**Problems with if & else**

**1. Write a program that reads 5 Random 3 Digit values and then outputs the minimum and the maximum value**

#!/bin/bash -x

num1=$((RANDOM%899+100))

num2=$((RANDOM%899+100))

num3=$((RANDOM%899+100))

num4=$((RANDOM%899+100))

num5=$((RANDOM%899+100))

if [ $num1 -gt $num2 ] && [ $num1 -gt $num2 ] && [ $num1 -gt $num3 ] && [ $num1 -gt $num4 ] && [ $num1 -gt $num5 ] ; then

echo "$num1 is a Greatest Number"

elif [ $num2 -gt $num3 ] && [ $num2 -gt $num3 ] && [ $num2 -gt $num4 ] && [ $num2 -gt $num5 ] ; then

echo "$num2 is a Greatest Number"

elif [ $num3 -gt $num4 ] && [ $num3 -gt $num5 ] ; then

echo "$num3 is a Greatest Number"

elif [ $num4 -gt $num5 ] ; then

echo "$num4 is a Greatest Number"

else

echo "$num5 is a Greatest Number"

fi

if [ $num1 -lt $num2 ] && [ $num1 -lt $num2 ] && [ $num1 -lt $num3 ] && [ $num1 -lt $num4 ] && [ $num1 -lt $num5 ] ; then

echo "$num1 is a smallestt Number"

elif [ $num2 -lt $num3 ] && [ $num2 -lt $num3 ] && [ $num2 -lt $num4 ] && [ $num2 -lt $num5 ] ; then

echo "$num2 is a smallest Number"

elif [ $num3 -lt $num4 ] && [ $num3 -lt $num5 ] ; then

echo "$num3 is a smallest Number"

elif [ $num4 -lt $num5 ] ; then

echo "$num4 is a smallest Number"

else

echo "$num5 is a smallest Number"

fi

output:

AKHIL@DESKTOP-43E18QK MINGW64 ~

$ ./ifelse.sh

+ num1=372

+ num2=146

+ num3=705

+ num4=150

+ num5=572

+ '[' 372 -gt 146 ']'

+ '[' 372 -gt 146 ']'

+ '[' 372 -gt 705 ']'

+ '[' 146 -gt 705 ']'

+ '[' 705 -gt 150 ']'

+ '[' 705 -gt 572 ']'

+ echo '705 is a Greatest Number'

705 is a Greatest Number

+ '[' 372 -lt 146 ']'

+ '[' 146 -lt 705 ']'

+ '[' 146 -lt 705 ']'

+ '[' 146 -lt 150 ']'

+ '[' 146 -lt 572 ']'

+ echo '146 is a smallest Number'

146 is a smallest Number

**2. Write a program that takes day and month from the command line and prints true if day of month is between March 20 and June 20, false otherwise.**

#!/bin/bash -x

read -p "Enter Date: " date

read -p "Enter Month: " Month

calculatedates=0

result="false"

flag=1

if [ $Month -ge 3 -a $Month -le 6 ]; then

datelimit=$((30 + (Month % 2)))

if [ $date -ge 1 -a $date -le $datelimit ]; then

calculatedates=$(((Month \* 100) + date))

#prints the dates between march 20 to june 20

if [ $calculatedates -ge 320 -a $calculatedates -le 620 ]; then

result="true"

flag=0

fi

fi

fi

echo "$Month/$date $result"

exit $flag

output:

AKHIL@DESKTOP-43E18QK MINGW64 ~

$ ./ifmonth.sh

Enter Date: 24

Enter Month: 5

5/24 true

AKHIL@DESKTOP-43E18QK MINGW64 ~

$ ./ifmonth.sh

Enter Date: 19

Enter Month: 3

3/19 false

**3. Write a program that takes a year as input and outputs the Year is a Leap Year or not a Leap Year. A Leap Year checks for 4 Digit Number, Divisible by 4 and not 100 unless divisible by 400.**

#!/bin/bash

yy=0

isleap="false"

echo "Enter year (yyyy) : "

read yy

if [ $((yy % 4)) -ne 0 ] ; then

echo "not a leap year"

elif [ $((yy % 400)) -eq 0 ] ; then

isleap="true"

elif [ $((yy % 100)) -eq 0 ] ; then

echo "is a leap year"

else

isleap="true"

fi

if [ "$isleap" == "true" ];

then

echo "$yy is leap year"

else

echo "$yy is NOT leap year"

fi

output:

AKHIL@DESKTOP-43E18QK MINGW64 ~

$ ./leap.sh

Enter year (yyyy) :

2014

not a leap year

2014 is NOT leap year

AKHIL@DESKTOP-43E18QK MINGW64 ~

$ ./leap.sh

Enter year (yyyy) :

2020

2020 is leap year

(OR)

#!/bin/bash

echo "Enter the year (YYYY)"

read year

if [ $((year % 4)) -eq 0 ]

then

if [ $((year % 400)) -eq 0 ]

then

if [ $((year % 100)) -eq 0 ]

then

echo "its a not leap year"

else

echo "its a leap year"

fi

else

echo "Its a leap year"

fi

else

echo "its not a leap year"

fi

output:

AKHIL@DESKTOP-43E18QK MINGW64 ~

$ ./ifmonth2.sh

Enter the year (YYYY)

2020

Its a leap year

AKHIL@DESKTOP-43E18QK MINGW64 ~

$ ./ifmonth2.sh

Enter the year (YYYY)

2018

its not a leap year

**4. Write a program to simulate a coin flip and print out "Heads" or "Tails" accordingly.**

AKHIL@DESKTOP-43E18QK MINGW64 ~

$ nano flipacoin.sh

#!/bin/bash

Result=$((RANDOM%2))

if [[ ${Result} -eq 0 ]]; then

echo HEADS

else

echo TAILS

fi

output:

AKHIL@DESKTOP-43E18QK MINGW64 ~

$ ./flipacoin.sh

TAILS

AKHIL@DESKTOP-43E18QK MINGW64 ~

$ ./flipacoin.sh

HEADS