**Coding Question 1**

The function ***def differenceofSum(n. m)*** accepts two integers n, m as arguments Find the sum of all numbers in range from 1 to m(both inclusive) that are not divisible by n. Return difference between sum of integers not divisible by n with sum of numbers divisible by n.

**Assumption:**

* n>0 and m>0
* Sum lies between integral range

**Example**

***Input***n:4  
m:20  
***Output***90

Solution:

import java.util.\*;

class HelloWorld {

public static int differenceOf(int n, int m)

{

int sum1 = 0, sum2 = 0;

for(int i = 1; i <= m ; i++)

{

if(i % n == 0)

{

sum1 += i;

}

else

{

sum2 += i;

}

}

int difference = sum2 - sum1;

return difference;

}

public static void main(String[] args) {

Scanner read = new Scanner(System.in);

System.out.println("Enter n : ");

int n = read.nextInt();

System.out.println("Enter m : ");

int m = read.nextInt();

System.out.println(differenceOf(n, m));

}

}

**Coding Question 2**

You are required to implement the following Function ***def LargeSmallSum(arr).***

The function accepts an integers arr of size ’length’ as its arguments you are required to return the sum of second largest largest element from the even positions and second smallest from the odd position of given ‘arr’.

**Assumption:**

* All array elements are unique
* Treat the 0th position a seven

**NOTE**

* Return 0 if array is empty
* Return 0, if array length is 3 or less than 3

**Example:-**

***Input***

arr:3 2 1 7 5 4

***Output***

7

Solution:

import java.util.\*;

class HelloWorld {

public static void main(String[] args) {

Scanner read = new Scanner(System.in);

System.out.println("Enter size of the array : ");

int n = read.nextInt();

int[] arr = new int[n];

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

{

arr[i] = read.nextInt();

}

System.out.println(""+LargeSmallSum(n, arr));

}

static int LargeSmallSum(int n, int arr[])

{

ArrayList<Integer> al1 = new ArrayList<>();

ArrayList<Integer> al2 = new ArrayList<>();

if(arr.length <= 3)

{

return 0;

}

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

{

if(i % 2 == 0)

{

al1.add(arr[i]);

}

else

{

al2.add(arr[i]);

}

}

Collections.sort(al1);

Collections.sort(al2);

int size1 = al1.size();// to get the size of arraylist 1

int size2 = al2.size();

int evenel = al1.get(size1 - 2);

int oddel = al2.get(size2 - 2);

int add = evenel + oddel;

return add;

}

}