, open a text editor program, open a new file to begin typing a shell script or shell programming, then give the shell permission to execute your shell script and put your script at the location from where the shell can find it.

Let us understand the steps in creating a Shell Script:

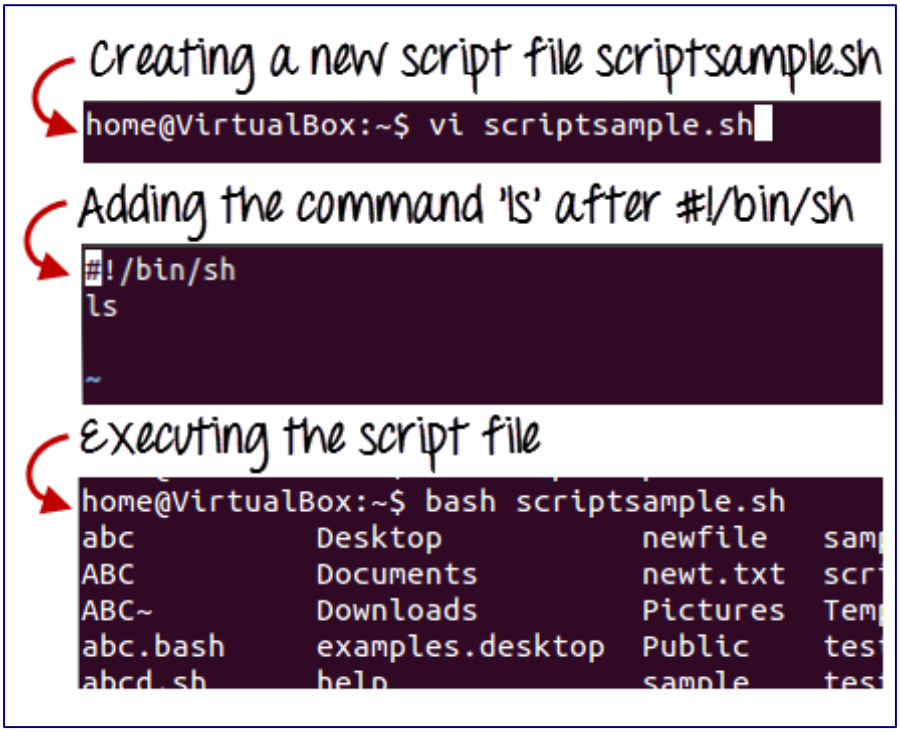
1. **Create a file using** a **vi** editor(or any other editor). Name script file with **extension .sh**
2. **Start** the script with **#! /bin/sh**
3. Write some code.
4. Save the script file as filename.sh
5. For **executing** the script type **bash filename.sh**

"#!" is an operator called shebang which directs the script to the interpreter location. So, if we use"#! /bin/sh" the script gets directed to the bourne-shell.

Let's create a small script -

```
#!/bin/sh
ls
```

Let's see the steps to create Shell Script Programs in Linux/Unix -



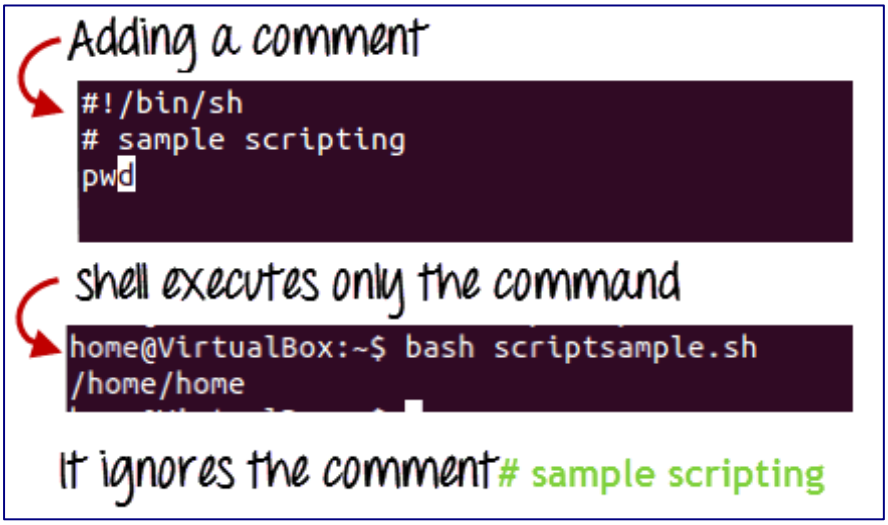
Command 'ls' is executed when we execute the scrip sample.sh file.

