



The image shows a screenshot of a Java IDE window. The title bar at the top has three colored buttons (red, yellow, green) and the text "untit". Below the title bar is a tab bar with three tabs: "Perfect.java", "OneOfEach.java", and "s.java". The "Perfect.java" tab is active. The code editor displays the following Java code:

```
1 public class Divisors {
2     public static void main(String[] args)
3     {
4         int a = Integer.parseInt(args[0]);
5         for (int i = 1; i <= a; i++) {
6             if ( (a % i) == 0){
7                 System.out.println(i);
8             }
9         }
10    }
11 }
12 }
13 }
```

The code is a Java program that finds and prints all divisors of a given number 'a'. The number 'a' is passed as a command-line argument. The program uses a for loop to iterate from 1 to 'a', and an if statement to check if 'a' is divisible by 'i'. If it is, 'i' is printed.

```
public class InOrder {  
    public static void main(String[] args) {  
  
        int randFirst = (int)(Math.random()*(10));  
        int randSecond = 0;  
  
        while (randFirst >= randSecond){  
  
            System.out.println(randFirst);  
            randSecond = randFirst;  
            randFirst = (int)(Math.random()*(10));  
        }  
    }  
}
```

```
public class Perfect {  
    public static void main(String[] args) {  
  
        int a = Integer.parseInt(args[0]);  
        int sum = 0;  
        String action = "";  
        String fin = "";  
  
        for (int i = 1; i < a; i++) {  
            if ( (a % i) == 0){  
                sum = sum + i;  
                action = action + i + " + " ;  
            }  
        }  
  
        if (sum == a) {  
  
            // remove of the last "+"  
  
            for (int i = 0 ; i < action.length() - 2; i++) {  
  
                fin = fin + action.charAt(i);  
  
            }  
  
            System.out.println( a + " is a perfect number since " + a + " = " + fin);  
  
        }  
        else {  
            System.out.println( a + " is not a perfect number");  
        }  
    }  
}
```

```

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

        String a = (args[0]);
        String aRvs = "";
        String middle = "";

        for (int i = (a.length() - 1); i >= 0; i--) {

            if (a.length() % 2 == 0) {

                if ( i == (a.length() / 2) - 1 ) {
                    middle = middle + a.charAt(i);

                }

            }
            else {

                if ( i == (a.length() / 2) ) {
                    middle = middle + a.charAt(i);

                }

            }

            char b = a.charAt(i);
            aRvs = aRvs + b;

        }

        System.out.println(aRvs);
        System.out.println("The middle character is " + middle);

    }
}

```

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

```
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]);
        Random generator = new Random (seed);

        int families2 = 0;
        int families3 = 0;
        int families4 = 0;
        int child = 0;
        int common = 0;
        int avg = 0;

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

            child = 1;

            double first = generator.nextDouble();
            double gender = first;

            while ((int)(gender*2) == (int)(first*2)) {

                gender = generator.nextDouble();

                child ++;
            }
            if (child == 2){
                families2 ++;
            }
            else if (child == 3){
                families3 ++;
            }
            else {
                families4 ++;
            }
            avg = avg + child;
        }
    }
}
```

```

        while ((int)(gender*2) == (int)(first*2)) {
            gender = generator.nextDouble();
            child ++;
        }
        if (child == 2){
            families2 ++;
        }
        else if (child == 3){
            families3 ++;
        }
        else {
            families4 ++;
        }
        avg = avg + child;
    }

    double avarege = ((double)(avg) / (double) (7));
    System.out.println("Average: " + avarege + " children to get at least one of each gender.");
    System.out.println("Number of families with 2 children: " + families2);
    System.out.println("Number of families with 3 children: " + families3);
    System.out.println("Number of families with 4 or more children: " + families4);

    if ((families2 > families3)&&(families2 > families4)) {
        System.out.println("The most common number of children is 2.");
    }
    else if ((families3 > families2)&&(families3 > families4)) {
        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.");
    }
}
}

```