

ASSIGNMENT – 5

Q.1) Write a shell program, which accepts the name of a file from the standard input and performs the following tests on it. A) File existence B) File readable C) File writeable D) Both Readable and writeable.

A) File existence :-

```
echo "Enter your File name to check it exist or not"
read fileName
if [ -f "$fileName" ]
then
    echo "File is found"
else
    echo "File is not found"
fi
```

```
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./Test.sh
Enter your File name
Abdeali.txt
File is found
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./Test.sh
Enter your File name
jhscg
File is not found
```

B) File Readable :-

```
echo "Enter your File name to check is it readable?"
read fileName
if [ -r "$fileName" ]
then
    echo "File is Readable"
else
    echo "File is not Readable"
fi
```

```
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ chmod +rwx isReadable.sh
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./isReadable.sh
Enter your File name to check is it readable?
Abdeali.txt
File is Readable
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ chmod +wx Abdeali.txt
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./isReadable.sh
Enter your File name to check is it readable?
Abdeali.txt
File is Readable
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ chmod -r Abdeali.txt
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./isReadable.sh
Enter your File name to check is it readable?
Abdeali.txt
File is not Readable
```

C) File Writable :-

```
echo "Enter your File name to check is it writable?"
read fileName
if [ -w "$fileName" ]
then
    echo "File is Writable"
else
    echo "File is not Writable"
fi
```

```
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ chmod +rw isWritable.sh
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./isWritable.sh
Enter your File name to check is it writable?
Abdeali.txt
File is Writable
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ chmod -w Abdeali.txt
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./isWritable.sh
Enter your File name to check is it writable?
Abdeali.txt
File is not Writable
```

D) File Both :-

```
echo "Enter your File name to check is it Both?"
read fileName
if [[ -r "$fileName" && -w "$fileName" ]]
then
    echo "File is Writable and Readable"
else
    echo "File is not Writable and Readable"
fi
```

```
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./isBoth.sh
Enter your File name to check is it Both?
Abdeali.txt
./isBoth.sh: line 3: [: missing `]'
File is not Writable and Readable
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./isBoth.sh
Enter your File name to check is it Both?
Abdeali.txt
File is not Writable and Readable
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ chmod +w Abdeali.txt
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./isBoth.sh
Enter your File name to check is it Both?
Abdeali.txt
File is not Writable and Readable
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ chmod +rw Abdeali.txt
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./isBoth.sh
Enter your File name to check is it Both?
Abdeali.txt
File is Writable and Readable
```

Q.2) Write a shell program using 3 arguments to take the pattern as well as input and the output file names. If the pattern is found display "Pattern found", else display "Error message". Also check if right number of arguments is entered.

Code :-

```
echo "Enter Substring : "  
read substring  
value=`cat Pattern.txt`  
  
if [[ $value == *"$substring"* ]];  
then  
    echo "String contains substring."  
else  
    echo "String does not contain substring."  
fi
```

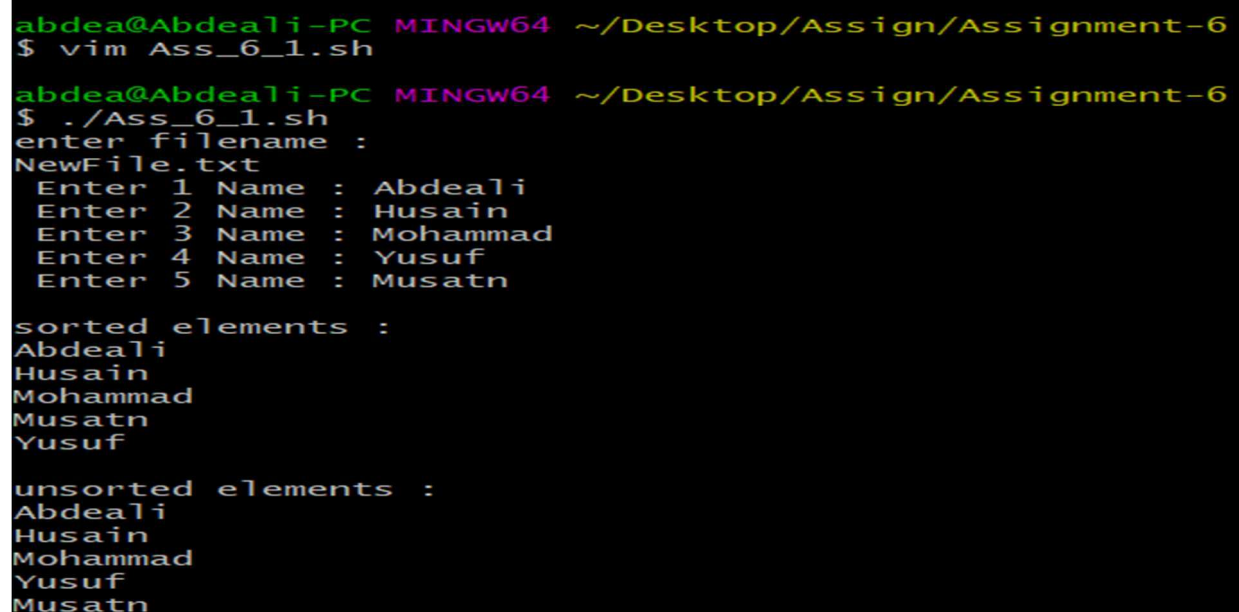
Output :-

```
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./Arguments.sh  
bash: ./Arguments.sh: Permission denied  
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ chmod +rwx Arguments.sh  
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./Arguments.sh  
Enter Substring :  
Hello  
String contains substring.  
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$ ./Arguments.sh  
Enter Substring :  
jhcfsdjf  
String does not contain substring.  
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali$
```