Adding shell comments

Commenting is important in any program. In Shell programming, the syntax to add a comment is

```
#comment
```

Let understand this with an example.



What are Shell Variables?

As discussed earlier, Variables store data in the form of characters and numbers. Similarly, Shell variables are used to store information and they can by the shell only.

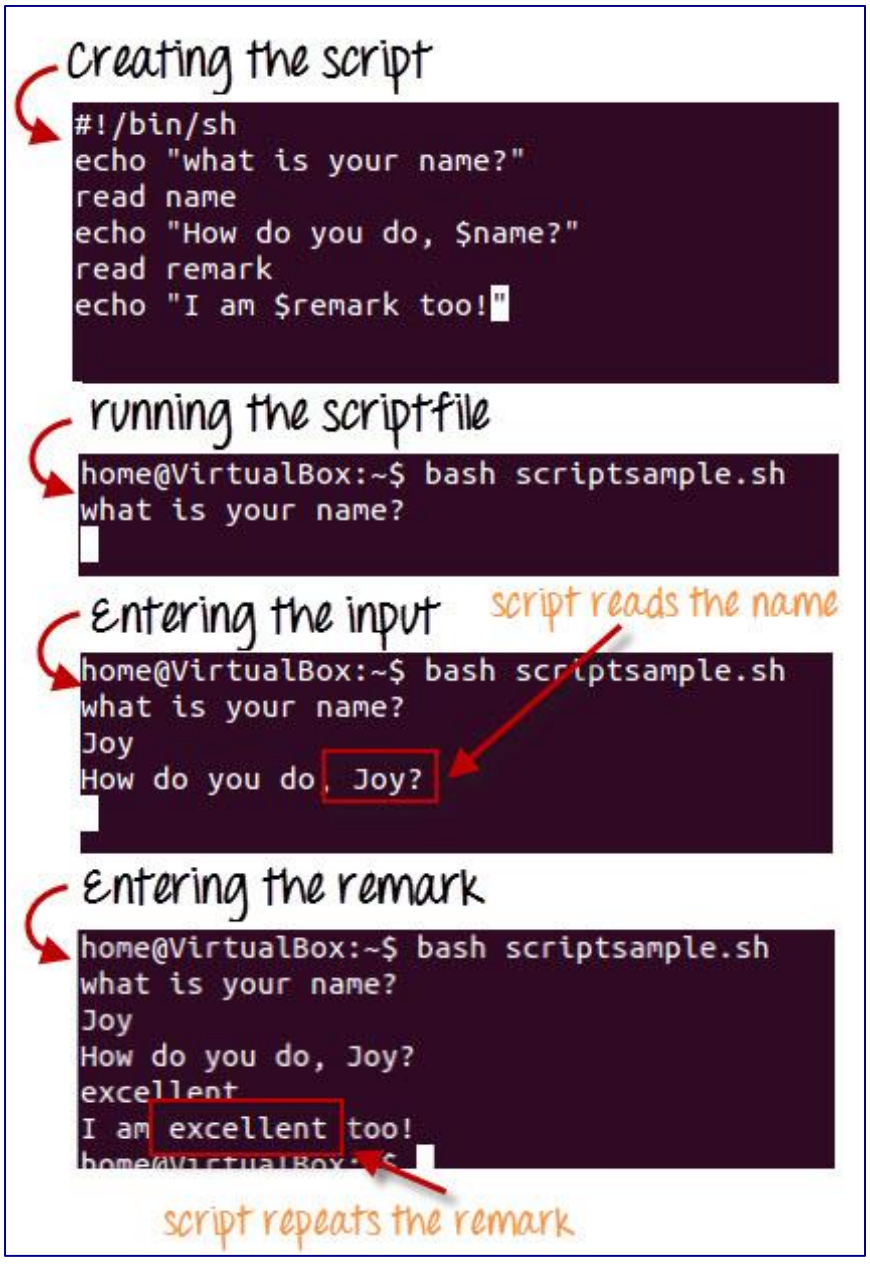
For example, the following creates a shell variable and then prints it:

```
variable ="Hello"
echo $variable
```

Below is a small script which will use a variable.

```
#!/bin/sh
echo "what is your name?"
read name
echo "How do you do, $name?"
read remark
echo "I am $remark too!"
```

Let's understand, the steps to create and execute the script



As you see, the program picked the value of the variable 'name' as Joy and 'remark' as excellent.

