Assignment :: 22nd April Thursday 2021

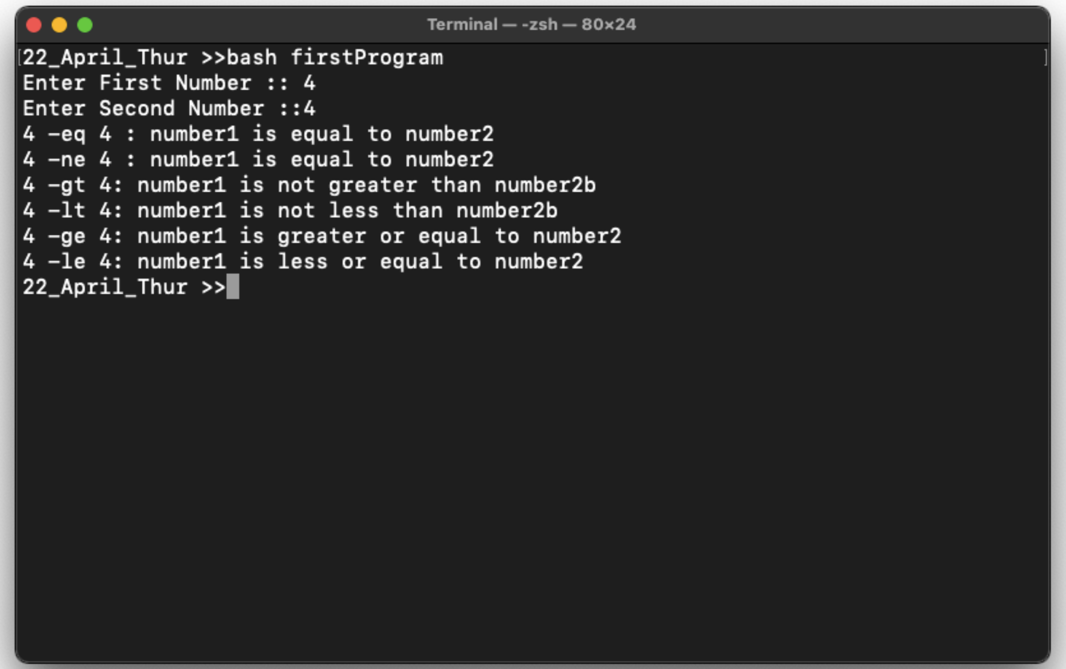
Total 12 Shell Script Programs

(some error related to nextLine or unsupported syntax may occur , change accordingly)

1 ::

```
#Program to perform function of  different
relational Operators
#!/bin/bash
echo -n "Enter First Number :: "
read number1
echo -n "Enter Second Number ::"
read number2

if [ $number1 -eq $number2 ]
then
 echo "$number1 -eq $number2 : number1 is
equal to number2"
else
 echo "$number1 -eq $number2: number1 is not
equal to number2"
fi
if [ $number1 -ne $number2 ]
then
echo "$number1 -ne $number2: number1 is not
equal to number2"
else
echo "$number1 -ne $number2 : number1 is equal
to number2"
fi
if [ $number1 -gt $number2 ]
then
```

```
echo "$number1 -gt $number2: number1 is
greater than number2"
else
echo "$number1 -gt $number2: number1 is not
greater than number2b"
fi
if [ $number1 -lt $number2 ]
then
echo "$number1 -lt $number2: number1a is less
than number2"
else
echo "$number1 -lt $number2: number1 is not
less than number2b"
fi
if [ $number1 -ge $number2 ]
then
echo "$number1 -ge $number2: number1 is
greater or equal to number2"
else
echo "$number1 -ge $number2: number1 is not
greater or equal to number2"
fi
if [ $number1 -le $number2 ]
then
echo "$number1 -le $number2: number1 is less
or equal to number2"
else
echo "$number1 -le $number2: number1 is not
less or equal to number2"
fi
```

