```bash
#!/bin/bash

function f2()
{
sum=$(($1 + $2))
echo "The sum is $sum"

mult=$(($3 * $4))
echo "The multiplication is $mult"
}
a=10
b=20
c=30
d=40

f2 $a $b $c $d
```

^G Help   ^O Write Out   ^W Where Is   ^K Cut   ^T Execute   ^C Location   M-U Undo   M-A Set Mark   M-] To Bracket   M-Q Previous   ^B Back   ^← Prev Word
^X Exit   ^R Read File   ^\ Replace   ^U Paste   ^J Justify   ^/ Go To Line M-E Redo   M-6 Copy   ^Q Where Was   M-W Next   ^F Forward   ^→ Next Word

```
sumukh_g@Sumukh:~$ nano function2.sh
sumukh_g@Sumukh:~$ ./function2.sh
The sum is 30
The multiplication is 1200
sumukh_g@Sumukh:~$
```

```bash
#!/bin/bash

function f1
{
        echo "Welcome to function 1"
}

f1
```

```
sumukh_g@Sumukh:~$ nano function1.sh
sumukh_g@Sumukh:~$ chmod +x function1.sh
sumukh_g@Sumukh:~$ ./function1.sh
Welcome to function 1
sumukh_g@Sumukh:~$
```

```
sumukh_g@Sumukh:~$ nano deletearr.sh
sumukh_g@Sumukh:~$ ./deletearr.sh
is
sumukh_g@Sumukh:~$ nano deletearr.sh
sumukh_g@Sumukh:~$ ./deletearr.sh
my is sumukh
sumukh_g@Sumukh:~$ 
```

```bash
#!/bin/bash

declare -a arr3=("my" "name" "is" "sumukh")

unset arr3[1]

echo "${arr3[@]}"
```

```
sumukh_g@Sumukh:~$ nano deletearr.sh
sumukh_g@Sumukh:~$ ./deletearr.sh
is
sumukh_g@Sumukh:~$
```

```bash
#!/bin/bash

declare -a arr2=("10","20","30","40")

length=${#arr2[@]}

for((i=0;i<${length};i++))
do
echo "${arr2[$i]}"
done
```

```
sumukh_g@Sumukh:~$ nano array2.sh
sumukh_g@Sumukh:~$ ./array2.sh
10,20,30,40
sumukh_g@Sumukh:~$
```

```bash
#!/bin/bash

#declaring an array

declare -a arr1=("this","is","a","line")

echo ${arr1[@]}
```
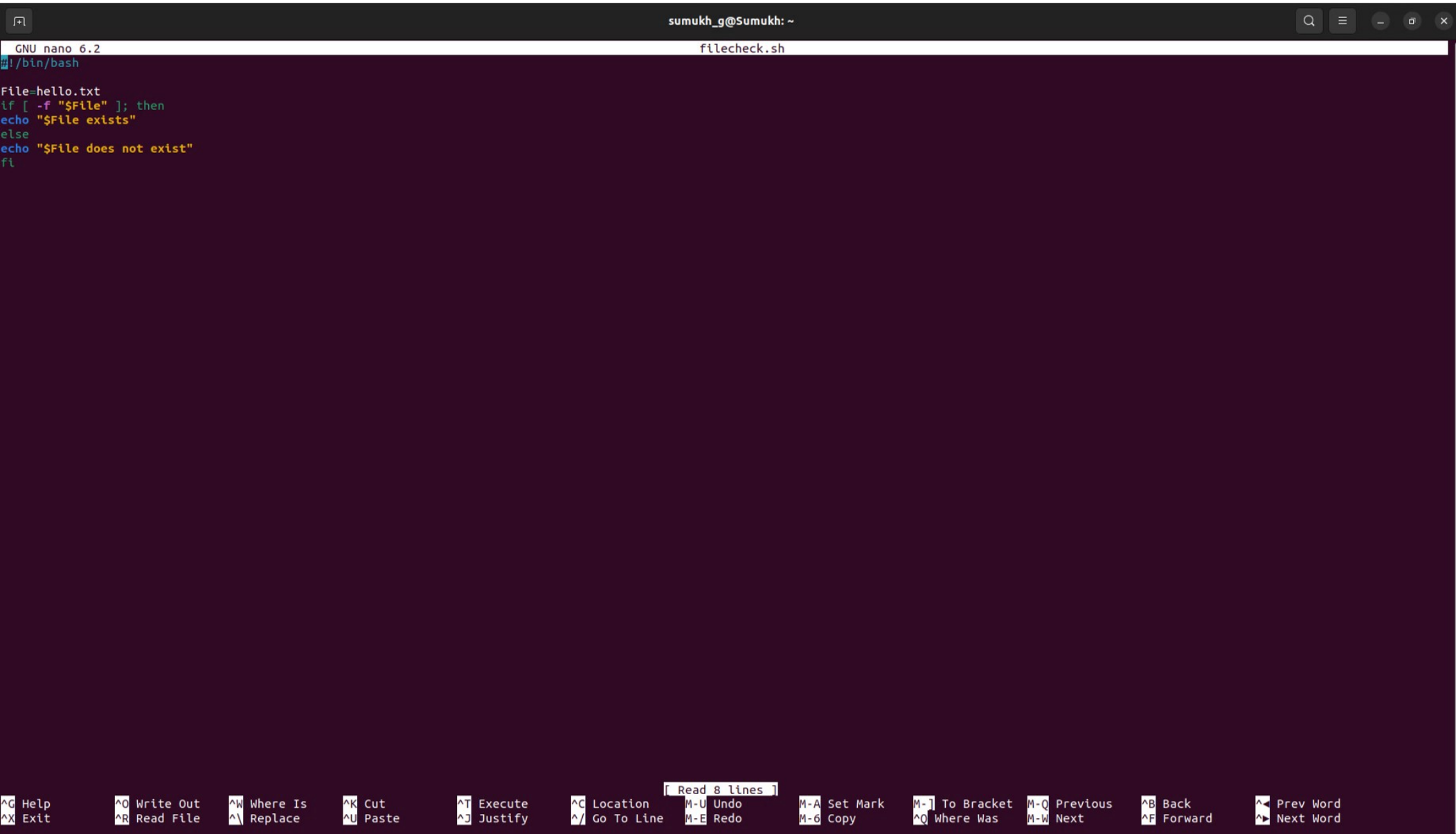
^G Help   ^O Write Out   ^W Where Is   ^K Cut   ^T Execute   ^C Location   M-U Undo   M-A Set Mark   M-] To Bracket M-Q Previous   ^B Back   ^← Prev Word
^X Exit   ^R Read File   ^\ Replace   ^U Paste   ^J Justify   ^/ Go To Line M-E Redo   M-6 Copy   ^Q Where Was   M-W Next   ^F Forward   ^→ Next Word

```
#!/bin/bash

File=hello.txt
if [ -f "$File" ]; then
echo "$File exists"
else
echo "$File does not exist"
fi
```

```
^G Help       ^O Write Out   ^W Where Is   ^K Cut      ^T Execute   ^C Location   M-U Undo   M-A Set Mark   M-] To Bracket   M-Q Previous   ^B Back        ^< Prev Word
^X Exit       ^R Read File   ^\ Replace    ^U Paste    ^J Justify   ^/ Go To Line M-E Redo   M-6 Copy       ^Q Where Was     M-W Next       ^F Forward     ^> Next Word
```

```
sumukh_g@Sumukh:~$ nano filecheck.sh
