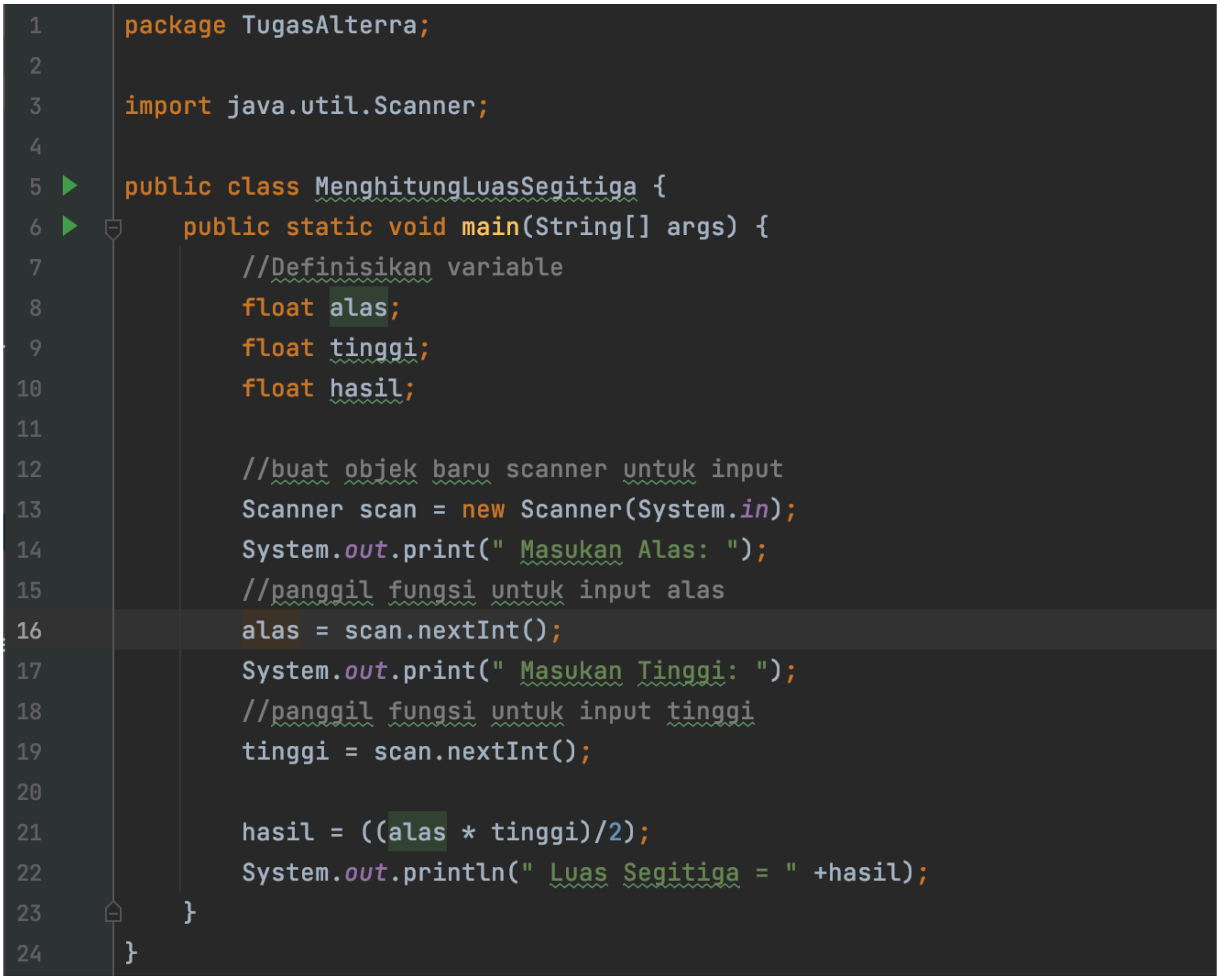
Java Programming

PART 1

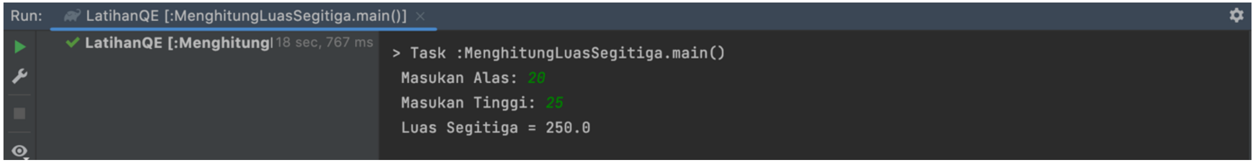
Problem 1: Menghitung Luas Segitiga

Input :



```java
package TugasAlterra;

import java.util.Scanner;

public class MenghitungLuasSegitiga {
    public static void main(String[] args) {
        //Definisikan variable
        float alas;
        float tinggi;
        float hasil;

        //buat objek baru scanner untuk input
        Scanner scan = new Scanner(System.in);
        System.out.print(" Masukan Alas: ");