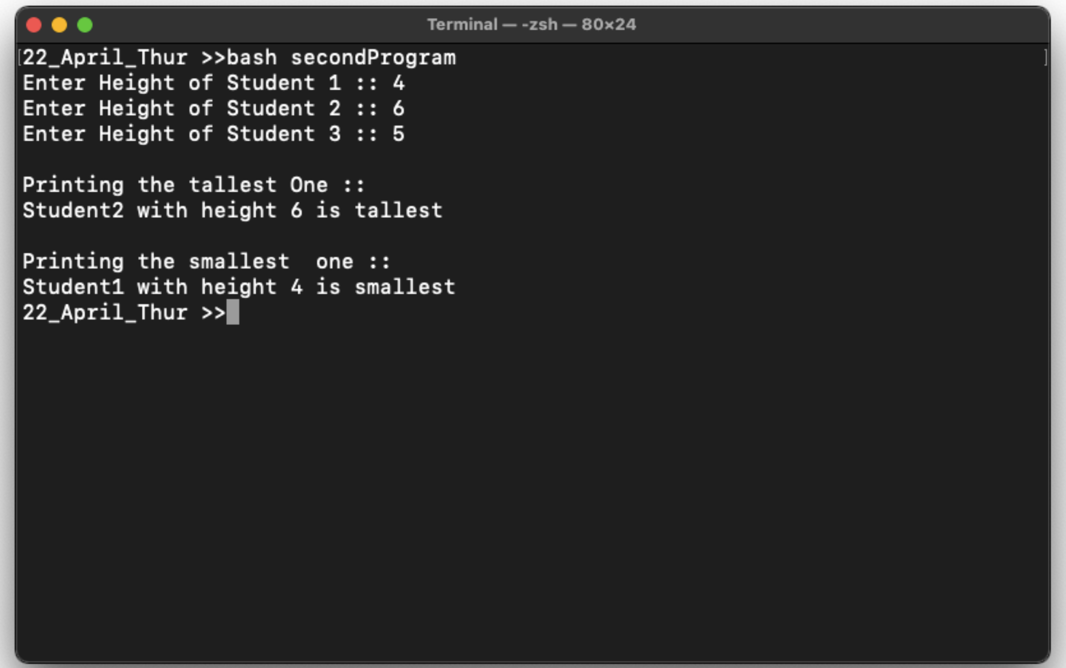


```
22_April_Thur >>bash firstProgram
Enter First Number :: 4
Enter Second Number ::4
4 -eq 4 : number1 is equal to number2
4 -ne 4 : number1 is equal to number2
4 -gt 4: number1 is not greater than number2b
4 -lt 4: number1 is not less than number2b
4 -ge 4: number1 is greater or equal to number2
4 -le 4: number1 is less or equal to number2
22_April_Thur >>
```

2 ::

```
#Program to compare student height with each
other and printing  the tallest and smallest

#!/bin/bash
echo -n "Enter Height of Student 1 :: "
read student1
echo -n "Enter Height of Student 2 :: "
read student2
echo -n "Enter Height of Student 3 :: "
read student3

echo  -e "\nPrinting the tallest One ::"
if [ $student1 -gt $student2 -a $student1 -gt
$student3 ]
then
echo "Student1 with height $student1 is
tallest"
```

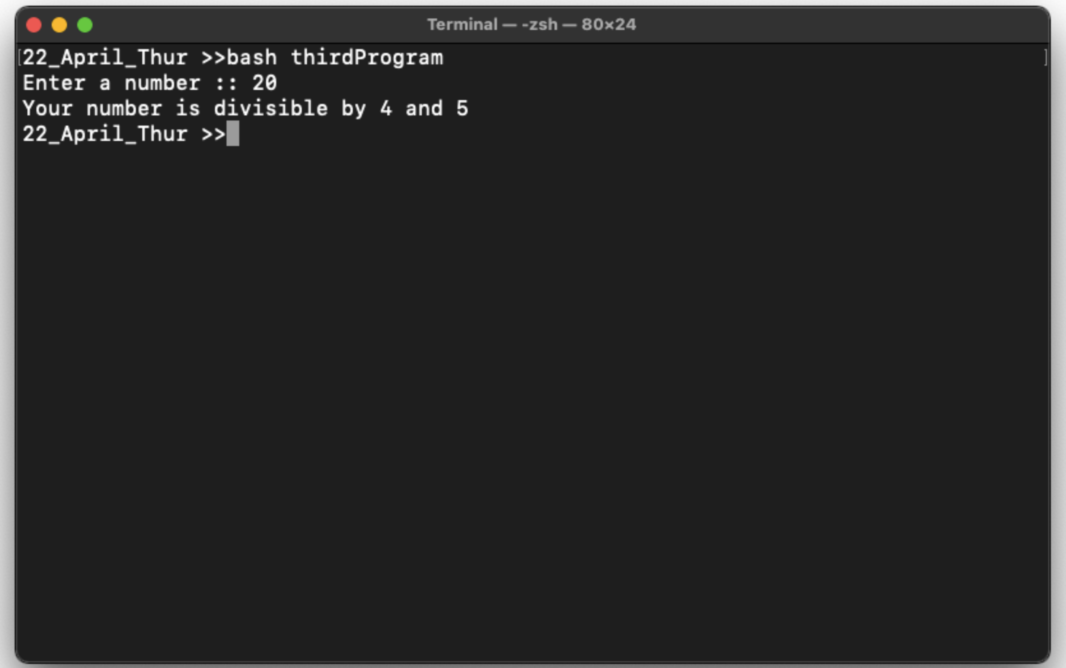
```
elif [ $student2 -gt $student1 -a $student2
-gt $student3 ]
then
echo "Student2 with height $student2 is
tallest"
else
echo "Student3 with height $student3 is
tallest"
fi


echo -e "\nPrinting the smallest  one :: "
if [ $student1 -lt $student2 -a $student1 -lt
$student3 ]
then
echo "Student1 with height $student1 is
smallest"
elif [ $student2 -lt $student1 -a $student2
-lt $student3 ]
then
echo "Student2 with height $student2 is
smallest"
else
echo "Student3 with height $student3 is
smallest"
fi
```



```
22_April_Thur >>bash secondProgram
Enter Height of Student 1 :: 4
Enter Height of Student 2 :: 6
Enter Height of Student 3 :: 5

Printing the tallest One ::
Student2 with height 6 is tallest

Printing the smallest  one ::
Student1 with height 4 is smallest
22_April_Thur >>
```

