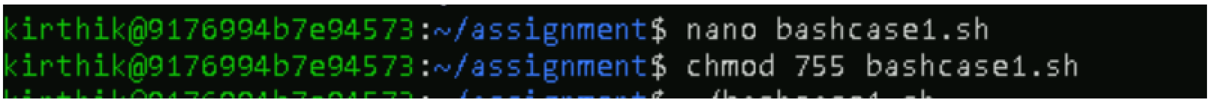
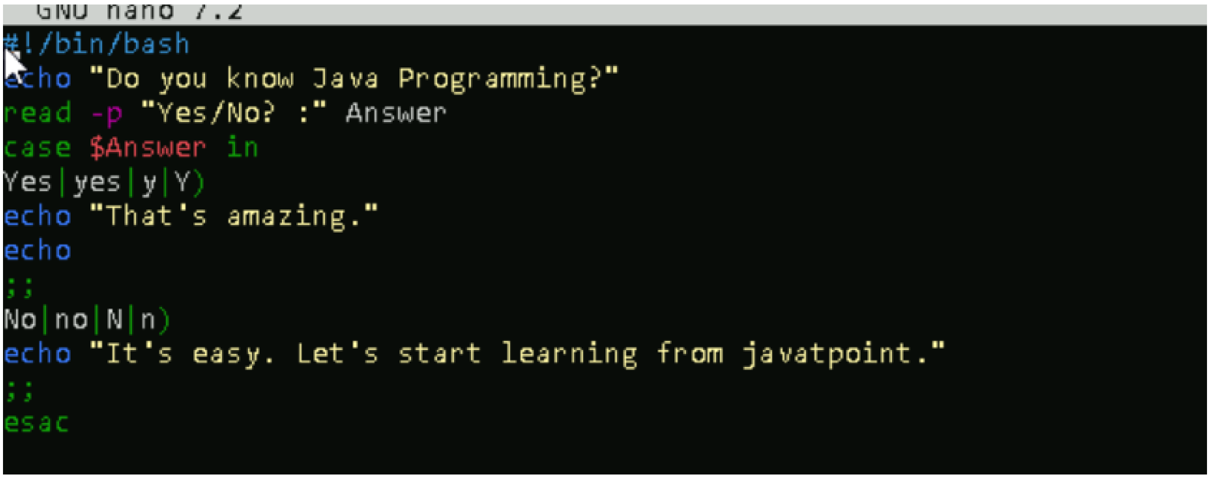
SWITCHCASE:

CREATING FILE:



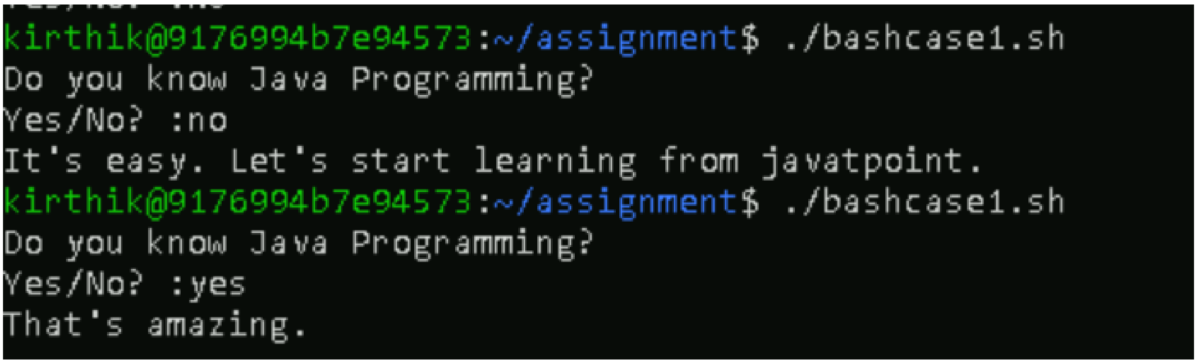
```
kirthik@9176994b7e94573:~/assignment$ nano bashcase1.sh
kirthik@9176994b7e94573:~/assignment$ chmod 755 bashcase1.sh
```

SCRIPT:



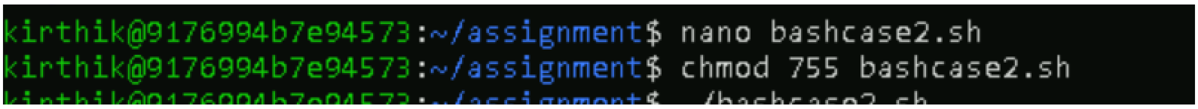
```
  GNU nano 7.2
#!/bin/bash
echo "Do you know Java Programming?"
read -p "Yes/No? :" Answer
case $Answer in
Yes|yes|y|Y)
echo "That's amazing."
echo
;;
No|no|N|n)
echo "It's easy. Let's start learning from javatpoint."
;;
esac
```

OUTPUT:



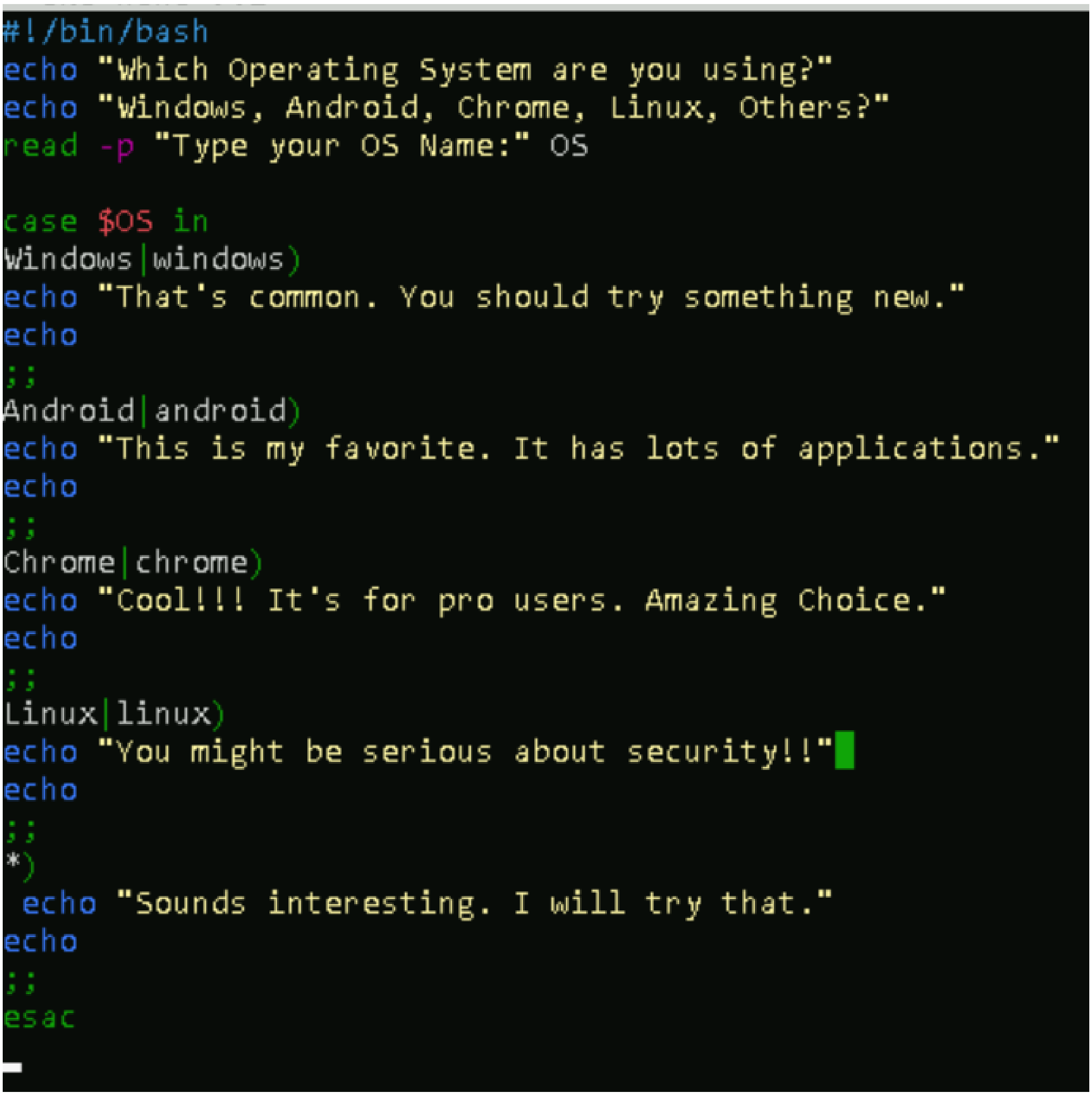
```
kirthik@9176994b7e94573:~/assignment$ ./bashcase1.sh
Do you know Java Programming?
Yes/No? :no
It's easy. Let's start learning from javatpoint.
kirthik@9176994b7e94573:~/assignment$ ./bashcase1.sh
Do you know Java Programming?
Yes/No? :yes
That's amazing.
```

CREATING FILE:



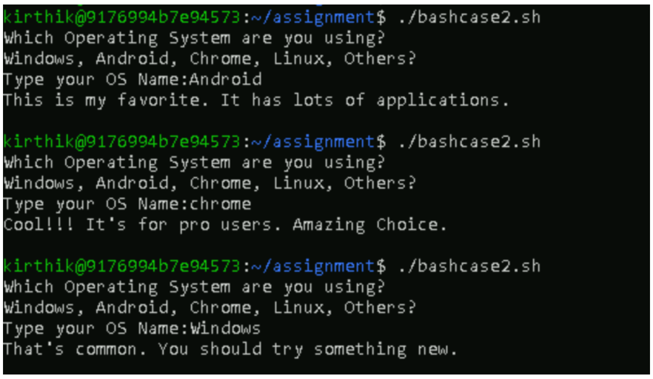
```
kirthik@9176994b7e94573:~/assignment$ nano bashcase2.sh
kirthik@9176994b7e94573:~/assignment$ chmod 755 bashcase2.sh
kirthik@9176994b7e94573:~/assignment$ ./bashcase2.sh
```

SCRIPT:



```
#!/bin/bash
echo "Which Operating System are you using?"
echo "Windows, Android, Chrome, Linux, Others?"
read -p "Type your OS Name:" OS

case $OS in
Windows|windows)
echo "That's common. You should try something new."
echo
;;
Android|android)
echo "This is my favorite. It has lots of applications."
echo
;;
Chrome|chrome)
echo "Cool!!! It's for pro users. Amazing Choice."
echo
;;
Linux|linux)
echo "You might be serious about security!!"
echo
;;
*)
 echo "Sounds interesting. I will try that."
echo
;;
esac
```

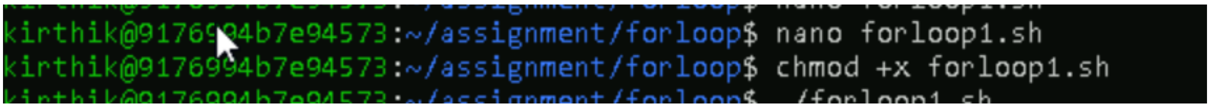
OUTPUT:



```
kirthik@9176994b7e94573:~/assignment$ ./bashcase2.sh
Which Operating System are you using?
Windows, Android, Chrome, Linux, Others?
Type your OS Name:Android
This is my favorite. It has lots of applications.

kirthik@9176994b7e94573:~/assignment$ ./bashcase2.sh
Which Operating System are you using?
Windows, Android, Chrome, Linux, Others?
Type your OS Name:chrome
Cool!!! It's for pro users. Amazing Choice.

kirthik@9176994b7e94573:~/assignment$ ./bashcase2.sh
Which Operating System are you using?
Windows, Android, Chrome, Linux, Others?
Type your OS Name:Windows
That's common. You should try something new.
```

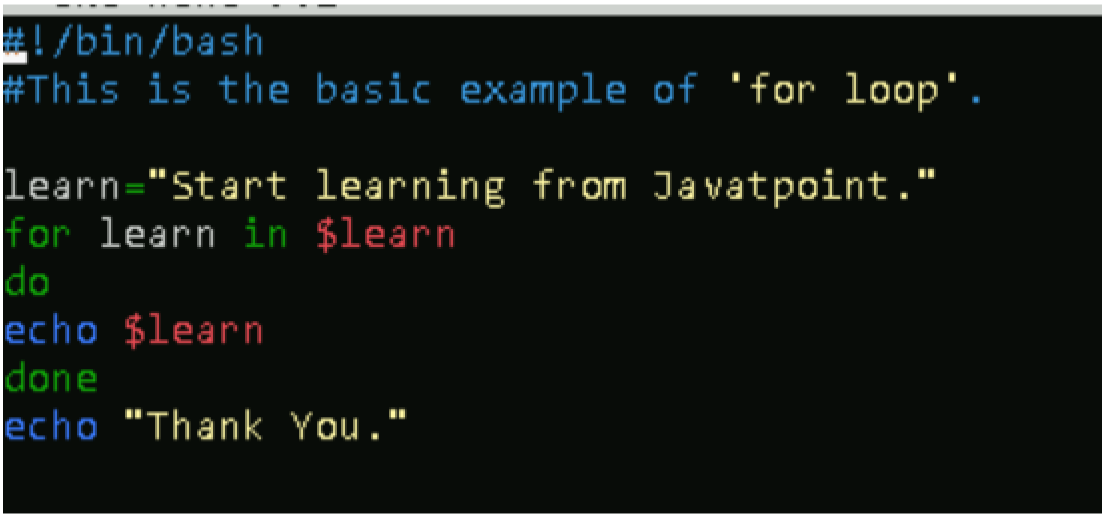
FOR LOOP:

CREATING FILE:



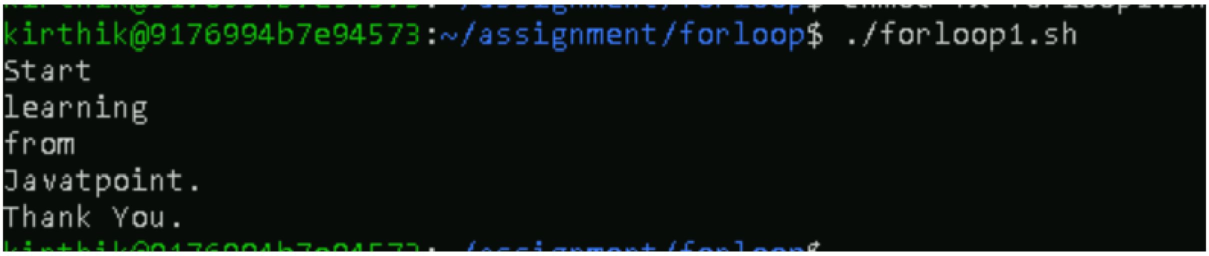
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano forloop1.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x forloop1.sh
```

SCRIPT:



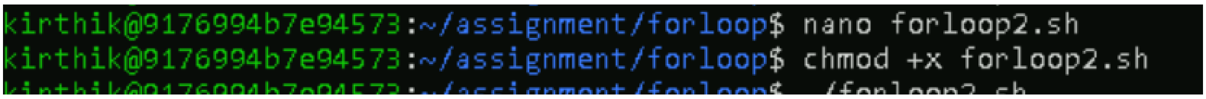
```
#!/bin/bash
#This is the basic example of 'for loop'.

learn="Start learning from Javatpoint."
for learn in $learn
do
echo $learn
done
echo "Thank You."
```

OUTPUT:



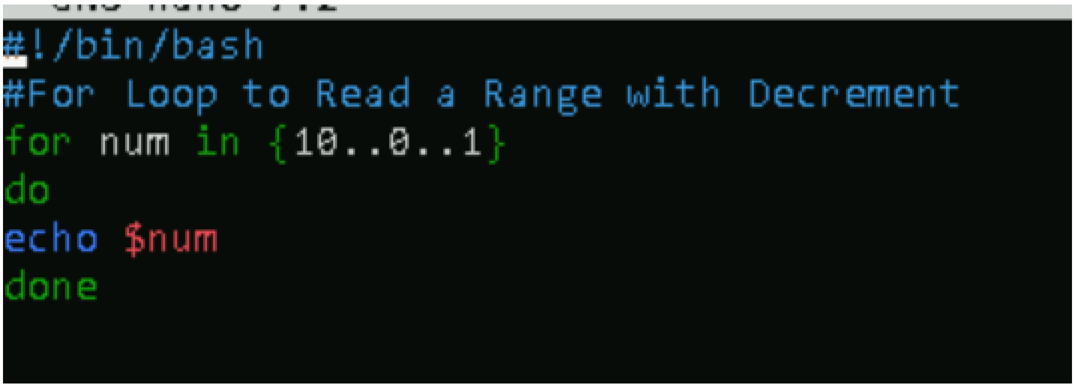
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./forloop1.sh
Start
learning
from
Javatpoint.
Thank You.
```

CREATING FILE:



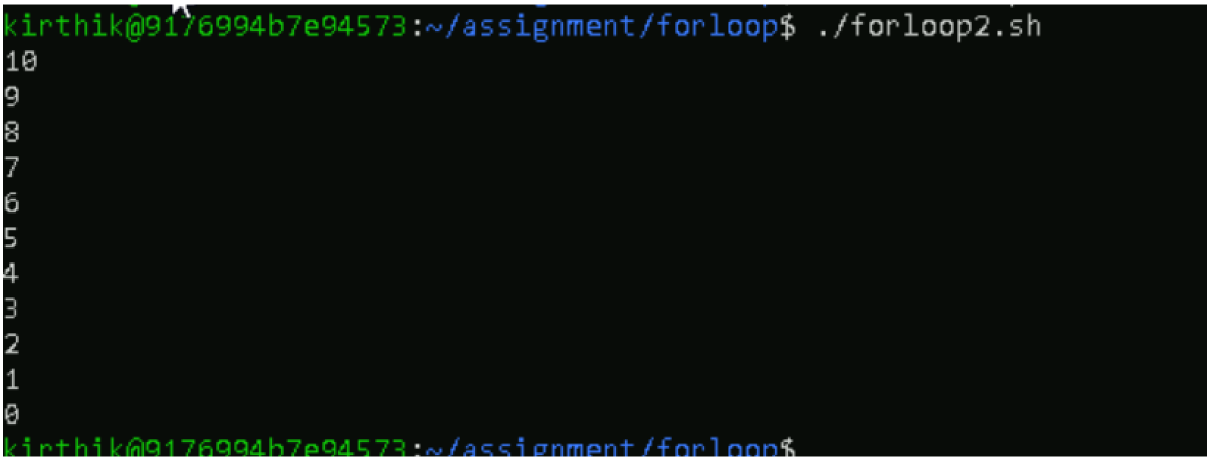
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano forloop2.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x forloop2.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./forloop2.sh
```

SCRIPT:



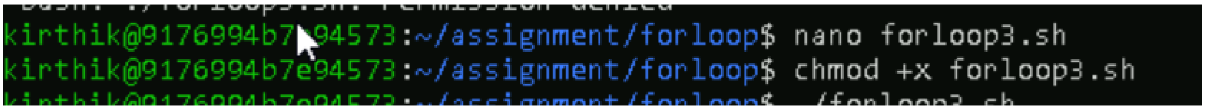
```
  GNU nano 7.2
#!/bin/bash
#For Loop to Read a Range with Decrement
for num in {10..0..1}
do
echo $num
done
```

OUTPUT:



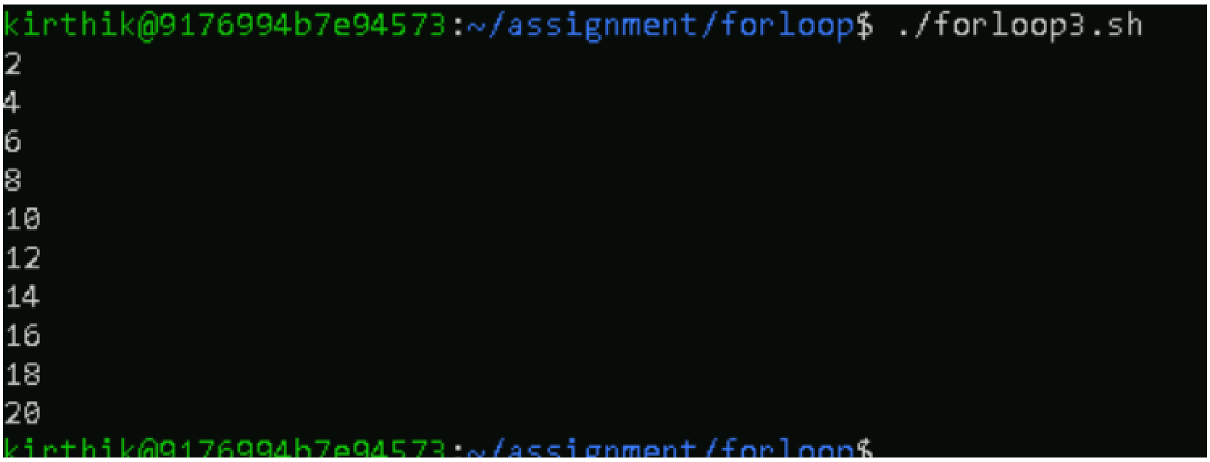
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./forloop2.sh
10
9
8
7
6
5
4
3
2
1
0
kirthik@9176994b7e94573:~/assignment/forloop$
```

CREATING FILE:



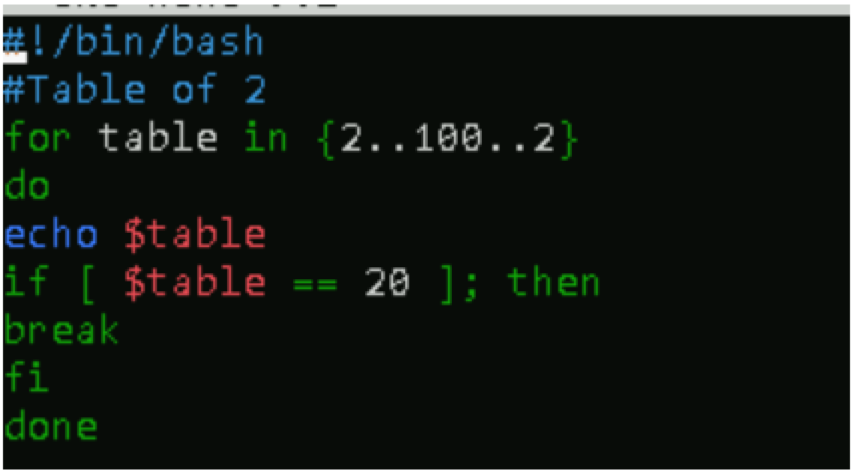
```
bash: ./forloop3.sh: Permission denied
kirthik@9176994b7e94573:~/assignment/forloop$ nano forloop3.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x forloop3.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./forloop3.sh
```

SCRIPT:



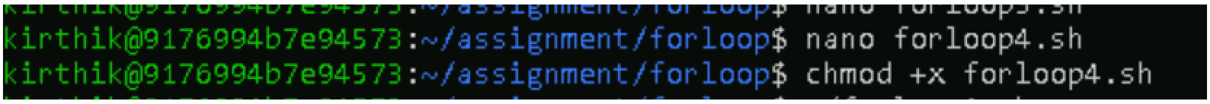
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./forloop3.sh
2
4
6
8
10
12
14
16
18
20
kirthik@9176994b7e94573:~/assignment/forloop$
```

OUTPUT:



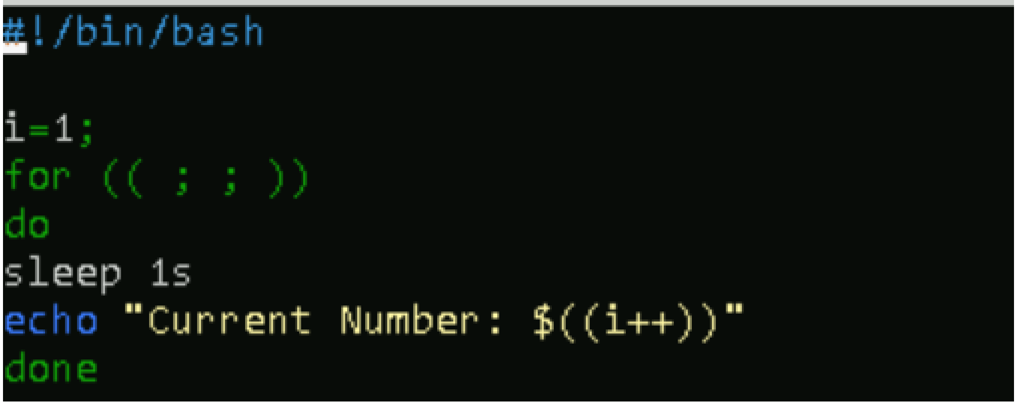
```
#!/bin/bash
#Table of 2
for table in {2..100..2}
do
echo $table
if [ $table == 20 ]; then
break
fi
done
```

CREATING FILE:



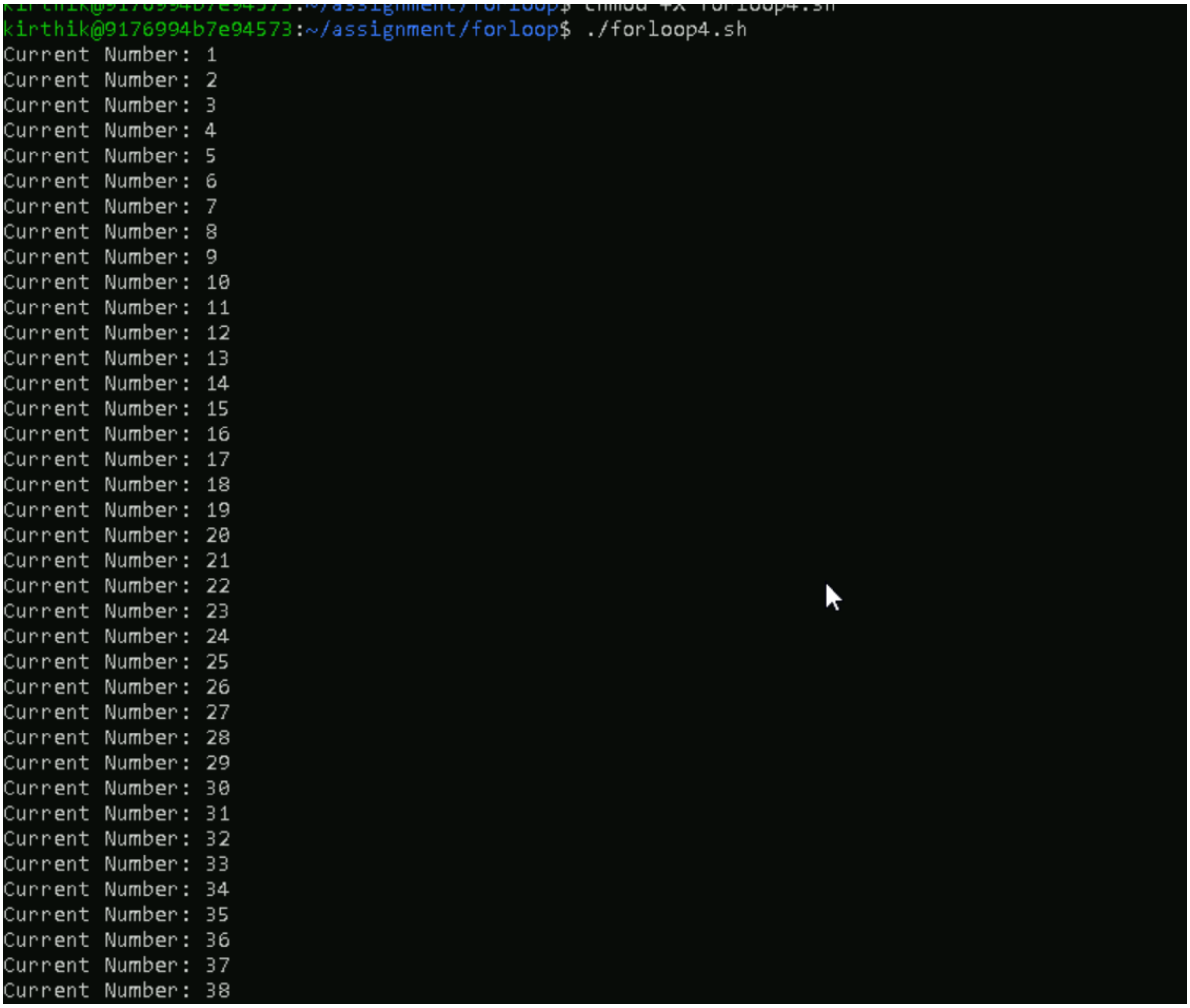
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano forloop3.sh
kirthik@9176994b7e94573:~/assignment/forloop$ nano forloop4.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x forloop4.sh
```

SCRIPT:



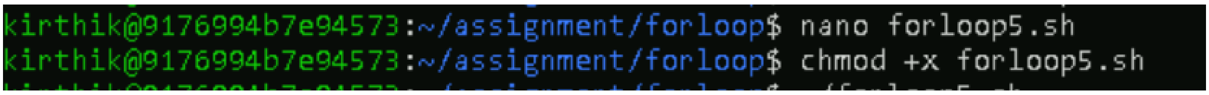
```
#!/bin/bash

i=1;
for (( ; ; ))
do
sleep 1s
echo "Current Number: $((i++))"
done
```

OUTPUT:



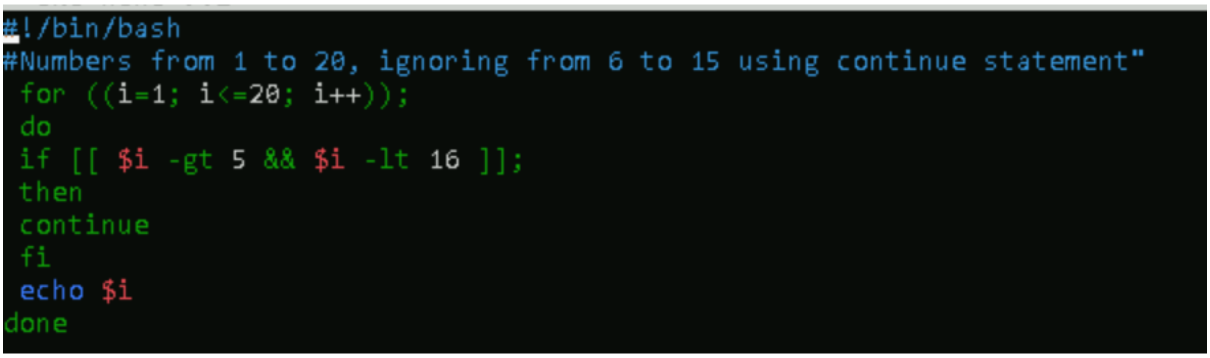
```
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x forloop4.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./forloop4.sh
Current Number: 1
Current Number: 2
Current Number: 3
Current Number: 4
Current Number: 5
Current Number: 6
Current Number: 7
Current Number: 8
Current Number: 9
Current Number: 10
Current Number: 11
Current Number: 12
Current Number: 13
Current Number: 14
Current Number: 15
Current Number: 16
Current Number: 17
Current Number: 18
Current Number: 19
Current Number: 20
Current Number: 21
Current Number: 22
Current Number: 23
Current Number: 24
Current Number: 25
Current Number: 26
Current Number: 27
Current Number: 28
Current Number: 29
Current Number: 30
Current Number: 31
Current Number: 32
Current Number: 33
Current Number: 34
Current Number: 35
Current Number: 36
Current Number: 37
Current Number: 38
```

CREATING FILE:



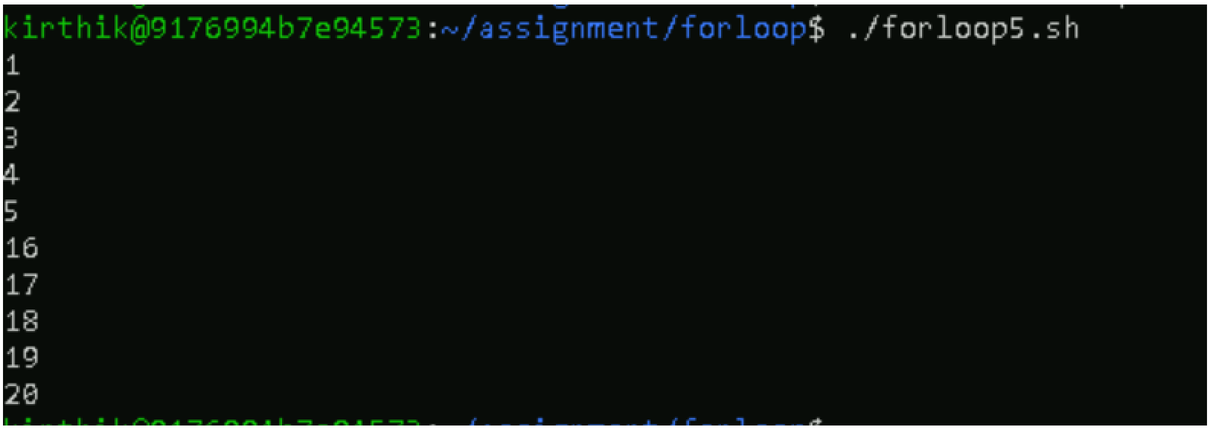
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano forloop5.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x forloop5.sh
```

SCRIPT:



```
#!/bin/bash
#Numbers from 1 to 20, ignoring from 6 to 15 using continue statement"
 for ((i=1; i<=20; i++));
 do
 if [[ $i -gt 5 && $i -lt 16 ]];
 then
 continue
 fi
 echo $i
done
```

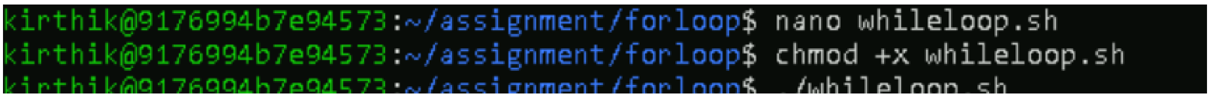
OUTPUT:



```
kirthik@9176994b7e94573:~/assignment/forloop$ ./forloop5.sh
1
2
3
4
5
16
17
18
19
20
```

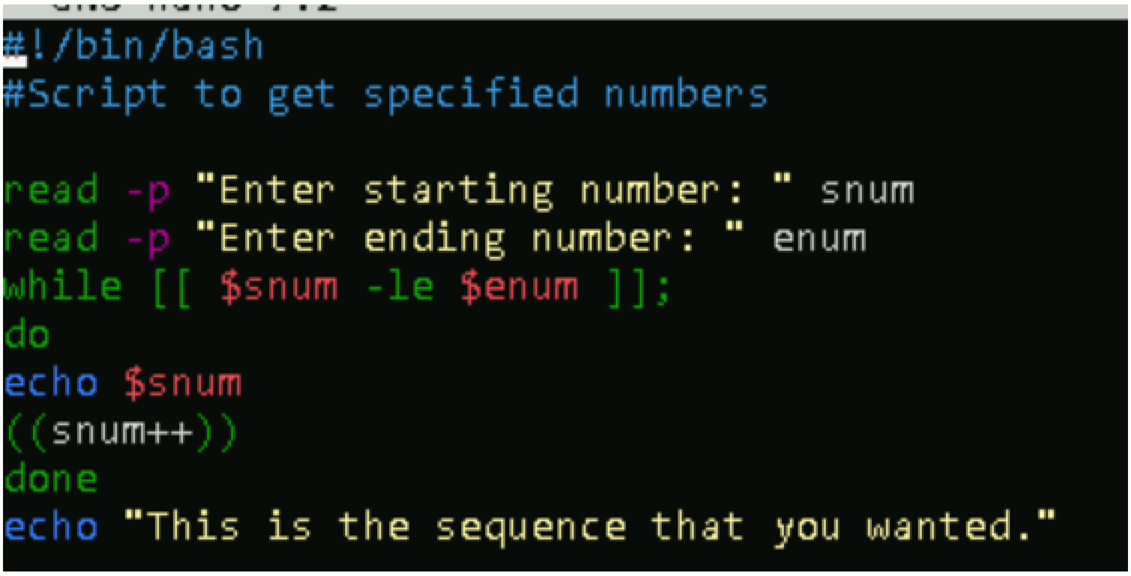
DO WHILE LOOP

CREATING FILE:



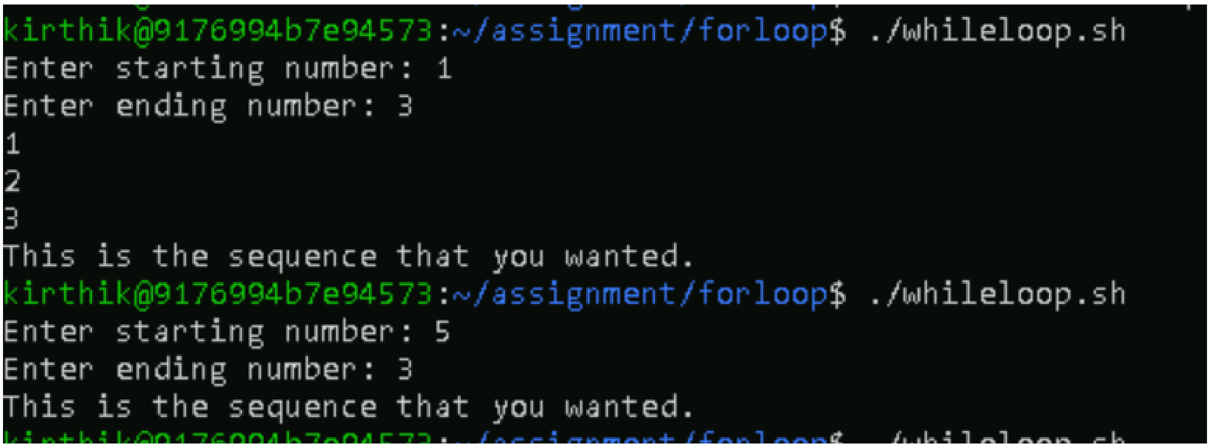
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano whileloop.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x whileloop.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./whileloop.sh
```

SCRIPT:



```
#!/bin/bash
#Script to get specified numbers

read -p "Enter starting number: " snum
read -p "Enter ending number: " enum
while [[ $snum -le $enum ]];
do
echo $snum
((snum++))
done
echo "This is the sequence that you wanted."
```

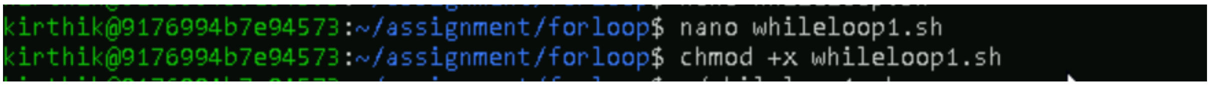
OUTPUT:



```
kirthik@9176994b7e94573:~/assignment/forloop$ ./whileloop.sh
Enter starting number: 1
Enter ending number: 3
1
2
3
This is the sequence that you wanted.
kirthik@9176994b7e94573:~/assignment/forloop$ ./whileloop.sh
Enter starting number: 5
Enter ending number: 3
This is the sequence that you wanted.
kirthik@9176994b7e94573:~/assignment/forloop$ ./whileloop.sh
```

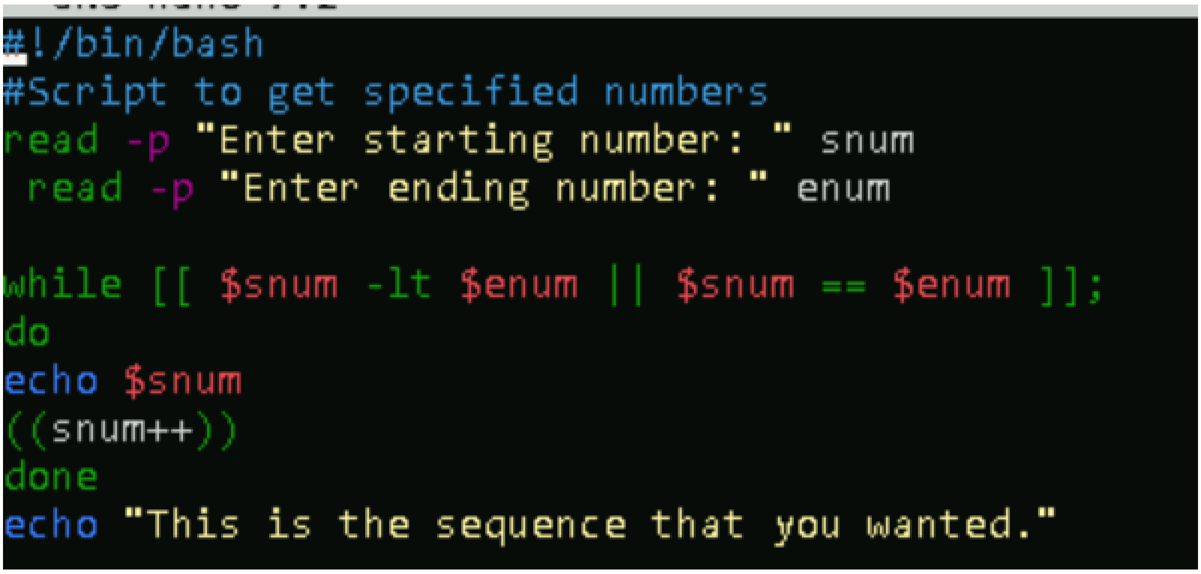
WHILELOOP

CREATING FILE:



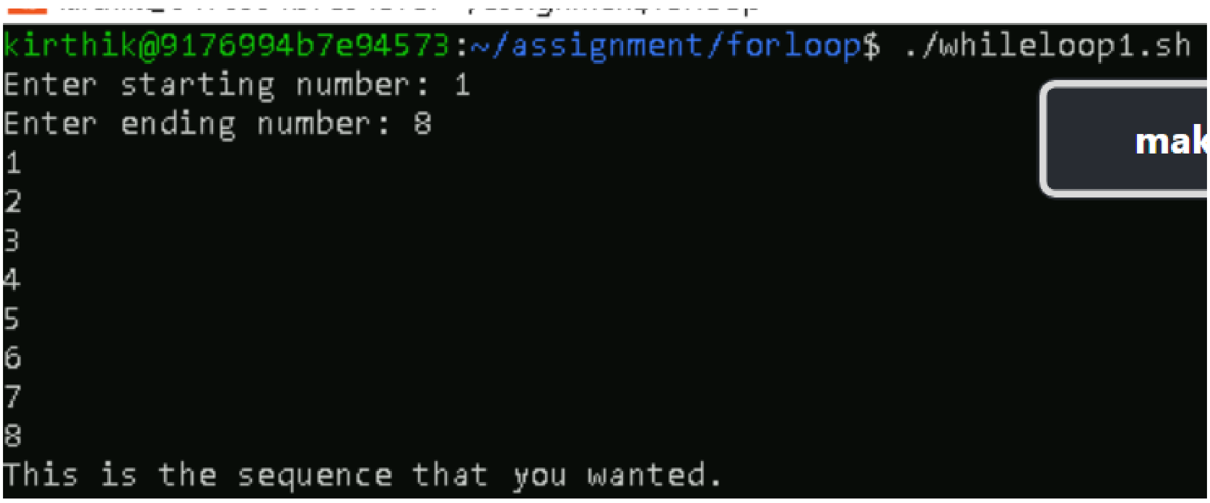
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano whileloop1.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x whileloop1.sh
```

SCRIPT:



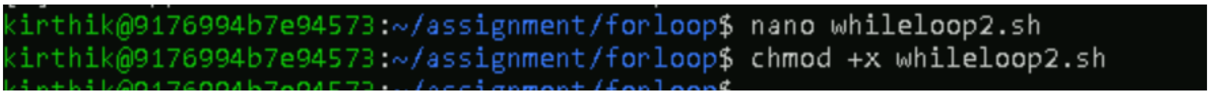
```
#!/bin/bash
#Script to get specified numbers
read -p "Enter starting number: " snum
 read -p "Enter ending number: " enum

while [[ $snum -lt $enum || $snum == $enum ]];
do
echo $snum
((snum++))
done
echo "This is the sequence that you wanted."
```

OUTPUT:



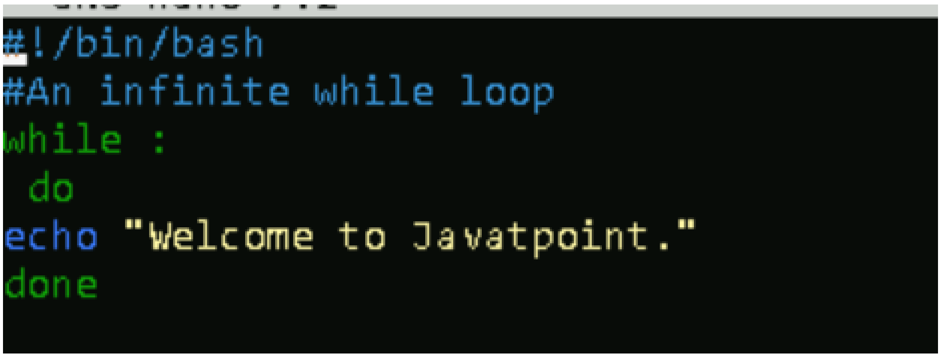
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./whileloop1.sh
Enter starting number: 1
Enter ending number: 8
1
2
3
4
5
6
7
8
This is the sequence that you wanted.
```

CREATING FILE:



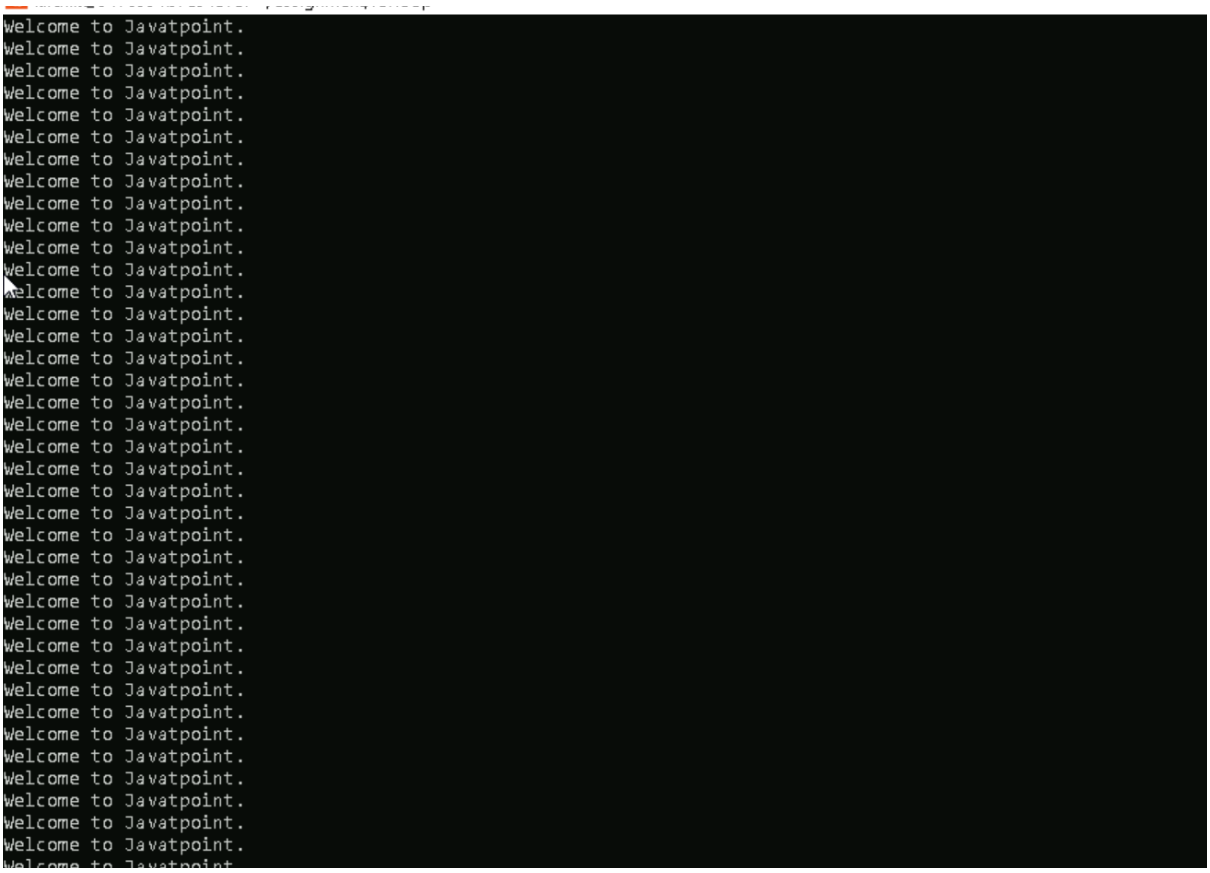
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano whileloop2.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x whileloop2.sh
kirthik@9176994b7e94573:~/assignment/forloop$
```

SCRIPT:



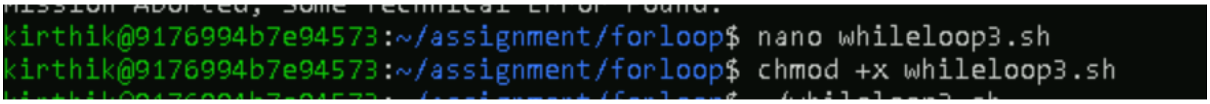
```
#!/bin/bash
#An infinite while loop
while :
 do
echo "Welcome to Javatpoint."
done
```

OUTPUT



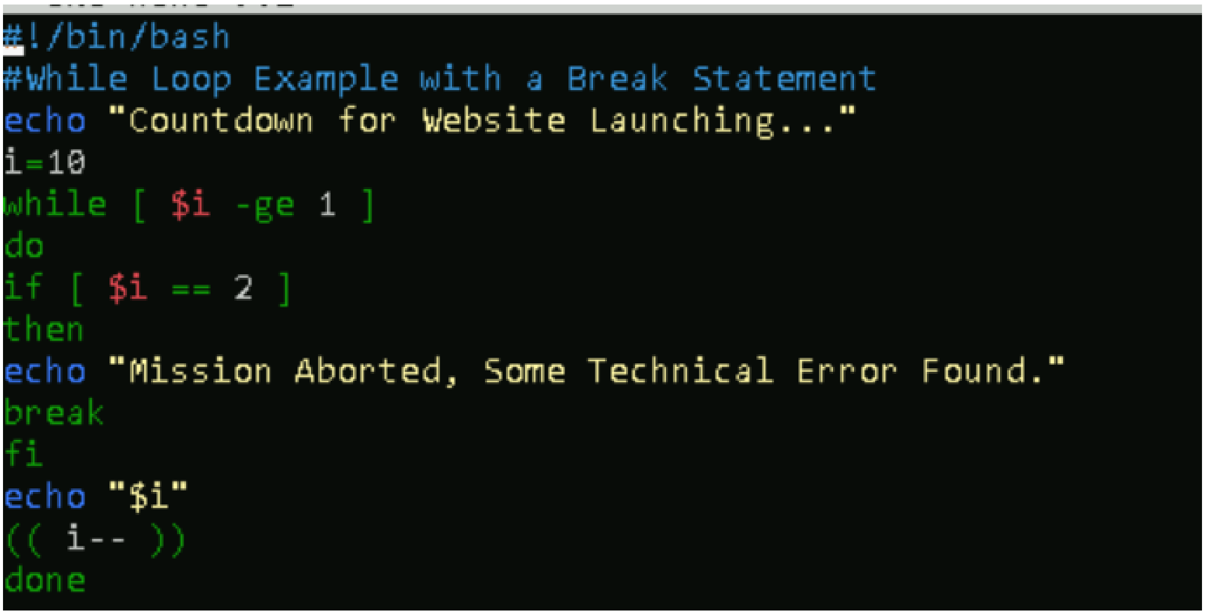
```
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
Welcome to Javatpoint.
```

CREATING FILE:



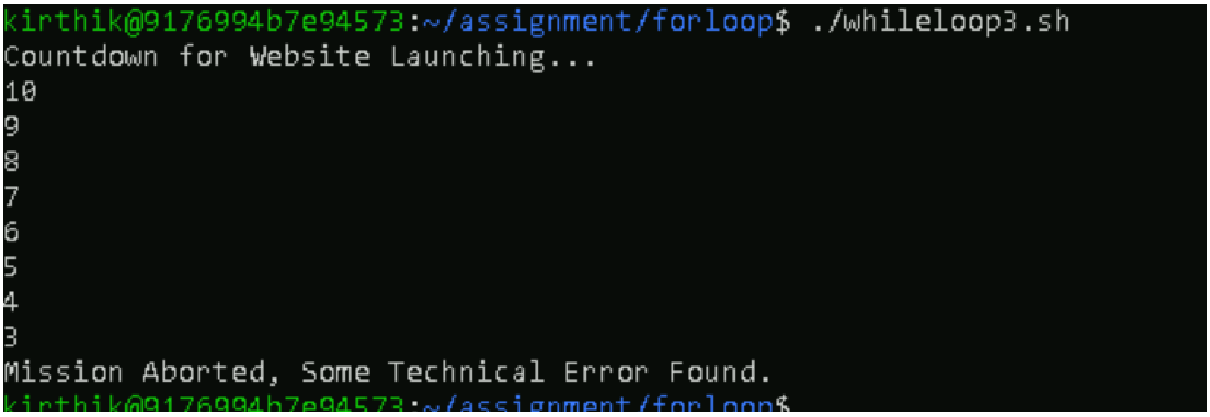
```
Mission Aborted, Some Technical Error Found.
kirthik@9176994b7e94573:~/assignment/forloop$ nano whileloop3.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x whileloop3.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./whileloop3.sh
```

SCRIPT:



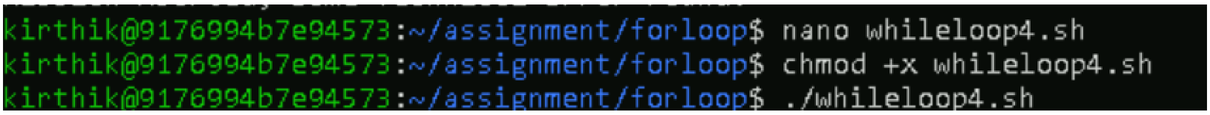
```
#!/bin/bash
#While Loop Example with a Break Statement
echo "Countdown for Website Launching..."
i=10
while [ $i -ge 1 ]
do
if [ $i == 2 ]
then
echo "Mission Aborted, Some Technical Error Found."
break
fi
echo "$i"
(( i-- ))
done
```

OUTPUT:



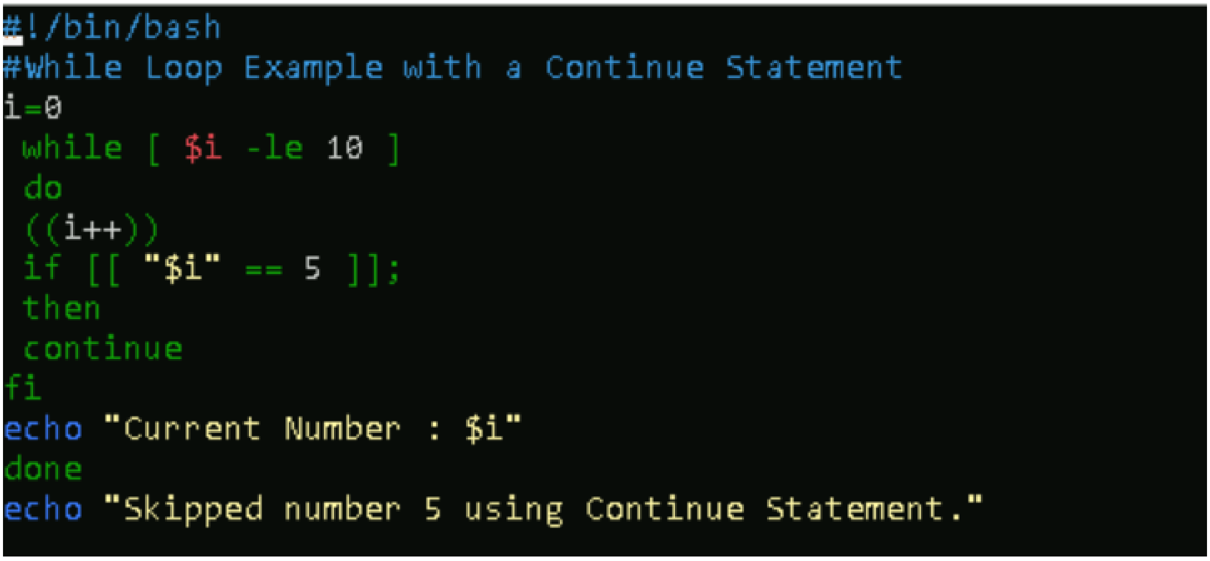
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./whileloop3.sh
Countdown for Website Launching...
10
9
8
7
6
5
4
3
Mission Aborted, Some Technical Error Found.
kirthik@9176994b7e94573:~/assignment/forloop$
```

CREATING FILES:



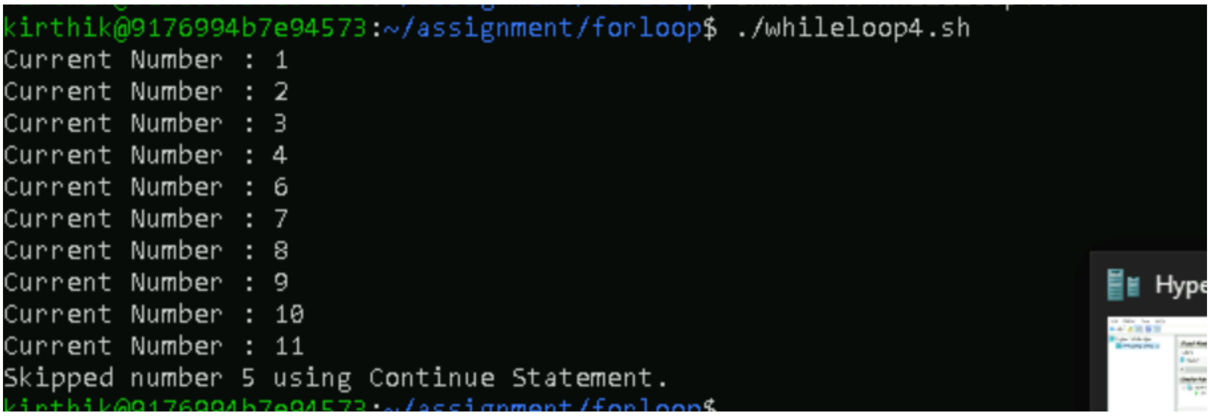
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano whileloop4.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x whileloop4.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./whileloop4.sh
```

SCRIPT:



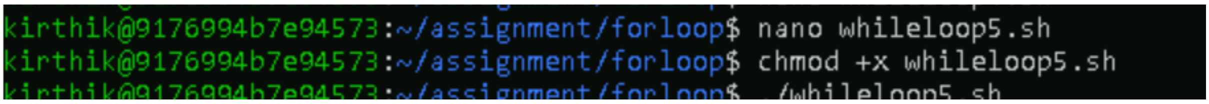
```
#!/bin/bash
#While Loop Example with a Continue Statement
i=0
 while [ $i -le 10 ]
 do
 ((i++))
 if [[ "$i" == 5 ]];
 then
 continue
fi
echo "Current Number : $i"
done
echo "Skipped number 5 using Continue Statement."
```

OUTPUT:



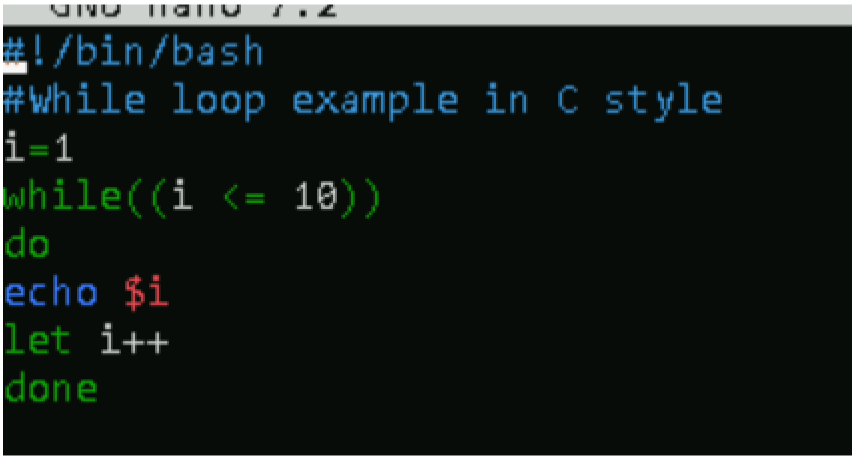
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./whileloop4.sh
Current Number : 1
Current Number : 2
Current Number : 3
Current Number : 4
Current Number : 6
Current Number : 7
Current Number : 8
Current Number : 9
Current Number : 10
Current Number : 11
Skipped number 5 using Continue Statement.
kirthik@9176994b7e94573:~/assignment/forloop$
```

CREATING FILE:



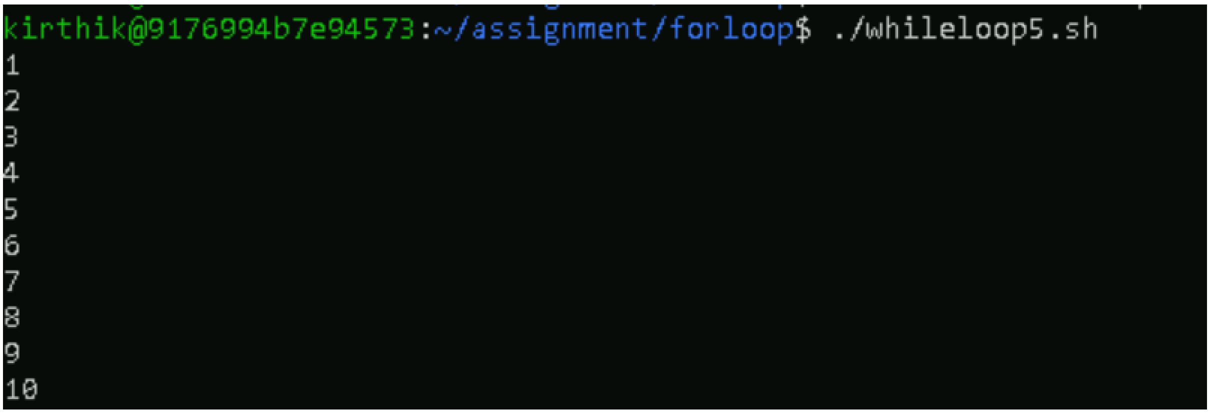
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano whileloop5.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x whileloop5.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./whileloop5.sh
```

SCRIPT:



```
  GNU nano 7.2
#!/bin/bash
#While loop example in C style
i=1
while((i <= 10))
do
echo $i
let i++
done
```

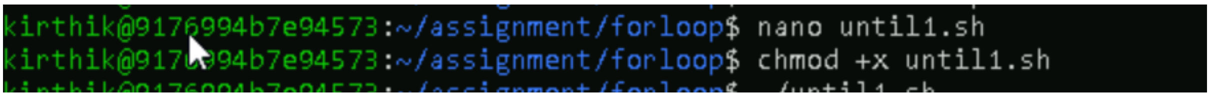
OUTPUT:



```
kirthik@9176994b7e94573:~/assignment/forloop$ ./whileloop5.sh
1
2
3
4
5
6
7
8
9
10
```

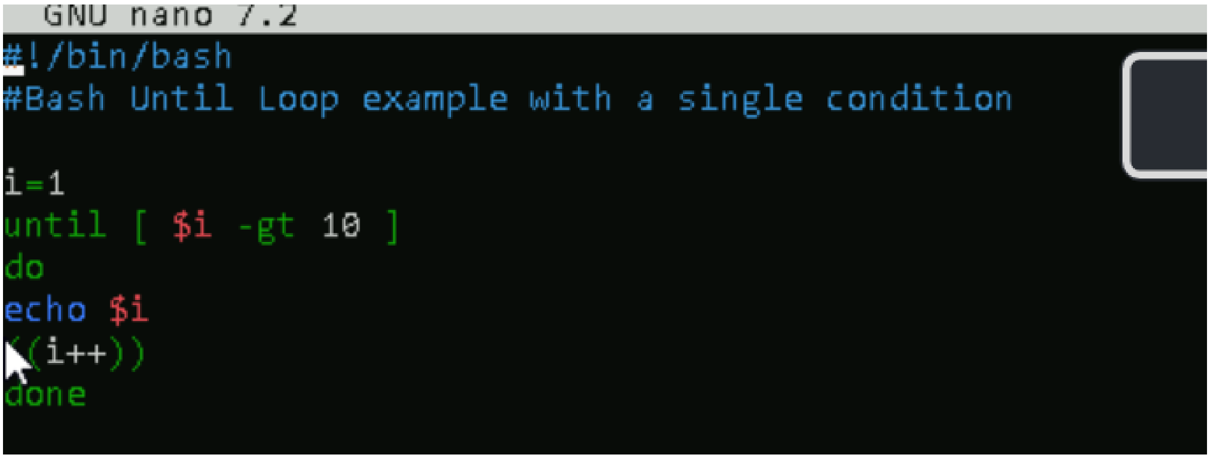
UNTIL

CREATING FILE:



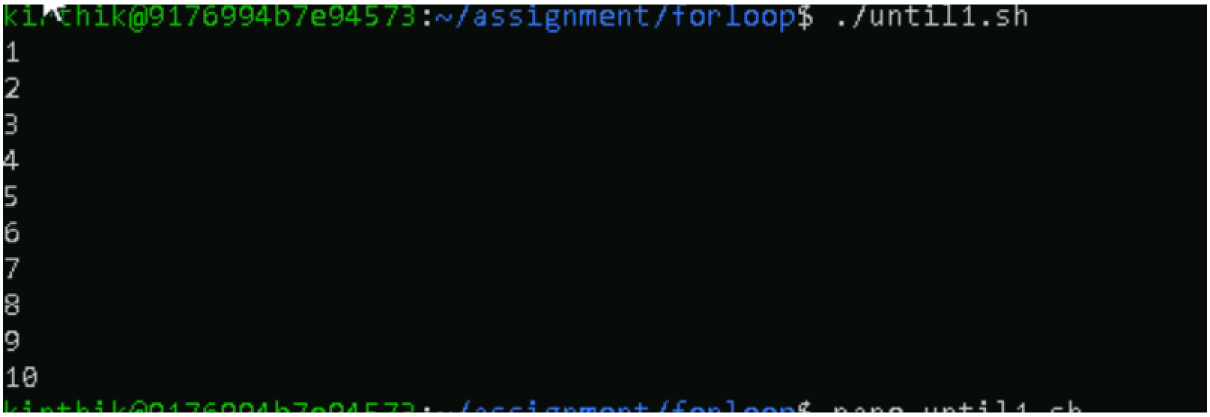
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano until1.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x until1.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./until1.sh
```

SCRIPT:



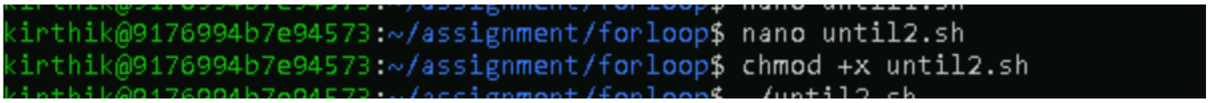
```
  GNU nano 7.2
#!/bin/bash
#Bash Until Loop example with a single condition

i=1
until [ $i -gt 10 ]
do
echo $i
((i++))
done
```

OUTPUT:



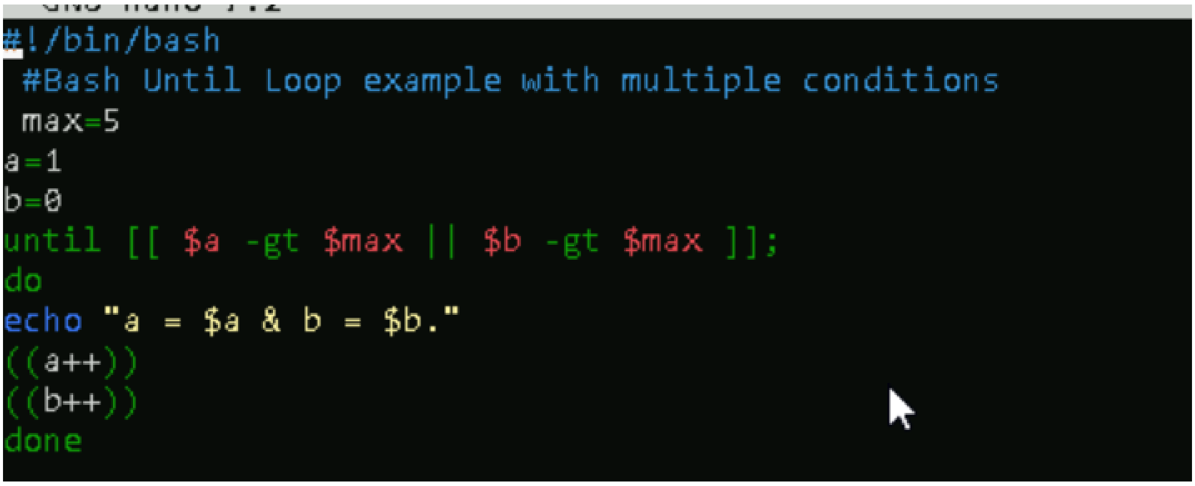
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./until1.sh
1
2
3
4
5
6
7
8
9
10
kirthik@9176994b7e94573:~/assignment/forloop$ nano until1.sh
```

CREATING FILE:



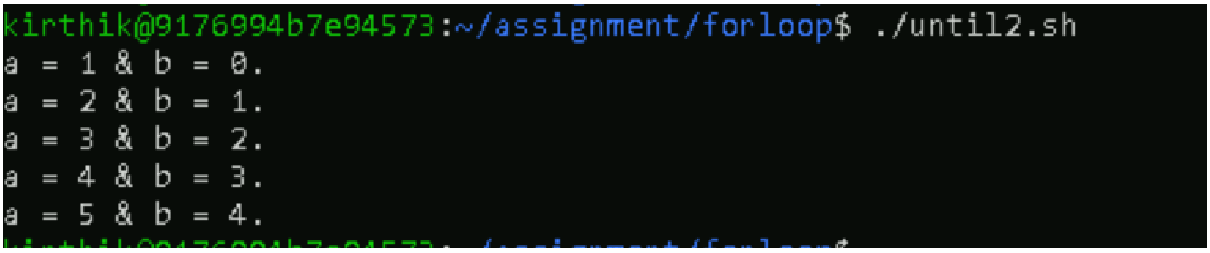
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano until1.sh
kirthik@9176994b7e94573:~/assignment/forloop$ nano until2.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x until2.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./until2.sh
```

SCRIPT:



```
  GNU nano 7.2
#!/bin/bash
 #Bash Until Loop example with multiple conditions
 max=5
a=1
b=0
until [[ $a -gt $max || $b -gt $max ]];
do
echo "a = $a & b = $b."
((a++))
((b++))
done
```

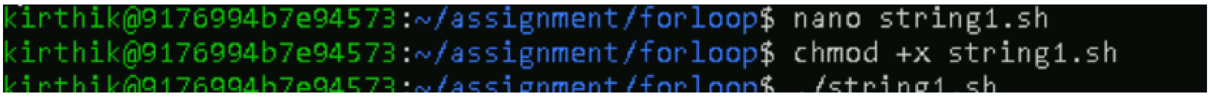
OUTPUT:



```
kirthik@9176994b7e94573:~/assignment/forloop$ ./until2.sh
a = 1 & b = 0.
a = 2 & b = 1.
a = 3 & b = 2.
a = 4 & b = 3.
a = 5 & b = 4.
kirthik@9176994b7e94573:~/assignment/forloop$
```

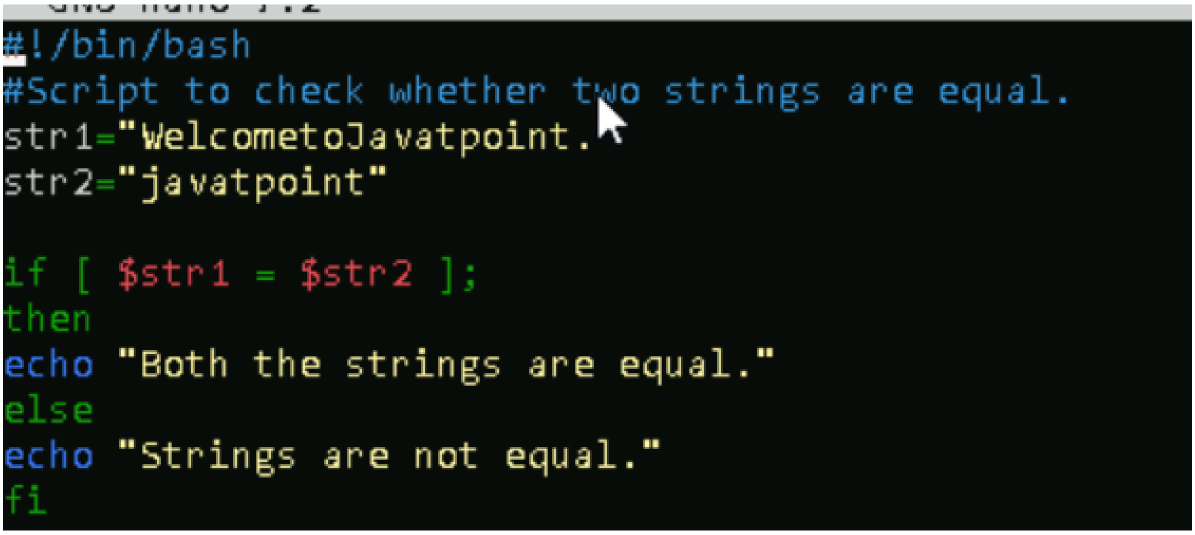
STRING:

CREATING FILE:



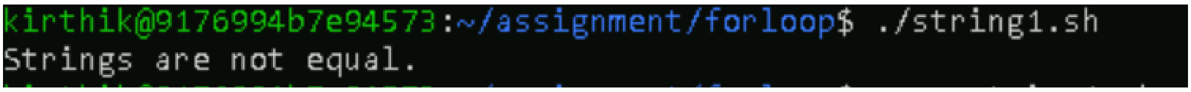
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano string1.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x string1.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./string1.sh
```

SCRIPT:



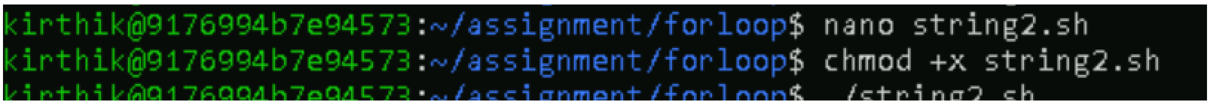
```
#!/bin/bash
#Script to check whether two strings are equal.
str1="WelcometoJavatpoint."
str2="javatpoint"

if [ $str1 = $str2 ];
then
echo "Both the strings are equal."
else
echo "Strings are not equal."
fi
```

OUTPUT:



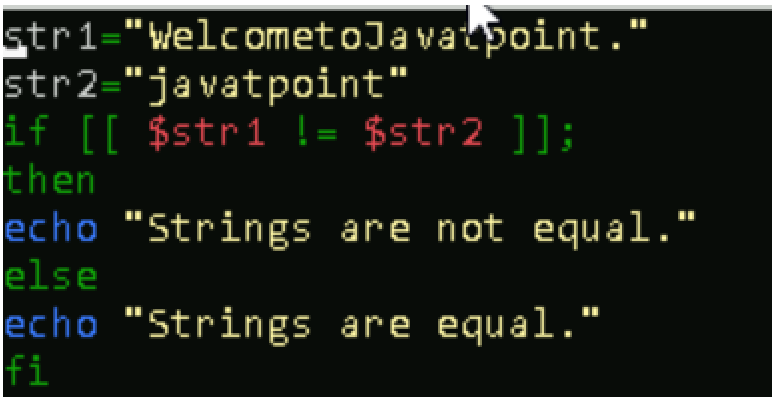
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./string1.sh
Strings are not equal.
```

CREATING FILE:



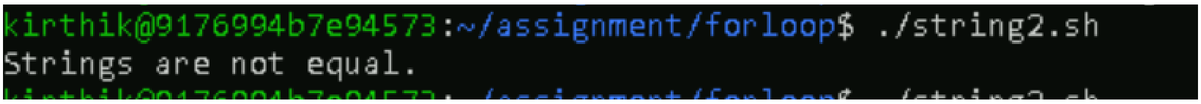
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano string2.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x string2.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./string2.sh
```

SCRIPT:



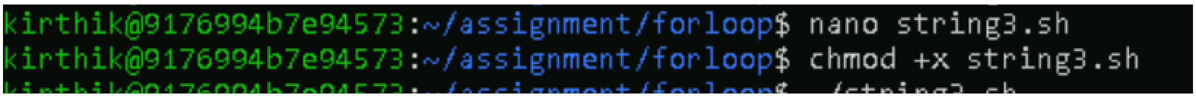
```
str1="WelcometoJavatpoint."
str2="javatpoint"
if [[ $str1 != $str2 ]];
then
echo "Strings are not equal."
else
echo "Strings are equal."
fi
```

OUTPUT:



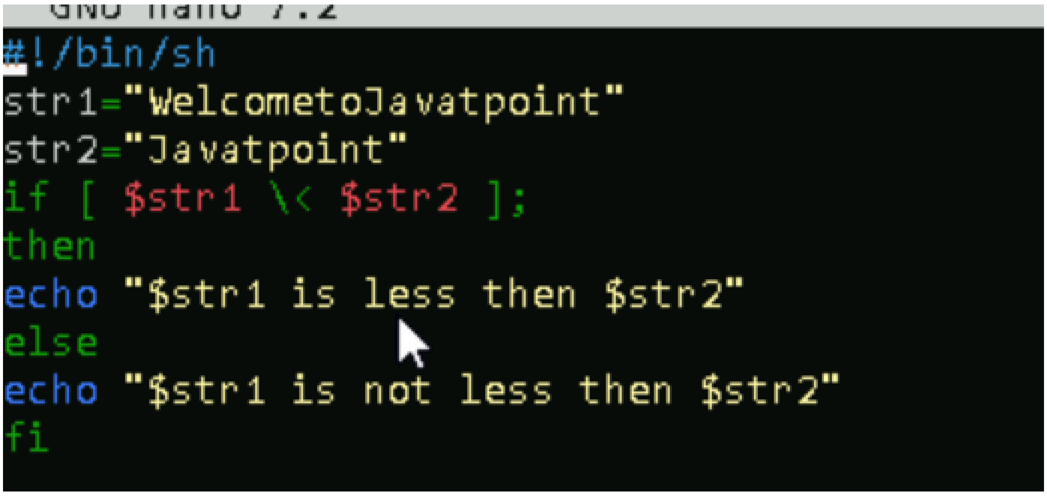
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./string2.sh
Strings are not equal.
kirthik@9176994b7e94573:~/assignment/forloop$ ./string2.sh
```

CREATING FILE:



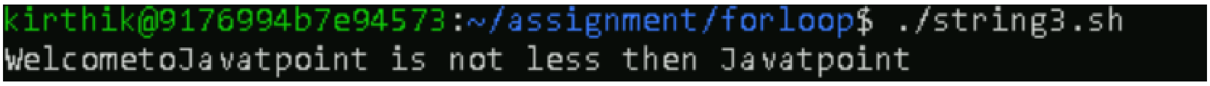
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano string3.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x string3.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./string3.sh
```

SCRIPT:



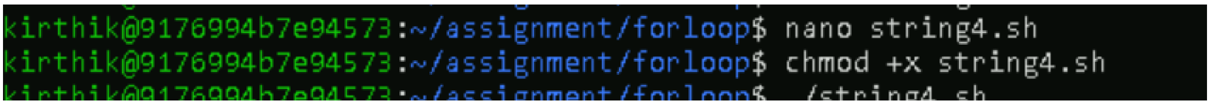
```
  GNU nano 7.2
#!/bin/sh
str1="WelcometoJavatpoint"
str2="Javatpoint"
if [ $str1 \< $str2 ];
then
echo "$str1 is less then $str2"
else
echo "$str1 is not less then $str2"
fi
```

OUTPUT:



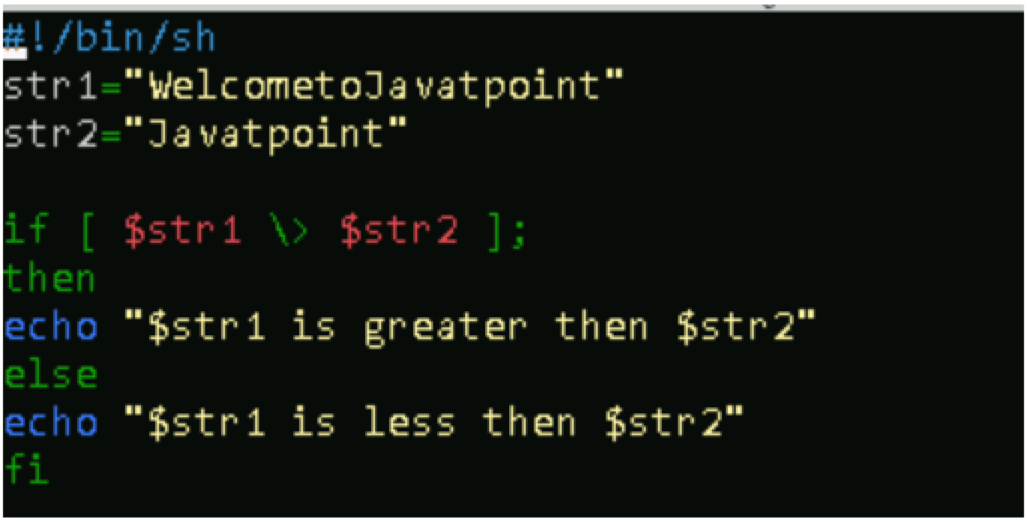
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./string3.sh
WelcometoJavatpoint is not less then Javatpoint
```

CREATING FILE:



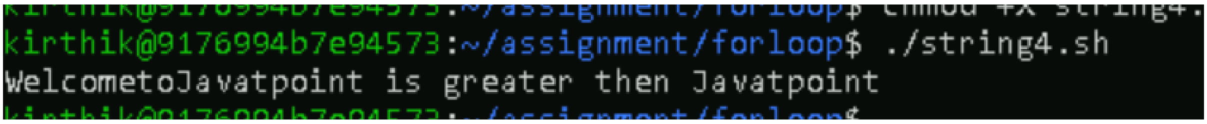
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano string4.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x string4.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./string4.sh
```

SCRIPT:



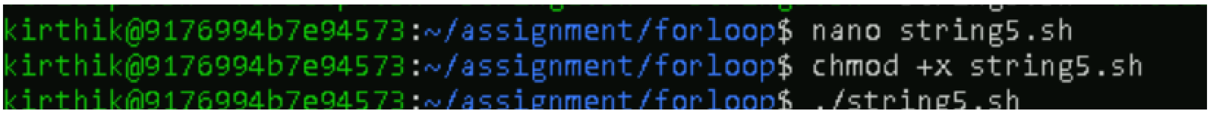
```
#!/bin/sh
str1="WelcometoJavatpoint"
str2="Javatpoint"

if [ $str1 \> $str2 ];
then
echo "$str1 is greater then $str2"
else
echo "$str1 is less then $str2"
fi
```

OUTPUT:



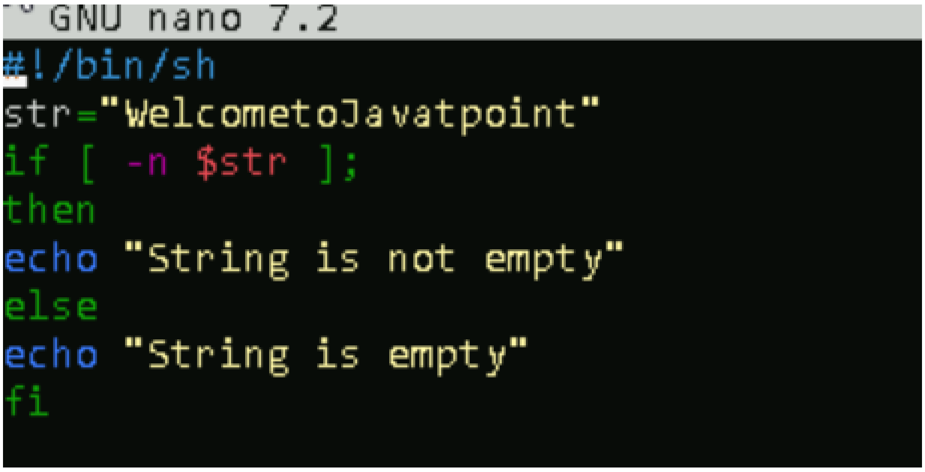
```
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x string4.
kirthik@9176994b7e94573:~/assignment/forloop$ ./string4.sh
WelcometoJavatpoint is greater then Javatpoint
kirthik@9176994b7e94573:~/assignment/forloop$
```

CREATING FILE:



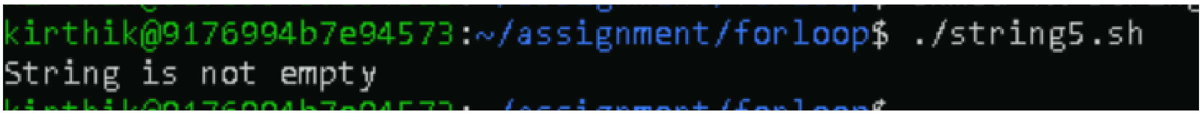
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano string5.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x string5.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./string5.sh
```

SCRIPT:



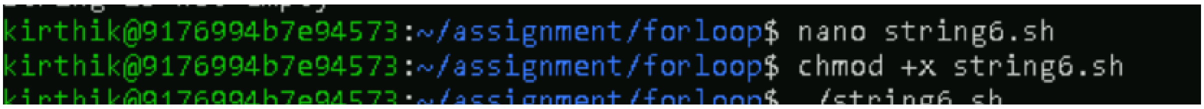
```
GNU nano 7.2
#!/bin/sh
str="WelcometoJavatpoint"
if [ -n $str ];
then
echo "String is not empty"
else
echo "String is empty"
fi
```

OUTPUT:



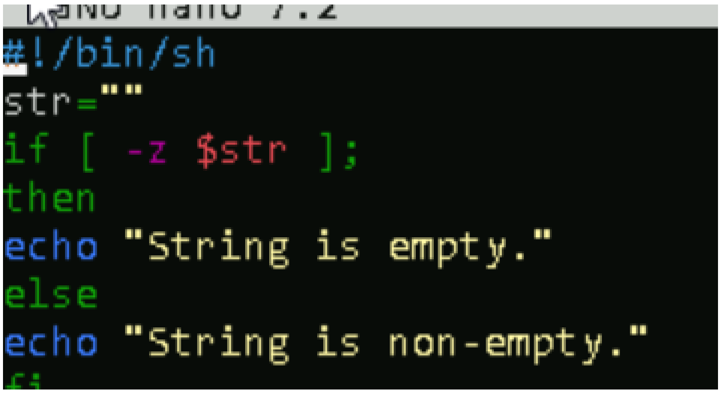
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./string5.sh
String is not empty
kirthik@9176994b7e94573:~/assignment/forloop$
```

CREATING FILE:



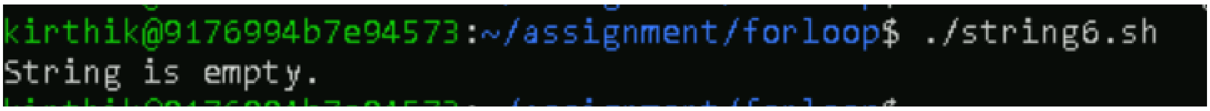
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano string6.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x string6.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./string6.sh
```

SCRIPT:



```
GNU nano 7.2
#!/bin/sh
str=""
if [ -z $str ];
then
echo "String is empty."
else
echo "String is non-empty."
fi
```

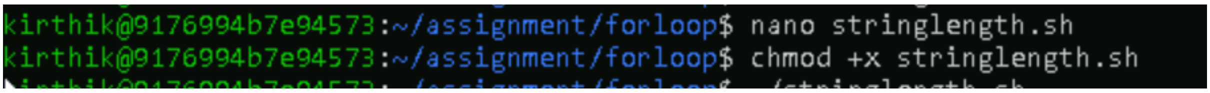
OUTPUT:



```
kirthik@9176994b7e94573:~/assignment/forloop$ ./string6.sh
String is empty.
kirthik@9176994b7e94573:~/assignment/forloop$
```

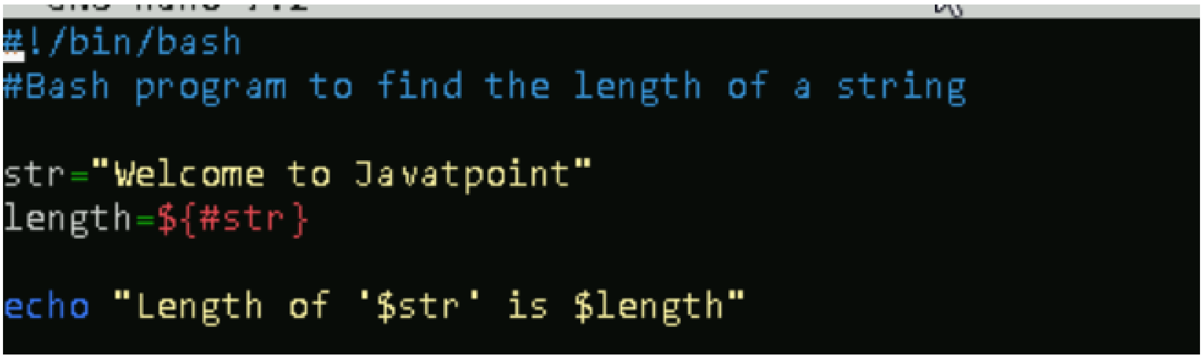
STRINGLENGTH:

CREATING FILE:



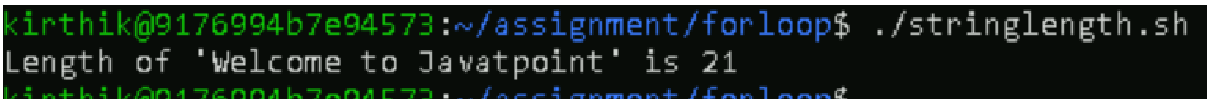
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano stringlength.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x stringlength.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./stringlength.sh
```

SCRIPT:



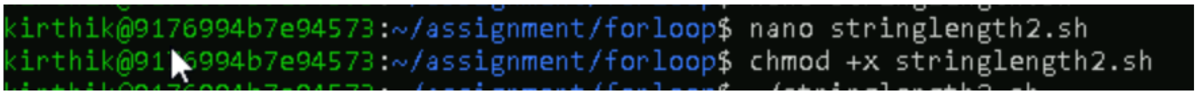
```
  GNU nano 7.2
#!/bin/bash
#Bash program to find the length of a string

str="Welcome to Javatpoint"
length=${#str}

echo "Length of '$str' is $length"
```

OUTPUT:



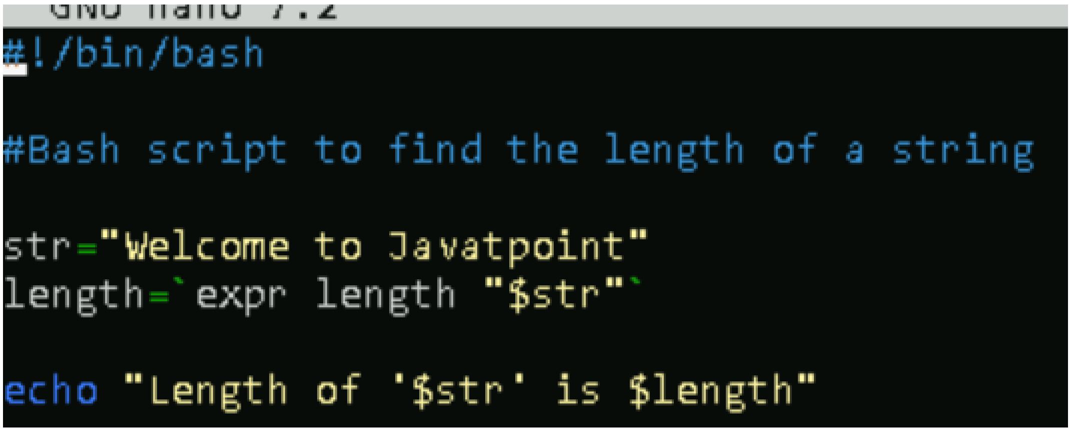
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./stringlength.sh
Length of 'Welcome to Javatpoint' is 21
kirthik@9176994b7e94573:~/assignment/forloop$
```

CREATING FILE:



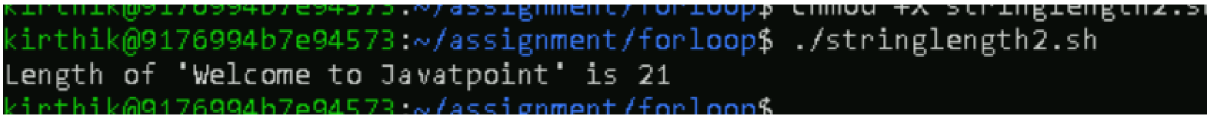
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano stringlength2.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x stringlength2.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./stringlength2.sh
```

SCRIPT:



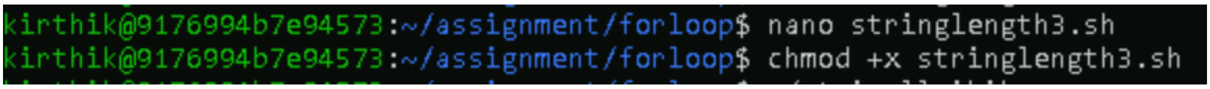
```
  GNU nano 7.2
#!/bin/bash

#Bash script to find the length of a string

str="Welcome to Javatpoint"
length=`expr length "$str"`

echo "Length of '$str' is $length"
```

OUTPUT:



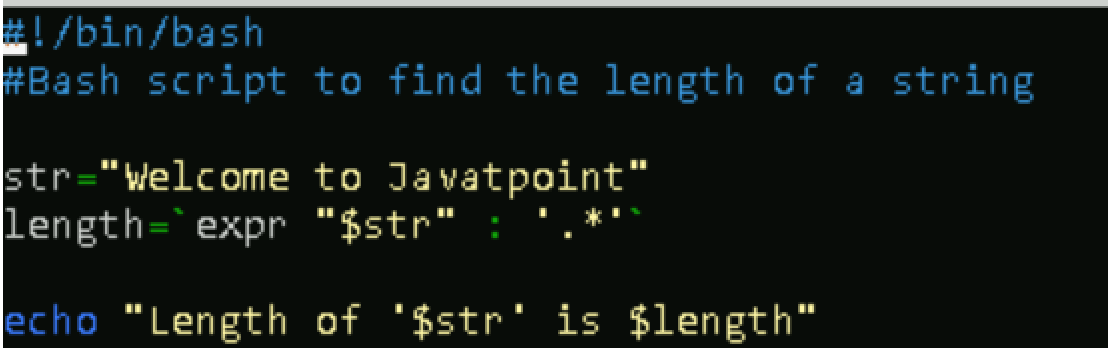
```
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x stringlength2.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./stringlength2.sh
Length of 'Welcome to Javatpoint' is 21
kirthik@9176994b7e94573:~/assignment/forloop$
```

CREATING FILE:



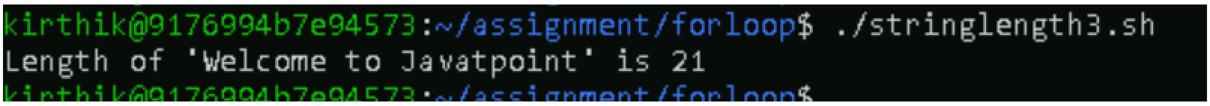
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano stringlength3.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x stringlength3.sh
```

SCRIPT:



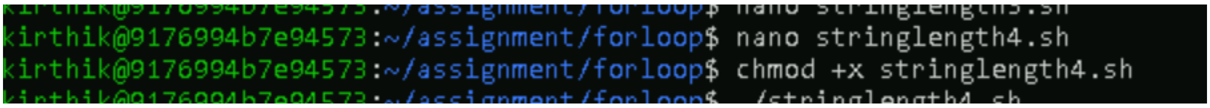
```
#!/bin/bash
#Bash script to find the length of a string

str="Welcome to Javatpoint"
length=`expr "$str" : '.*'`

echo "Length of '$str' is $length"
```

OUTPUT:



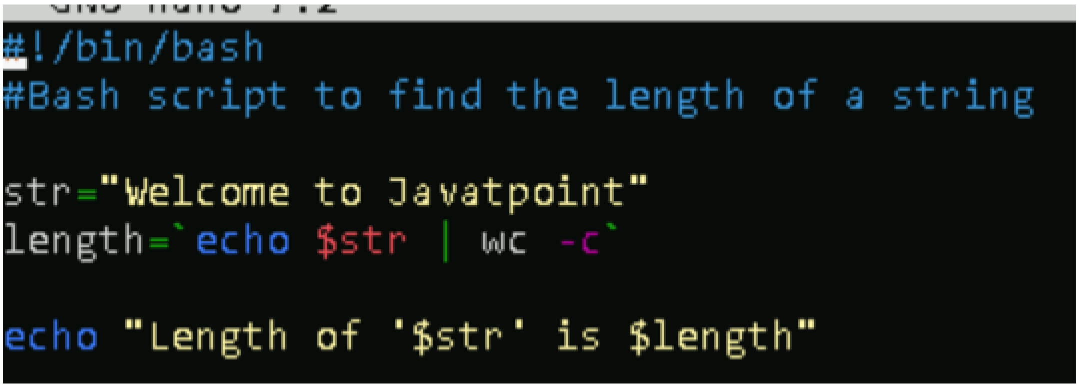
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./stringlength3.sh
Length of 'Welcome to Javatpoint' is 21
kirthik@9176994b7e94573:~/assignment/forloop$
```

CREATING FILE:



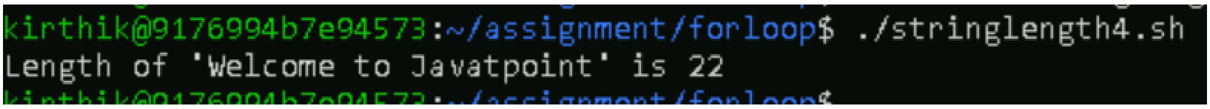
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano stringlength3.sh
kirthik@9176994b7e94573:~/assignment/forloop$ nano stringlength4.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x stringlength4.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./stringlength4.sh
```