        //panggil fungsi untuk input alas
        alas = scan.nextInt();
        System.out.print(" Masukan Tinggi: ");
        //panggil fungsi untuk input tinggi
        tinggi = scan.nextInt();

        hasil = ((alas * tinggi)/2);
        System.out.println(" Luas Segitiga = " +hasil);
    }
}
```

Output:

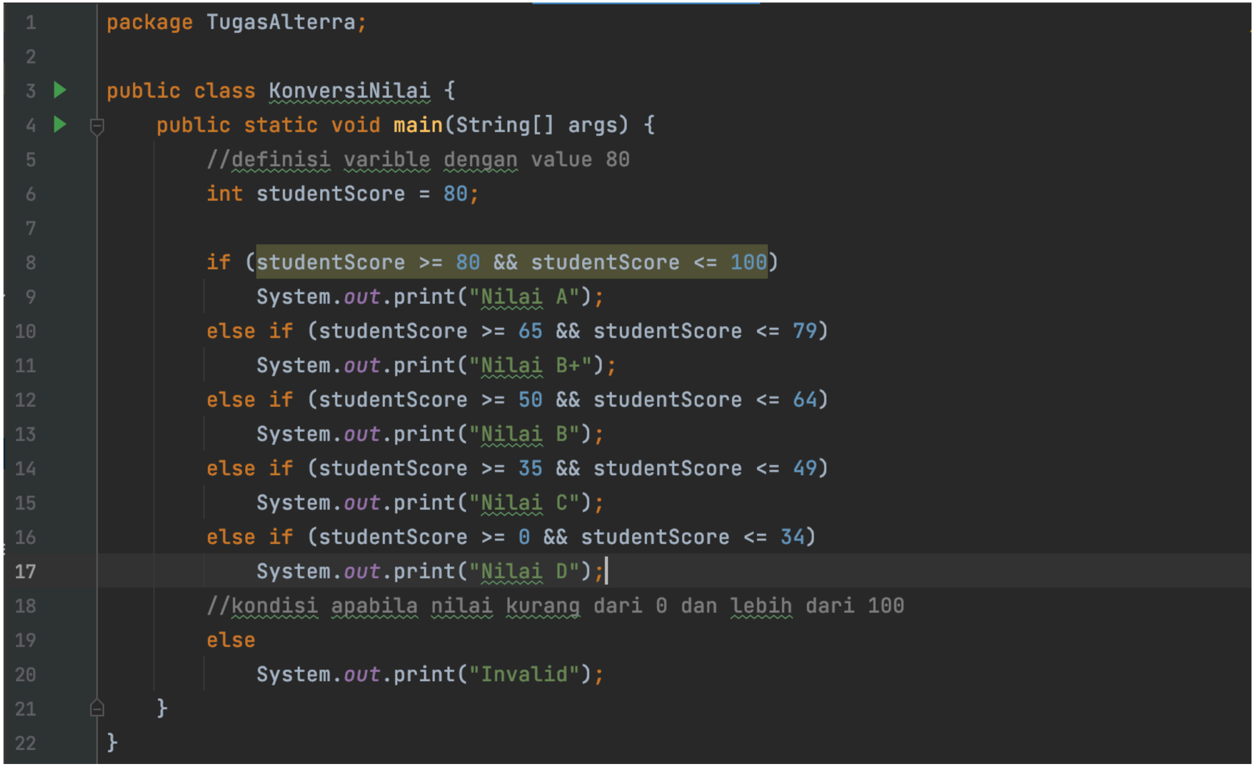


Run:
LatihanQE [:MenghitungLuasSegitiga.main()]
LatihanQE [:Menghitung| 18 sec, 767 ms

```
> Task :MenghitungLuasSegitiga.main()
 Masukan Alas: 20
 Masukan Tinggi: 25
 Luas Segitiga = 250.0
```