3 ::

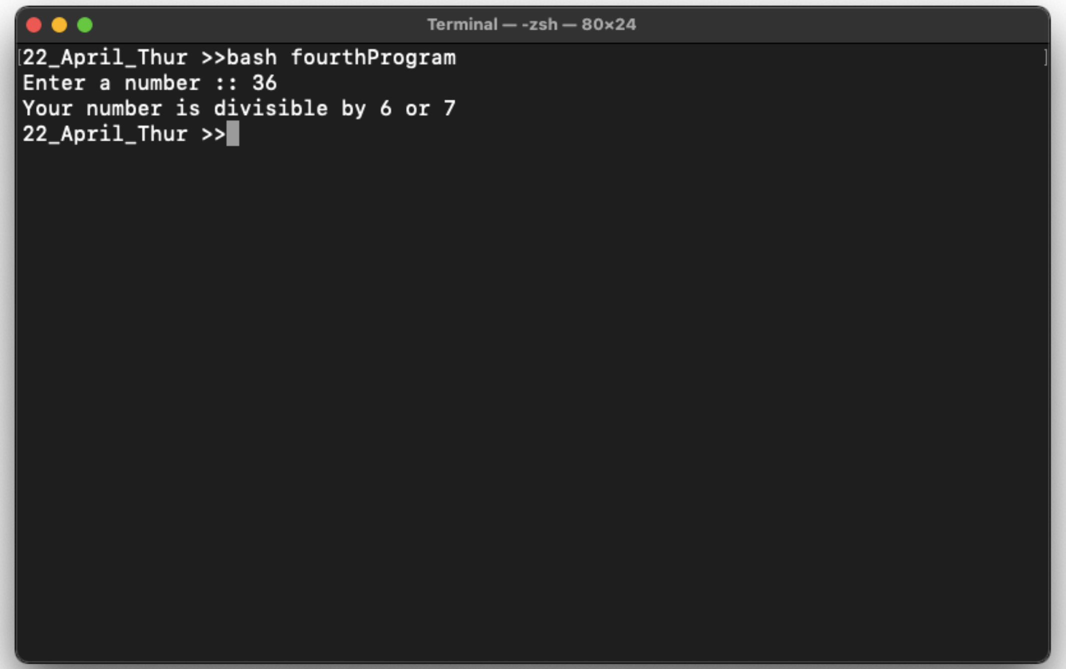
```
#!/bin/bash
echo -n "Enter a number :: "
read number

if [ $(( $number % 4 )) -eq 0 ] && [ $
(( $number % 5 )) -eq 0 ]
then
    echo "Your number is divisible by 4 and 5"
else
    echo "Your number is not divisible by 4
and 5"
fi
```



```
22_April_Thur >>bash thirdProgram
Enter a number :: 20
Your number is divisible by 4 and 5
22_April_Thur >>
```

4 ::

