Ron Eliav

Divisors

```java
/**
 *  Gets a command-line argument (int), and prints all the divisors
of the given number.
 */
public class Divisors {
    public static void main (String[] args) {
        int num = Integer.parseInt(args[0]); // gets a number from
the user
        for(int i = 1; i <= num; i++){ // the loop passes all the
numbers between 1 - the number that the user gave, and print all the
divisors.
            if(num % i == 0){ // check if i+1 is a divisor of the
number that the user gave.
                System.out.println(i);

            }
        }

        }
}
```

Reverse

```
/**
 * Prints a given string, backward. Then prints the middle character
in the string.
 * The program expects to get one command-line argument: A string.
 */
public class Reverse {
    public static void main (String[] args) {
        String userString = args[0];    // Gets a number from the
user
        int wordLength = userString.length(); // wordLength contain
the String Length
        for(int i = wordLength; i > 0; i--) {  // The loop passes on
all the String array and print the string in revearse order.
            System.out.print(userString.charAt(i - 1));
        }

        System.out.println(); // print the new line
        if(wordLength % 2 == 1){ // If the string is odd print the
middle char
            System.out.println(userString.charAt(wordLength / 2));
        } else {    // If the string is even - print the first
middle char
            System.out.println(userString.charAt((wordLength / 2) -
1));

        }


    }
}
```

InOrder

```java
/**
 *  Generates and prints random integers in the range [0,10),
 *  as long as they form a non-decreasing sequence.
 */
public class InOrder {
    public static void main (String[] args) {
        int previoustNum = (int)(Math.random() * 10); // Generates
the previoust number beteen 0 to 9
        System.out.print(previoustNum + " "); // print the previoust
number
        boolean loopBool = true; // Boolean variable that indicates
the continuity of the loop
        do{
            int currentNum = (int)(Math.random() * 10); // Generates
a new number between 0 to 9
            if(currentNum >= previoustNum) { // check if the new
number is bigger or equals to the current number
                System.out.print(currentNum + " "); // print the new
number
                previoustNum = currentNum;
            } else {
                loopBool = false;
            }

        } while(loopBool);

    }
}
```

DamkaBoard

```java
/**
 *  Gets a command-line argument n (int), and prints an n-by-n damka board.
 */
public class DamkaBoard {
    public static void main(String[] args) {
        int n = Integer.parseInt(args[0]); // Gets a number from the user
        for(int i = 0; i < n; i++) { // The loop print * or space relate to the number the user gave.
            for(int j = 0; j < (n * 2); j++ ) { // In Each lime the loop print n timed '*', and n times ' '
                if((i + j) % 2 == 0){
                    System.out.print("*");
                } else {
                    System.out.print(" ");
                }
            }
            System.out.println();
        }
    }
}
```

Perfect

```java
/**
 *  Gets a command-line argument (int), and chekcs if the given
number is perfect.
 */
public class Perfect {
    public static void main (String[] args) {
        int userNum = Integer.parseInt(args[0]); // Gest a number
from the user
        int sumOfDivisors = 1; // Create a variable that save the
sum of the divisors
        String printIfTheNumIsPerfect = userNum + " is a perfect
number since " + userNum + " = 1";

        for(int i = 2; i < userNum; i++ ) { // The loop search on
the divisors of the number, calculates the sum into a variable
            if(userNum % i == 0) {          // and adds the number
into the String that will be printed in the end of the program
                sumOfDivisors += i;
                printIfTheNumIsPerfect = printIfTheNumIsPerfect + "
+ "+ i;
            }
        }
        if(sumOfDivisors == userNum){ // Check if the sum of the
divisors is equal to the first numbrt
            System.out.println(printIfTheNumIsPerfect);
        } else {
            System.out.println(userNum + " is not a perfect
number");
        }
    }
}
```

OneOfEachStats

```java
import java.util.Random;

public class OneOfEachStats {
    public static void main (String[] args) {
        // Gets the two command-line arguments
        int T = Integer.parseInt(args[0]);
        int seed = Integer.parseInt(args[1]);
        // Initailizes a random numbers generator with the given seed value
        Random generator = new Random(seed);
        int countSumOfAllChildren = 0;
        int family2 = 0;
        int family3 = 0;
        int family4 = 0;

        for(int i = 0; i < T; i ++) {
            boolean isBoy = false;
            boolean isGirl = false;
            int countChildren = 0; // count the sum of the children.

            while(!isBoy || !isGirl) { // The loop check if the next children
is the same like before. If so, she continues if not ends the loop.
                double rnd = generator.nextDouble();
                if(rnd < 0.5) {
                    isGirl = true;
                } else {
                    isBoy = true;
                }
                countChildren++;
            }
            countSumOfAllChildren += countChildren; //Adds the quantity of the
current family to the sum of all families.
            if(countChildren == 2) {
                family2++;
            } else {
                if(countChildren == 3){
                    family3++;
                } else {
                    family4++;
                }
            }
        }
        System.out.println("Average: " + (double)countSumOfAllChildren / T + "
children to get at least one of each gender.");
        System.out.println("Number of families with 2 children: " + family2);
        System.out.println("Number of families with 3 children: " + family3);
        System.out.println("Number of families with 4 or more children: " +
family4);
```

```
        int max = Math.max(Math.max(family2, family3), family4);
        int average = 0;
        if(max == family2){
            average = 2;
        } else {
            if(max == family3) {
                average = 3;
            } else {
                average = 4;
            }
        }
        System.out.println("The most common number of children is " + average
+ ".");
    }
}
```