# Day 5 Practice Problems

**Solved by:** SAPTAK DAS(saptakds@gmail.com)

# Sequences Practice Problems

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

```
$ echo $((RANDOM%9))

7
```

2. Use Random to get Dice Number between 1 to 6

```
$ echo $((1+RANDOM%6))

3
```

3. Add two Random Dice Number and Print the Result

```
#!/bin/bash -x
random1=$((1+RANDOM%6))
random2=$((1+RANDOM%6))
sum=$(($random1 + $random2))
echo $sum

++ expr 1
+ random1=1
++ expr 6
+ random2=6
++ expr 1 + 6
+ sum=7
+ echo 7
7
```

4. Write a program that reads 5 Random 2 Digit values, then find their sum and the average

```
#!/bin/bash -x
sum=0
for num in `seq 5`
do
   randomNo=$((10+RANDOM%99))
   sum=$(($sum + $randomNo))
done
avg=$(echo "scale=2;$sum/$num" | bc)
echo "Sum: $sum"
echo "Average: $avg"
```

```
+ sum=0
++ seq 5
+ for num in `seq 5`
+ randomNo=13
+ sum=13
+ for num in `seq 5`
+ randomNo=35
+ sum=48
+ for num in `seq 5`
+ randomNo=71
+ sum=119
+ for num in `seq 5`
+ randomNo=66
+ sum=185
+ for num in `seq 5`
+ randomNo=86
+ sum=271
++ echo 'scale=2;271/5'
++ bc
+ avg=54.20
+ echo 'Sum: 271'
Sum: 271
+ echo 'Average: 54.20'
Average: 54.20
```

5. Unit Conversion
   a. 1ft = 12 in then 42 in = ? ft
   b. Rectangular Plot of 60 feet x 40 feet in meters
   c. Calculate area of 25 such plots in acres

```
#!/bin/bash -x
inch=42
feet=$(echo "scale-2;$inch/12" | bc)
echo "$inch in = $feet ft"
```

```
+ inch=42
```

```
++ echo 'scale-2;42/12'
++ bc
+ feet='-2
3'
+ echo '42 in = -2
3 ft'
42 in = -2
3 ft
```

```
#!/bin/bash -x
length=60
breadth=40
plots=25
area=$(echo "scale=2;$length*$breadth*$plots" | bc)
finalAreaAcres=$(echo "scale=2;$area/43560" | bc)
echo "Area in acres: $finalAreaAcres"
```

```
+ length=60
+ breadth=40
+ plots=25
++ echo 'scale=2;60*40*25'
++ bc
+ area=60000
++ echo 'scale=2;60000/43560'
++ bc
+ finalAreaAcres=1.37
+ echo 'Area in acres: 1.37'
Area in acres: 1.37
```

# 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
max=100
min=999
for num in `seq 5`
do
   randomNo=$((100+RANDOM%999))
   if [ $randomNo -gt $max ]
   then
      max=$randomNo
   fi
   if [ $randomNo -lt $min ]
   then
      min=$randomNo
   fi
done
echo "Maximum: $max"
echo "Minimum: $min"
```

```
+ max=100
+ min=999
++ seq 5
+ for num in `seq 5`
+ randomNo=314
+ '[' 314 -gt 100 ']'
+ max=314
+ '[' 314 -lt 999 ']'
+ min=314
+ for num in `seq 5`
+ randomNo=436
+ '[' 436 -gt 314 ']'
+ max=436
+ '[' 436 -lt 314 ']'
+ for num in `seq 5`
+ randomNo=337
+ '[' 337 -gt 436 ']'
+ '[' 337 -lt 314 ']'
+ for num in `seq 5`
+ randomNo=403
+ '[' 403 -gt 436 ']'
+ '[' 403 -lt 314 ']'
+ for num in `seq 5`
+ randomNo=930
+ '[' 930 -gt 436 ']'
+ max=930
+ '[' 930 -lt 314 ']'
+ echo 'Maximum: 930'
Maximum: 930
+ echo 'Minimum: 314'
Minimum: 314
```

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
if [ $2 = "March"  -a $1 -ge 20  ]
then
    echo "true"
elif [ $2 = "June" -a $1 -le 20 ]
then
    echo "true"
elif [ $2 = "April" -o $2 = "May" ]
then
    echo "true"
else
    echo "false"
fi
```

```
$ ./myScript22.sh 27 "May"
+ '[' May = March -a 27 -ge 20 ']'
+ '[' May = June -a 27 -le 20 ']'
+ '[' May = April -o May = May ']'
+ echo true
true
```