SCRIPT:



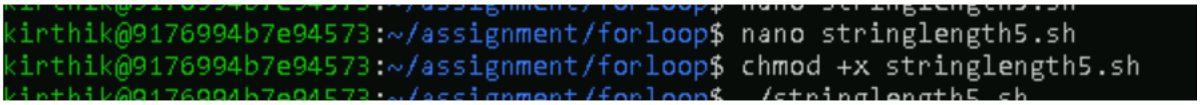
```
GNU nano 7.2
#!/bin/bash
#Bash script to find the length of a string

str="Welcome to Javatpoint"
length=`echo $str | wc -c`

echo "Length of '$str' is $length"
```

OUTPUT:



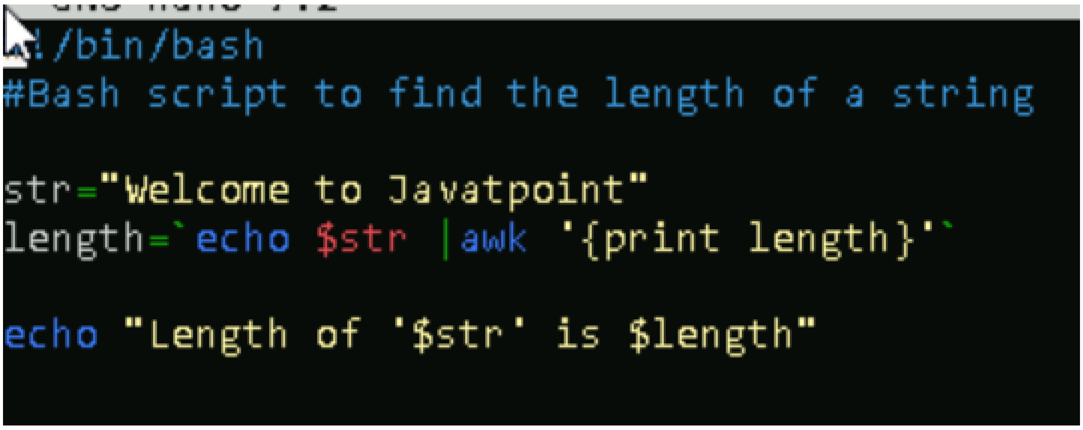
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./stringlength4.sh
Length of 'Welcome to Javatpoint' is 22
kirthik@9176994b7e94573:~/assignment/forloop$
```

CREATING FILE:



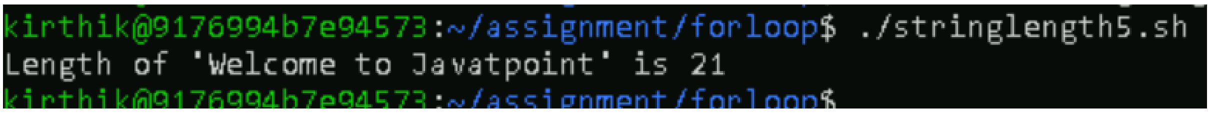
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano stringlength5.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x stringlength5.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./stringlength5.sh
```

SCRIPT:



```
#!/bin/bash
#Bash script to find the length of a string

str="Welcome to Javatpoint"
length=`echo $str |awk '{print length}'`

echo "Length of '$str' is $length"
```

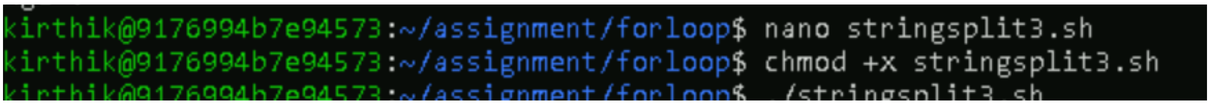
OUTPUT:



```
kirthik@9176994b7e94573:~/assignment/forloop$ ./stringlength5.sh
Length of 'Welcome to Javatpoint' is 21
kirthik@9176994b7e94573:~/assignment/forloop$
```

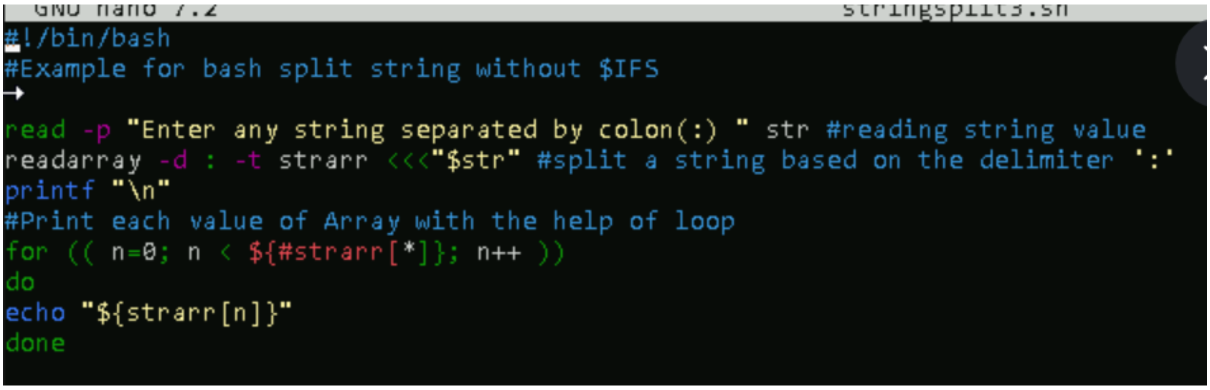
STRINGSPLIT:

CREATING FILE:



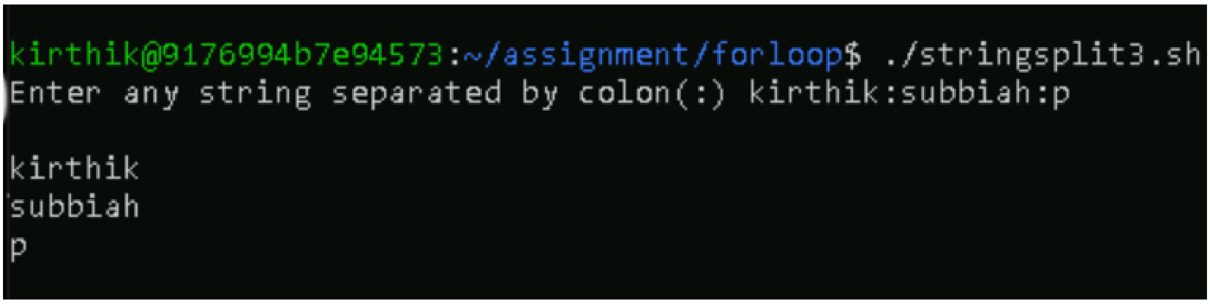
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano stringsplit3.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x stringsplit3.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./stringsplit3.sh
```

SCRIPT:



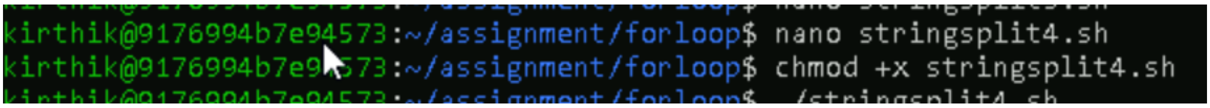
```
  GNU nano 7.2                                  stringsplit3.sh
#!/bin/bash
#Example for bash split string without $IFS

read -p "Enter any string separated by colon(:) " str #reading string value
readarray -d : -t strarr <<<"$str" #split a string based on the delimiter ':'
printf "\n"
#Print each value of Array with the help of loop
for (( n=0; n < ${#strarr[*]}; n++ ))
do
echo "${strarr[n]}"
done
```

OUTPUT:



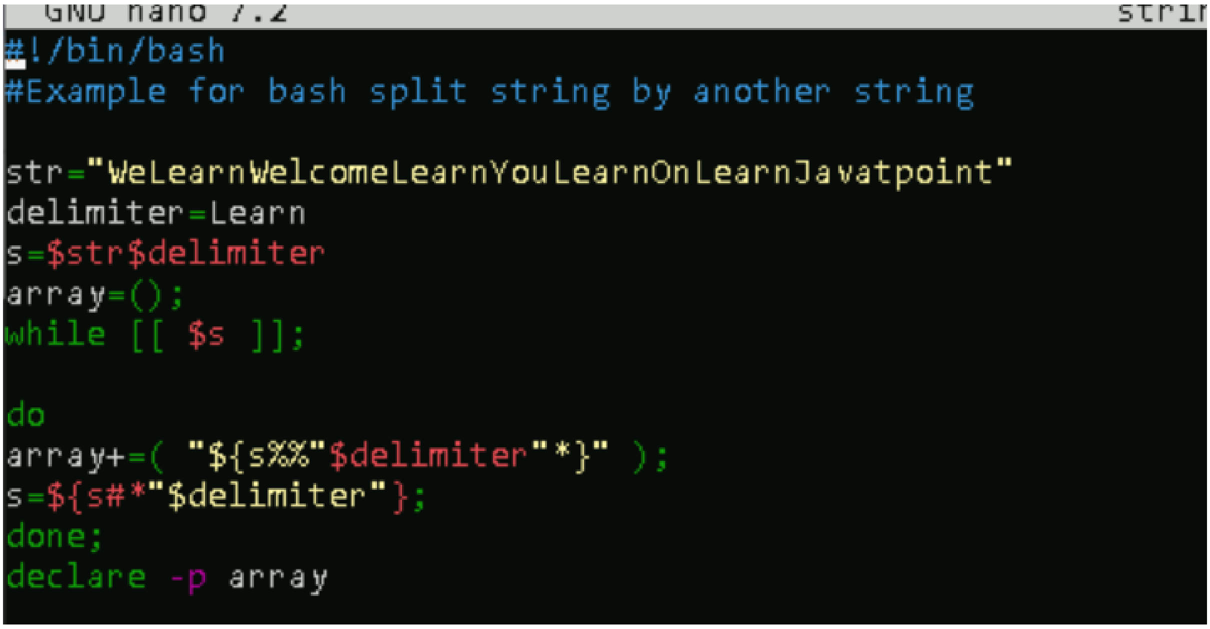
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./stringsplit3.sh
Enter any string separated by colon(:) kirthik:subbiah:p

kirthik
subbiah
p
```

CREATING FILE:



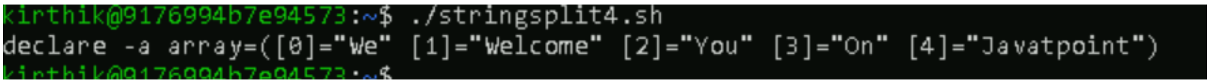
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano stringsplit3.sh
kirthik@9176994b7e94573:~/assignment/forloop$ nano stringsplit4.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x stringsplit4.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./stringsplit4.sh
```

SCRIPT:



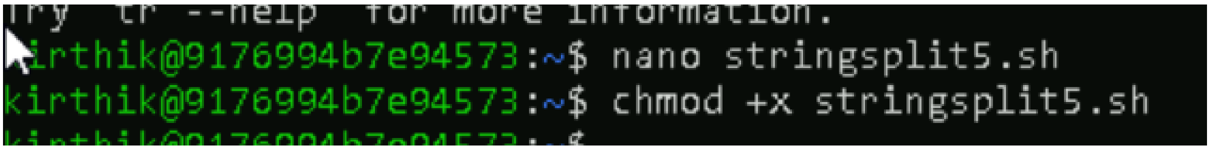
```
  GNU nano 7.2                                                    strin
#!/bin/bash
#Example for bash split string by another string

str="WeLearnWelcomeLearnYouLearnOnLearnJavatpoint"
delimiter=Learn
s=$str$delimiter
array=();
while [[ $s ]];

do
array+=( "${s%%"$delimiter"*}" );
s=${s#*"$delimiter"};
done;
declare -p array
```

OUTPUT:



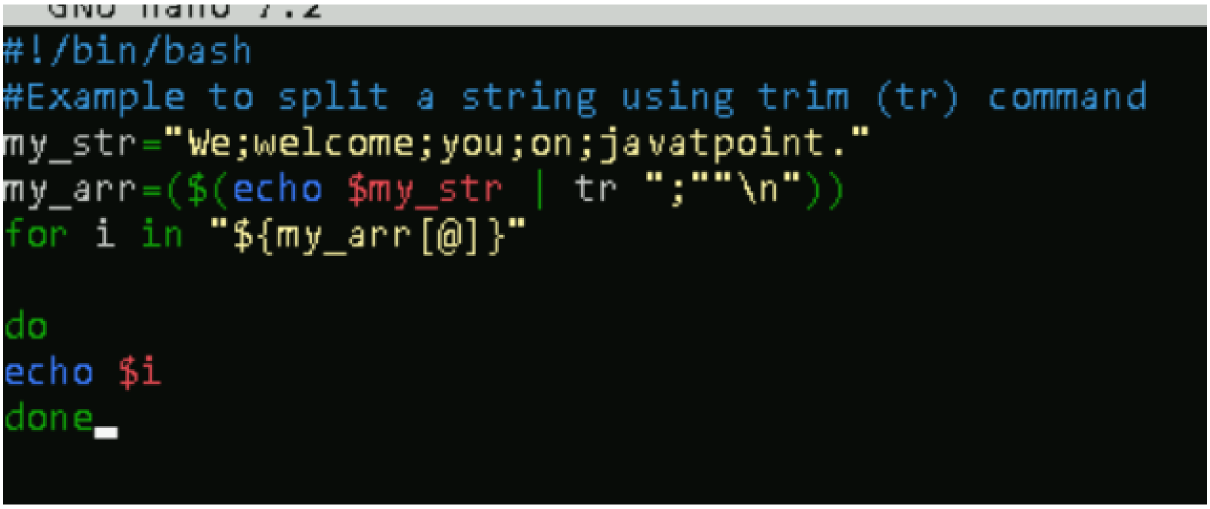
```
kirthik@9176994b7e94573:~$ ./stringsplit4.sh
declare -a array=([0]="We" [1]="Welcome" [2]="You" [3]="On" [4]="Javatpoint")
kirthik@9176994b7e94573:~$
```

CREATING FILE:



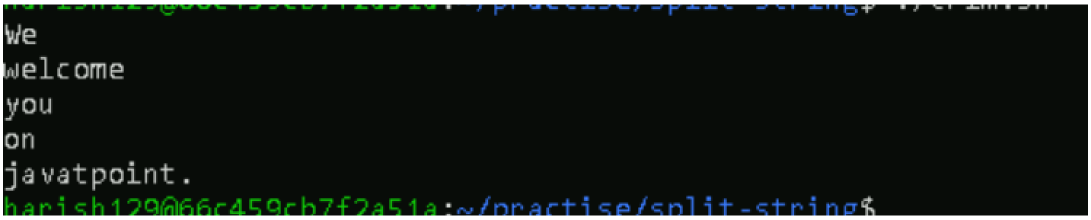
```
Try  tr --help  for more information.
kirthik@9176994b7e94573:~$ nano stringsplit5.sh
kirthik@9176994b7e94573:~$ chmod +x stringsplit5.sh
kirthik@9176994b7e94573:~$
```

SCRIPT:



```
  GNU nano 7.2
#!/bin/bash
#Example to split a string using trim (tr) command
my_str="We;welcome;you;on;javatpoint."
my_arr=($(echo $my_str | tr ";""\n"))
for i in "${my_arr[@]}"

do
echo $i
done
```

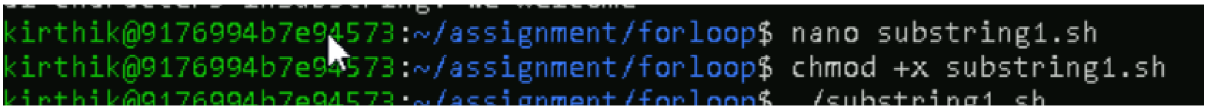
OUTPUT:



```
harish129@66c459cb7f2a51a:~/practise/split-string$ ./trim.sh
We
welcome
you
on
javatpoint.
harish129@66c459cb7f2a51a:~/practise/split-string$
```

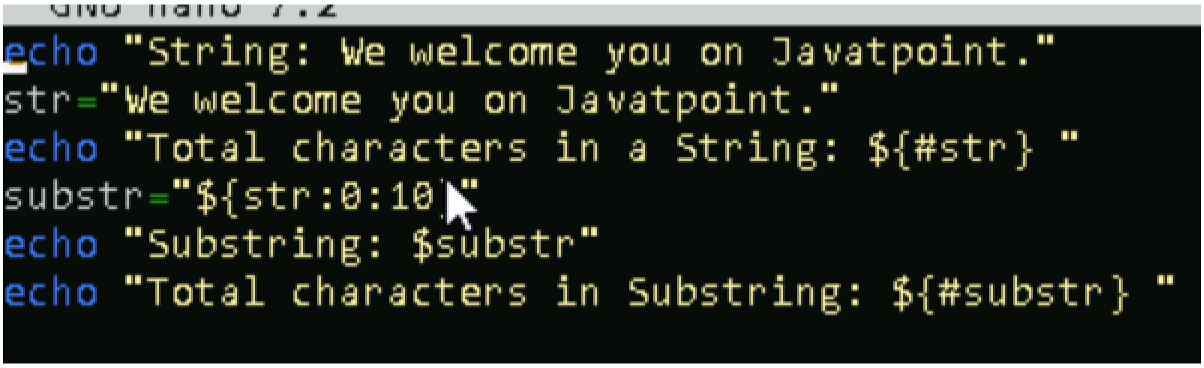
SUBSTRINGS:

CREATING FILE:



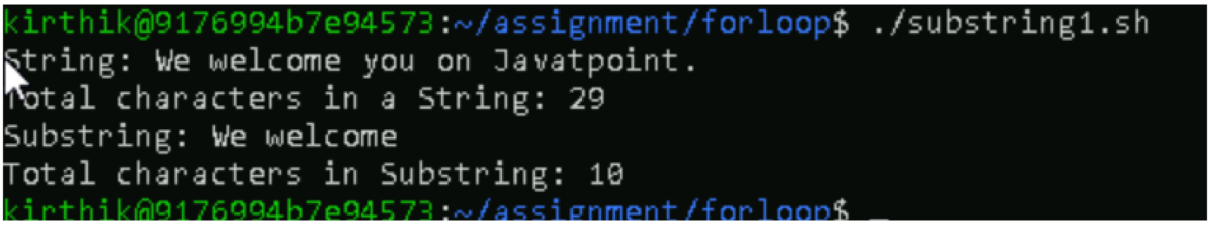
```
kirthik@9176994b7e94573:~/assignment/forloop$ nano substring1.sh
kirthik@9176994b7e94573:~/assignment/forloop$ chmod +x substring1.sh
kirthik@9176994b7e94573:~/assignment/forloop$ ./substring1.sh
```

SCRIPT:



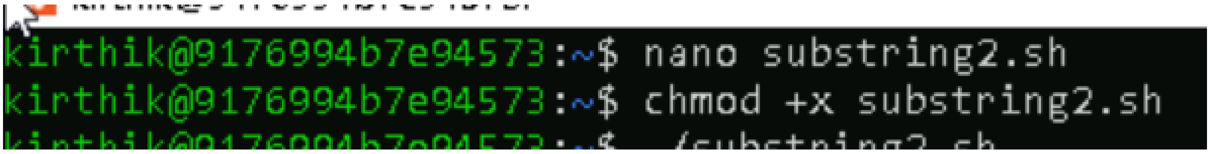
```
  GNU nano 7.2
echo "String: We welcome you on Javatpoint."
str="We welcome you on Javatpoint."
echo "Total characters in a String: ${#str} "
substr="${str:0:10}"
echo "Substring: $substr"
echo "Total characters in Substring: ${#substr} "
```

OUTPUT:



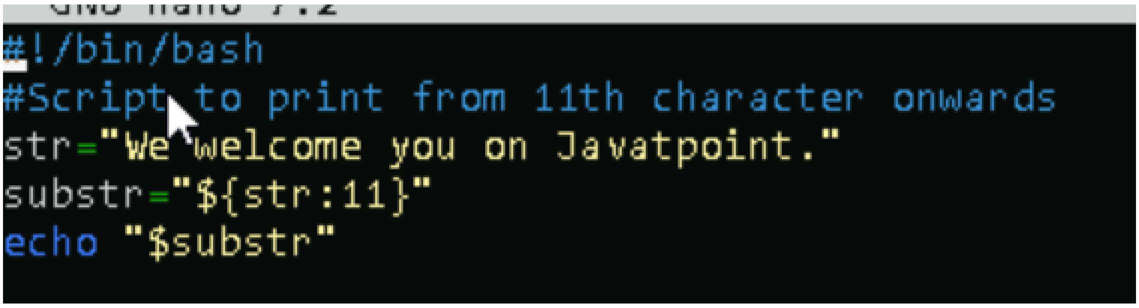
```
kirthik@9176994b7e94573:~/assignment/forloop$ ./substring1.sh
String: We welcome you on Javatpoint.
Total characters in a String: 29
Substring: We welcome
Total characters in Substring: 10
kirthik@9176994b7e94573:~/assignment/forloop$ _
```

CREATING FILE:



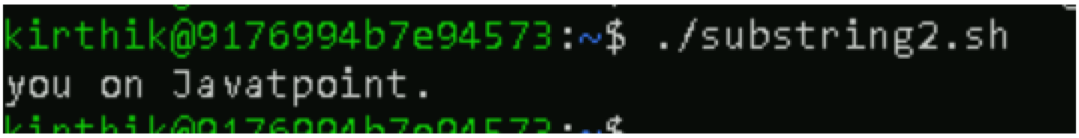
```
kirthik@9176994b7e94573:~$ nano substring2.sh
kirthik@9176994b7e94573:~$ chmod +x substring2.sh
kirthik@9176994b7e94573:~$ ./substring2.sh
```

SCRIPT:



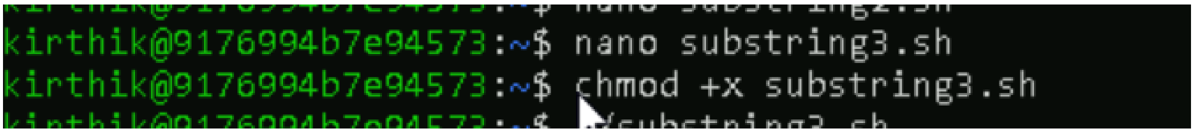
```
GNU nano 7.2
#!/bin/bash
#Script to print from 11th character onwards
str="We welcome you on Javatpoint."
substr="${str:11}"
echo "$substr"
```

OUTPUT:



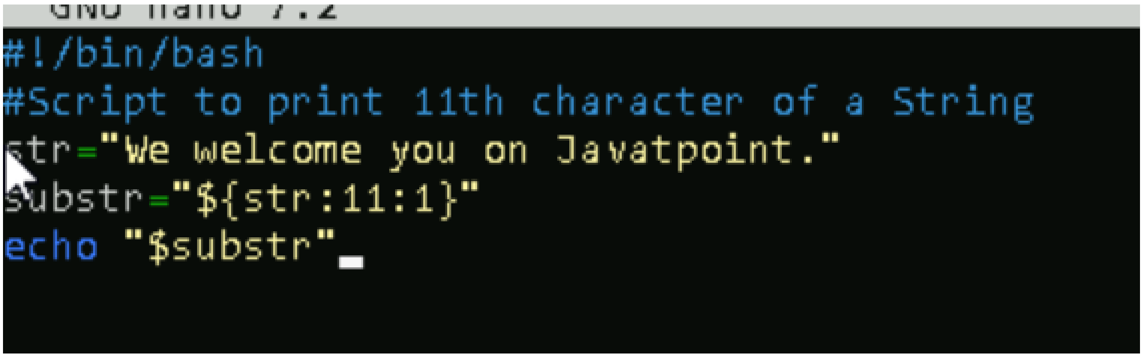
```
kirthik@9176994b7e94573:~$ ./substring2.sh
you on Javatpoint.
kirthik@9176994b7e94573:~$
```

CREATING FILE:



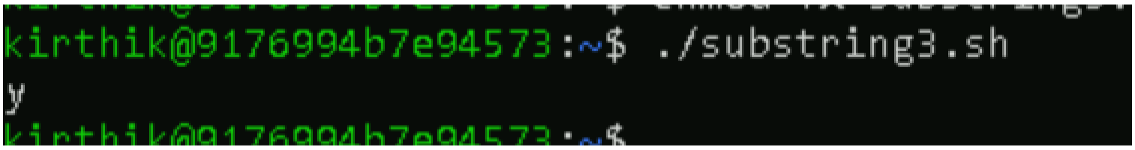
```
kirthik@9176994b7e94573:~$ nano substring2.sh
kirthik@9176994b7e94573:~$ nano substring3.sh
kirthik@9176994b7e94573:~$ chmod +x substring3.sh
kirthik@9176994b7e94573:~$ ./substring3.sh
```

SCRIPT:



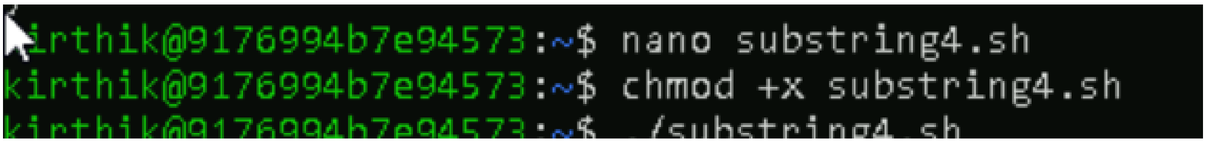
```
  GNU nano 7.2
#!/bin/bash
#Script to print 11th character of a String
str="We welcome you on Javatpoint."
substr="${str:11:1}"
echo "$substr"
```

OUTPUT:



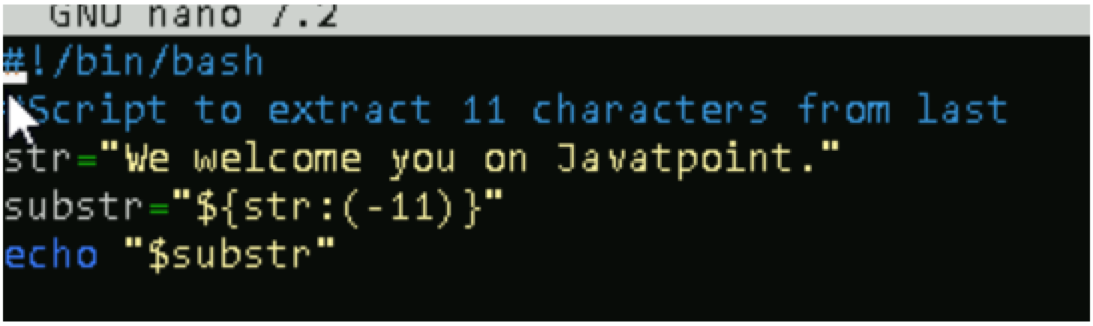
```
kirthik@9176994b7e94573:~$ chmod +x substring3.sh
kirthik@9176994b7e94573:~$ ./substring3.sh
y
kirthik@9176994b7e94573:~$
```

CREATING FILE:



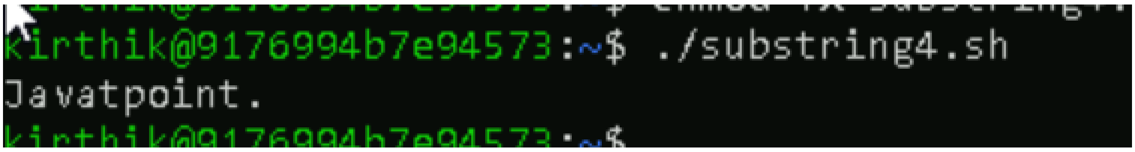
```
kirthik@9176994b7e94573:~$ nano substring4.sh
kirthik@9176994b7e94573:~$ chmod +x substring4.sh
kirthik@9176994b7e94573:~$ ./substring4.sh
```

SCRIPT:



```
  GNU nano 7.2
#!/bin/bash
#Script to extract 11 characters from last
str="We welcome you on Javatpoint."
substr="${str:(-11)}"
echo "$substr"
```

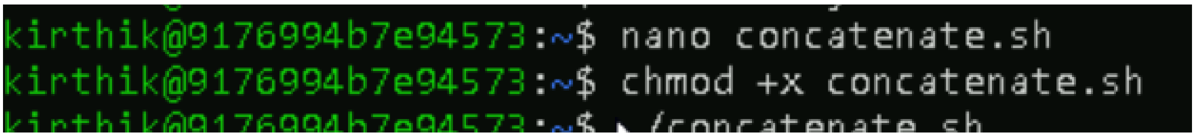
OUTPUT:



```
kirthik@9176994b7e94573:~$ chmod +x substring4.sh
kirthik@9176994b7e94573:~$ ./substring4.sh
Javatpoint.
kirthik@9176994b7e94573:~$
```

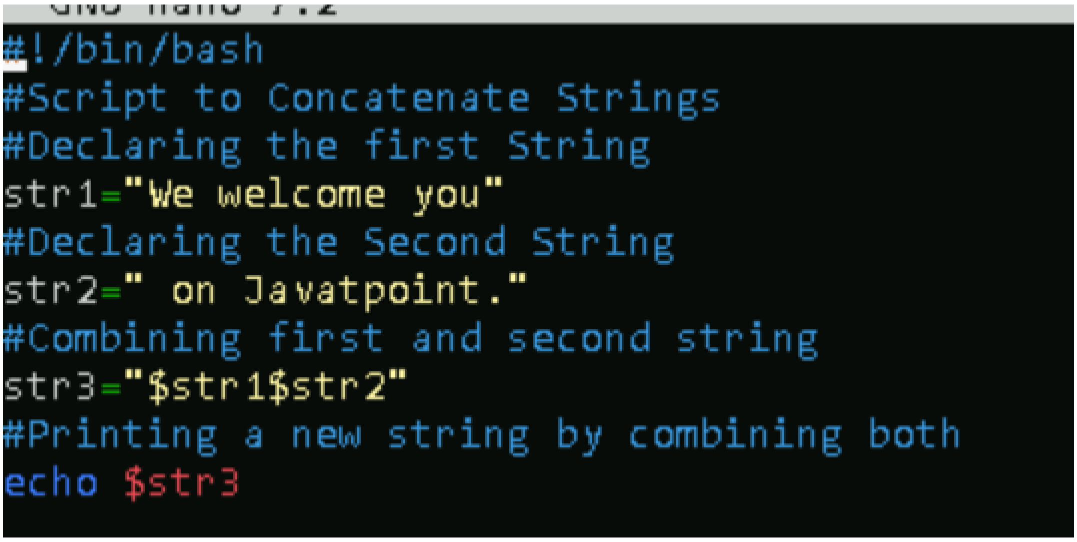
CONCATENATE

CREATING FILE:



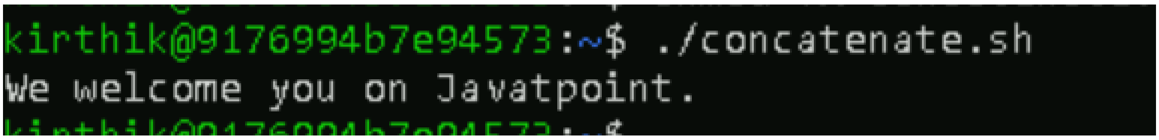
```
kirthik@9176994b7e94573:~$ nano concatenate.sh
kirthik@9176994b7e94573:~$ chmod +x concatenate.sh
kirthik@9176994b7e94573:~$ ./concatenate.sh
```

SCRIPT:



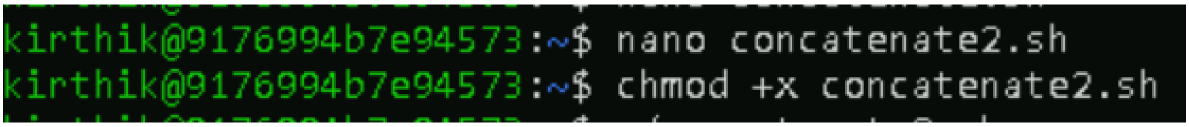
```
  GNU nano 7.2
#!/bin/bash
#Script to Concatenate Strings
#Declaring the first String
str1="We welcome you"
#Declaring the Second String
str2=" on Javatpoint."
#Combining first and second string
str3="$str1$str2"
#Printing a new string by combining both
echo $str3
```

OUTPUT:



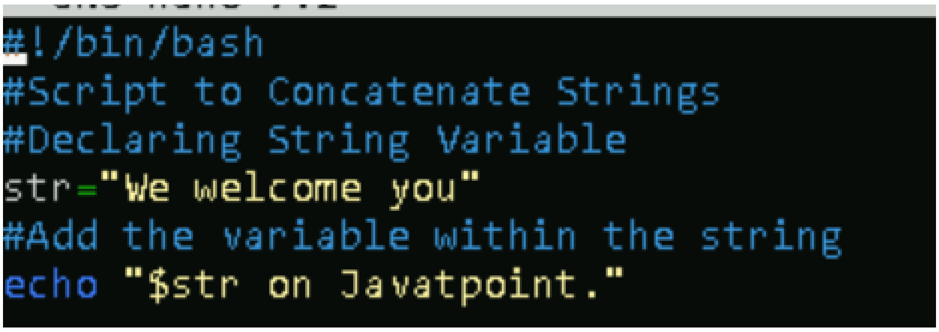
```
kirthik@9176994b7e94573:~$ ./concatenate.sh
We welcome you on Javatpoint.
kirthik@9176994b7e94573:~$
```

CREATING FILE:



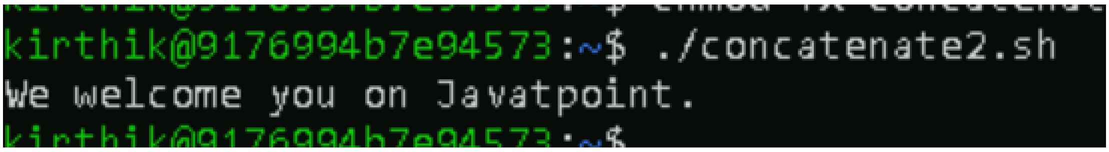
```
kirthik@9176994b7e94573:~$ nano concatenate2.sh
kirthik@9176994b7e94573:~$ chmod +x concatenate2.sh
```

SCRIPT:



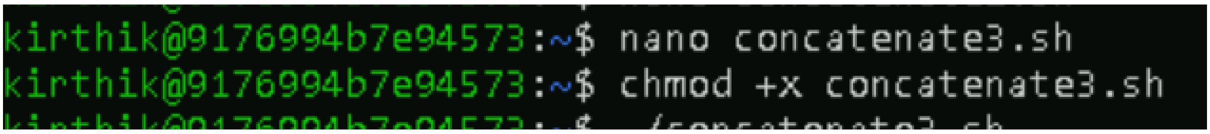
```
#!/bin/bash
#Script to Concatenate Strings
#Declaring String Variable
str="We welcome you"
#Add the variable within the string
echo "$str on Javatpoint."
```

OUTPUT:



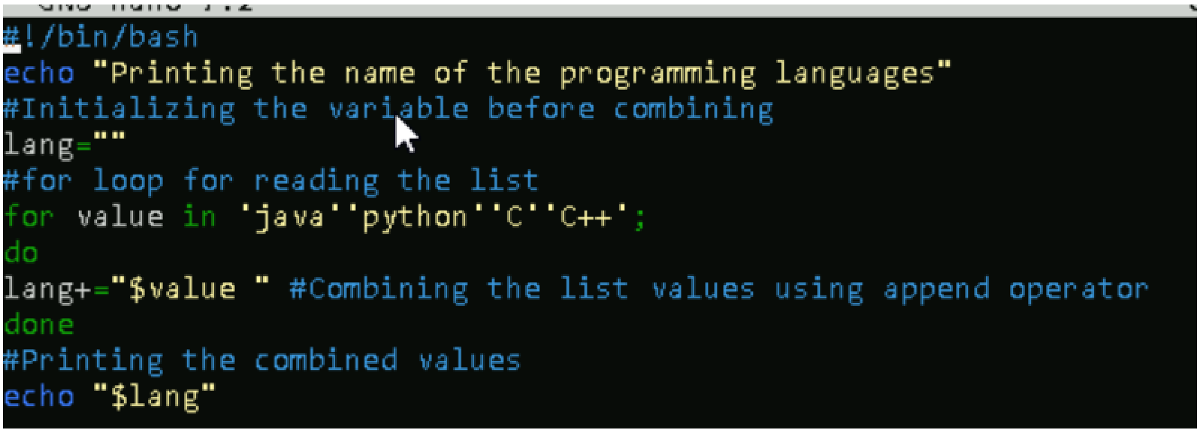
```
kirthik@9176994b7e94573:~$ ./concatenate2.sh
We welcome you on Javatpoint.
kirthik@9176994b7e94573:~$
```

CREATING FILE:



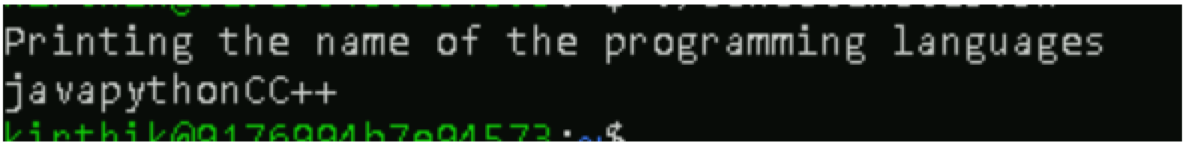
```
kirthik@9176994b7e94573:~$ nano concatenate3.sh
kirthik@9176994b7e94573:~$ chmod +x concatenate3.sh
kirthik@9176994b7e94573:~$ ./concatenate3.sh
```

SCRIPT:



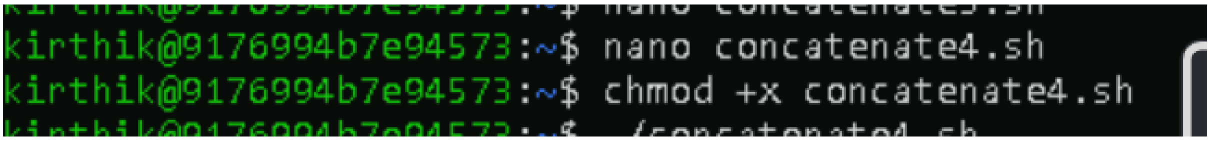
```
  GNU nano 7.2
#!/bin/bash
echo "Printing the name of the programming languages"
#Initializing the variable before combining
lang=""
#for loop for reading the list
for value in 'java''python''C''C++';
do
lang+="$value " #Combining the list values using append operator
done
#Printing the combined values
echo "$lang"
```

OUTPUT:



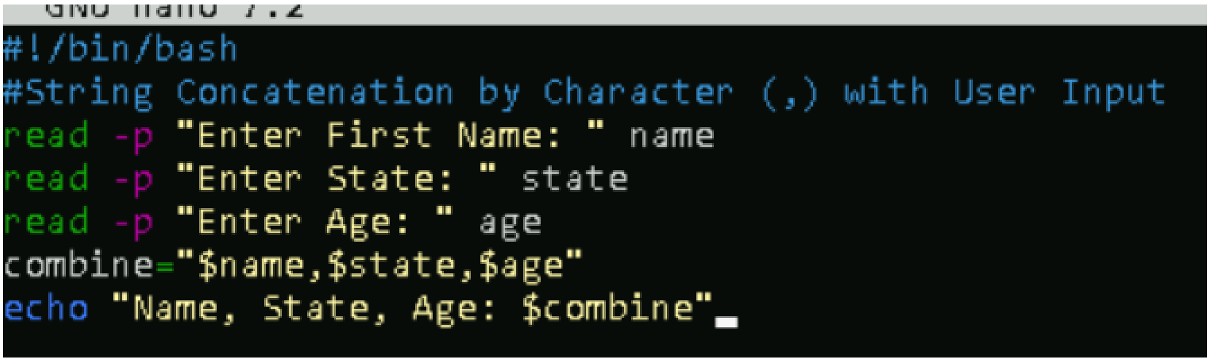
```
Printing the name of the programming languages
javapythonCC++
kirthik@9176994b7e94573:~$
```

CREATING FILE:



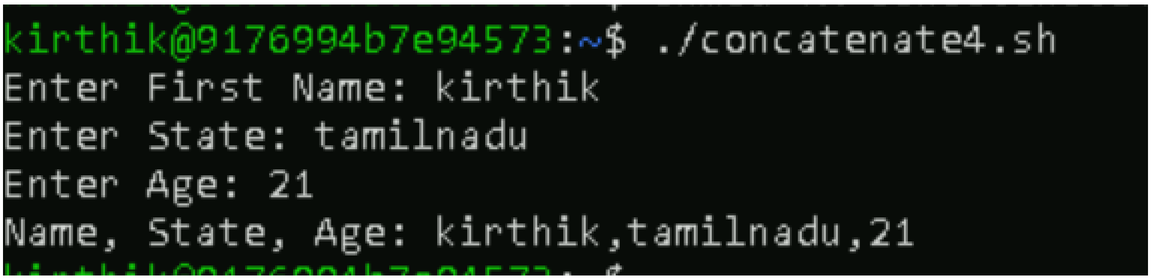
```
kirthik@9176994b7e94573:~$ nano concatenate3.sh
kirthik@9176994b7e94573:~$ nano concatenate4.sh
kirthik@9176994b7e94573:~$ chmod +x concatenate4.sh
kirthik@9176994b7e94573:~$ ./concatenate4.sh
```

SCRIPT:



```
  GNU nano 7.2
#!/bin/bash
#String Concatenation by Character (,) with User Input
read -p "Enter First Name: " name
read -p "Enter State: " state
read -p "Enter Age: " age
combine="$name,$state,$age"
echo "Name, State, Age: $combine"
```

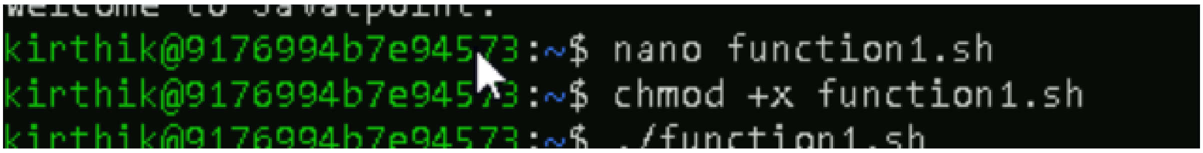
OUTPUT:



```
kirthik@9176994b7e94573:~$ ./concatenate4.sh
Enter First Name: kirthik
Enter State: tamilnadu
Enter Age: 21
Name, State, Age: kirthik,tamilnadu,21
kirthik@9176994b7e94573:~$
```

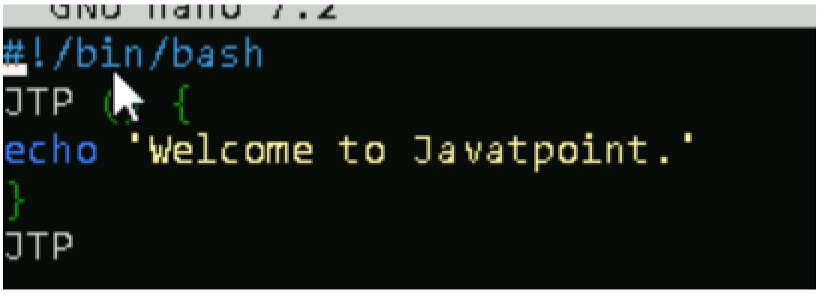
FUNCTIONS:

CREATING FILE:



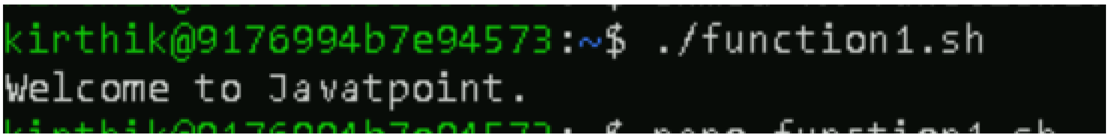
```
Welcome to Javatpoint.
kirthik@9176994b7e94573:~$ nano function1.sh
kirthik@9176994b7e94573:~$ chmod +x function1.sh
kirthik@9176994b7e94573:~$ ./function1.sh
```

SCRIPT:



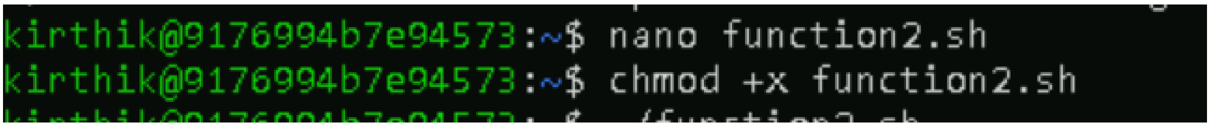
```
  GNU nano 7.2
#!/bin/bash
JTP () {
echo 'Welcome to Javatpoint.'
}
JTP
```

OUTPUT:



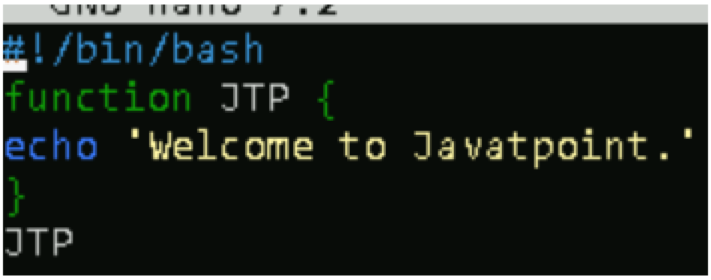
```
kirthik@9176994b7e94573:~$ ./function1.sh
Welcome to Javatpoint.
kirthik@9176994b7e94573:~$ nano function1.sh
```

CREATING FILE:



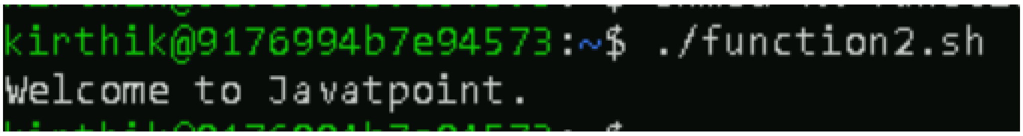
```
kirthik@9176994b7e94573:~$ nano function2.sh
kirthik@9176994b7e94573:~$ chmod +x function2.sh
kirthik@9176994b7e94573:~$ ./function2.sh
```

SCRIPT:



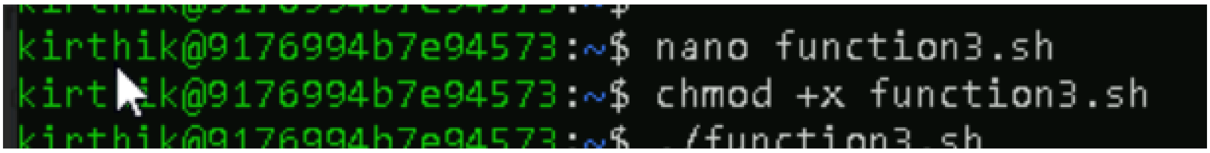
```
GNU nano 7.2
#!/bin/bash
function JTP {
echo 'Welcome to Javatpoint.'
}
JTP
```

OUTPUT:



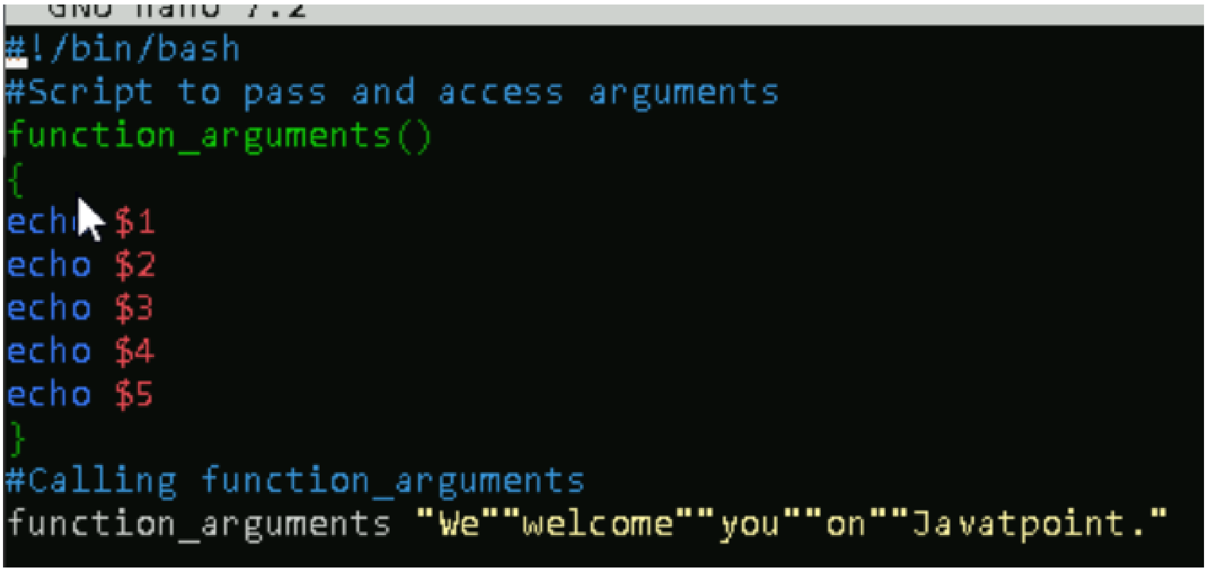
```
kirthik@9176994b7e94573:~$ chmod +x functio
kirthik@9176994b7e94573:~$ ./function2.sh
Welcome to Javatpoint.
kirthik@9176994b7e94573:~$
```

CREATING FILE:



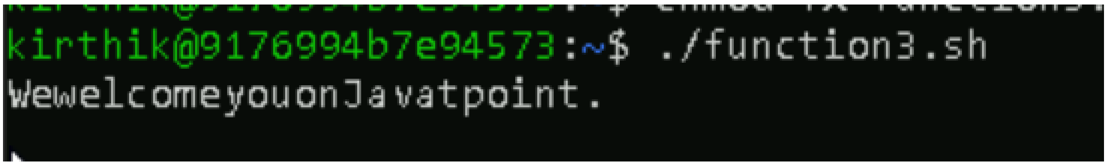
```
kirthik@9176994b7e94573:~$ nano function3.sh
kirthik@9176994b7e94573:~$ chmod +x function3.sh
kirthik@9176994b7e94573:~$ ./function3.sh
```

SCRIPT:



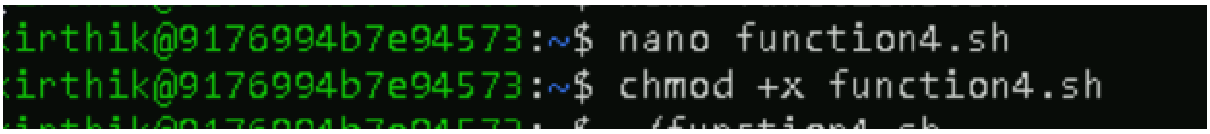
```
  GNU nano 7.2
#!/bin/bash
#Script to pass and access arguments
function_arguments()
{
echo $1
echo $2
echo $3
echo $4
echo $5
}
#Calling function_arguments
function_arguments "We""welcome""you""on""Javatpoint."
```

OUTPUT:



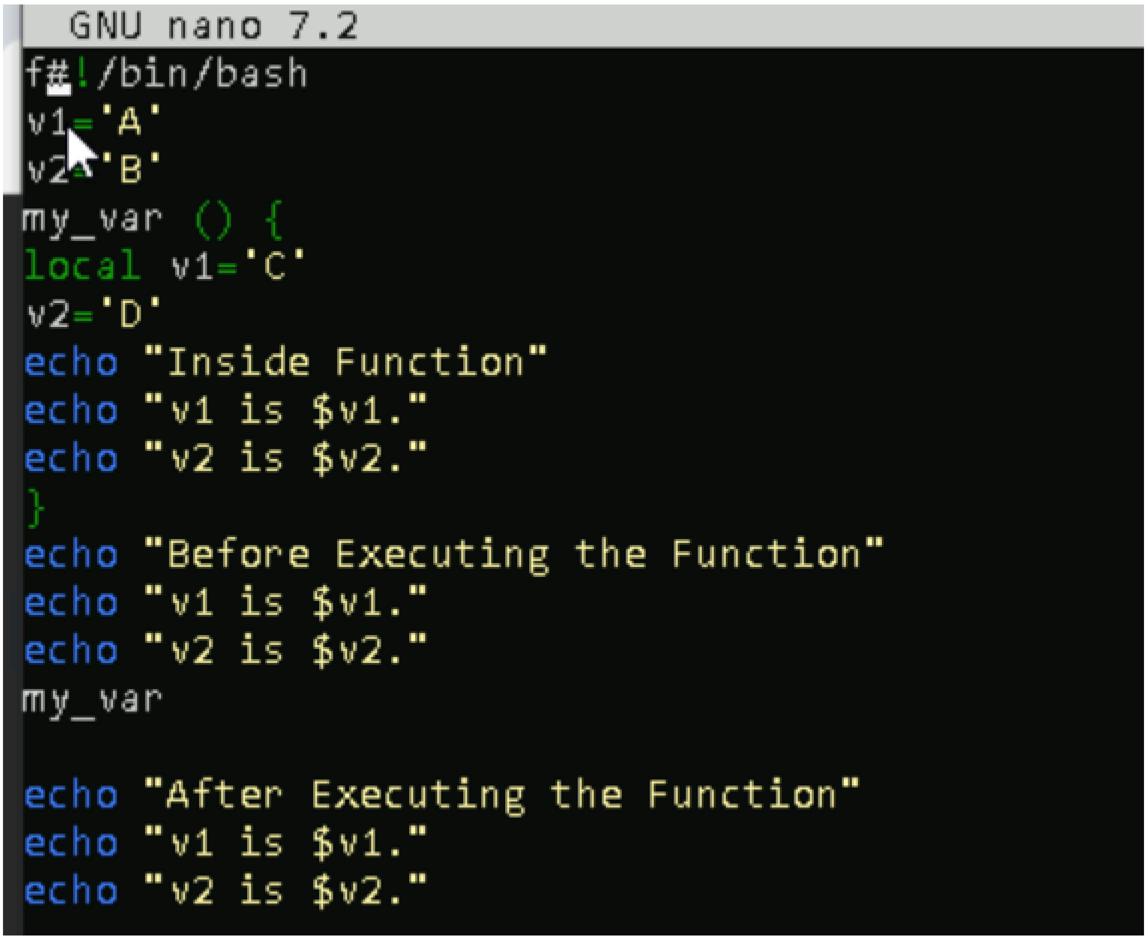
```
kirthik@9176994b7e94573:~$ ./function3.sh
WewelcomeyouonJavatpoint.
```

CREATING FILE:



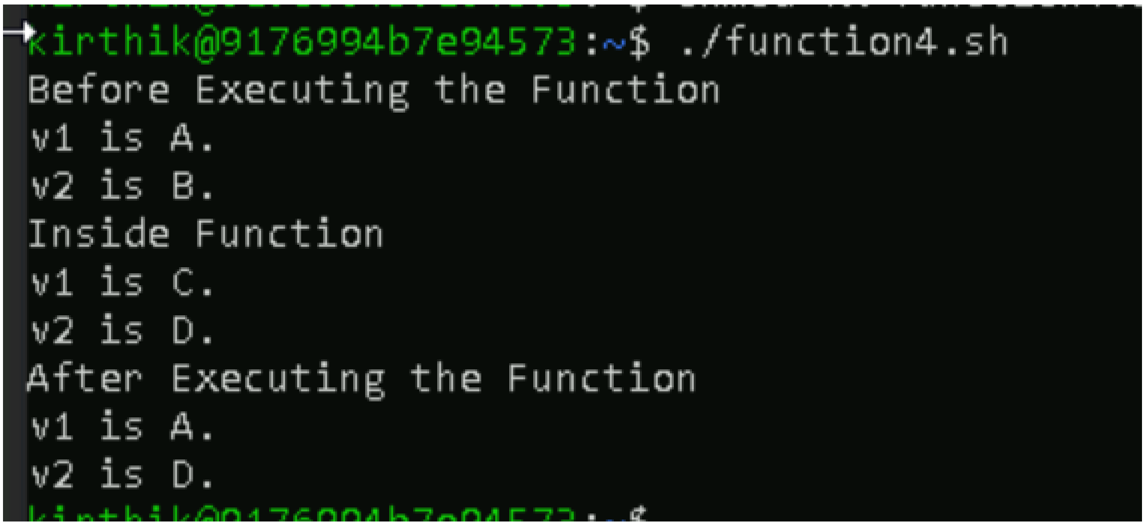
```
irthik@9176994b7e94573:~$ nano function4.sh
irthik@9176994b7e94573:~$ chmod +x function4.sh
```

SCRIPT:



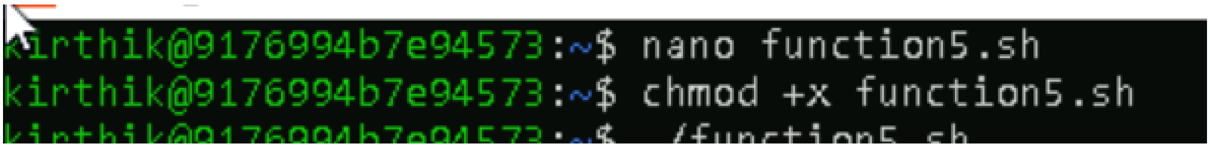
```
f#!/bin/bash
v1='A'
v2='B'
my_var () {
local v1='C'
v2='D'
echo "Inside Function"
echo "v1 is $v1."
echo "v2 is $v2."
}
echo "Before Executing the Function"
echo "v1 is $v1."
echo "v2 is $v2."
my_var

echo "After Executing the Function"
echo "v1 is $v1."
echo "v2 is $v2."
```

OUTPUT:



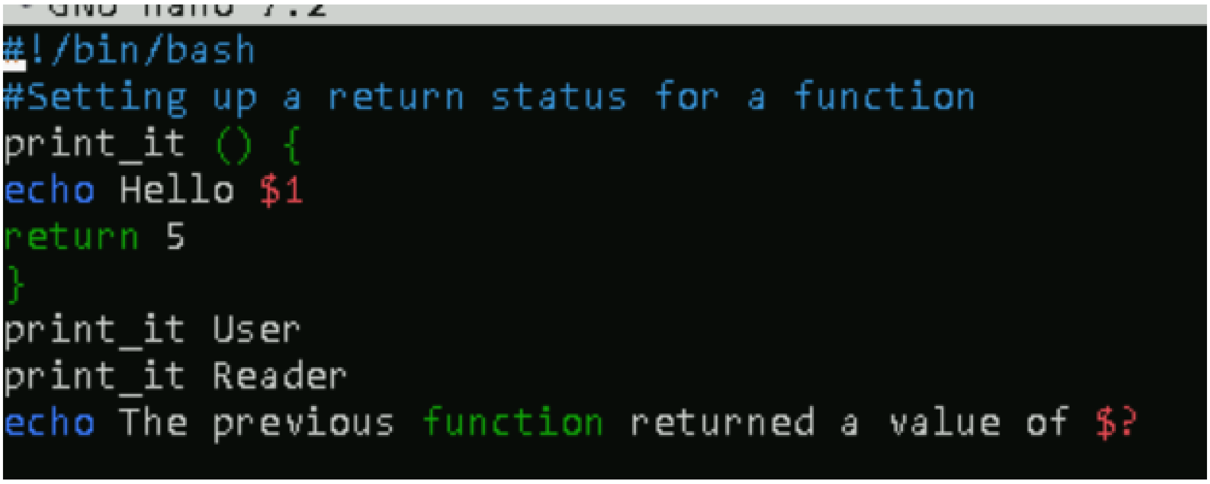
```
kirthik@9176994b7e94573:~$ ./function4.sh
Before Executing the Function
v1 is A.
v2 is B.
Inside Function
v1 is C.
v2 is D.
After Executing the Function
v1 is A.
v2 is D.
kirthik@9176994b7e94573:~$
```

CREATING FILE:



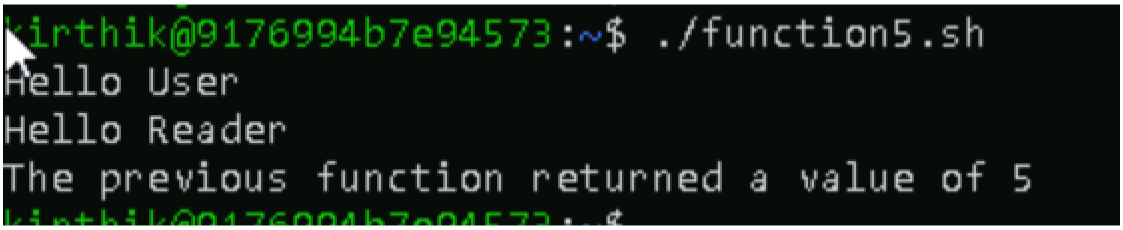
```
kirthik@9176994b7e94573:~$ nano function5.sh
kirthik@9176994b7e94573:~$ chmod +x function5.sh
kirthik@9176994b7e94573:~$ ./function5.sh
```

SCRIPT:



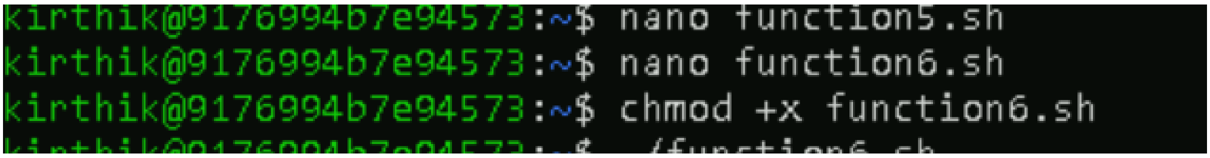
```
GNU nano 7.2
#!/bin/bash
#Setting up a return status for a function
print_it () {
echo Hello $1
return 5
}
print_it User
print_it Reader
echo The previous function returned a value of $?
```

OUTPUT:



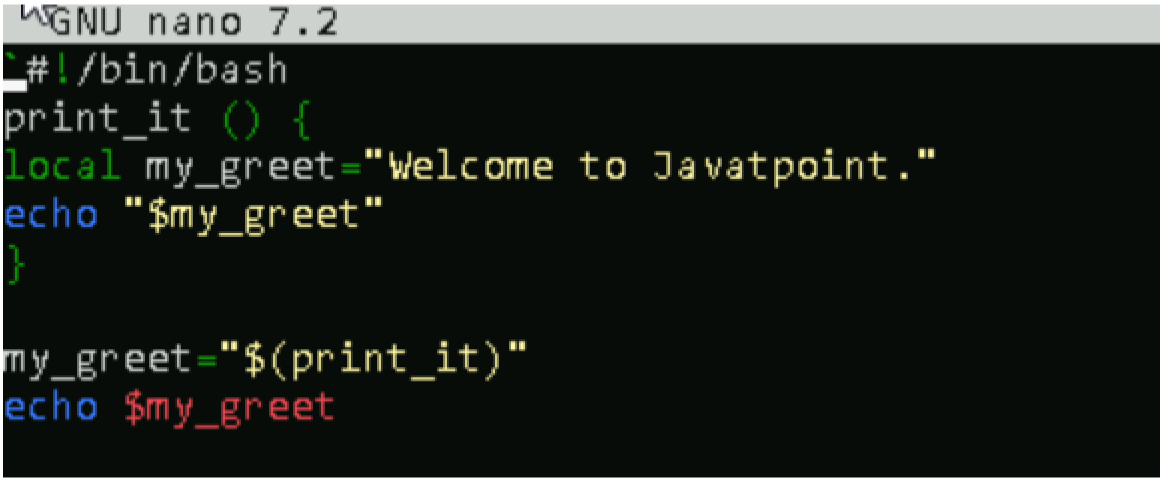
```
kirthik@9176994b7e94573:~$ ./function5.sh
Hello User
Hello Reader
The previous function returned a value of 5
kirthik@9176994b7e94573:~$
```

CREATING FILE:



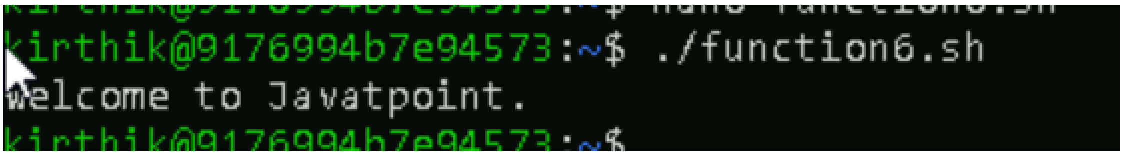
```
kirthik@9176994b7e94573:~$ nano function5.sh
kirthik@9176994b7e94573:~$ nano function6.sh
kirthik@9176994b7e94573:~$ chmod +x function6.sh
kirthik@9176994b7e94573:~$ ./function6.sh
```

SCRIPT:



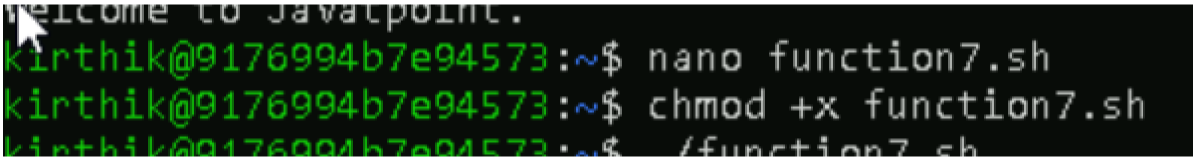
```
GNU nano 7.2
#!/bin/bash
print_it () {
local my_greet="Welcome to Javatpoint."
echo "$my_greet"
}

my_greet="$(print_it)"
echo $my_greet
```

OUTPUT:



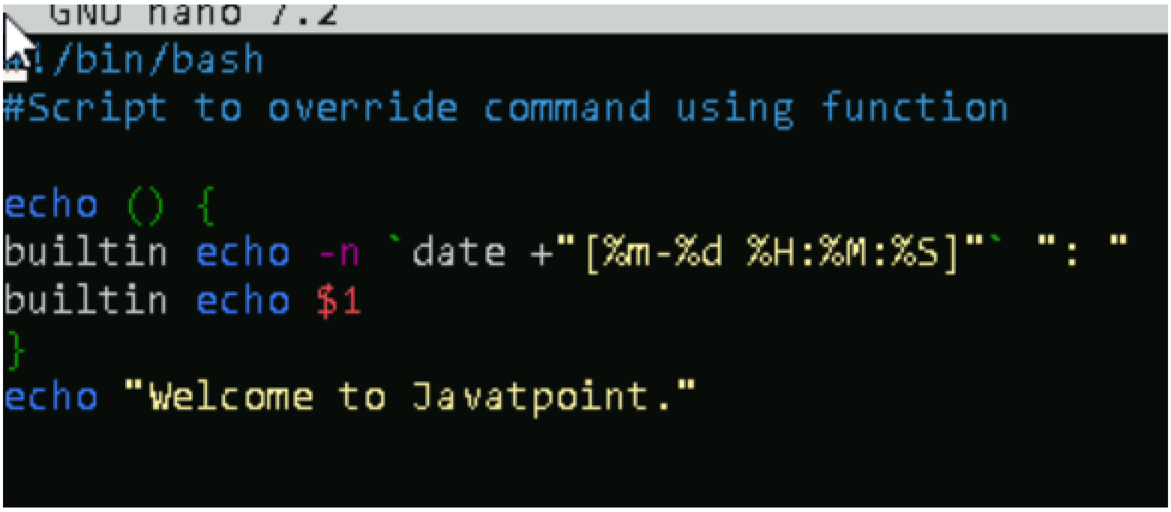
```
kirthik@9176994b7e94573:~$ nano function6.sh
kirthik@9176994b7e94573:~$ ./function6.sh
Welcome to Javatpoint.
kirthik@9176994b7e94573:~$
```

CREATING FILE:



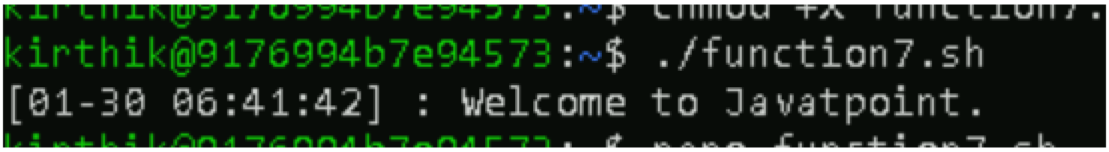
```
Welcome to Javatpoint.
kirthik@9176994b7e94573:~$ nano function7.sh
kirthik@9176994b7e94573:~$ chmod +x function7.sh
kirthik@9176994b7e94573:~$ ./function7.sh
```

SCRIPT:



```
  GNU nano 7.2
#!/bin/bash
#Script to override command using function

echo () {
builtin echo -n `date +"[%m-%d %H:%M:%S]"` ": "
builtin echo $1
}
echo "Welcome to Javatpoint."
```

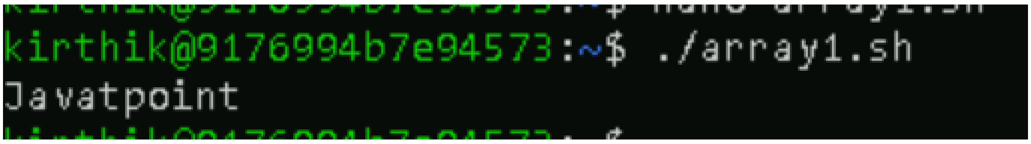
OUTPUT:



```
kirthik@9176994b7e94573:~$ chmod +x function7.
kirthik@9176994b7e94573:~$ ./function7.sh
[01-30 06:41:42] : Welcome to Javatpoint.
kirthik@9176994b7e94573:~$ nano function7.sh
```

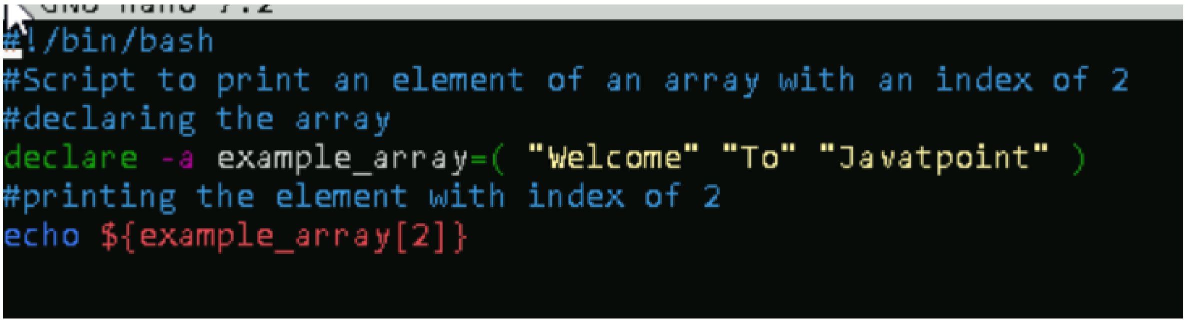
ARRAY:

CREATING FILE:



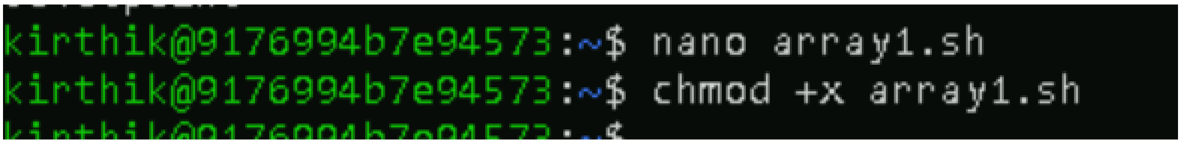
```
kirthik@9176994b7e94573:~$ nano array1.sh
kirthik@9176994b7e94573:~$ ./array1.sh
Javatpoint
kirthik@9176994b7e94573:~$
```

SCRIPT:



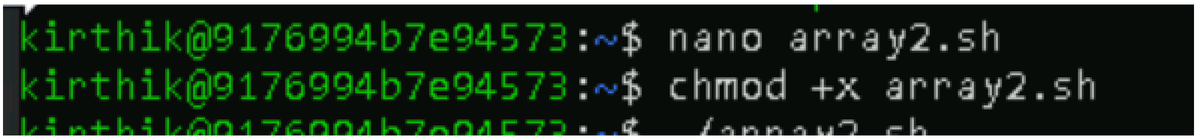
```
GNU nano 7.2
#!/bin/bash
#Script to print an element of an array with an index of 2
#declaring the array
declare -a example_array=( "Welcome" "To" "Javatpoint" )
#printing the element with index of 2
echo ${example_array[2]}
```

OUTPUT:



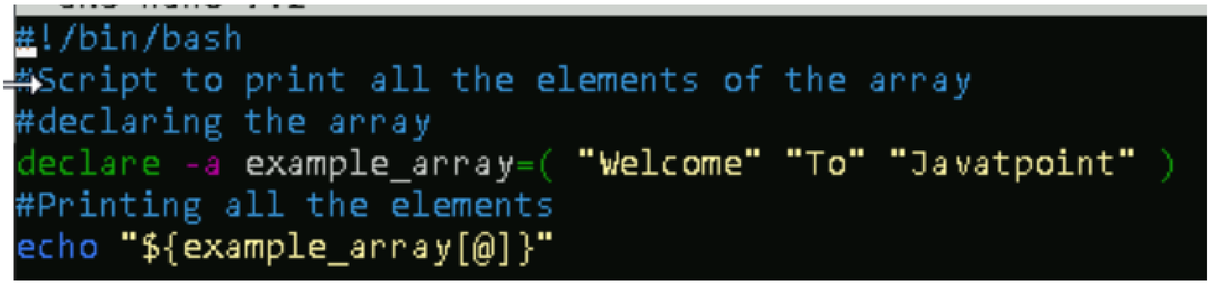
```
Javatpoint
kirthik@9176994b7e94573:~$ nano array1.sh
kirthik@9176994b7e94573:~$ chmod +x array1.sh
kirthik@9176994b7e94573:~$
```

CREATING FILE:



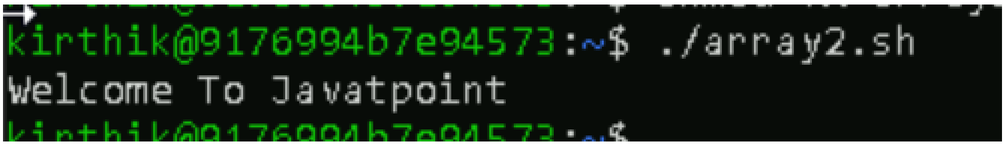
```
kirthik@9176994b7e94573:~$ nano array2.sh
kirthik@9176994b7e94573:~$ chmod +x array2.sh
kirthik@9176994b7e94573:~$ ./array2.sh
```

SCRIPT:



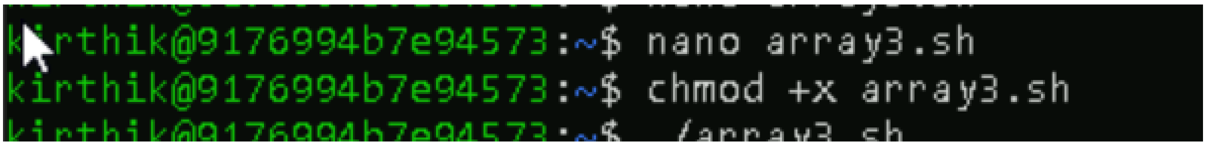
```
#!/bin/bash
#Script to print all the elements of the array
#declaring the array
declare -a example_array=( "Welcome" "To" "Javatpoint" )
#Printing all the elements
echo "${example_array[@]}"
```

OUTPUT:



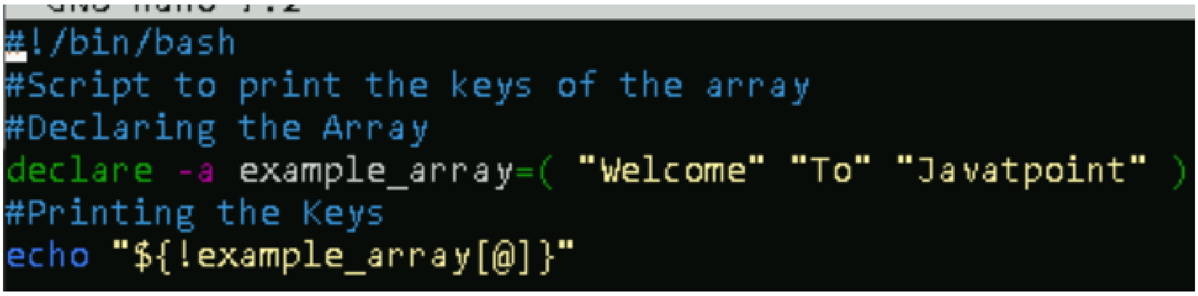
```
kirthik@9176994b7e94573:~$ chmod +x array2.sh
kirthik@9176994b7e94573:~$ ./array2.sh
Welcome To Javatpoint
kirthik@9176994b7e94573:~$
```

CREATING FILE:



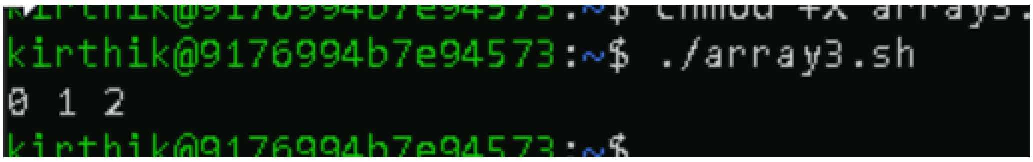
```
kirthik@9176994b7e94573:~$ nano array3.sh
kirthik@9176994b7e94573:~$ chmod +x array3.sh
kirthik@9176994b7e94573:~$ ./array3.sh
```

SCRIPT:



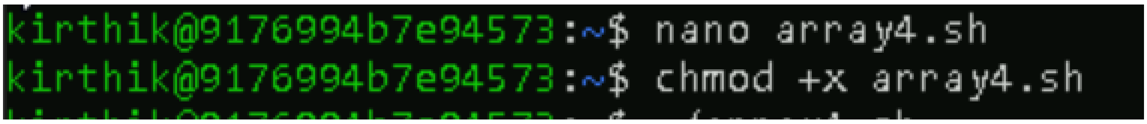
```
  GNU nano 7.2
#!/bin/bash
#Script to print the keys of the array
#Declaring the Array
declare -a example_array=( "Welcome" "To" "Javatpoint" )
#Printing the Keys
echo "${!example_array[@]}"
```

OUTPUT:



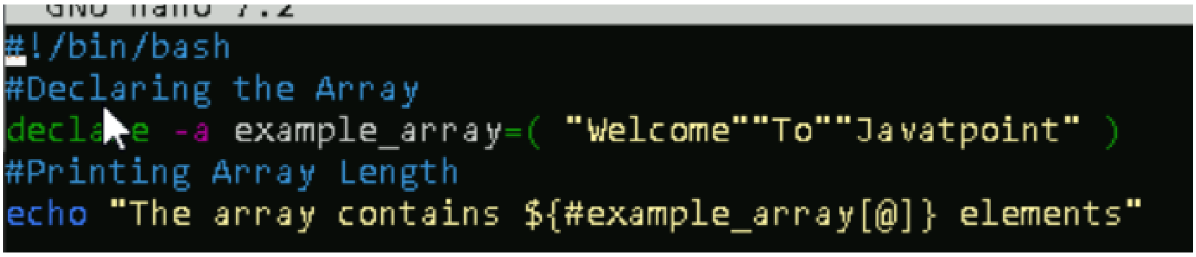
```
kirthik@9176994b7e94573:~$ chmod +x array3.
kirthik@9176994b7e94573:~$ ./array3.sh
0 1 2
kirthik@9176994b7e94573:~$
```

CREATING FILE:



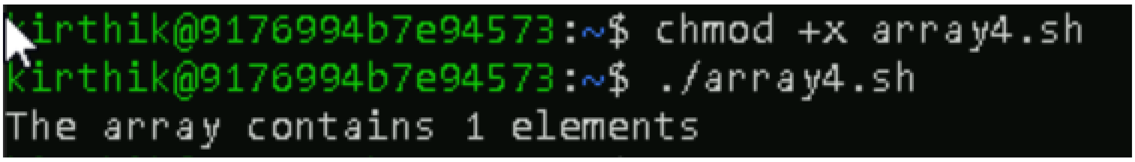
```
kirthik@9176994b7e94573:~$ nano array4.sh
kirthik@9176994b7e94573:~$ chmod +x array4.sh
```

SCRIPT:



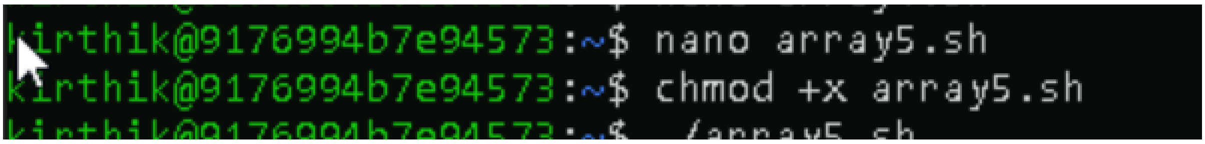
```
  GNU nano 7.2
#!/bin/bash
#Declaring the Array
declare -a example_array=( "Welcome""To""Javatpoint" )
#Printing Array Length
echo "The array contains ${#example_array[@]} elements"
```

OUTPUT:



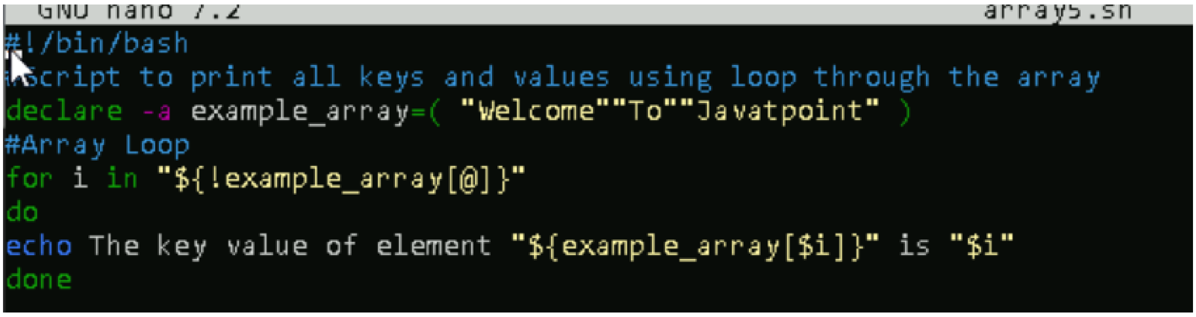
```
kirthik@9176994b7e94573:~$ chmod +x array4.sh
kirthik@9176994b7e94573:~$ ./array4.sh
The array contains 1 elements
```

CREATING FILE:



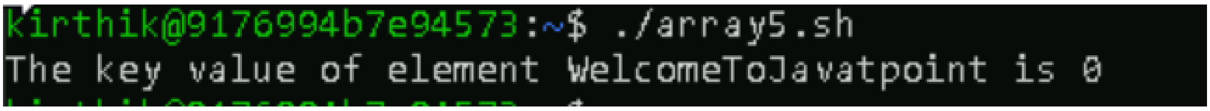
```
kirthik@9176994b7e94573:~$ nano array5.sh
kirthik@9176994b7e94573:~$ chmod +x array5.sh
kirthik@9176994b7e94573:~$ ./array5.sh
```

SCRIPT:



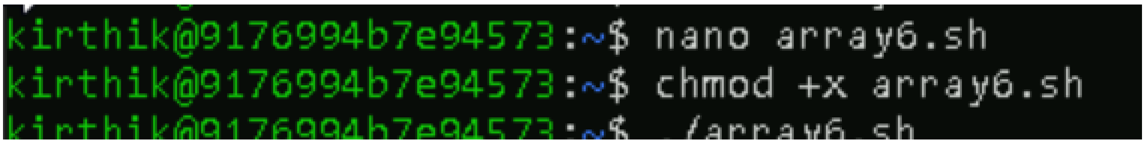
```
  GNU nano 7.2                                                    array5.sh
#!/bin/bash
#Script to print all keys and values using loop through the array
declare -a example_array=( "Welcome""To""Javatpoint" )
#Array Loop
for i in "${!example_array[@]}"
do
echo The key value of element "${example_array[$i]}" is "$i"
done
```

OUTPUT:



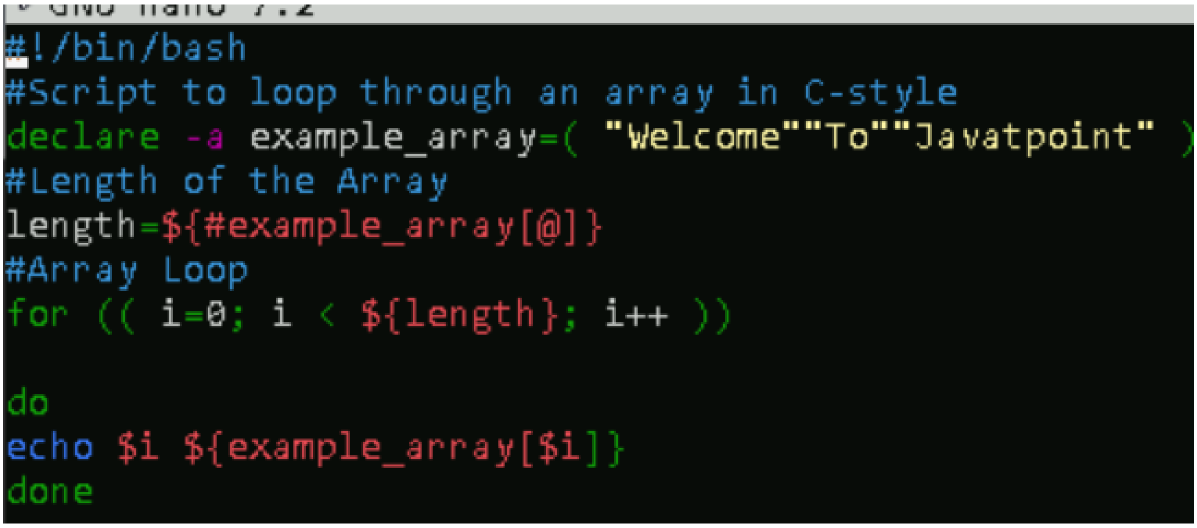
```
kirthik@9176994b7e94573:~$ ./array5.sh
The key value of element WelcomeToJavatpoint is 0
kirthik@9176994b7e94573:~$
```

CREATING FILE:



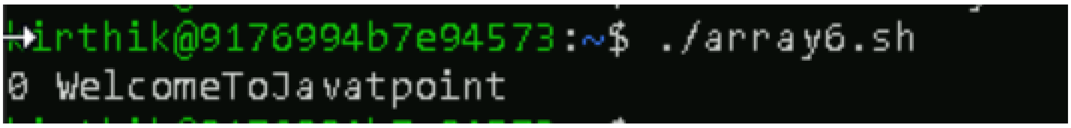
```
kirthik@9176994b7e94573:~$ nano array6.sh
kirthik@9176994b7e94573:~$ chmod +x array6.sh
kirthik@9176994b7e94573:~$ ./array6.sh
```

SCRIPT:



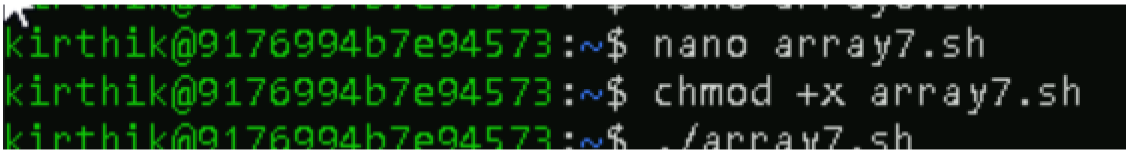
```
GNU nano 7.2
#!/bin/bash
#Script to loop through an array in C-style
declare -a example_array=( "Welcome""To""Javatpoint" )
#Length of the Array
length=${#example_array[@]}
#Array Loop
for (( i=0; i < ${length}; i++ ))

do
echo $i ${example_array[$i]}
done
```

OUTPUT:



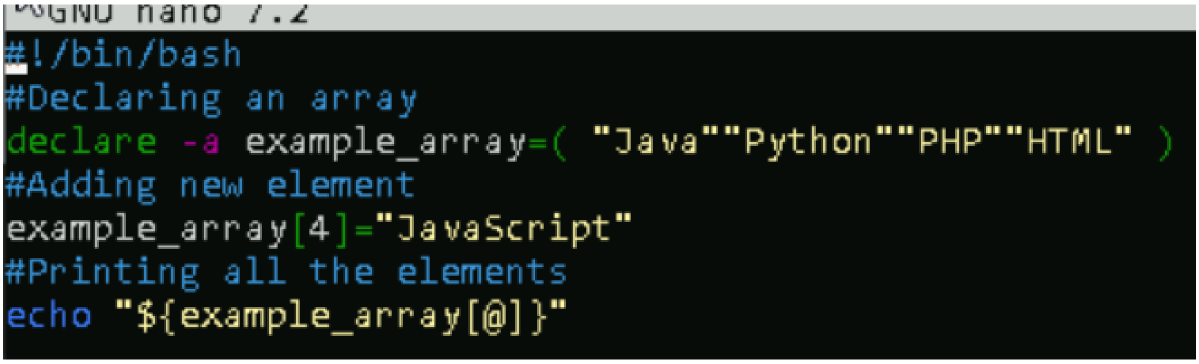
```
kirthik@9176994b7e94573:~$ ./array6.sh
0 WelcomeToJavatpoint
```

CREATING FILE:



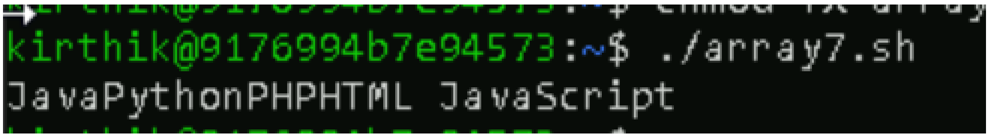
```
kirthik@9176994b7e94573:~$ nano array7.sh
kirthik@9176994b7e94573:~$ chmod +x array7.sh
kirthik@9176994b7e94573:~$ ./array7.sh
```

SCRIPT:



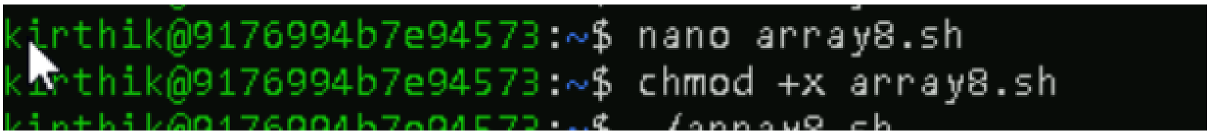
```
GNU nano 7.2
#!/bin/bash
#Declaring an array
declare -a example_array=( "Java""Python""PHP""HTML" )
#Adding new element
example_array[4]="JavaScript"
#Printing all the elements
echo "${example_array[@]}"
```

OUTPUT:



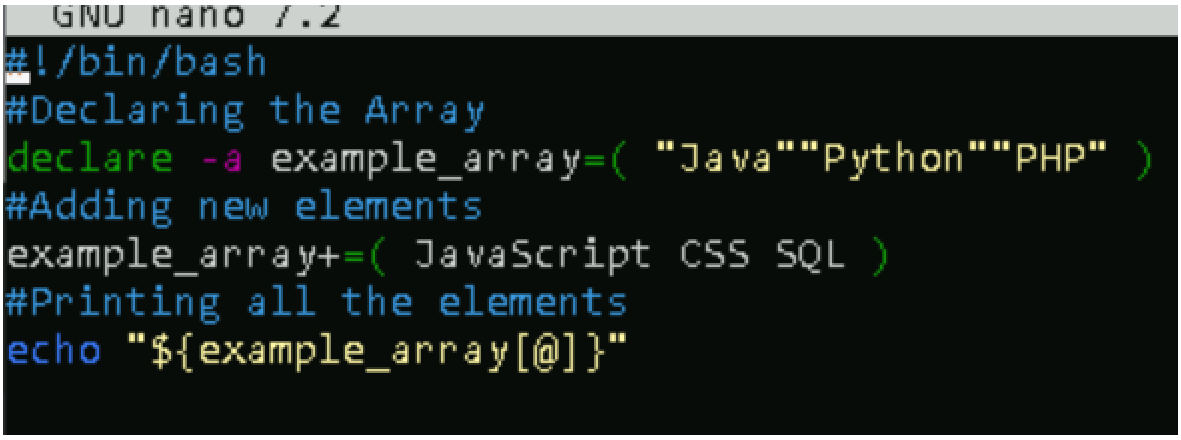
```
kirthik@9176994b7e94573:~$ ./array7.sh
JavaPythonPHPHTML JavaScript
```

CREATING FILE:



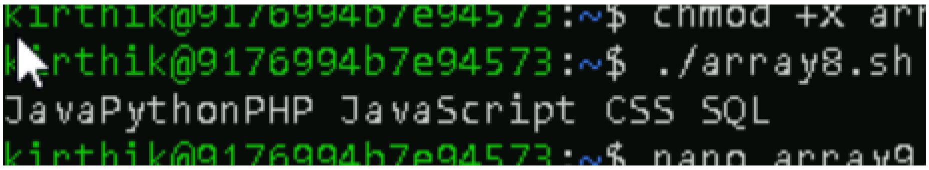
```
kirthik@9176994b7e94573:~$ nano array8.sh
kirthik@9176994b7e94573:~$ chmod +x array8.sh
kirthik@9176994b7e94573:~$ ./array8.sh
```

SCRIPT:



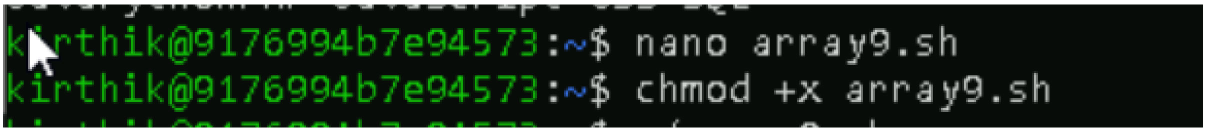
```
#!/bin/bash
#Declaring the Array
declare -a example_array=( "Java""Python""PHP" )
#Adding new elements
example_array+=( JavaScript CSS SQL )
#Printing all the elements
echo "${example_array[@]}"
```

OUTPUT:



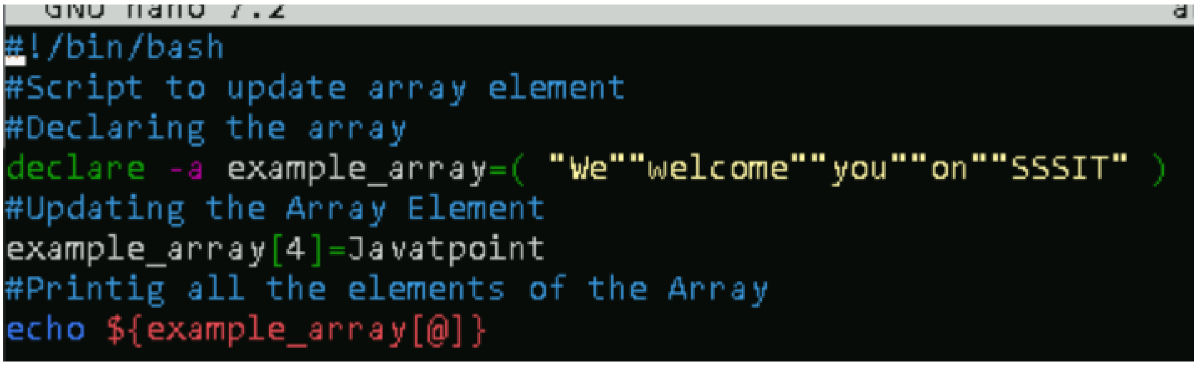
```
kirthik@9176994b7e94573:~$ chmod +x arr
kirthik@9176994b7e94573:~$ ./array8.sh
JavaPythonPHP JavaScript CSS SQL
kirthik@9176994b7e94573:~$ nano array9
```

CREATING FILE:



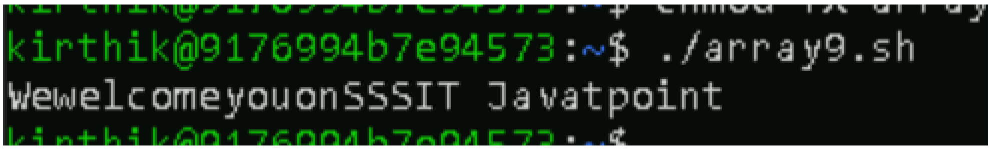
```
kirthik@9176994b7e94573:~$ nano array9.sh
kirthik@9176994b7e94573:~$ chmod +x array9.sh
```

SCRIPT:



```
  GNU nano 7.2
#!/bin/bash
#Script to update array element
#Declaring the array
declare -a example_array=( "We""welcome""you""on""SSSIT" )
#Updating the Array Element
example_array[4]=Javatpoint
#Printig all the elements of the Array
echo ${example_array[@]}
```

OUTPUT:



```
kirthik@9176994b7e94573:~$ ./array9.sh
WewelcomeyouonSSSIT Javatpoint
kirthik@9176994b7e94573:~$
```