ASSIGNMENT – 6

Q.1) Write a shell program, which accepts the name of the file from the standard input and then performs the following operations: i) Enter the 5 names in the file ii) Sort the name in existing file iii) List unsorted and sorted file

```
echo "enter filename : "  
read fileName  
  
for i in {1..5}  
do  
read -p " Enter $i Name : " newword  
echo "$newword" >> $fileName  
done  
  
echo ""  
echo "sorted elements : "  
sort $fileName  
  
echo ""  
echo "unsorted elements : "  
cat $fileName
```



```
abdea@Abdeali-PC MINGW64 ~/Desktop/Assign/Assignment-6  
$ vim Ass_6_1.sh  
  
abdea@Abdeali-PC MINGW64 ~/Desktop/Assign/Assignment-6  
$ ./Ass_6_1.sh  
enter filename :  
NewFile.txt  
Enter 1 Name : Abdeali  
Enter 2 Name : Husain  
Enter 3 Name : Mohammad  
Enter 4 Name : Yusuf  
Enter 5 Name : Musatn  
  
sorted elements :  
Abdeali  
Husain  
Mohammad  
Musatn  
Yusuf  
  
unsorted elements :  
Abdeali  
Husain  
Mohammad  
Yusuf  
Musatn
```

Q.2) Write a menu driven shell program to copy, edit, rename and delete a file.

Code :-

```
read -p "Enter FileName: " fileName

if [ -f "$fileName" ]
then
    op=0
    echo "1 - Copy file"
    echo "2 - Edit file"
    echo "3 - Rename file"
    echo "4 - Delete file"
    echo "5 - Exit"

    while [ $op -lt 5 ]
    do
        echo ""
        read -p "Select an option(1/2/3/4/5): " op

        if [ $op == 1 ]
        then
            read -p "Enter the fileName to be copied in
: " copyFileName
            cp $fileName $copyFileName
            echo "Copied sucessfully"
        fi

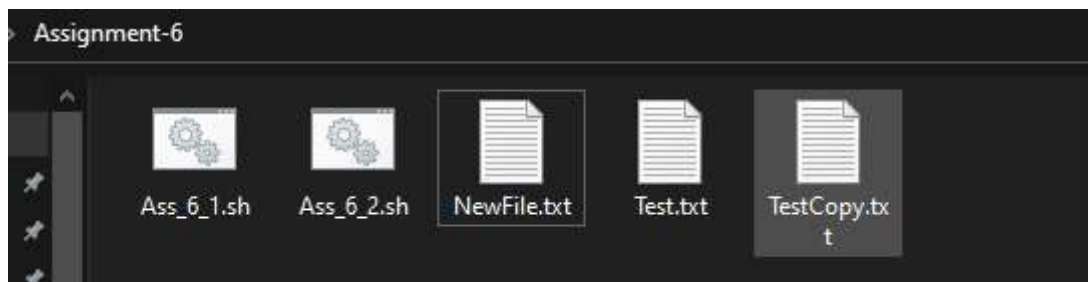
        if [ $op == 2 ]
        then
            read -p "Enter the text to append : " text
            echo "$text" >> $fileName
            echo "Edited sucessfully"
        fi
    done
fi
```

```
if [ $op == 3 ]
then
    read -p "Enter new fileName : " newFileName
    mv $fileName $newFileName
    echo "Renamed sucessfully"
fi

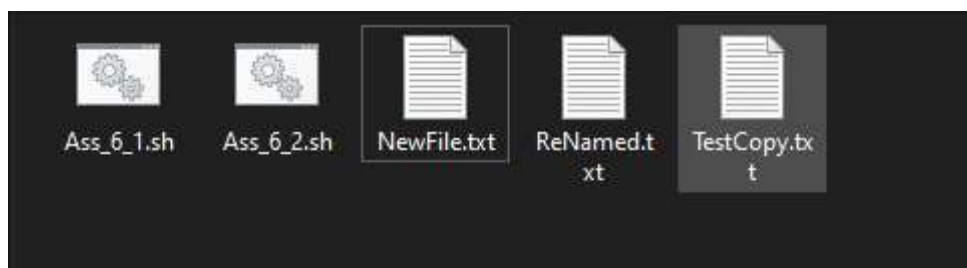
if [ $op == 4 ]
then
    rm $fileName
    echo "Delete sucessfully"
fi
done

else
    echo "File not found"
fi
```