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 -x
read -p "Enter a year:" year
if [ $year -lt 1000 -o $year -gt 9999 ]
then
    echo "not a 4 digit year"
elif [ $(($year % 4)) -eq 0 -a $(($year % 100)) -ne 0  -o  $(($year % 400)) -eq>
then
    echo "$year is a Leap Year"
else
    echo "$year not a Leap Year"
fi
```

```
+ read -p 'Enter a year:' year
Enter a year:2018
```

```
+ '[' 2018 -lt 1000 -o 2018 -gt 9999 ']'
+ '[' 2 -eq 0 -a 18 -ne 0 -o 18 -eq 0 ']'
+ echo '2018 not a Leap Year'
2018 not a Leap Year
```

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

```
#!/bin/bash -x
echo "Flipping coin..."
toss=$((1+RANDOM%2))
if [ $toss -eq 1 ]
then
   echo "Heads"
else
   echo "Tails"
fi
```

```
+ echo 'Flipping coin...'
Flipping coin...
+ toss=2
+ '[' 2 -eq 1 ']'
+ echo Tails
Tails
```

# Selection Practice Problems with if, elif and else

1. Read a single digit number and write the number in word

```
#!/bin/bash -x
read -p "Enter the sigle digit no.: " n
if [ $n -eq 0 ]
then
   echo "Zero"
elif [ $n -eq 1 ]
then
   echo "One"
elif [ $n -eq 2 ]
then
   echo "Two"
elif [ $n -eq 3 ]
then
   echo "Three"
elif [ $n -eq 4 ]
then
   echo "Four"
elif [ $n -eq 5 ]
then
   echo "Five"
elif [ $n -eq 6 ]
then
   echo "Six"
elif [ $n -eq 7 ]
then
   echo "Seven"
elif [ $n -eq 8 ]
then
   echo "Eight"
elif [ $n -eq 9 ]
then
   echo "Nine"
else
   echo "Invalid no."
fi
```

```
+ read -p 'Enter the sigle digit no.: ' n
Enter the sigle digit no.: 7
+ '[' 7 -eq 0 ']'
+ '[' 7 -eq 1 ']'
+ '[' 7 -eq 2 ']'
+ '[' 7 -eq 3 ']'
+ '[' 7 -eq 4 ']'
+ '[' 7 -eq 5 ']'
+ '[' 7 -eq 6 ']'
+ '[' 7 -eq 7 ']'
+ echo Seven
Seven
```

2. Read a Number and Display the week day (Sunday, Monday,...)

```
#!/bin/bash -x
read -p "Enter the day no.: " n
if [ $n -eq 1 ]
then
   echo "Monday"
elif [ $n -eq 2 ]
then
   echo "Tuesday"
elif [ $n -eq 3 ]
then
   echo "Wednesday"
elif [ $n -eq 4 ]
then
   echo "Thursday"
elif [ $n -eq 5 ]
then
   echo "Friday"
elif [ $n -eq 6 ]
then
   echo "Saturday"
elif [ $n -eq 7 ]
then
   echo "Sunday"
else
   echo "Invalid no."
fi
```

```
+ read -p 'Enter the sigle digit no.: ' n
Enter the sigle digit no.: 3
+ '[' 3 -eq 0 ']'
+ '[' 3 -eq 1 ']'
+ '[' 3 -eq 2 ']'
+ '[' 3 -eq 3 ']'
+ echo Three
Three
```

3. Read a Number 1, 10, 100, 1000, etc and display unit, ten, hundred,...

```
#!/bin/bash -x
read -p "Enter the no.: " n
```

```
if [ $n -eq 1 ]
then
   echo "Unit"
elif [ $n -eq 10 ]
then
   echo "Ten"
elif [ $n -eq 100 ]
then
   echo "Hundred"
elif [ $n -eq 1000 ]
then
   echo "Thousand"
else
   echo "Invalid no."
fi
```

```
+ read -p 'Enter the no.: ' n
Enter the no.: 100
+ '[' 100 -eq 1 ']'
+ '[' 100 -eq 10 ']'
+ '[' 100 -eq 100 ']'
+ echo Hundred
Hundred
```

4. Enter 3 Numbers do following arithmetic operation and find the one that is maximum and minimum
   1. a + b * c 3. c + a / b
   2. a % b + c 4. a * b + c

