**Day 5 Practice Problems**

1**)Use Random Function (RANDOM) to get Single Digit**

#!/bin/bash -x

echo "Getting single digit using RANDOM function"

echo $(( RANDOM % 10 + 1 ))

**OUTPUT**

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./Single\_Digit.sh

+ echo 'Getting single digit using RANDOM function'

Getting single digit using RANDOM function

+ echo 8

8

**2) Use Random to get Dice Number between 1 to 6 ?**

#!/bin/bash -x

echo "Dice Number between 1 to 6"

echo $(( RANDOM % 6 + 1 ))

**OUTPUT**

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./Dice\_Number.sh

+ echo 'Dice Number between 1 to 6'

Dice Number between 1 to 6

+ echo 2

2

Add two Random Dice Number and Print the Result

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ nano Add\_Dice\_Number.sh

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./Add\_Dice\_Number.sh

+ echo 'Add two Random Dice Number'

Add two Random Dice Number

+ Dice1=13

+ Dice2=10

+ Add=23

+ echo ' Addition of two dice number' '' 10

Addition of two dice number 10

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

**3) Write a program that reads 5 Random 2 Digit values, then find their sum and average**

#!/bin/bash -x

read -p " Enter first number" first

read -p " Enter second number"second

read -p " Enter third number"third

read -p " Enter fourth number" fourth

read -p " Enter fifth number " fifth

add=$(( $first+$second+$third+$fourth+$fifth ))

avg=$(( $add/5 ))

**OUTPUT**

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./Add\_Avg.sh

+ read -p ' Enter first number' first

Enter first number1

+ read -p ' Enter second numbersecond'

Enter second numbersecond2

+ read -p ' Enter third numberthird'

Enter third numberthird3

+ read -p ' Enter fourth number' fourth

Enter fourth number4

+ read -p ' Enter fifth number ' fifth

Enter fifth number 5

+ add=10

+ avg=2

**4) Unit Cunversion:**

1. **1ft=12 in then 42 in=?ft**
2. **Rectangular plot of 60 feet\* 40feet in meters**
3. **Calculate area of 25 such plots in acres**

#!/bin/bash -x

read -p " Enter inch" inch

ft=$(($inch\*12 ))

echo "$inch inch= $ft ft"

**OUTPUT**

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./Unit\_Conversion.sh

+ read -p ' Enter inch' inch

Enter inch4

+ ft=48

+ echo '4 inch= 48 ft'

4 inch= 48 ft

**Selection Practice 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

min\_val=100

max\_val=999

echo "5 Random 5 digit values"

Value=$(( RANDOM % 999 + 5 ))

mid=$(( $min\_val + $max\_val % 2 ))

if [ $Value -lt $mid ]

then

echo "Value is minimum" $min\_val

else

echo " Value is maximum" $max\_val

fi

**OUTPUT**

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./Min\_Max\_value.sh

+ min\_val=100

+ max\_val=999

+ echo '5 Random 5 digit values'

5 Random 5 digit values

+ Value=403

+ mid=101

+ '[' 403 -lt 101 ']'

+ echo ' Value is maximum' 999

Value is maximum 999

**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 day which u want to display:" day

read -p "Enter month which u want to display:" month

day1=20

m1=may

m2=june

if [ $day -eq $day1 ] && [ $day -eq $m1 ] && [ $day -eq $m2 ]

then

echo " true"

else

echo "false"

fi

**OUTPUT**

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./day\_month1.sh

+ read -p 'Enter day which u want to display:' day

Enter day which u want to display:2

+ read -p 'Enter month which u want to display:' month

Enter month which u want to display:may

+ day1=20

+ m1=may

+ m2=june

+ '[' 2 -eq 20 ']'

+ echo false

false

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

#!/bin/bash -x

read -p "Enter the year u want to check" year

d=`expr $year % 4 `

if [ $d -eq 0 ]

then

echo "$Year is leap"

else

echo " $Year is not leap"

fi

**OUTPUT**

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./leap\_year1.sh

+ read -p 'Enter the year u want to check' year

Enter the year u want to check 1994

++ expr 1994 % 4

+ d=2

+ '[' 2 -eq 0 ']'

+ echo ' is not leap'

is not leap

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$

**4) Write a progrsm to simulate a coin flip and print out “HEAD” and “TAIL”**

#!/bin/bash -x

head=1

randomCheck=$(( RANDOM%2 ))

if [ $head -eq $randomCheck ]

then

echo "its head"

else

echo "its tail"

fi

**OUTPUT**

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./Coin\_flip.sh

+ head=1

+ randomCheck=0

+ '[' 1 -eq 0 ']'