sumukh_g@Sumukh:~$ chmod +x filecheck.sh
sumukh_g@Sumukh:~$ ./filecheck.sh
hello.txt exists
sumukh_g@Sumukh:~$
```

```bash
#!/bin/bash

value = `cat hello.txt`
echo "$value"
```

```bash
#!/bin/bash

a=1

until [ $a -gt 10 ]
do
        echo "a is $a"
        ((a++))
done
```

```
^G Help      ^O Write Out   ^W Where Is   ^K Cut      ^T Execute   ^C Location   M-U Undo   M-A Set Mark   M-] To Bracket   M-Q Previous   ^B Back      ^◄ Prev Word
^X Exit      ^R Read File   ^\ Replace    ^U Paste    ^J Justify   ^/ Go To Line M-E Redo   M-6 Copy       ^Q Where Was     M-W Next       ^F Forward   ^► Next Word
```

```
sumukh_g@Sumukh:~$ nano until.sh
sumukh_g@Sumukh:~$ chmod +x until.sh
sumukh_g@Sumukh:~$ ./until.sh
a is 1
a is 2
a is 3
a is 4
a is 5
a is 6
a is 7
a is 8
a is 9
a is 10
sumukh_g@Sumukh:~$
```

```bash
#!/bin/bash

a=5

while [[ $a != 0 ]]
do
        echo "Value of a is $a";
        ((a--));
done
```

^G Help   ^O Write Out   ^W Where Is   ^K Cut   ^T Execute   ^C Location   M-U Undo   M-A Set Mark   M-] To Bracket   M-Q Previous   ^B Back   ^← Prev Word
^X Exit   ^R Read File   ^\ Replace   ^U Paste   ^J Justify   ^/ Go To Line M-E Redo   M-6 Copy   ^Q Where Was   M-W Next   ^F Forward   ^→ Next Word

```
sumukh_g@Sumukh:~$ nano while.sh
sumukh_g@Sumukh:~$ ./while.sh
Value of a is 5
Value of a is 4
Value of a is 3
Value of a is 2
Value of a is 1
sumukh_g@Sumukh:~$
```

```bash
#!/bin/bash

a=10
echo "The value of a is $a before while loop"
while [ $a -gt 6 ]
do
        a--
done
echo "The value of a is $a after while loop"
```

^G Help   ^O Write Out   ^W Where Is   ^K Cut   ^T Execute   ^C Location   M-U Undo   M-A Set Mark   M-] To Bracket M-Q Previous   ^B Back   ^← Prev Word
^X Exit   ^R Read File   ^\ Replace   ^U Paste   ^J Justify   ^/ Go To Line   M-E Redo   M-6 Copy   ^Q Where Was   M-W Next   ^F Forward   ^→ Next Word

```
#!/bin/bash

a=10
echo "The value of a is $a before while loop"
while [ $a -gt 6 ]
do
        a++
done
echo "The value of a is $a after while loop"
```

```bash
#!/bin/bash

a=10
echo "The value of a is $a before while loop"
while [ a -gt 6 ]
do
        a++
done
echo "The value of a is $a after while loop"
```

```
sumukh_g@Sumukh:~$ nano condition1.sh
sumukh_g@Sumukh:~$ chmod +x condition1.sh
sumukh_g@Sumukh:~$ ./condition1.sh
./condition1.sh: line 7: [: missing `]'
./condition1.sh: line 10: [: missing `]'
c is greatest
sumukh_g@Sumukh:~$ nano condition1.sh