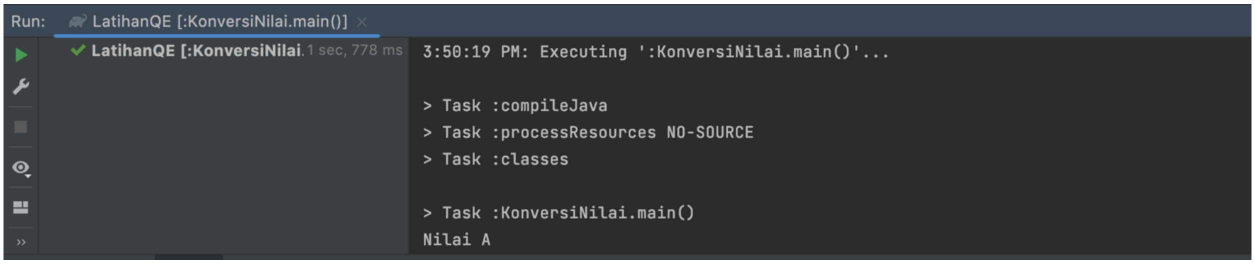
Problem 2: Konversi Nilai

Input:



```java
package TugasAlterra;

public class KonversiNilai {
    public static void main(String[] args) {
        //definisi varible dengan value 80
        int studentScore = 80;

        if (studentScore >= 80 && studentScore <= 100)
            System.out.print("Nilai A");
        else if (studentScore >= 65 && studentScore <= 79)
            System.out.print("Nilai B+");
        else if (studentScore >= 50 && studentScore <= 64)
            System.out.print("Nilai B");
        else if (studentScore >= 35 && studentScore <= 49)
            System.out.print("Nilai C");
        else if (studentScore >= 0 && studentScore <= 34)
            System.out.print("Nilai D");
        //kondisi apabila nilai kurang dari 0 dan lebih dari 100
        else
            System.out.print("Invalid");
    }
}
```

Output:

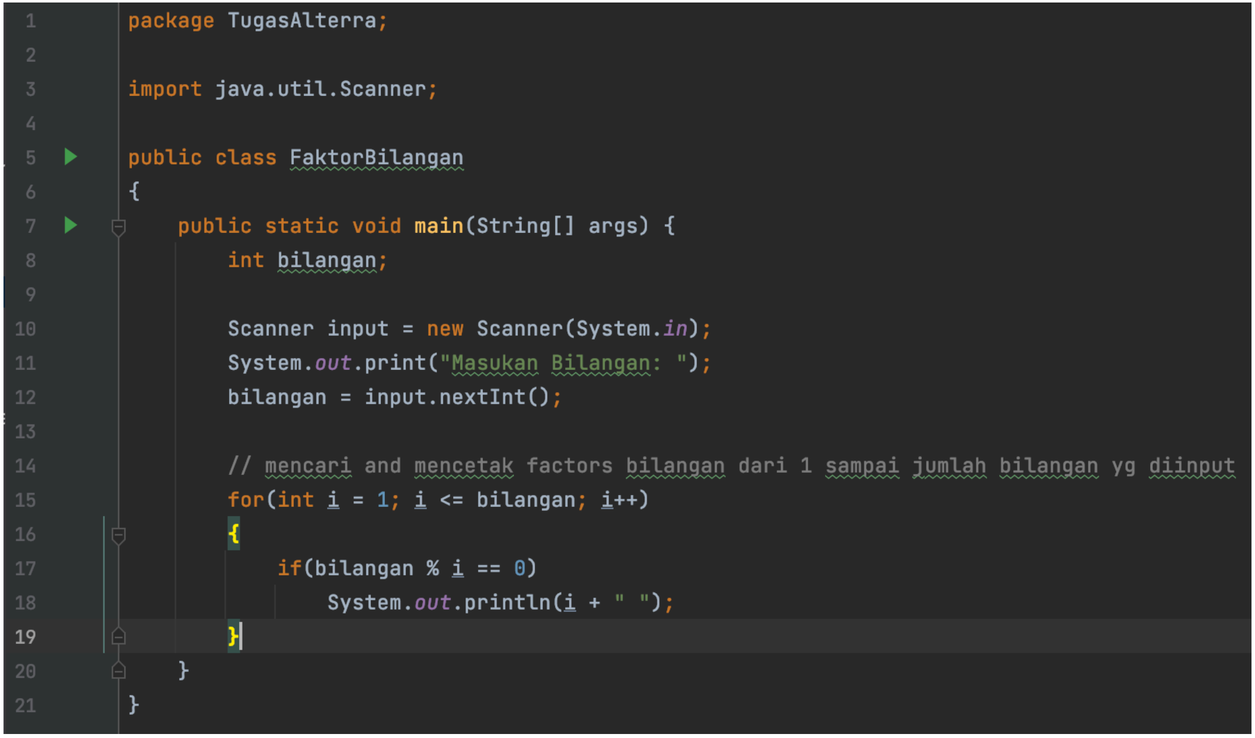


Run:
LatihanQE [:KonversiNilai.main()]
LatihanQE [:KonversiNilai 1 sec, 778 ms

```
3:50:19 PM: Executing ':KonversiNilai.main()'...

> Task :compileJava
> Task :processResources NO-SOURCE
> Task :classes

> Task :KonversiNilai.main()
Nilai A
```

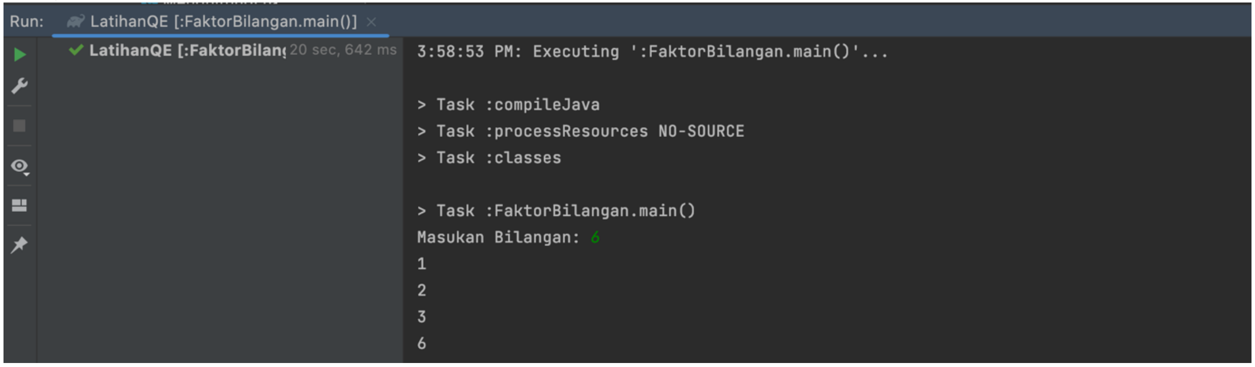
Problem 3 : Faktor Bilangan

Input :



```java
package TugasAlterra;

import java.util.Scanner;

public class FaktorBilangan
{
    public static void main(String[] args) {
        int bilangan;

        Scanner input = new Scanner(System.in);
        System.out.print("Masukan Bilangan: ");
        bilangan = input.nextInt();

        // mencari and mencetak factors bilangan dari 1 sampai jumlah bilangan yg diinput
        for(int i = 1; i <= bilangan; i++)
        {
            if(bilangan % i == 0)
                System.out.println(i + " ");
        }
    }
}
```

Output:

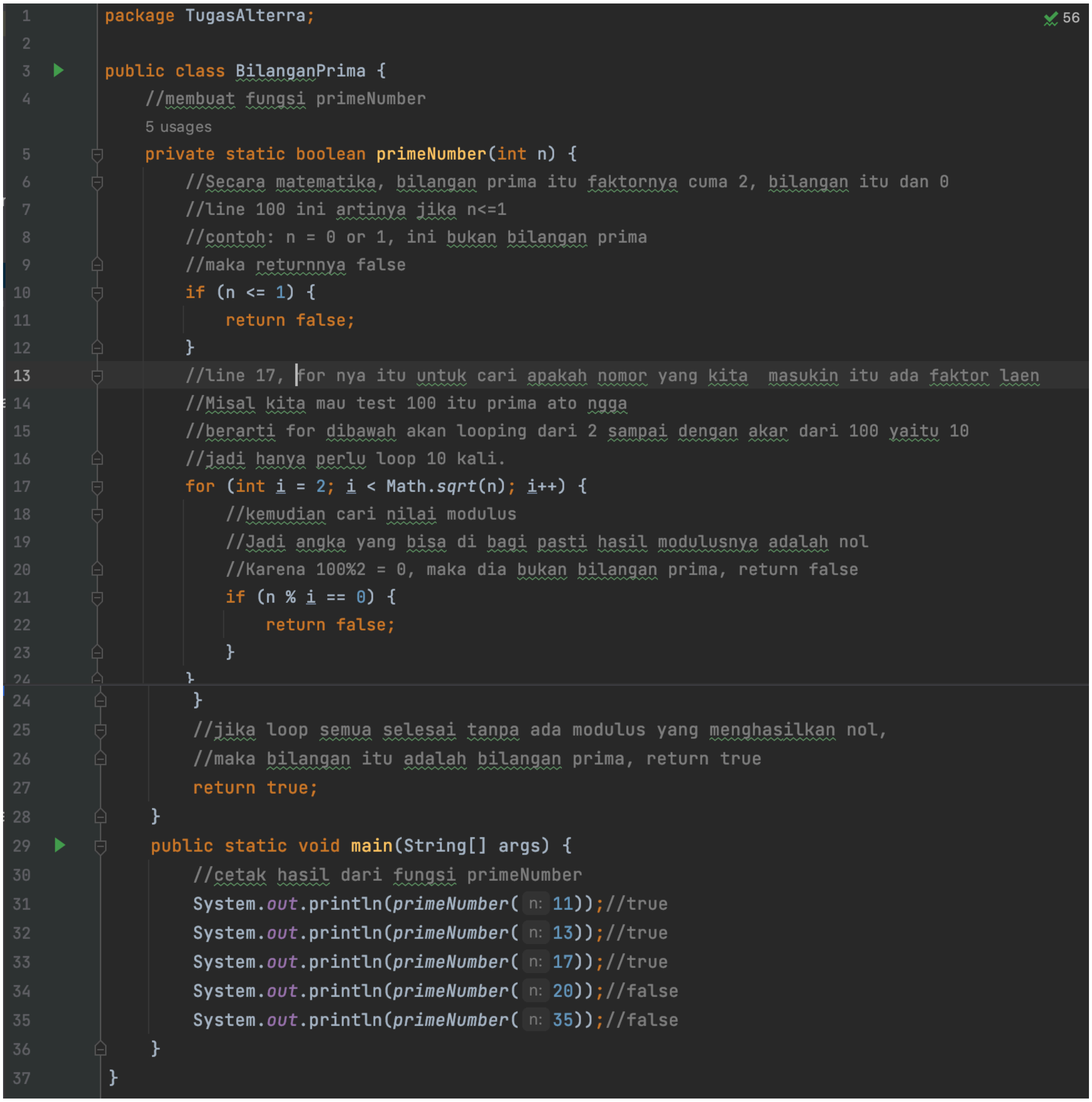


```
3:58:53 PM: Executing ':FaktorBilangan.main()'...

> Task :compileJava
> Task :processResources NO-SOURCE
> Task :classes

> Task :FaktorBilangan.main()
Masukan Bilangan: 6
1
2
3
6
```