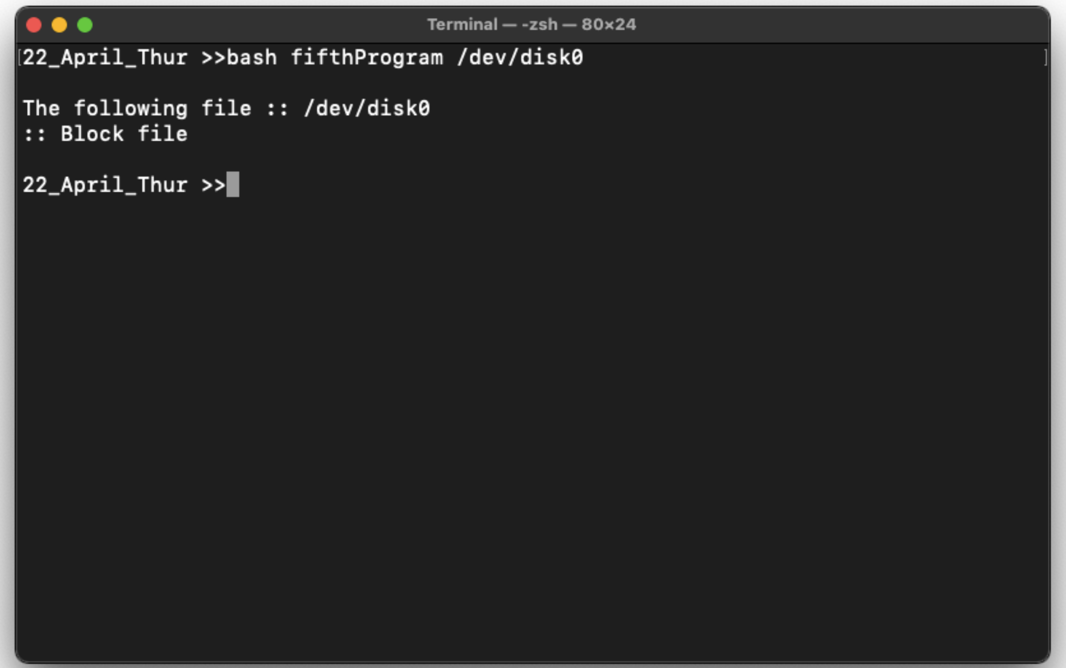
```
#!/bin/bash
echo -n "Enter a number :: "
read number

if [ $(( $number % 6 )) -eq 0 ] || [ $
(( $number % 7 )) -eq 0 ]
then
    echo "Your number is divisible by 6 or 7"
else
    echo "Your number is not divisible by 6 or
7"
fi
```



```
22_April_Thur >>bash fourthProgram
Enter a number :: 36
Your number is divisible by 6 or 7
22_April_Thur >>
```

5 ::

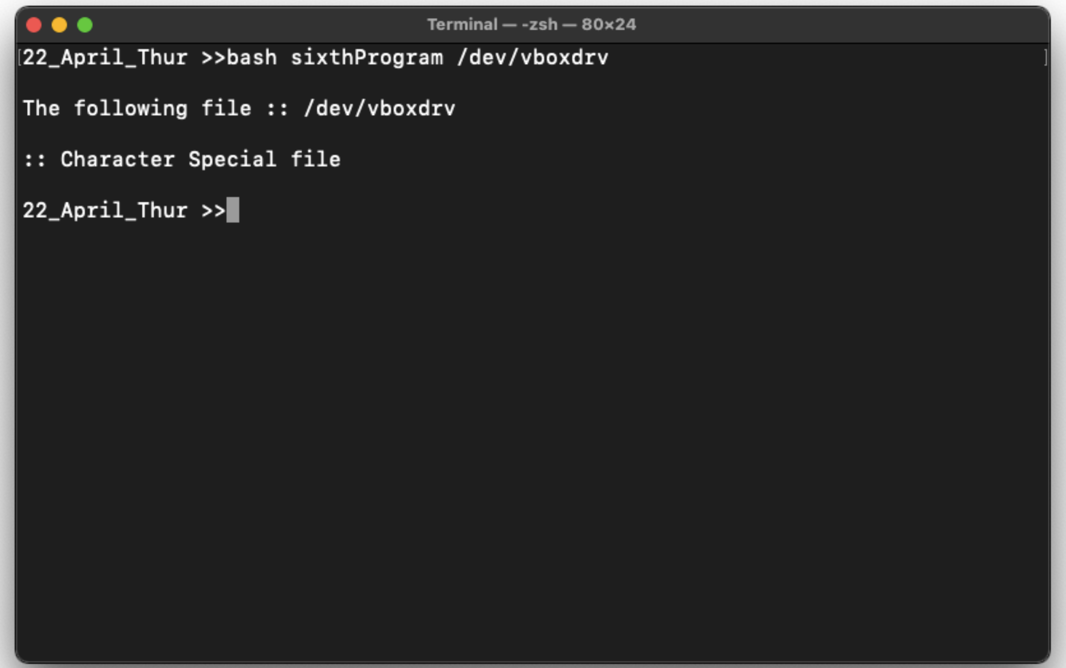
```
#!/bin/bash
fileName=$1
echo -e  "\nThe following file :: $fileName "
if [ -b $fileName ]
then
echo -e ":: Block file\n"
else
echo -e  ":: Not a Block  file\n"
fi
```



```
22_April_Thur >>bash fifthProgram /dev/disk0

The following file :: /dev/disk0
:: Block file

22_April_Thur >>
```

