

## Divisors

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

        while (i<x)
        {
            int d = i+1;
            if (x%d == 0)
            {
                System.out.println(i+1);
                i ++;
            }
            else
                { i++; }
        }
    }
}
```

## Reverse

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

    {
        String s = args[0];
        int l = s.length()-1;

        while ( l >= 0)
        {
            System.out.print(s.charAt(l));

            l--;

        }

        int x = s.length();
        System.out.println("");

        if ((x%2)==0)
        {
            System.out.println("The middle character is " +
s.charAt((x/2)-1));
        }

        else
        {
            System.out.println("The middle character is " +
s.charAt((x-1)/2));
        }
    }
}
```

InOrder

```
public class InOrder {
    public static void main (String[] args)
    {
        int a1 = ((int)(Math.random() * 10));
        int a2 = ((int)(Math.random() * 10));

        if(a1>=a2)
        {
            System.out.print(a1);

        }

        else
        {
            System.out.print(a1 + " " + a2);

        }

        int a3 = ((int)(Math.random() * 10));
        if (a3>=a2)
        {
            System.out.print(" " + a3);
            a2=a3;
        }

        a3 = ((int)(Math.random() * 10));
        while (a3>=a2)
        {
            {
                System.out.print(" " + a3);
                a2=a3;
            }
            a3= ((int)(Math.random() * 10));
        }

    }
}
```

DamkaBoard

```
public class DamkaBoard {
    public static void main(String[] args)
    {
        int n = Integer.parseInt(args[0]);
        int i = 1;

        while (i<=n)
        {
            int j = 0;

            if((i%2)!=0)
            {
                while (j<n)
                {
                    System.out.print("* ");
                    j ++;
                }
                System.out.println();
                i++;
            }

            else
            {
                while (j<n)
                {
                    System.out.print(" *");
                    j ++;
                }
                System.out.println();
                i++;
            }
        }
    }
}
```

Perfect

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

        int i = 2;
        int x = 1;
        String s = ("1");

        while (i<n)
        {
            if((n%i) == 0)
            {
                s = (s + " + " + i);
                x = x + i;
            }

            i ++;
        }

        if((x==n)&&(n!=0)&&(n!=1))
        {
            System.out.print(n + " is a perfect number since " + n
+ " = " + s);
        }

        else
        {
            System.out.print(n + " is not a perfect number");
        }

    }
}
```

## OneOfEachStats

```
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 i = 0;
        int masterSum = 0;
        double avg = 0;
        int f2 = 0;
        int f3 = 0;
        int f4 = 0;

        while (i< T)
        {
            int x1 = ((int)(Math.random()*2));
            int x2 = x1;
            int sum = 1;

            while (x1==x2)
            {
                x2= ((int)(Math.random()*2));
                sum ++;
            }

            masterSum = (masterSum + sum);
            if (sum == 2)
                {f2++;}
            if (sum == 3)
                {f3++;}
            if (sum >= 4)
                {f4++;}

            i++;
        }
        avg = (masterSum/T);

        System.out.println("Average: " + avg + " children to get at
least on of each gender.");
    }
}
```

```

        System.out.println("Number of families with 2 children: " +
f2);
        System.out.println("Number of families with 3 children: " +
f3);
        System.out.println("Number of families with 4 or more
children: " + f4);

        if((f2>f3)&&(f2>f4))
        {System.out.println("The most common number of children is
2.");}

        if((f3>f2)&&(f3>f4))
        {System.out.println("The most common number of children is
3.");}

        if((f4>f2)&&(f4>f3))
        {System.out.println("The most common number of children is 4
or more.");}

    }
}

```