sumukh_g@Sumukh:~$ ./condition1.sh
./condition1.sh: line 10: [: missing `]'
c is greatest
sumukh_g@Sumukh:~$ nano condition1.sh
sumukh_g@Sumukh:~$ ./condition1.sh
c is greatest
sumukh_g@Sumukh:~$
```

```bash
#!/bin/bash

a=10
b=20
c=30

if [ $a -gt $b ] && [ $a -gt $c ]
then
echo "a is greatest"
elif [ $b -gt $a ] && [ $b -gt $c ]
then
echo "b is greatest"
else
echo "c is greatest"
fi
```

```bash
#!/bin/bash

a=10
b=20
c=30

if [ $a -gt $b ] && [ $a -gt $c ]
then
echo "a is greatest"
elif [ $b -gt $a && $b -gt $c ]
then
echo "b is greatest"
else
echo "c is greatest"
fi
```

```bash
#!/bin/bash

a=10
b=20
c=30

if [ $a -gt $b && $a -gt $c ]
then
echo "a is greatest"
elif [ $b -gt $a && $b -gt $c ]
then
echo "b is greatest"
else
echo "c is greatest"
fi
```

```
sumukh_g@Sumukh:~$ nano name.sh
sumukh_g@Sumukh:~$ ./name.sh
bash: ./name.sh: Permission denied
sumukh_g@Sumukh:~$ chmod +x name.sh
sumukh_g@Sumukh:~$ ./name.sh
Enter your name
Sumukh
Your Name is : Sumukh
sumukh_g@Sumukh:~$
```

```
#!/bin/bash

echo "Enter your name"
read
echo "Your Name is : $REPLY"
```

```
sumukh_g@Sumukh:~$ nano case.sh
sumukh_g@Sumukh:~$ chmod +x case.sh
sumukh_g@Sumukh:~$ ./case.sh
Kashmir is famous for apple
sumukh_g@Sumukh:~$
```

```
GNU nano 6.2                                    case.sh *
#!/bin/bash

FRUIT="apple"

        case "$FRUIT" in
                "apple") echo "Kashmir is famous for apple"
                ;;
                "kiwi") echo "NewZealand is famous for kiwi"
                ;;
                "cherry") echo "Middle East is famous for cherry"
                ;;
                *) echo "Nothing found"
                ;;
        esac
```

```
sumukh_g@Sumukh:~$ nano table.sh
sumukh_g@Sumukh:~$ ./table.sh
Enter the number 10
10 x 1 = 10
10 x 2 = 20
10 x 3 = 30
10 x 4 = 40
10 x 5 = 50
10 x 6 = 60
10 x 7 = 70
10 x 8 = 80
10 x 9 = 90
10 x 10 = 100
sumukh_g@Sumukh:~$
```

```bash
#!/bin/bash

read -p "Enter the number" num

#This is file for tables of a number

for i in {1..10}
do
        echo "$num x $i = $((num * i))"
done
```

```
sumukh_g@Sumukh:~$ cat /etc/shells
# /etc/shells: valid login shells
/bin/sh
/bin/bash
/usr/bin/bash
/bin/rbash
/usr/bin/rbash
/usr/bin/sh
/bin/dash
/usr/bin/dash
sumukh_g@Sumukh:~$ which bash
/usr/bin/bash
sumukh_g@Sumukh:~$ nano hello.sh
sumukh_g@Sumukh:~$ ./hello.sh
bash: ./hello.sh: Permission denied
sumukh_g@Sumukh:~$ chmod +x hello.h