+ echo 'its tail'

its tail

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./Coin\_flip.sh

+ head=1

+ randomCheck=1

+ '[' 1 -eq 1 ']'

+ echo 'its head'

its head

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$

**1)Read a single digit and write number in word**

#! /bin/bash -x

read -p "Enter single digit:" n

len=$(echo $n | wc -c)

len=$(( $len - 1 ))

echo "number in words $n"

for (( i=1; i<=$len; i++ ))

do

digit=$(echo $n | cut -c $i)

case $digit in

0) echo "zero" ;;

1) echo "one" ;;

2) echo "two" ;;

3) echo "three" ;;

4) echo "four" ;;

5) echo "five" ;;

6) echo "six" ;;

7) echo "seven" ;;

8) echo "eight" ;;

9) echo "nine" ;;

esac

done

**OUTPUT**

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./Num\_in\_word.sh

+ read -p 'Enter single digit:' n

Enter single digit:5

++ echo 5

++ wc -c

+ len=2

+ len=1

+ echo 'number in words 5'

number in words 5

+ (( i=1 ))

+ (( i<=1 ))

++ echo 5

++ cut -c 1

+ digit=5

+ case $digit in

+ echo five

five

+ (( i++ ))

+ (( i<=1 ))

**2)Read a number and display a week day**

#! /bin/bash -x

read -p "Enter the number :" n

len=$(echo $n | wc -c)

len=$(( $len - 1 ))

echo "number is $n in word:"

for ((i=1; i<=$len; i++ ))

do

digit=$(echo $n | cut -c $i)

case $digit in

1) echo "Sunday" ;;

2) echo "Monday" ;;

3) echo "Tuesday" ;;

4) echo "Wedensday" ;;

5) echo "Thursday" ;;

6) echo "Friday" ;;

7) echo "Sunday" ;;

esac

done

**OUTPUT**

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./day.sh

+ read -p 'Enter the number :' n

Enter the number :2

++ echo 2

++ wc -c

+ len=2

+ len=1

+ echo 'number is 2 in word:'

number is 2 in word:

+ (( i=1 ))

+ (( i<=1 ))

++ echo 2

++ cut -c 1

+ digit=2

+ case $digit in

+ echo Monday

Monday

+ (( i++ ))

+ (( i<=1 ))

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$

**3)Read a number 1,10,100,1000, etc and display unit ten, hundred, thousand.**

#! /bin/bash -x

read -p "Enter the no " n

len=$(echo $n | wc -c)

len=$(( $len - 1 ))

echo "Enter the no:"

for (( i=1; i<=$len; i++ ))

do

digit=$(echo $n | cut -c $i)

case $digit in

1) echo "unit" ;;

10)echo "Ten" ;;

100)echo "Hundred" ;;

1000)echo "thousand" ;;

esac

done

**OUTPUT**

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./Unit\_digit.sh

+ read -p 'Enter the no ' n

Enter the no 100

++ echo 100

++ wc -c

+ len=4

+ len=3

+ echo 'Enter the no:'

Enter the no:

+ (( i=1 ))

+ (( i<=3 ))

++ echo 100

++ cut -c 1

+ digit=1

+ case $digit in

+ echo unit

unit

+ (( i++ ))

+ (( i<=3 ))

++ echo 100

++ cut -c 2

+ digit=0

+ case $digit in

+ (( i++ ))

+ (( i<=3 ))

++ echo 100

++ cut -c 3

+ digit=0

+ case $digit in

+ (( i++ ))

+ (( i<=3 ))

**4)Enter 3 numbers do fillowing arithmetic operation and find one that**

1. **A + b\* c**
2. **A%b+c**
3. **C+a/b**
4. **A\*b+c**

**OUTPUT**

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$ ./arithmetic\_operation.sh

+ echo ' Enter value of a, b & c'

Enter value of a, b & c

+ read a b c

1 2 3

+ a=7

+ b=6

+ c=4

+ d=46

+ '[' 7 -gt 6 ']'

+ '[' 7 -gt 4 ']'

+ '[' 7 -gt 46 ']'

+ '[' 6 -gt 7 ']'

+ '[' 4 -gt 7 ']'

+ '[' 46 -gt 7 ']'

+ '[' 46 -gt 6 ']'

+ '[' 46 -gt 4 ']'

+ echo '46 id greated'

46 id greated

+ '[' 7 -le 6 ']'

+ '[' 6 -le 7 ']'

+ '[' 6 -le 4 ']'

+ '[' 4 -le 7 ']'

+ '[' 4 -le 6 ']'

+ '[' 4 -le 46 ']'

+ echo '4 is smaller'

4 is smaller

Admin@Admin-PC MINGW64 ~/TerminalCommands12/linux-content (master)

$