Part 2

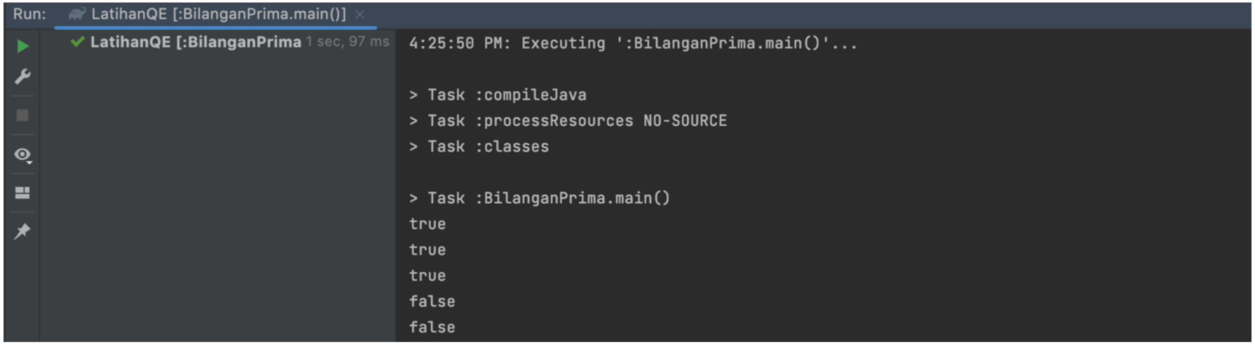
Problem 4. Bilangan Prima

Input :



```
1   package TugasAlterra;
2
3   public class BilanganPrima {
4       //membuat fungsi primeNumber
        5 usages
5       private static boolean primeNumber(int n) {
6           //Secara matematika, bilangan prima itu faktornya cuma 2, bilangan itu dan 0
7           //line 100 ini artinya jika n<=1
8           //contoh: n = 0 or 1, ini bukan bilangan prima
9           //maka returnnya false
10          if (n <= 1) {
11              return false;
12          }
13          //line 17, for nya itu untuk cari apakah nomor yang kita  masukin itu ada faktor laen
14          //Misal kita mau test 100 itu prima ato ngga
15          //berarti for dibawah akan looping dari 2 sampai dengan akar dari 100 yaitu 10
16          //jadi hanya perlu loop 10 kali.
17          for (int i = 2; i < Math.sqrt(n); i++) {
18              //kemudian cari nilai modulus
19              //Jadi angka yang bisa di bagi pasti hasil modulusnya adalah nol
20              //Karena 100%2 = 0, maka dia bukan bilangan prima, return false
21              if (n % i == 0) {
22                  return false;
23              }
24          }
24          }
25          //jika loop semua selesai tanpa ada modulus yang menghasilkan nol,
26          //maka bilangan itu adalah bilangan prima, return true
27          return true;
28      }
29      public static void main(String[] args) {
30          //cetak hasil dari fungsi primeNumber
31          System.out.println(primeNumber( n: 11));//true
32          System.out.println(primeNumber( n: 13));//true
33          System.out.println(primeNumber( n: 17));//true
34          System.out.println(primeNumber( n: 20));//false
35          System.out.println(primeNumber( n: 35));//false
36      }
37  }
```

Output :

