```
public class Divisors {
      public static void main (String[] args) {
           int x = Integer.parseInt(args[0]);
           int d = x;

           while (0 < d){
               if ((x % d) == 0){
                    System.out.println(x / d);
                 }
                 d = d - 1;
           }
      }
}
```

```
public class Reverse {
    public static void main (String[] args){
        String s = args[0];
        String sOut = "";
        int i = (s.length() - 1);
        int m = (i / 2);


        while (i >= 0){
            char c = s.charAt(i);
            sOut = sOut + c;
            i = (i - 1);
        }
        System.out.println(sOut);
        System.out.println("The middle character is " + s.charAt(m));


    }
}
```

```
public class InOrder {

        public static void main (String[] args) {

                int  i = (int)(10 * Math.random());

     System.out.println();

     int newnum;


     do{

      System.out.print(i + " ");

      newnum = (int)(10 * Math.random());

      if (newnum >= i){

        i = newnum;

      }


    }while(newnum >= i);


  }



}
```

```
public class DamkaBoard {

    public static void main(String[] args) {

        int n = Integer.parseInt(args[0]);



        for (int i = 1;i <= n; i++) { //prints number of rows until n


            for (int j = 1;j < n; j++){ // prints the row until j=n

                if(i % 2 == 0){

                    System.out.print(" *");

                }else {

                    System.out.print("* ");

                }



            } System.out.println();



        }

    }

}
```

```
public class Perfect {

    public static void main (String[] args) {

        int N = Integer.parseInt(args[0]);

        int d = N - 1;

        String s = N + " is a perfect number since " + N + " = " + 1;

        int sum = 1;


        while (1 < d){

            if ((N % d) == 0){

                s = s + " + " + (N / d);

                sum = sum + (N / d);

            }d = d - 1;

        }


        if (sum == N){

            System.out.println(s);

        } else {

            System.out.println(N + " is not a perfect number");

        }





    }

}
```

```
import java.util.Random;

public class OneOfEachStats {
        public static void main (String[] args) {
                int T = Integer.parseInt(args[0]);
                int seed = Integer.parseInt(args[1]);
    int count;
    int countT = 0;
    int counttwo = 0;
    int countthree = 0;
    int countfour = 0;

    // Set a fixed seed to make random numbers predictable
    Random generetor = new Random(seed);

    for (int rep = 0; T > rep; rep++) {
      count = 1;
      int boy = 0;
      int girl;
      girl = boy;


      while ( (girl<=0) || (boy<=0)) {
        double check = generetor.nextDouble();
        if (check > 0.5){
          boy++;
        }else{
           girl++;
        }
      }
      count = boy + girl;


      if (count == 2) {
        counttwo++;
      } else if (count == 3) {
        countthree++;
      } else {
        countfour++;
      }
      countT += count;


    }
    double avg = ((double) countT) /((double) T);
    System.out.println("Average: " + avg + " children to get at least one of each gender.");
    System.out.println("number of families with 2 children: " + counttwo);
    System.out.println("number of families with 3 children:  " + countthree);
    System.out.println("number of families with 4 or more children:  " + countfour);

    int common = Math.max(Math.max(counttwo, countthree), countfour);
    String commonstr;

    if (common == counttwo) {
      commonstr = "2";
    } else if (common == countthree) {
      commonstr = "3";
```

```
        } else {
            commonstr = "4 or more";
        }
        System.out.println("The most common number of children is " + commonstr);
    }
}
```