Need of Shell Scripts

Some time we want to execute a bunch of commands routinely, so we have to 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.

Introduction

- Shell Scripts are also known as '*Shell Programs*' or '*Shell Procedures*'.
- Shell Script file means, a file which contains a set of commands within it. If any file contains commands, then that file can be executable file.
- Shell scripts are similar to the '***batch file***' in windows environment.
- It can be useful to execute the set of commands at a single moment of time, we will get our required outputs and those can also be saved under a file.
- Executing the commands separately will consume more time, Using shell scripts we can reduce this time at a greater extent.
- Shell scripts can also take arguments, which are known as command line arguments.

Steps to Execute Shell Script

Step-1: Write a shell script file using either an editor or a cat command and save the file with '***.sh***' extension.
Step-2: Add an executable permission to that shell script file.
Step-3: Execute the shell script file by the command '***. /filename.sh***'.

Types of Shell Scripts

1. Static Scripts (Non-Interactive Scripts)
2. Dynamic Scripts (Interactive Shell Scripts)

Static Scripts

It does not require any input from the user once the execution has started.

Program-1: Write a static script using the cat command to execute the following commands- ls, date, cal, who.

Solution-1:
Step-1:

```
$cat > script1.sh
ls
date
cal
who
ctrl+d
```

Step-2:

```
chmod u+x script1.sh
```

Step-3:
./script1.sh

Program-2: Write a static script using the cat command to execute the following commands- ls, date, cal, who along with separator or appropriate message.

Solution-2:
Step-1:

```
$cat > script2.sh
echo "******************************************************"
ls
echo "******************************************************"
echo "The Date command `date`"
echo "******************************************************"
cal
echo "******************************************************"
who
echo "******************************************************"
ctrl+d
```

Step-2:
chmod u+x script2.sh

Step-3:
./script2.sh

Student Task: Write a static script using the cat command to execute the following commands- clear, whoami, cal 2020, banner <Your Name> with appropriate titles.

Dynamic Script

It requires input from the user once the execution has started.

Program-3: Write a dynamic script to find list of files or directories from a given directory.

Solution-3:
Step-1:

```
$cat > script3.sh
echo “Enter any directory Path”
read dirpath
echo “Contents of directory $dirpath”
ls $dirpath
ctrl+d
```

Step-2:
chmod u+x script3.sh

Step-3:
./script3.sh

Student Task: Write a dynamic script using the cat command to take month number and year from user and display the calendar of the specified month & year.

Program-4: Write a dynamic script to perform arithmetic calculations on given values.

Solution-4:
Step-1:

```
$cat > script4.sh
echo "Enter any 2 numbers"
read x
read y
clear
echo "Given values"
echo "x: $x"
echo "y: $y"
echo "Addition=`expr $x + $y`"
echo "Substration=`expr $x - $y`"
echo "Multiplication=`expr $x * $y`"
echo "Division=`expr $x / $y`"
echo "Remainder=`expr $x % $y`"
```

Step-2:

```
chmod u+x script4.sh
```

Step-3:

```
./script4.sh
```

Program-5: Write a dynamic script to compare whether the two strings are equal or not.

Solution-5:
Step-1:

```
$cat > script5.sh
echo "Enter any 2 string"
read x
read y
clear
echo "Given values"
echo "String1: $x"
echo "String2: $y"
echo "String comparison result=`expr $x = $y`"
```

Step-2:

```
chmod u+x script5.sh
```

Step-3:

```
./script5.sh
```