knóldus

Count Divisor

knólˣ

Question

1. You have been given 3 integers: A, B and C. Find how many numbers between A and B (both inclusive) are divisible by C. You do not need to print these numbers, you just have to find their count.

Sample Input: A = 1, B = 10, C = 2
Sample Output: 5

knóldus

Program

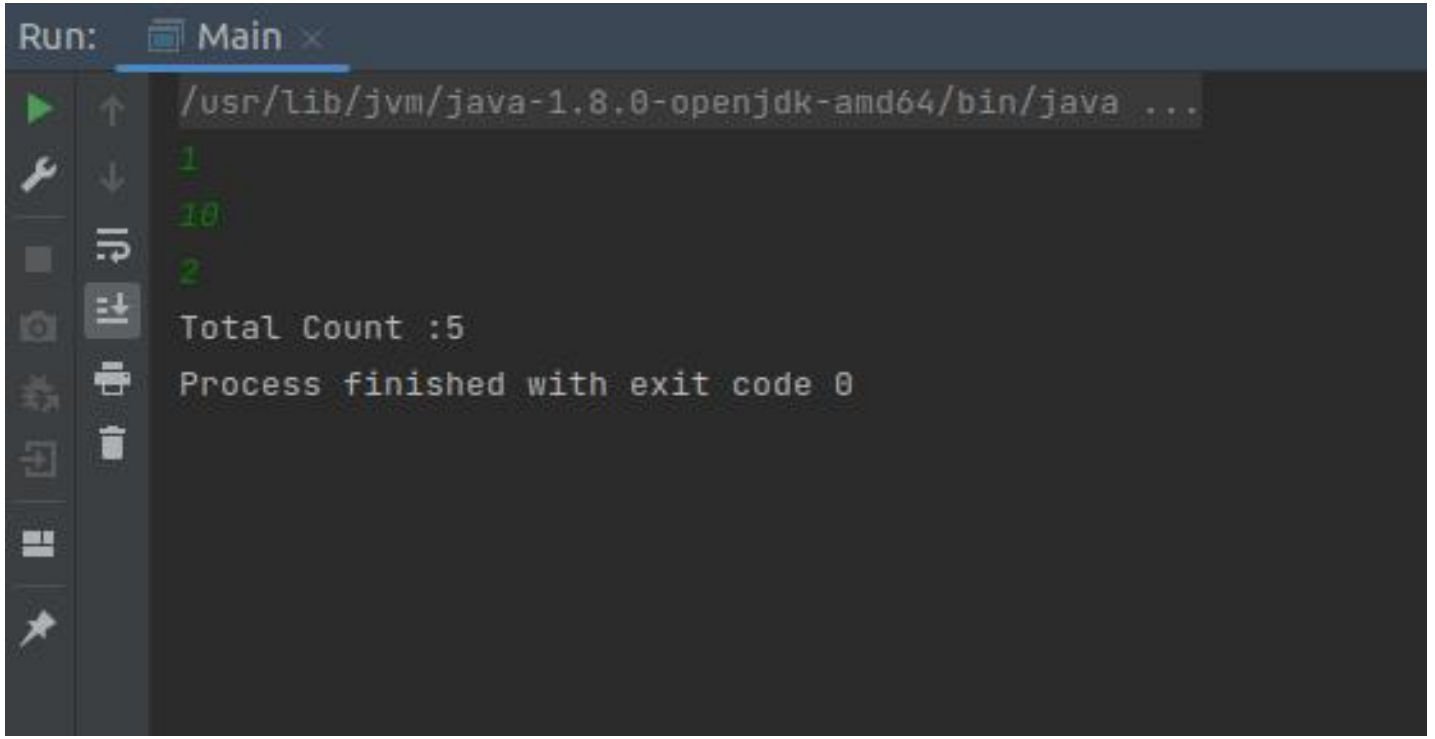
Main.scala | MainTest.scala | build.sbt | build.properties

```scala
import scala.io.StdIn.readInt
 Manish Mishra *
class Count{
  //countDivisor Method to count the value and return
  Manish Mishra *
  def countDivisor(firstNum: Int, secondNum: Int, thirdNum: Int): Int = {
    var count: Int = 0
    for(index <- firstNum ≤ to ≤ secondNum){
      if(index % thirdNum == 0){
        count+=1
      }
    }
    return count
  }
}
```

Program

```
14  //singleton object
    Manish Mishra *
15  object Main extends App{
16    private val obj = new Count
17    private val firstNum: Int = readInt()
18    private val secondNum: Int = readInt()
19    private val thirdNum: Int = readInt()
20    //try catch exception handling
21    try{
22      if(thirdNum == 0){
23        throw new ArithmeticException
24      }
25      else{
26        val count: Int = obj.countDivisor(firstNum, secondNum, thirdNum)
27        print("Total Count :" + count)
28      }
29    }
30    catch{
31      case ex: ArithmeticException =>{
32        print(" Divide by Zero Exception [Divisor Cannot be Zero]")
33      }
34    }
35  }
36
```

Output


Run:
Main
/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...
1
10
2
Total Count :5
Process finished with exit code 0

knóldus

THANK YOU

knólˣ