chmod: cannot access 'hello.h': No such file or directory
sumukh_g@Sumukh:~$
sumukh_g@Sumukh:~$ chmod +x hello.sh
sumukh_g@Sumukh:~$ ./hello.sh
welcome to hello world
sumukh_g@Sumukh:~$ nano tables.sh
sumukh_g@Sumukh:~$
sumukh_g@Sumukh:~$ nano table.sh
sumukh_g@Sumukh:~$ nano tables.sh
sumukh_g@Sumukh:~$ nano table.sh
sumukh_g@Sumukh:~$ chmod +x table.sh
sumukh_g@Sumukh:~$ ./table.sh
num 10
 * 1 = 0
 * 2 = 0
 * 3 = 0
 * 4 = 0
 * 5 = 0
 * 6 = 0
 * 7 = 0
 * 8 = 0
 * 9 = 0
 * 10 = 0
sumukh_g@Sumukh:~$
```

```bash
#!/bin/bash

read -p num

#This is file for tables of a number

for i in {1..10}
do
        echo "$num * $i = $((num * i))"
done
```

```
#!/bin/bash

echo "welcome to hello world"
```

```bash
#!/bin/bash

str1="hello"
str2="world"

str3=$str1$str2

echo "$str3"
```

^G Help   ^O Write Out   ^W Where Is   ^K Cut   ^T Execute   ^C Location   M-U Undo   M-A Set Mark   M-] To Bracket   M-Q Previous   ^B Back   ^← Prev Word
^X Exit   ^R Read File   ^\ Replace   ^U Paste   ^J Justify   ^/ Go To Line M-E Redo   M-6 Copy   ^Q Where Was   M-W Next   ^F Forward   ^→ Next Word

```
sumukh_g@Sumukh:~$ nano strcont.sh
sumukh_g@Sumukh:~$ chmod +x strcont.sh
sumukh_g@Sumukh:~$ ./strcont.sh
helloworld
sumukh_g@Sumukh:~$
```

```
sumukh_g@Sumukh:~$ nano substring.sh
sumukh_g@Sumukh:~$ chmod +x substring.sh
sumukh_g@Sumukh:~$ ./substring.sh
Welcome to shell scripting
String : Welcome to shell scripting
Substr : Welcome to
sumukh_g@Sumukh:~$ nano substring.sh
sumukh_g@Sumukh:~$ ./substring.sh
Welcome to shell scripting
String : Welcome to shell scripting
Substr :  shell scripting
sumukh_g@Sumukh:~$ nano substring.sh
sumukh_g@Sumukh:~$ ./substring.sh
Welcome to shell scripting
String : Welcome to shell scripting
Substr :
sumukh_g@Sumukh:~$ nano substring.sh
sumukh_g@Sumukh:~$ ./substring.sh
Welcome to shell scripting
String : Welcome to shell scripting
Substr : s
sumukh_g@Sumukh:~$ nano substring.sh
sumukh_g@Sumukh:~$
```

```bash
#!/bin/bash

#to get one character
echo "Welcome to shell scripting"

str="Welcome to shell scripting"

substr="${str:11:1}"

echo "String : $str"
echo "Substr : $substr"
```

```
^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark   M-] To Bracket  M-Q Previous   ^B Back        ^← Prev Word
^X Exit        ^R Read File    ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy       ^Q Where Was    M-W Next       ^F Forward     ^→ Next Word
```

```bash
#!/bin/bash

#to get characters from 10
