Sum of Even Numbers Till N

```
import java.util.* ;
import java.io.*;
public class Solution {
    public static long evenSumTillN(int n) {
        // Write your code here.
        long num = 0;
        for(int i=0; i<=n; i=i+2)
        {
            num = num + i;
        }
        return num;
    }
}
```

Fahrenheit to Celsius

```
import java.util.* ;
import java.io.*;
class Solution {

    public static void main(String args[]) {

        // Write code here
        Scanner user_input = new Scanner(System.in);
        int start = user_input.nextInt();
        int end = user_input.nextInt();
        int gap = user_input.nextInt();
        for(int i=start; i<=end; i=i+gap)
        {
            System.out.println(i + " " + ((i-32)*5)/9 + "");
        }
    }
}
```

Sum of even & odd

```
import java.util.*;
class Solution {
```

```
    public static void main(String args[]) {

        // Write code here
        Scanner user_input = new Scanner(System.in);
        int N = user_input.nextInt();
        int even = 0;
        int odd = 0;
        int temp = 0;
        while(N>0)
        {
            temp = N%10;
            N =N/10;
            if(temp%2==0)
            even = even + temp;
            else
            odd = odd + temp;
        }
        System.out.println(even + " " + odd + "");
    }
}
```

Find power of a number

```
import java.util.* ;
import java.io.*;
class Solution {

    public static void main(String args[]) {

        // Write code here
        Scanner user_input = new Scanner(System.in);
        int num = user_input.nextInt();
        int power = user_input.nextInt();
        int ans = 1;
        if(power==0)
        ans = 1;
        else
        for(int i=1; i<=power; i++)
        {
            ans = ans*num;
        }
        System.out.println((int)ans);
    }
```

```
}
```

Factorial of a Number

```
import java.util.*;
class Solution {

    public static void main(String args[]) {

        Scanner user_input = new Scanner(System.in);
        long num = user_input.nextLong();
        if(num==0 || num==1)
        num = 1;
        else
        for(long i=num-1; i>0; i--)
        {
            num = num*i;
        }
        System.out.println(num);

    }
}
```

N-th Fibonacci Number

```
import java.util.*;
import java.io.*;

public class Solution {

   public static long[][] multiply(long[][] a, long[][] b, long mod)

    {

        int n = a.length;

        int m = a[0].length;

        int p = b[0].length;

        long[][] c = new long[n][p];

        for (int i = 0; i < n; i++) {
```

```
      for (int j = 0; j < p; j++) {
       for (int k = 0; k < m; k++) {
       c[i][j] = (c[i][j] + (a[i][k] * b[k][j]) % mod) % mod;
      }
      }
      }
     return c;
      }
public static long[][] power(long[][] base, long exp, long mod) {
 if (exp == 0) {
       int n = base.length;
       long[][] ans = new long[n][n];
       for (int i = 0; i < n; i++)
       {
           ans[i][i] = 1;
       }
       return ans;
    }
     long[][] res = power(base, exp / 2, mod);
      res = multiply(res, res, mod);
     if (exp % 2 != 0) {
```

```
        res = multiply(res, base, mod);

          }

        return res;

           }

    public static int fibonacciNumber(int n)

     {

          long mod = (long) 1e9 + 7;

         long[][] dp = {{1, 1}, {1, 0}};

          long[][] ans = power(dp, n - 1, mod);

        return (int) (ans[0][0] % mod);

        }

         }
```