6 ::

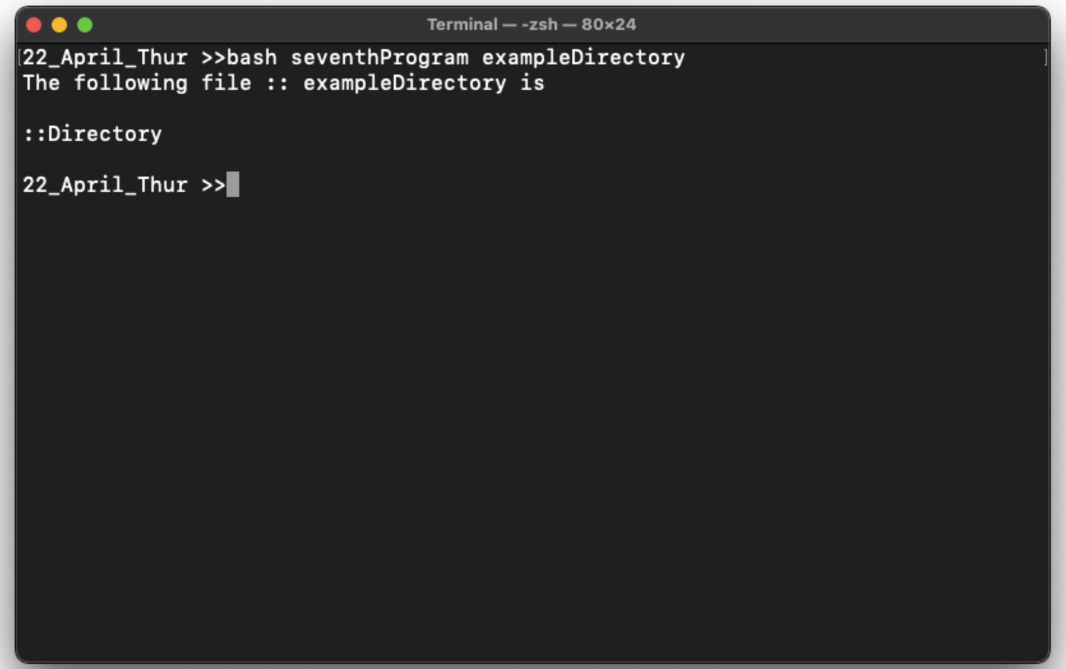
```
#!/bin/bash
fileName=$1
echo -e  "\nThe following file :: $fileName "
if [ -c $fileName ]
then
echo -e "\n:: Character Special file\n"
else
echo -e  "\n:: Not a Character Special file\n"
fi
```



```
22_April_Thur >>bash sixthProgram /dev/vboxdrv

The following file :: /dev/vboxdrv

:: Character Special file

22_April_Thur >>
```

7 ::