echo "Welcome to shell scripting"

str="Welcome to shell scripting"

substr="${str:10}"

echo "String : $str"
echo "Substr : $substr"
```

```bash
#!/bin/bash

#to get first 10 characters
echo "Welcome to shell scripting"

str="Welcome to shell scripting"

substr="${str:0:10}"

echo "String : $str"
echo "Substr : $substr"
```

^G Help   ^O Write Out   ^W Where Is   ^K Cut   ^T Execute   ^C Location   M-U Undo   M-A Set Mark   M-] To Bracket   M-Q Previous   ^B Back   ^◄ Prev Word
^X Exit   ^R Read File   ^\ Replace   ^U Paste   ^J Justify   ^/ Go To Line M-E Redo   M-6 Copy   ^Q Where Was   M-W Next   ^F Forward   ^► Next Word

```
sumukh_g@Sumukh:~$ nano strln.sh
sumukh_g@Sumukh:~$ ./strln.sh
The length of Hello World is 11
sumukh_g@Sumukh:~$ nano strln.sh
sumukh_g@Sumukh:~$ ./strln.sh
The length of How are you doing? is 18
sumukh_g@Sumukh:~$ nano strln.sh
sumukh_g@Sumukh:~$
sumukh_g@Sumukh:~$ ./strln.sh
./strln.sh: line 14: unexpected EOF while looking for matching `` ' ''
./strln.sh: line 16: syntax error: unexpected end of file
sumukh_g@Sumukh:~$ nano strln.sh
sumukh_g@Sumukh:~$ ./strln.sh
expr: syntax error: unexpected argument ':.*'
The length of How are you doing? is
sumukh_g@Sumukh:~$ nano strln.sh
sumukh_g@Sumukh:~$ ./strln.sh
./strln.sh: line 19: wc-c: command not found
The length of How are you doing? is
sumukh_g@Sumukh:~$ nano strln.sh
sumukh_g@Sumukh:~$ ./strln.sh
The length of How are you doing? is 1
sumukh_g@Sumukh:~$ nano strln.sh
sumukh_g@Sumukh:~$
```

```bash
#!/bin/bash

#str1="Hello World"
#length=${#str1}
#echo "The length of $str1 is $length"


#str2="How are you doing?"
#length=`expr length "$str2"`
#echo "The length of $str2 is $length"


#str2="How are you doing?"
#length=`expr "$str2" :'.*'`
#echo "The length of $str2 is $length"


#str2="How are you doing?"
#length=`echo $str|wc -c`
#echo "The length of $str2 is $length"
```

```
#!/bin/bash

echo  "Username : $1"
echo  "Age : $2"
echo  "Name: $3"
```

^G Help   ^O Write Out   ^W Where Is   ^K Cut   ^T Execute   ^C Location   M-U Undo   M-A Set Mark   M-] To Bracket M-Q Previous   ^B Back   ^← Prev Word
^X Exit   ^R Read File   ^\ Replace   ^U Paste   ^J Justify   ^/ Go To Line M-E Redo   M-6 Copy   ^Q Where Was   M-W Next   ^F Forward   ^→ Next Word

```
sumukh_g@Sumukh:~$ nano cmdline.sh
sumukh_g@Sumukh:~$ chmod +x cmdline.sh
sumukh_g@Sumukh:~$ ./cmdline.sh abc 23 sumukh
Username : abc
Age : 23
Name: sumukh
sumukh_g@Sumukh:~$
```