Copy Output :



Rename Output :



Whole Execution :

```
abdea@Abdeali-PC MINGW64 ~/Desktop/Assign/Assignment-6
$ ./Ass_6_2.sh
Enter FileName: Test.txt
1 - Copy file
2 - Edit file
3 - Rename file
4 - Delete file
5 - Exit

Select an option(1/2/3/4/5): 1
Enter the fileName to be copied in : TestCopy.txt
Copied sucessfully

Select an option(1/2/3/4/5): 2
Enter the text to append : Hello!!
Edited sucessfully

Select an option(1/2/3/4/5): 3
Enter new fileName : ReNamed.txt
./Ass_6_2.sh: line 35: Test.txt: command not found
Renamed sucessfully

Select an option(1/2/3/4/5): 4
rm: cannot remove 'Test.txt': No such file or directory
Delete sucessfully

Select an option(1/2/3/4/5): 5

abdea@Abdeali-PC MINGW64 ~/Desktop/Assign/Assignment-6
$ ./Ass_6_2.sh
Enter FileName: TestCopy.txt
1 - Copy file
2 - Edit file
3 - Rename file
4 - Delete file
5 - Exit

Select an option(1/2/3/4/5): 4
Delete sucessfully

Select an option(1/2/3/4/5): |
```

ASSIGNMENT – 7

Q.1) Write a menu driven shell program to perform the following tasks

I. Enter the sentence in file

II. Search a given whole word in an existing file

III. Quit

Code :-

```
while :
do

echo "enter your choice :"
echo "1 : enter sentence in a file"
echo "2 : find a word in a file"
echo "3 : quit"
echo ""
echo "select option : "

echo "number : "
read num

case $num in

"1")echo "enter filename :"
read srcfile

echo "enter the sentence : "
read sentence

echo "$sentence" >> $srcfile
echo "your sentence is added in $srcfile"
echo ""
;;

"2")echo "enter filename :"
read filename

echo "enter word :"
read word

echo "$str" >> $filename
echo "word count = $(grep -o -i $word $filename | wc -l)"
echo ""
```



