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

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

            for (int i = 1; i <= number; i++) {
                  if (number % i == 0) {
                        System.out.println(i);
                  }
            }

      }
}
```

```java
public class Reverse {
    public static void main (String[] args){
        String str = args[0];

        for (int i = str.length() - 1; i >= 0; i--) {
            System.out.print(str.charAt(i));
        }

        System.out.println();

        // if str.length-1 is odd - making sure the "first middle" char will be printed
        int middleCharIndex = (int)(Math.floor((str.length() - 1)/ 2));

        System.out.println("The middle character is " + str.charAt(middleCharIndex));

    }
}
```

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

        int num = (int)(Math.random() * 10);
        int temp = 0; // will storage the old value of num

        do {
            System.out.print(num + " ");
            temp = num;
            num = (int)(Math.random() * 10);
        }
        while (num >= temp);

    }
}
```

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

        int N = Integer.parseInt(args[0]);
        String isPerfect = N + " is a perfect number since " + N
+ " = 1";
        int check = 1; // to validate later if N is indeed
perfect. 1 is already a divisor

        for (int i = 2; i < N; i++) {
            if (N % i == 0) {
                isPerfect = isPerfect + " + " + i;
                check += i;
            }
        }

        if (check == N) {
            System.out.println(isPerfect);
        } else {
            System.out.println(N + " is not a perfect number");
        }

    }
}
```

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

        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                    if ((i+1) % 2 != 0) { // for odd rows, the space is after the *. started from row 1
                        System.out.print("* ");
                    } else {
                        System.out.print(" *");
                    }
            }

        System.out.println();

        }

    }
}
```

```
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

        double totalChildren = 0.0; // To calculate the avg later

        int twoChildrenFamilies = 0;
        int threeChildrenFamilies = 0;
        int fourAndMoreChildrenFamilies = 0;
        int mostCommonNumberOFChildren = 0;
        Random generator = new Random(seed);

        boolean isGirl = true;
        boolean isBoy = true;
        int childCountForOneFamily = 0;

        for (int i = 0; i < T; i++) {

            isBoy = true;
            isGirl = true;

            while (isGirl || isBoy) {

                double probabilty = generator.nextDouble();

                if (probabilty >= 0.5) {
                    isBoy = false;
                    // System.out.print("g ");
                } else {
                    isGirl = false;
                    // System.out.print("b ");

                }

                childCountForOneFamily++;

            }

            totalChildren += childCountForOneFamily;

            if (childCountForOneFamily == 2) {
                twoChildrenFamilies++;
            } else if (childCountForOneFamily == 3) {
                threeChildrenFamilies++;
            } else if (childCountForOneFamily >= 4) {
                fourAndMoreChildrenFamilies++;
            }

            childCountForOneFamily = 0; // Must initialized at
the end of each test

        }

        double avgChildrenInAllFamilies = totalChildren / T;
        System.out.println("Average: " + avgChildrenInAllFamilies
+ " children to get at least one of each gender.");
```

```
        System.out.println("Number of families with 2 children: "
+ twoChildrenFamilies);

        System.out.println("Number of families with 3 children: "
+ threeChildrenFamilies);

        System.out.println("Number of families with 4 or more
children: " + fourAndMoreChildrenFamilies);

        String commonNumber; // Temp variable
        if (twoChildrenFamilies >= threeChildrenFamilies) {
            if (twoChildrenFamilies >=
fourAndMoreChildrenFamilies) {
                commonNumber = "2";
            } else {
                commonNumber = "4 or more";
            }
        } else {
            if (threeChildrenFamilies >=
fourAndMoreChildrenFamilies) {
                commonNumber = "3";
            } else {
                commonNumber = "4 or more";
            }
        }

        System.out.println("The most common number of children is
" + commonNumber + ". ");



    }
}
```