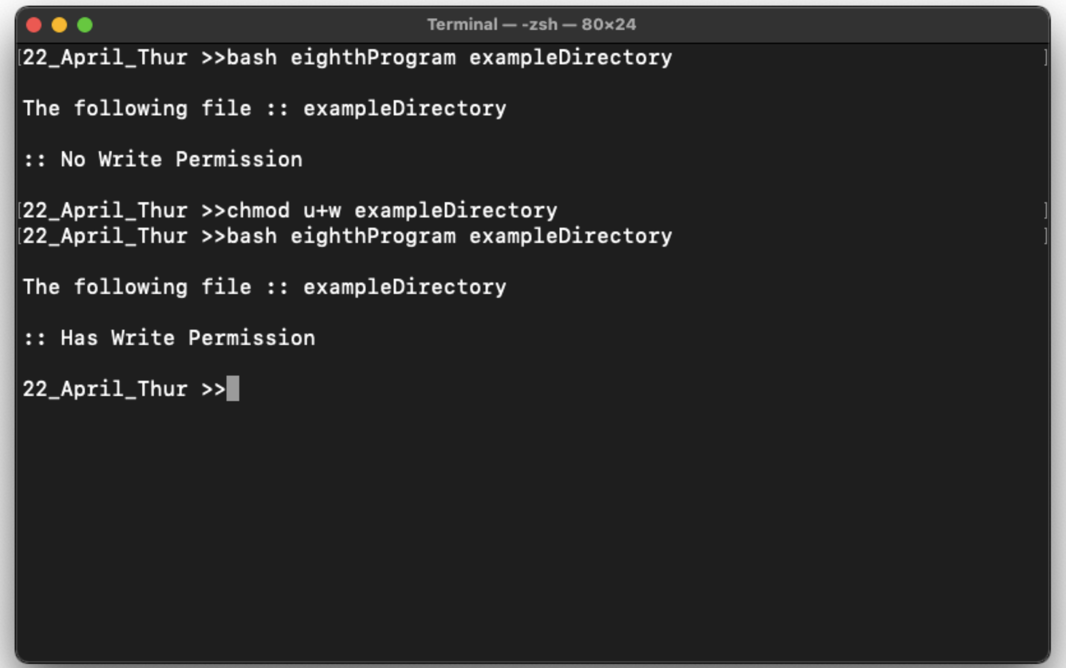
```
#!/bin/bash
fileName=$1
echo "The following file :: $fileName is "
if [ -d $fileName ]
then
echo -e  "\n::Directory\n";
else
echo -e "\n::Not a Directory\n";
fi
```



```
22_April_Thur >>bash seventhProgram exampleDirectory
The following file :: exampleDirectory is

::Directory

22_April_Thur >>
```

8 ::

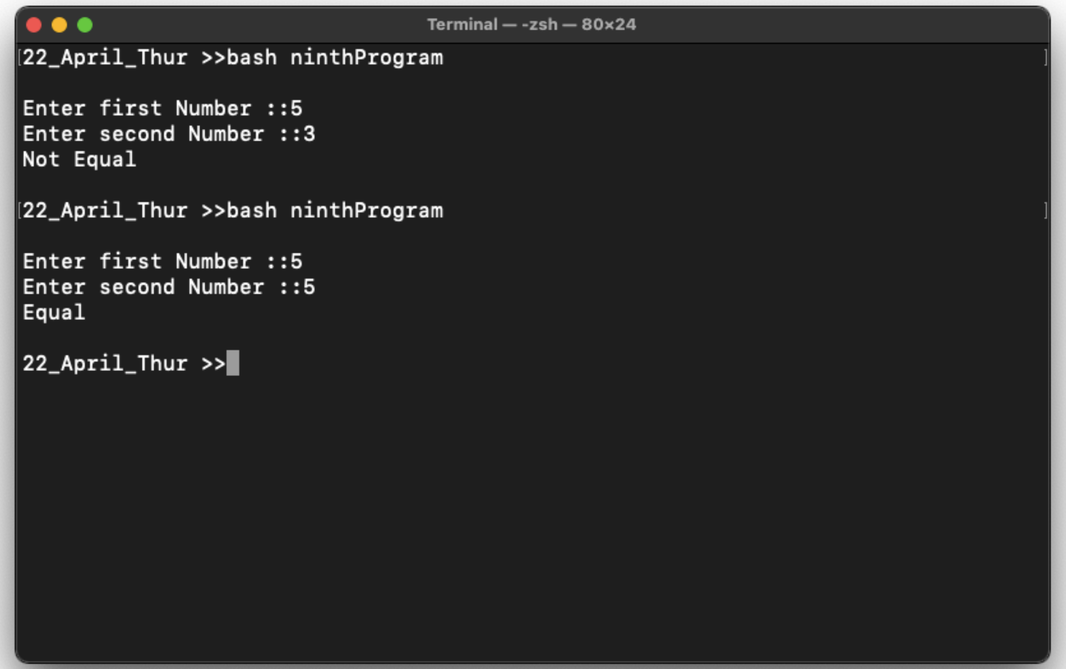
```
#!/bin/bash
fileName=$1
echo -e  "\nThe following file :: $fileName "
if [ -w $fileName ]
then
echo -e "\n:: Has Write Permission\n"
else
echo -e  "\n:: No Write Permission\n"
fi
```



```
Terminal — -zsh — 80×24
22_April_Thur >>bash eighthProgram exampleDirectory

The following file :: exampleDirectory

:: No Write Permission

22_April_Thur >>chmod u+w exampleDirectory
22_April_Thur >>bash eighthProgram exampleDirectory

The following file :: exampleDirectory

:: Has Write Permission

22_April_Thur >>
```

