Extra Programs

Q. Write a Java program to print “CSE” 100 times.

```
package recursion;

public class Q2
{
    public void print(int n)
    {
          if(n>=1)
          {
                  System.out.println("CSE \t");
                  print(n-1);
          }
    }
    public static void main(String[] args)
    {
          Q2 ob = new Q2();
          ob.print(100);
    }
}
```

Q. Write a Java program to calculate the digit sum of a number.

```
package recursion;
import java.util.Scanner;

public class Q
{
    public static int digitSum(int n)
    {
          if(n==0)
                  return 0;
          else
                  return n%10 + digitSum(n/10);
    }
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number: ");
        int no = sc.nextInt();
        System.out.println("Digit sum is "+digitSum(no));
    }
}
```

Q. Write a Java program to calculate the sum of natural numbers.

```
package recursion;
import java.util.*;

public class Q
{
    public static int sumNaturalNumber(int n)
    {
        if(n==1)
            return 1;
        else
            return n+sumNaturalNumber(n-1);
  }
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the last terms of natural no.: ");
        int n = sc.nextInt();
        System.out.println("Sum: "+sumNaturalNumber(n));

    }
}
```

Q. Write a Java program to calculate the GCD of two numbers.

```
package recursion;
import java.util.*;

public class Q
```

```
{
        public static int GCD(int n,int m)
        {
                if(n==m)
                        return n;
                else if(n>m)
                        return GCD(n-m,m);
                else
                        return GCD(n,m-n);
   }
   public static void main(String[] args)
   {
         Scanner sc = new Scanner(System.in);
         System.out.println("Enter the first number: ");
         int n = sc.nextInt();
        System.out.println("Enter the second number: ");
         int m= sc.nextInt();
        System.out.println("GCD: "+GCD(n,m));
   }
}
```

Q. Write a Java program to calculate the Binary equivalent Decimal.

```
package recursion;
import java.util.*;

public class Q
{
        public static int BtoD(int n,int m)
        {
                if(n<10)
                        return n*(int)Math.pow(2, m);
                else
                        return (n%10)*(int)Math.pow(2, m)+BtoD(n/10,m+1);
        }
   public static void main(String[] args)
   {
         Scanner sc = new Scanner(System.in);
         System.out.println("Enter the binary  number: ");
         int n = sc.nextInt();
        System.out.println("Decimal: "+BtoD(n,0));
   }
}
```

Q. Write a Java program to calculate the value of ’a’ to the power of ’b’ using recursion.

```
package recursion;
import java.util.*;

public class Q
{
     public static int power(int n,int pow)
       {
              if(pow==1)
                     return n;
              else
                     return n*power(n,pow-1);
       }
  public static void main(String[] args)
  {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the base: ");
        int n = sc.nextInt();
       System.out.println("Enter the power: ");
        int power = sc.nextInt();
        System.out.println("power value is : "+power(n,power));

  }
}
```

######### using iteration ##########

Q. Given an unsorted array, A, of integers and an integer k, write a program in Java for rearranging the elements in A so that all elements less than or equal to k come before any elements larger than k.

```
class Q
{
    public static int[] swap(int arr[], int start, int end, int item)
    {
        while(start<=end)
        {
            while(arr[start]<=item)
                start++;
            while(arr[end]>item)
                end--;
            int temp = arr[start];
            arr[start] = arr[end];
            arr[end] = temp;
            start++;
            end--;
        }
        return arr;
    }
    public static void display(int arr[])
    {
        for(int i = 0;i<arr.length;i++)
        {
            System.out.print(arr[i]+" ");
        }
    }
    public static void main(String[] args)
    {
        int arr[]={3, 9, 4, 11, 5, 1, 7};
        arr = swap(arr,0,arr.length-1,5);
        display(arr);
    }
}
```

Q. Write a java generics method that accepts an array of any data type and returns the reverse of the.

```
class Q
{
    public static <T> T[] reverse(T[] arr)
    {
        T temp;
         int n = arr.length;
        for(int i = 0;i<n/2;i++)
        {
            temp = arr[i];
            arr[i] = arr[n-1-i];
            arr[n-1-i] = temp;
        }
        return arr;
    }
    public static <T>void display(T[] arr)
    {
        for(int i = 0;i<arr.length;i++)
        {
            System.out.print(arr[i]+"  ");
        }
    }
    public static void main(String[] args)
    {
        Integer[] a = {10,20,30,40,50,20,10,20};
        System.out.print("Original Array:  ");
        a = reverse(a);
        System.out.print("\nReverse Array:  ");
        display(a);
        System.out.print("Original Array:  ");
        String[] b = {"IGT","DSA","UPM","Calculus","DSA"};
        b = reverse(b);
        System.out.print("\nReverse Array:  ");
         display(b);
    }
}
```