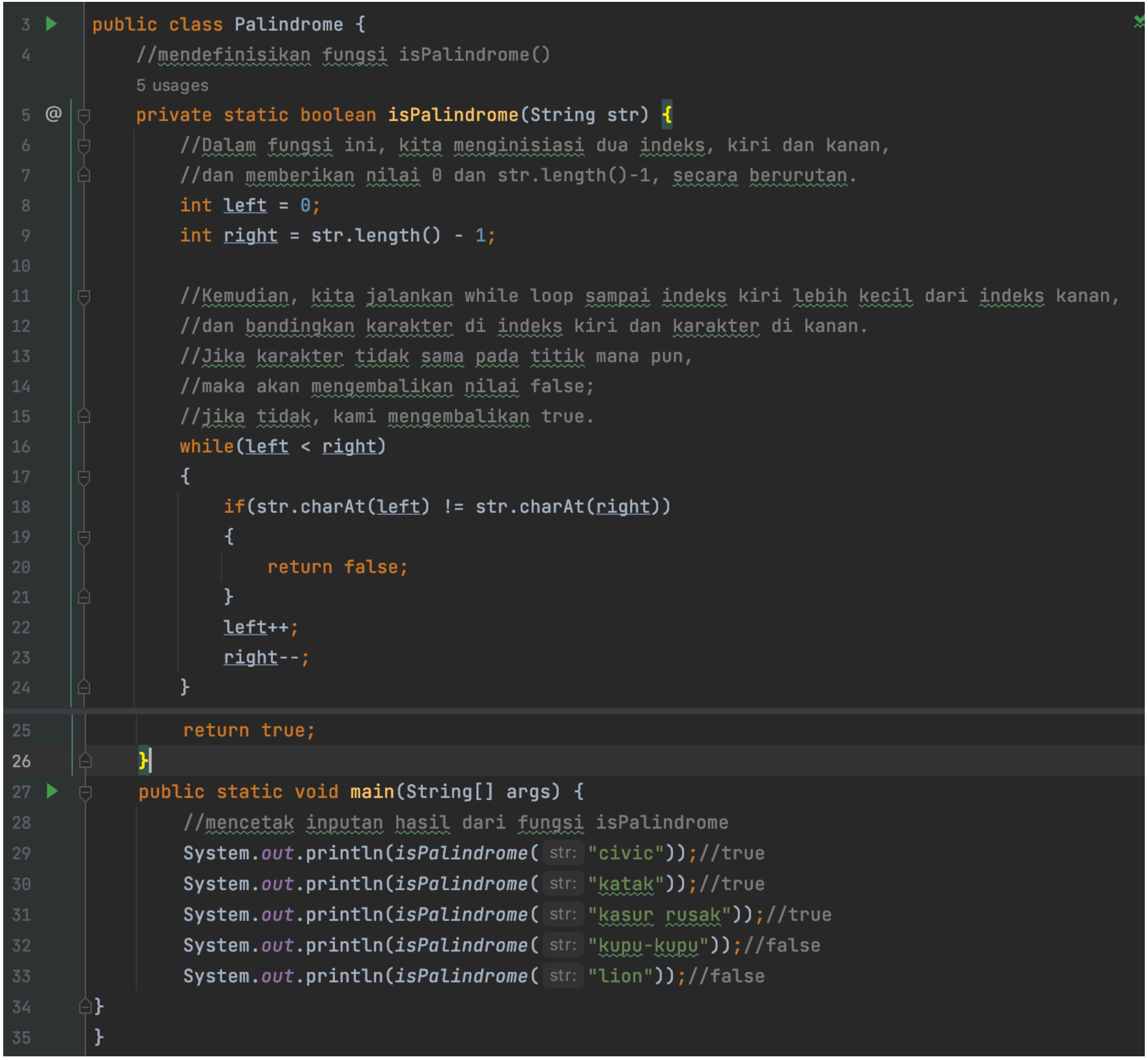


Run: LatihanQE [:BilanganPrima.main()]
LatihanQE [:BilanganPrima 1 sec, 97 ms

```
4:25:50 PM: Executing ':BilanganPrima.main()'...

> Task :compileJava
> Task :processResources NO-SOURCE
> Task :classes

> Task :BilanganPrima.main()
true
true
true
false
false
```

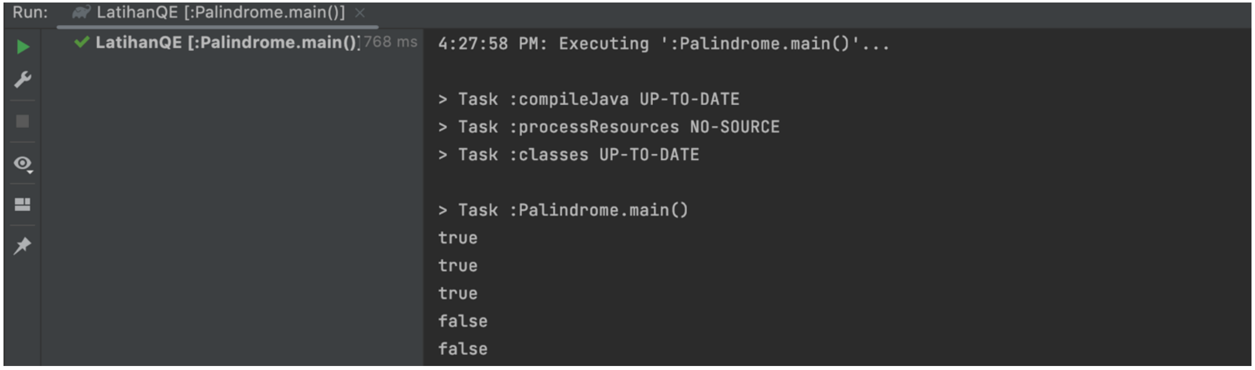
Problem 5. Palindrome

Input:



```java
public class Palindrome {
    //mendefinisikan fungsi isPalindrome()
    private static boolean isPalindrome(String str) {
        //Dalam fungsi ini, kita menginisiasi dua indeks, kiri dan kanan,
        //dan memberikan nilai 0 dan str.length()-1, secara berurutan.
        int left = 0;
        int right = str.length() - 1;

        //Kemudian, kita jalankan while loop sampai indeks kiri lebih kecil dari indeks kanan,
        //dan bandingkan karakter di indeks kiri dan karakter di kanan.
        //Jika karakter tidak sama pada titik mana pun,
        //maka akan mengembalikan nilai false;
        //jika tidak, kami mengembalikan true.
        while(left < right)
        {
            if(str.charAt(left) != str.charAt(right))
            {
                return false;
            }
            left++;
            right--;
        }
        return true;
    }
    public static void main(String[] args) {
        //mencetak inputan hasil dari fungsi isPalindrome
        System.out.println(isPalindrome("civic"));//true
        System.out.println(isPalindrome("katak"));//true
        System.out.println(isPalindrome("kasur rusak"));//true
        System.out.println(isPalindrome("kupu-kupu"));//false
        System.out.println(isPalindrome("lion"));//false
    }
}
```

