

## SWITCHCASE:

### CREATING FILE:

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

CREATING FILE:

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

SCRIPT:

```
#!/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
```



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.
```

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

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

[illegible]

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$
```

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:

```
#!/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
```