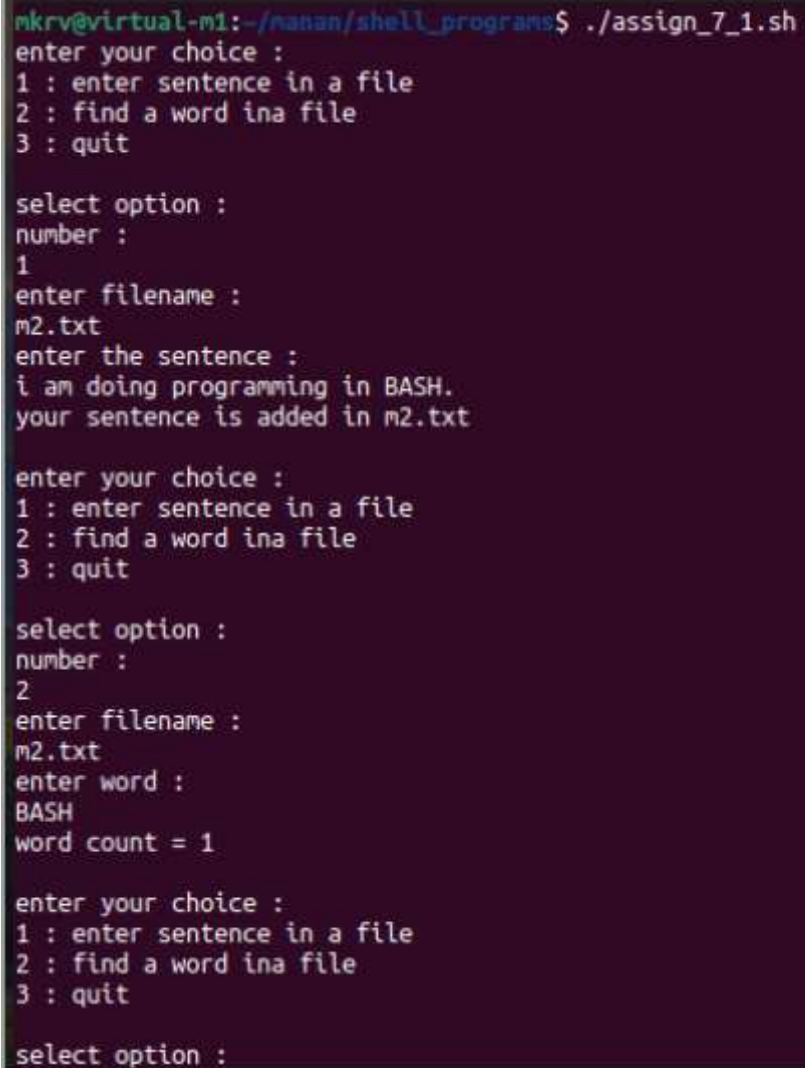
```
;;
```

```
"3")exit
```

```
;;
```

```
esac
```

```
done
```



```
mkrv@virtual-m1:~/nanan/shell_programs$ ./assign_7_1.sh
enter your choice :
1 : enter sentence in a file
2 : find a word in a file
3 : quit

select option :
number :
1
enter filename :
m2.txt
enter the sentence :
i am doing programming in BASH.
your sentence is added in m2.txt

enter your choice :
1 : enter sentence in a file
2 : find a word in a file
3 : quit

select option :
number :
2
enter filename :
m2.txt
enter word :
BASH
word count = 1

enter your choice :
1 : enter sentence in a file
2 : find a word in a file
3 : quit

select option :
```

Q.2) Write a shell program to prepare the electricity bill based on the following rules:

- I. For first 100 units ----- Rs. 1.00/unit
- II. For next 100 units ---- Rs. 2.00/unit
- III. Above 200 units ----- Rs. 3.00/unit

Code :-

```
read -p "Enter Total unit : " totalUnit
[[ $totalUnit =~ ^[0-9]+$ ]] || { echo "Enter a valid number";
continue; }

rupees=0

if (( totalUnit <= 100 ))
then

    rupees=1;
    echo "Charge per unit is Rs. 1"
fi

if ((totalUnit >= 101 && totalUnit <= 200))
then

    rupees=2;
    echo "Charge per unit is Rs. 2"
fi

if (( totalUnit >= 201))
then

    rupees=3;
    echo "Charge per unit is Rs. 3"
fi

bill=`expr $rupees \* $totalUnit`

echo "Total bill : " $bill
```

```
abdea@Abdeali-PC MINGW64 ~/Desktop/Assign/Assignment-7
$ ./ElectricityBill.sh
Enter Total unit : 100
Charge per unit is Rs. 1
Total bill : 100

abdea@Abdeali-PC MINGW64 ~/Desktop/Assign/Assignment-7
$ ./ElectricityBill.sh
Enter Total unit : 150
Charge per unit is Rs. 2
Total bill : 300

abdea@Abdeali-PC MINGW64 ~/Desktop/Assign/Assignment-7
$ ./ElectricityBill.sh
Enter Total unit : 250
Charge per unit is Rs. 3
Total bill : 750
```

Q.3) Write a shell program to prepare the electricity bill based on the following rules:

- I. For first 100 units ----- Rs. 0.75 /unit
- II. For next 100 units ---- Rs. 1.50/unit
- III. Above 200 units ----- Rs. 3.00/unit

```
read -p "Enter Total unit : " totalUnit
[[ $totalUnit =~ ^[0-9]+$ ]] || { echo "Enter a valid number";
continue; }

rupees=0.0

#totalUnit=`expr $totalUnit \* 1.0 | bc`

if (( totalUnit <= 100 ))
then

    rupees=0.75
    echo "Charge per unit is Rs. 0.75"
fi

if ((totalUnit >= 101 && totalUnit <= 200))
then

    rupees=1.50
    echo "Charge per unit is Rs. 1.50"
```

```
fi

if (( totalUnit >= 201))
then

    rupees=3.0
    echo "Charge per unit is Rs. 3"
fi

    bill=`expr "$totalUnit * $rupees" | bc -l`

echo "Total bill : " $bill
```

Output :-

```
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali/Assignment-7$ ./Electricity
-3.sh
Enter Total unit : 100
Charge per unit is Rs. 0.75
Total bill : 75.00
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali/Assignment-7$ ./Electricity
-3.sh
Enter Total unit : 120
Charge per unit is Rs. 1.50
Total bill : 180.00
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali/Assignment-7$ ./Electricity
-3.sh
Enter Total unit : 320
Charge per unit is Rs. 3
Total bill : 960.0
guest@matrix-ideacentre-510S-08ISH:~/Desktop/Abdeali/Assignment-7$ █
```

ASSIGNMENT – 8

Q.1) Write a shell script to sum up the following series.

$$1/1! + 2/2! + 3/3! + \dots$$

Code :-


```
echo "Enter the number of terms:"
read n

sum=0
fact=1

for ((i=1;i<=n;i++))
do
    fact=$((fact * i))
    term=$(echo "scale=4; $i/$fact" | bc)
    sum=$(echo "scale=4; $sum+$term" | bc)
done

echo "The sum of the series is: $sum"
```

Output :-



```
Enter the number of terms:
4
The sum of the series is: 2.7083
```

Q.2) Write a shell script to display the result "Pass" or "Fail" using the information given below. Student name, student register no, mark1, mark2, mark3, mark4. The minimum pass for each subject is 50.

Code :-

```
echo "Enter student name:"
read name

echo "Enter student register no:"
read regno

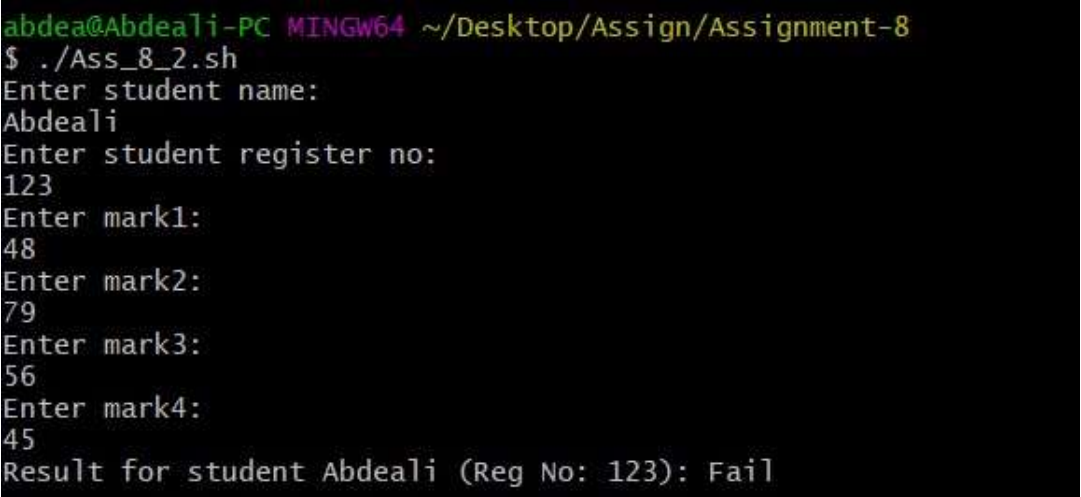
echo "Enter mark1:"
read mark1

echo "Enter mark2:"
read mark2

echo "Enter mark3:"
read mark3

echo "Enter mark4:"
read mark4

if ((mark1 >= 50 && mark2 >= 50 && mark3 >= 50 && mark4 >= 50))
then
    echo "Result for student $name (Reg No: $regno): Pass"
else
    echo "Result for student $name (Reg No: $regno): Fail"
fi
```



```
abdea@Abdeali-PC MINGW64 ~/Desktop/Assign/Assignment-8
$ ./Ass_8_2.sh
Enter student name:
Abdeali
Enter student register no:
123
Enter mark1:
48
Enter mark2:
79
Enter mark3:
56
Enter mark4:
45
Result for student Abdeali (Reg No: 123): Fail
```

Q.3) Write a shell script to display the result of the student in neat format using the information given below.