```
#!/bin/bash -x
read -p "Enter a: " a
read -p "Enter b: " b
read -p "Enter c: " c
A=$((a + b * c))
B=$((a % b + c))
C=$((c + a / b))
D=$((a * b + c))
max=$A
min=$A
if [ $B -gt $max ]
then
   max=$B
fi
if [ $C -gt $max ]
then
   max=$C
fi
if [ $D -gt $max ]
then
   max=$D
```

```
fi
if [ $B -lt $min ]
then
    min=$B
fi
if [ $C -lt $min ]
then
    min=$C
fi
if [ $D -lt $min ]
then
    min=$D
fi
echo "Max: $max"
echo "Min: $min"
```

```
+ read -p 'Enter a: ' a
Enter a: 1
+ read -p 'Enter b: ' b
Enter b: 2
+ read -p 'Enter c: ' c
Enter c: 3
+ A=7
+ B=4
+ C=3
+ D=5
+ max=7
+ min=7
+ '[' 4 -gt 7 ']'
+ '[' 3 -gt 7 ']'
+ '[' 5 -gt 7 ']'
+ '[' 4 -lt 7 ']'
+ min=4
+ '[' 3 -lt 4 ']'
+ min=3
+ '[' 5 -lt 3 ']'
+ echo 'Max: 7'
Max: 7
+ echo 'Min: 3'
Min: 3
```

# Selection Practice Problems with case statement

1. Read a single digit number and write the number in word using Case

```
#!/bin/bash -x
read -p "Enter the single digit no.: " n
case $n 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"
   ;;
   *)
      echo "Invalid no."
   ;;
esac
```

```
+ read -p 'Enter the sigle digit no.: ' n
Enter the sigle digit no.: 5
+ case $n in
+ echo Five
Five
```

2. Read a Number and Display the week day (Sunday, Monday,…)

```
#!/bin/bash -x
read -p "Enter the day no.: " n
case $n in
  1)
    echo "Monday"
  ;;
  2)
    echo "Tuesday"
  ;;
  3)
    echo "Wednesday"
  ;;
  4)
    echo "Thursday"
  ;;
  5)
    echo "Friday"
  ;;
  6)
    echo "Saturday"
  ;;
  7)
    echo "Sunday"
  ;;
  *)
    echo "Invalid no."
  ;;
esac
```

```
+ read -p 'Enter the day no.: ' n
Enter the day no.: 3
+ case $n in
+ echo Wednesday
Wednesday
```

3. Read a Number 1, 10, 100, 1000, etc and display unit, ten, hundred,…

```
#!/bin/bash -x
read -p "Enter the no.: " n
case $n in
  1)
    echo "Unit"
  ;;
  10)
    echo "Ten"
  ;;
  100)
    echo "Hundred"
  ;;
  1000)
    echo "Thousand"
  ;;
  *)
    echo "Invalid no."
  ;;
esac
```

```
+ read -p 'Enter the no.: ' n
Enter the no.: 10
+ case $n in
+ echo Ten
Ten
```

4. Write a program that takes User Inputs and does Unit Conversion of different Length units
   1. Feet to Inch 3. Inch to Feet
   2. Feet to Meter 4. Meter to Feet

```
#!/bin/bash -x
echo "1. Feet to inch"
echo "2. Feet to Meter"
echo "3. Inch to Feet"
echo "4. Meter to Feet"
read -p "Enter your choice: " choice
case $choice in
    1)
        read -p "Enter in feet: " n
        res=$(($n*12))
        echo "Inches: $res"
    ;;
    2)
        read -p "Enter in feet: " n
        res=$(echo "scale=2;$n/3.281" | bc)
        echo "Meters: $res"
    ;;
    3)
        read -p "Enter in inch: " n
        res=$(echo "scale=2;$n/12" | bc)
        echo "Feet $res"
    ;;
    4)
        read -p "Enter in meter: " n
        res=$(echo "scale=2;$n*3.28084" | bc)
        echo "Feet $res"
    ;;
    *)
        echo "Invalid choice"
    ;;
esac
```

```
+ echo '1. Feet to inch'
1. Feet to inch
+ echo '2. Feet to Meter'
2. Feet to Meter
+ echo '3. Inch to Feet'
3. Inch to Feet
+ echo '4. Meter to Feet'
4. Meter to Feet
+ read -p 'Enter your choice: ' choice
Enter your choice: 3
+ case $choice in
+ read -p 'Enter in inch: ' n
Enter in inch: 3
++ echo 'scale=2;3/12'
++ bc
+ res=.25
+ echo 'Feet .25'
Feet .25
```