9 ::

```
#!/bin/bash

echo -n -e  "\nEnter first Number ::"
read number1
echo -n "Enter second Number ::"
read number2

if [ $number1 == $number2 ]
then
echo -e "Equal\n"
else
echo -e "Not Equal\n"
fi
```

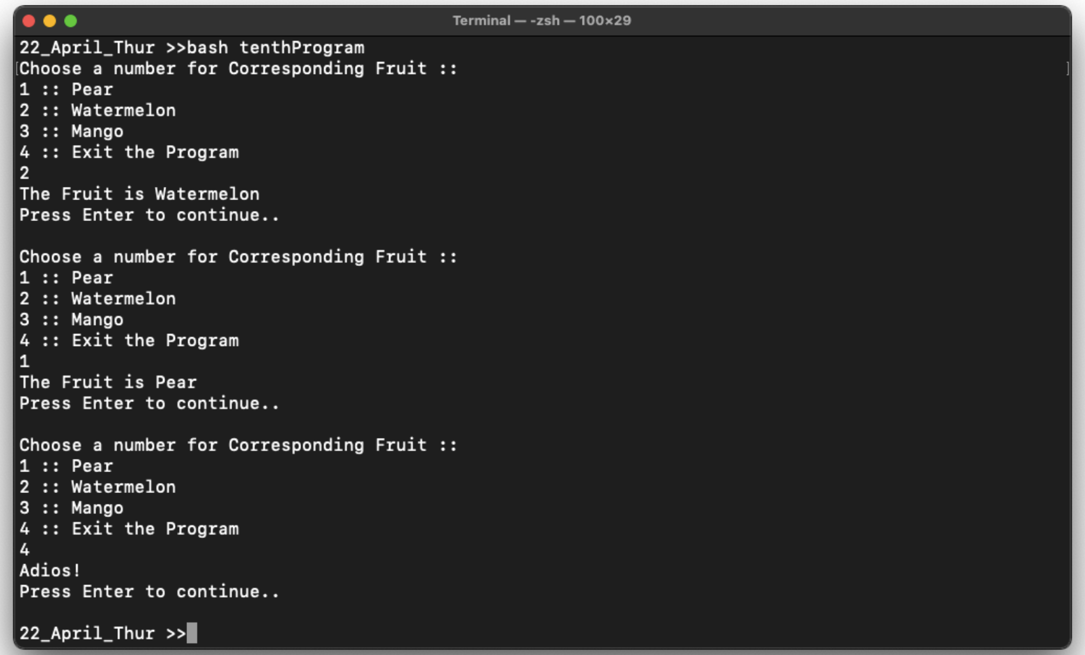


```
22_April_Thur >>bash ninthProgram

Enter first Number ::5
Enter second Number ::3
Not Equal

22_April_Thur >>bash ninthProgram

Enter first Number ::5
Enter second Number ::5
Equal

22_April_Thur >>
```

10 ::

```
#!/bin/bash
CHOICE=5
until [ $CHOICE -eq 4 ]
do

echo -n -e "Choose a number for Corresponding
Fruit ::\n"
echo "1 :: Pear"
echo "2 :: Watermelon"
echo "3 :: Mango"
echo "4 :: Exit the Program"
read CHOICE
case $CHOICE in
1) echo "The Fruit is Pear";;
2) echo "The Fruit is Watermelon";;
3) echo "The Fruit is Mango";;
4) echo "Adios!";;
```

```
*) echo "Invalid Input !";;
esac
echo "Press Enter to continue.."
read ENTER
done
```