Student name, student register no, mark1, mark2, mark3, mark4. The minimum pass for each subject is 50.

Code :-

```
echo "Enter student name:"
read name

echo "Enter student register no:"
read regno

echo "Enter mark1:"
read mark1

echo "Enter mark2:"
read mark2

echo "Enter mark3:"
read mark3

echo "Enter mark4:"
read mark4

if ((mark1 >= 50 && mark2 >= 50 && mark3 >= 50 && mark4 >= 50))
then
    result="Pass"
else
    result="Fail"
fi

# Print the result in a neat format
echo "===== "
echo "          STUDENT RESULT          "
echo "===== "
echo "Name:          $name"
echo "Reg No:       $regno"
echo "----- "
echo "Subject 1:    $mark1"
echo "Subject 2:    $mark2"
echo "Subject 3:    $mark3"
```



```
echo "Subject 4:   $mark4"
echo "-----"
echo "Result:      $result"
echo "=====
```

Output :-

```
abdea@Abdeali-PC MINGW64 ~/Desktop/Assign/Assignment-8
$ ./Ass_8_3.sh
Enter student name:
Abdeali
Enter student register no:
123456
Enter mark1:
78
Enter mark2:
89
Enter mark3:
56
Enter mark4:
96
=====
                        STUDENT RESULT
=====
Name:      Abdeali
Reg No:    123456
-----
Subject 1: 78
Subject 2: 89
Subject 3: 56
Subject 4: 96
-----
Result:    Pass
=====
```

ASSIGNMENT – 9

Q.1) Write a menu driven shell script for converting all the capital letters in a file to small case letters and vice versa.

Code :-

```
# Display the menu options
echo "1. Convert all capital letters to small case"
echo "2. Convert all small case letters to capital case"
echo "Enter your choice:"

# Read the user's choice
read choice

# Prompt the user to enter the file name
echo "Enter the name of the file:"
read filename

# Check if the file exists
if [ ! -f "$filename" ]
then
    echo "File not found!"
    exit 1
fi

# Perform the selected conversion
case $choice in

    1) # Convert all capital letters to small case
        tr '[:upper:]' '[:lower:]' < "$filename" >
        "${filename}_smallcase"
        echo "All capital letters in $filename converted to small case
        and saved to ${filename}_smallcase"
        ;;

    2) # Convert all small case letters to capital case
        tr '[:lower:]' '[:upper:]' < "$filename" >
        "${filename}_capitalcase"
```

```
        echo "All small case letters in $filename converted to capital
case and saved to ${filename}_capitalcase"
        ;;
    *) # Invalid choice
        echo "Invalid choice!"
        exit 1
        ;;
esac
```

Output :-

```
Welcome to the Capitalization Converter!
1. Convert all capital letters to small case
2. Convert all small case letters to capital
3. Quit
Enter your choice (1-3): 1
Enter input file name: input.txt
Enter output file name: output.txt
Capital letters converted to small case in file: input.txt
1. Convert all capital letters to small case
2. Convert all small case letters to capital
3. Quit
Enter your choice (1-3): 3
Exiting the Capitalization Converter...
```

Q.2) Write a shell script for a file contains records with each record containing name of city, name of state and name of country. How would you sort this file with country as the primary sort key and state as the secondary sort key.

Code :-

```
echo "Enter filename having data of country state and city: "  
read file  
if [ -e "$file" ]  
then  
  
    echo  
    echo "-----Unsorted-----"  
    cat $file  
    echo  
    echo "*****Sorted*****"  
    sort -k3,3 -k2,2 $file  
else  
    echo "File Not Found"  
fi
```

Output :-



```
devendra@devendra-VirtualBox: ~/Desktop/OS$ bash -f example92.sh  
Enter filename having data of country state and city:  
country.txt  
  
-----Unsorted-----  
BOTad      Gujarat      India  
Surat      Gujarat      India  
kabul      kabul       Afghanistan  
london     london      UK  
Washington Washington  US  
beijing    beijing     China  
  
*****Sorted*****  
kabul      kabul       Afghanistan  
beijing    beijing     China  
Surat      Gujarat      India  
BOTad      Gujarat      India  
london     london      UK  
Washington Washington  US  
devendra@devendra-VirtualBox: ~/Desktop/OS$
```