

The topics:

- Shell test for file and directory types
- Shell Script Exercises

Shell test for file and directory types:

Test	Meaning
-s file	Non empty file
-f file	Is File exist or normal file and not a directory
-d dir	Is Directory exist and not a file
-w file	Is writeable file
-r file	Is read-only file
-x file	Is file is executable

Examples:

-f file

```
ubuntu@ubuntu:~$ touch file1
ubuntu@ubuntu:~$ if test -f file1 ; then echo "FILE EXIST" ; fi
FILE EXIST
ubuntu@ubuntu:~$ if test -f file2 ; then echo "FILE EXIST" ; fi
ubuntu@ubuntu:~$
```

-s file

```
ubuntu@ubuntu:~$ touch myFile
ubuntu@ubuntu:~$ echo " This is my file * * " > myFile
ubuntu@ubuntu:~$ cat myFile
 This is my file * *
ubuntu@ubuntu:~$ touch myFile2
ubuntu@ubuntu:~$ if test -s myFile;then echo "NON EMPTY FILE";else echo "EMPTY FILE";fi
NON EMPTY FILE
ubuntu@ubuntu:~$ if test -s myFile2;then echo "NON EMPTY FILE";else echo "EMPTY FILE";fi
EMPTY FILE
```

Add this string to the file
"myFile"

-d dir

```
ubuntu@ubuntu:~$ mkdir pnu
ubuntu@ubuntu:~$ if test -d pnu;then echo "THIS IS DIRECTORY";else echo "NOT a DIRECTORY";fi
THIS IS DIRECTORY
ubuntu@ubuntu:~$ if test -d myFile;then echo "THIS IS DIRECTORY";else echo "NOT a DIRECTORY";fi
NOT a DIRECTORY
```

-x file

```
ubuntu@ubuntu:~$ if test -x myFile;then echo "File is Executable";else echo "Not executable";fi
Not executable
ubuntu@ubuntu:~$ chmod 755 myFile
```

After give x
permission

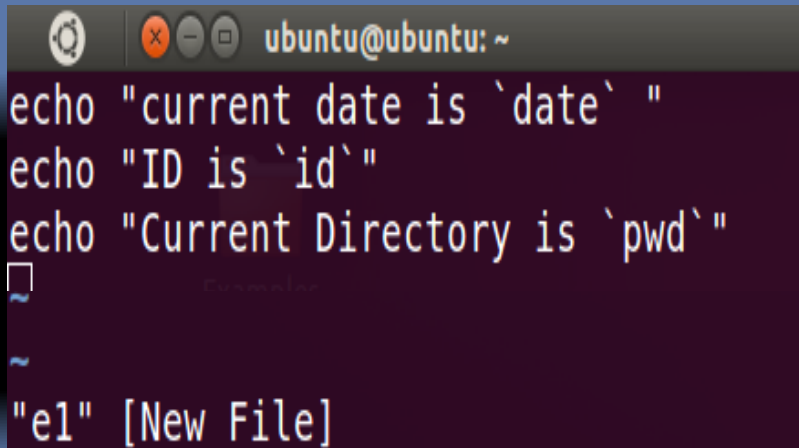
```
ubuntu@ubuntu:~$ if test -x myFile;then echo "File is Executable";else echo "Not executable";fi
File is Executable
```

Exercises:

1- Write shell script to determine whether given file exist or not.

```
if [ $# -ne 1 ]
then
    echo "Enter the file-name"
exit 1
fi
if [ -f $1 ]
then
    echo "$1 file exist"
else
    echo "Sorry, $1 file does not exist"
fi
```

2. Write Script to see current date & time, display information about your identity and current directory.



```
ubuntu@ubuntu: ~
echo "current date is `date` "
echo "ID is `id`"
echo "Current Directory is `pwd`"
~
~
"e1" [New File]
```

3. Write shell script that will add two numbers, which are supplied as command line argument, and if this two numbers are not given show error message.

```
if [ $# -ne 2 ]
then
    echo " Enter 2 number to find the sum"
else
    echo "Sum of $1 and $2 is `expr $1 + $2` "
fi
```

4- find out biggest number from given three numbers.

```
if [ $# -ne 3 ]
then
    echo "enter 3 nums to compare them"
    exit 1
fi

if [ $1 -gt $2 -a $1 -gt $3 ]
then
    echo "$1 is biggest num"
else
    if [ $2 -gt $1 -a $2 -gt $3 ]
    then
        echo " $2 is biggest num "
    else
        echo "$3 is biggest num"
    fi
fi
```

```
ubuntu@ubuntu:~$ sh e3 4 5
enter 3 nums to compare them
```

```
ubuntu@ubuntu:~$ sh e3 2 5 1
5 is biggest num
```

5. Write shell script using for loop to print the following patterns

```
1
11
111
1111
11111
```

```
ubuntu@ubuntu: ~
for (( i=1 ; i<=5 ; i++ ))
do
    for (( j=1 ; j<=i ; j++ ))
    do
        echo -n "1"
    done
    echo ""
done

ubuntu@ubuntu:~$ vi myFor
ubuntu@ubuntu:~$ bash myFor
1
11
111
1111
11111
```

6- Write script to print numbers as 5,4,3,2,1, the biggest number is supplied as command line argument. (Ex. Script-name 5)

```
ubuntu@ubuntu: ~
n=$1

while test $n != 0
do
    echo -n "$n,"
    n=`expr $n - 1`
done
echo ""
```

```
ubuntu@ubuntu:~$ sh e4 7
7,6,5,4,3,2,1,
ubuntu@ubuntu:~$ bash e4 5
5,4,3,2,1,
ubuntu@ubuntu:~$
```

5. Write script to print reverse order number of all digit, For eg. If no is 123 it will be 321.

take the number from the user at run time

```
echo "Enter any number :"  
read n  
n=$1  
rev=0  
sd=0  
while [ $n -gt 0 ]  
do  
sd=`expr $n % 10`  
rev=`expr $rev \* 10 + $sd`  
n=`expr $n / 10`  
done  
echo "Reverse number is $rev"
```