

1. Divisors

```
public class Divisors {  
    public static void main (String[] args) {  
        int x = Integer.parseInt(args[0]);  
        for (int i = 1; i <= x; i++) {  
            if (x%i==0) {  
                System.out.println(i);  
            }  
        }  
    }  
}
```

2. Reverse

```
public class Reverse {  
    public static void main (String[] args){  
        String s = args[0];  
        int i = 0;  
        int n = s.length();  
        String reverse = "";  
        char m = s.charAt((n-1)/2);  
        while (i<n) {  
            char c = s.charAt(n-1-i);  
            reverse = reverse + c;  
            i++;  
        }  
        System.out.println(reverse);  
        System.out.println("The middle character is " + m);  
    }  
}
```

3. InOrder

```
public class InOrder {  
    public static void main (String[] args) {  
        int x = (int)(Math.random() * 10);  
        int i = 0;  
        do {  
            i=x;  
            String z = x + " ";  
            x = (int)(Math.random() * 10);  
            System.out.print(z);  
  
        } while (i<=x);  
    }  
}
```

4. DamkaBoard

```
public class DamkaBoard {
    public static void main(String[] args) {
        int n = Integer.parseInt(args[0]);
        int i = 0;
        while (i < n) {
            int j = 0;
            while (j < n) {
                if (i % 2 == 1) {
                    System.out.print(" *");
                } else {
                    System.out.print("* ");
                }
                j++;
            }
            System.out.println();
            i++;
        }
    }
}
```

5. Perfect

```
public class Perfect {  
    public static void main (String[] args) {  
        int N = Integer.parseInt(args[0]);  
        int z=0;  
        String s = "1";  
        for (int i = 1; i<N; i++) {  
            if (N%i==0) {  
                z = z + i;  
                if (i!=1) {  
                    s = s + " + " + i;  
                }  
            }  
        }  
        if (N==z) {  
            System.out.println(N + " is a perfect number since " + N + " = " + s);  
        }else {  
            System.out.println(N + " is not a perfect number");  
        }  
    }  
}
```

6. OneOfEachStats

```
import java.util.Random;
/**
 * Computes some statistics about families in which the parents decide
 * to have children until they have at least one child of each gender.
 * The program expects to get two command-line arguments: an int value
 * that determines how many families to simulate, and an int value
 * that serves as the seed of the random numbers generated by the program.
 * Example usage: % java OneOfEachStats 1000 1
 */
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]);
        // Initializes a random numbers generator with the given seed value
        Random generator = new Random(seed);
        int two = 0;
        int three = 0;
        int four = 0;
        int total = 0;

        for (int i=0; i<T;i++) {
            Boolean g = true;
            Boolean b = true;
            int n = 0;

            while ( b || g) {
                double x = generator.nextDouble();
                if (x>=0.5) {
                    b = false;
                } if (x<0.5) {
                    g = false;
                }
                n++;
                total++;
                if ((n==2) && (g==false) && (b==false)) {
                    two++;
                } else if ((n==3) && (g==false) && (b==false)) {
                    three++;
                } else if ((n==4) && (g==false) && (b==false)) {
                    four++;
                }
            }
        }
    }
}
```

```

    }
    }
    }
    double average = (double) total/T;
    System.out.println("Average: " + average + " children to get at least one of each
gender." );
    System.out.println("Number of families with 2 children: " + two);
    System.out.println("Number of families with 3 children: " + three);
    System.out.println("Number of families with 4 or more children: " + four);
    if (two>=three && two>=four) {
        System.out.println("The most common number of children is 2.");
    } else if (three>two && three>=four) {
        System.out.println("The most common number of children is 3.");
    } else {
        System.out.println("The most common number of children is 4 or more.");
    }

    //// In the previous version of this program, you used a statement like:
    //// double rnd = Math.random();
    //// Where "rnd" is the variable that stores the generated random value.
    //// In this version of the program, replace this statement with:
    //// double rnd = generator.nextDouble();
    //// This statement will generate a random value in the range [0,1),
    //// just like you had in the previous version, except that the
    //// randomization will be based on the given seed.
    //// This is the only change that you have to do in the program.

    }
}

```