Name: Prathamesh Sudhir Paraswar

Class:TE9

Roll NO:33155

Batch:L9

Assignment 1B:

1) What is shell Scripting??

Being a Linux user means you play around with the command-line. Like it or not, there are just some things that are done much more easily via this interface than by pointing and clicking. The more you use and learn the command-line, the more you see its potential. Well, the command-line itself is a program: the shell. Most Linux distros today use Bash, and this is what you're really entering commands into.

Shell scripts allow us to program commands in chains and have the system execute them as a scripted event, just like batch files. They also allow for far more useful functions, such as command substitution. You can invoke a command, like date, and use its output as part of a file-naming scheme. You can automate backups and each copied file can have the current date appended to the end of its name. Scripts aren't just invocations of commands, either. They're programs in their own right. Scripting allows you to use programming functions – such as 'for' loops, if/then/else statements, and so forth – directly within your operating system's interface. And, you don't have to learn another language because you're using what you already know: the command-line.

Shell scripts are executed in a separate child shell process. This is done by providing special interpreter line at the beginning (starting with #!).

To run the script we make it executable and then invoke the script name.

$ chmod +x script.sh or $ chmod 755 script.sh

$ script.sh

Use:

o Perform arithmetic operations on integers

o Determine the length of the string.

o Extract a sub-string.

o Locate the position of a character in a string

Code:

```
#!/bin/bash


filename=""
create()
{
        echo -e "Create..."
        echo -e "\n Enter the name of the file : \c"
        read filename
        len=$(echo -n $filename | wc -m)
        if test $len -gt 0;then
                if [ -f "$filename" ]; then
                   echo "$filename exists."
                else
                   echo "$filename does not exist.Creating it..."
                   touch $filename
                   echo "File with name $filename created successfully."
                fi


        else
                echo "\n Filename cannot be empty."
```

```
        fi
}
readf()
{
        echo -e "read..."
        cat $filename
}
insertinto()
{
        echo -e "insert.."
        echo -e "Enter your roll number:"
        read r_no
        echo -n $r_no >> $filename
        echo -n " " >> $filename
        echo -e "Enter your name:"
        read name
        echo -n $name >> $filename
        echo -n " " >> $filename
        echo -e "Enter your department:"
        read d_name
        echo -n $d_name >> $filename
        echo -n " " >> $filename
        echo -e "Enter your phone number:"
        read p_no
        echo -n $p_no >> $filename
        echo -n " " >> $filename
        echo -e "" >> $filename
```

```
}

deletefrom()
{
        echo "delete..."
        echo -e "Enter your roll number:"
        read r_no
        sed -i "/$r_no/d" $filename
}

modify()
{
        echo "modify..."
        echo -e "Enter your id:"
        read r_no
        if ( grep -q $r_no $filename )
        then
           echo "Enter the name:"
           read name
           echo "Enter the department:"
           read dept
           echo "Enter the phone number:"
           read p_no
           sed -i "/$r_no/c\ $r_no\ $name\ $dept\ $p_no" $filename
        else
           echo "Your record not found."
        fi
```

```
}

search()
{
        echo "Search..."
        echo -e "Enter your id:"
        read r_no
        if ( grep -q $r_no $filename )
        then
                echo "Search Successfull..."
        else
           echo "Search Unsuccessfull..."
        fi
}
flag=0
while [ flag==0 ]

do
echo -e "\nEnter your choice : \c"
echo -e "\n1.Create DB: \c"
echo -e "\n2.Read DB : \c"
echo -e "\n3.Insert DB : \c"
echo -e "\n4:Delete DB: \c"
echo -e "\n5:Modify: \c"
echo -e "\n6:Search: \c"
echo -e "\n7:exit DB: \c"
read x
```

```
case $x in
        1 )
                create
                ;;

        2 )
                readf
                ;;
        3 )
                insertinto
                ;;
        4 )
                deletefrom
                ;;
        5 )
                modify
                ;;
        6 )
                search
                ;;
        7 )
                echo "exit..."
                break ;;
esac

done
```