This is a simple script. You can develop advanced scripts which contain conditional statements, loops, and functions. Shell scripting will make your life easy and Linux administration a breeze.

Summary:

- Kernel is the nucleus of the operating systems, and it communicates between hardware and software
- Shell is a program which interprets user commands through CLI like Terminal
- The Bourne shell and the C shell are the most used shells in Linux
- Linux Shell scripting is writing a series of command for the shell to execute
- Shell variables store the value of a string or a number for the shell to read
- Shell scripting in Linux can help you create complex programs containing conditional statements, loops, and functions
- Basic Shell Scripting Commands in Linux: cat, more, less, head, tail, mkdir, cp, mv, rm, touch, grep, sort, wc, cut and, more.

Example

Basic Calculator
Aim:
To Develop a basic math calculator using case statement.
Apparatus Required:
Hardware Requirements: Intel core II CPU
Software Requirements: Red-Hat Linux

Procedure:
1) Create a new file.
2) Read the operands.
3) Select any one

```
# Implementation of Calculator application
#!bin/bash
j=1
while [ $j -eq 1]
do
echo "Enter the First Operand;"
read f1
echo "Enter the second operand:"
read f2
echo "1-> Addition"
echo "2-> Subtraction"
echo "3-> Multiplication"
echo "4-> Division"
echo "Enter your choice"
read n
```

```
case "$n" in
1) echo "Addition"
f3=$((f1+f2))
echo "The result is:$f3";;
2)
echo "Subtraction"
let "f4=$f1 -$f2"
echo "The result is:$f4";;
3)
echo "Multiplication"
let "f5=$f1 * $f2"
echo "The result is:$f5";;
4)
echo "Division"
let "f6=$f1 / $f2"
echo "The result is:$f6";;
esac
echo "Do you want to continue (press:1 otherwise press any key
to quit) "
read j
done
OUTPUT:[su@localhost su]$ bash u
Enter the First Operand;23
Enter the second operand:23
1-> Addition
2-> Subtraction
3-> Multiplication
4-> Division

Enter your choice
1
Addition
The result is:46
Do you want to continue (press:1 otherwise press any key to
quit)
```

Vi shortcuts

The best way to learn Vi is to create a new file and try it out for yourself. Feel free to use the common keyboard shortcut list below to help you learn Vi's extensive vocabulary. This list of shortcuts is by no means exhaustive, but they will enable you to edit files and learn Vi in a short amount of time.

- `$ vi <filename>` — Open or edit a file.
- `i` — Switch to Insert mode.
- **Esc** — Switch to Command mode.
- `:w` — Save and continue editing.
- `:wq` or `ZZ` — Save and quit/exit vi.
- `:q!` — Quit vi and do not save changes.
- `yy` — Yank (copy) a line of text.
- `p` — Paste a line of yanked text below the current line.
- `o` — Open a new line under the current line.
- `O` — Open a new line above the current line.
- `A` — Append to the end of the line.
- `a` — Append after the cursor's current position.
- `I` — Insert text at the beginning of the current line.
- `b` — Go to the beginning of the word.
- `e` — Go to the end of the word.
- `x` — Delete a single character.
- `dd` — Delete an entire line.
- `Xdd` — Delete X number of lines.
- `Xyy` — Yank X number of lines.
- `G` — Go to the last line in a file.
- `XG` — Go to line X in a file.
- `gg` — Go to the first line in a file.
- `:num` — Display the current line's line number.
- `h` — Move left one character.
- `j` — Move down one line.
- `k` — Move up one line.
- `l` — Move right one character.