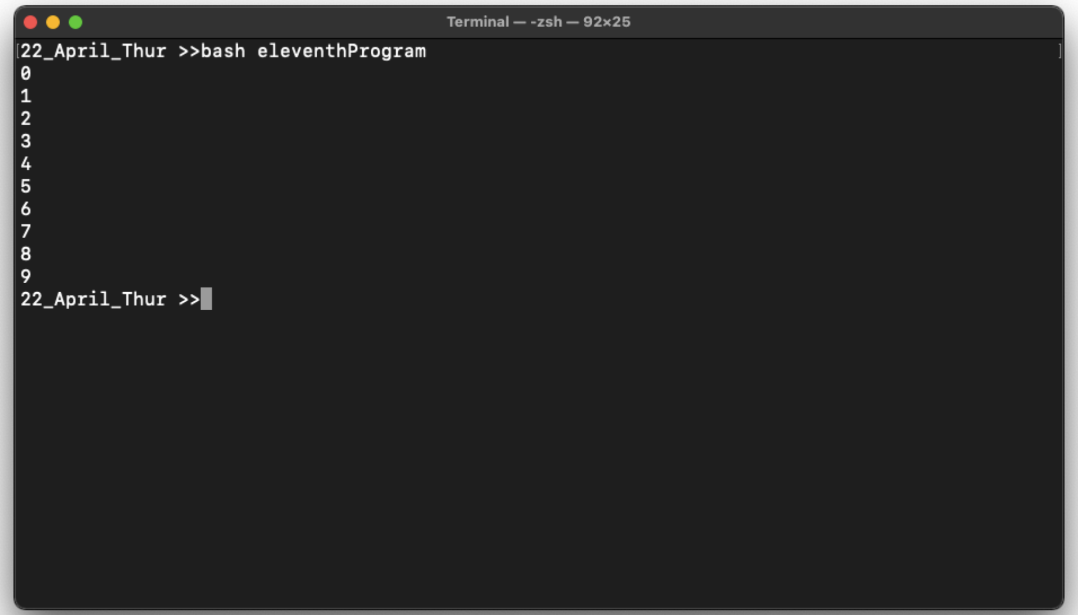
Terminal — -zsh — 100×29

```
22_April_Thur >>bash tenthProgram
Choose a number for Corresponding Fruit ::
1 :: Pear
2 :: Watermelon
3 :: Mango
4 :: Exit the Program
2
The Fruit is Watermelon
Press Enter to continue..

Choose a number for Corresponding Fruit ::
1 :: Pear
2 :: Watermelon
3 :: Mango
4 :: Exit the Program
1
The Fruit is Pear
Press Enter to continue..

Choose a number for Corresponding Fruit ::
1 :: Pear
2 :: Watermelon
3 :: Mango
4 :: Exit the Program
4
Adios!
Press Enter to continue..

22_April_Thur >>
```

11 ::

```
#!/bin/bash

number=0
while [ "$number" -lt 10 ]
do
echo $number
```

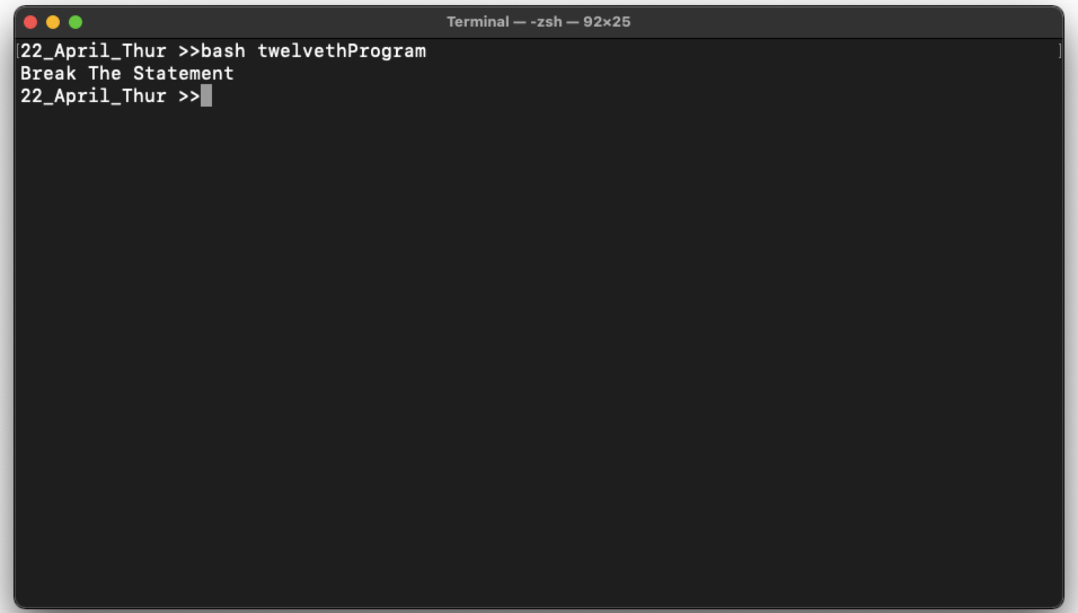
```
sleep 1
number=`expr $number + 1`
done
```



```
22_April_Thur >>bash eleventhProgram
0
1
2
3
4
5
6
7
8
9
22_April_Thur >>
```

12 ::

```
#!/bin/bash
a=5
while [ $a -lt 10 ]
do
echo "Break The Statement"
sleep 1
break
done
```



```
22_April_Thur >>bash twelvethProgram
Break The Statement
22_April_Thur >>
```