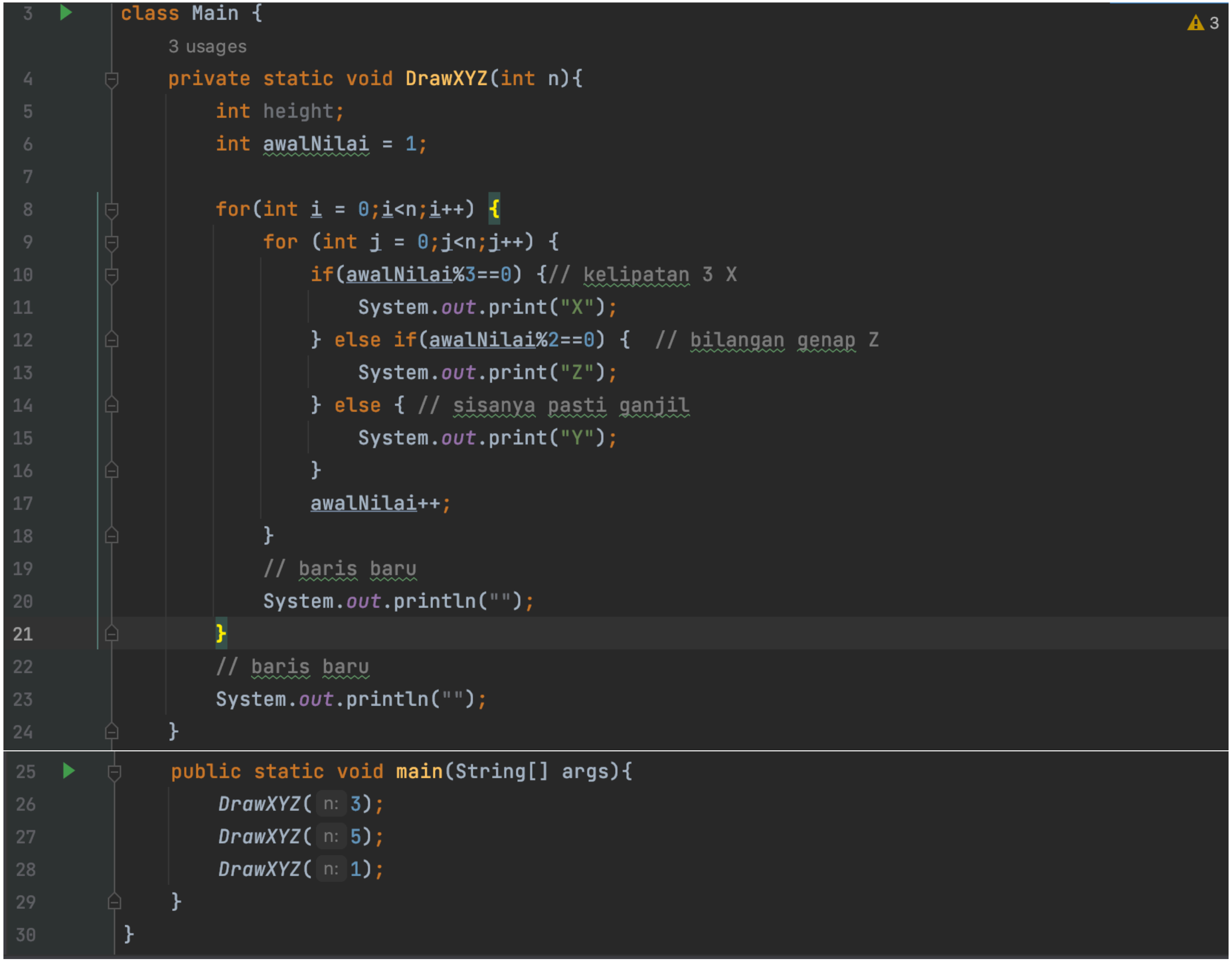
Output:


Run: LatihanQE [:Palindrome.main()]
LatihanQE [:Palindrome.main()] 768 ms

```
4:27:58 PM: Executing ':Palindrome.main()'...

> Task :compileJava UP-TO-DATE
> Task :processResources NO-SOURCE
> Task :classes UP-TO-DATE

> Task :Palindrome.main()
true
true
true
false
false
```

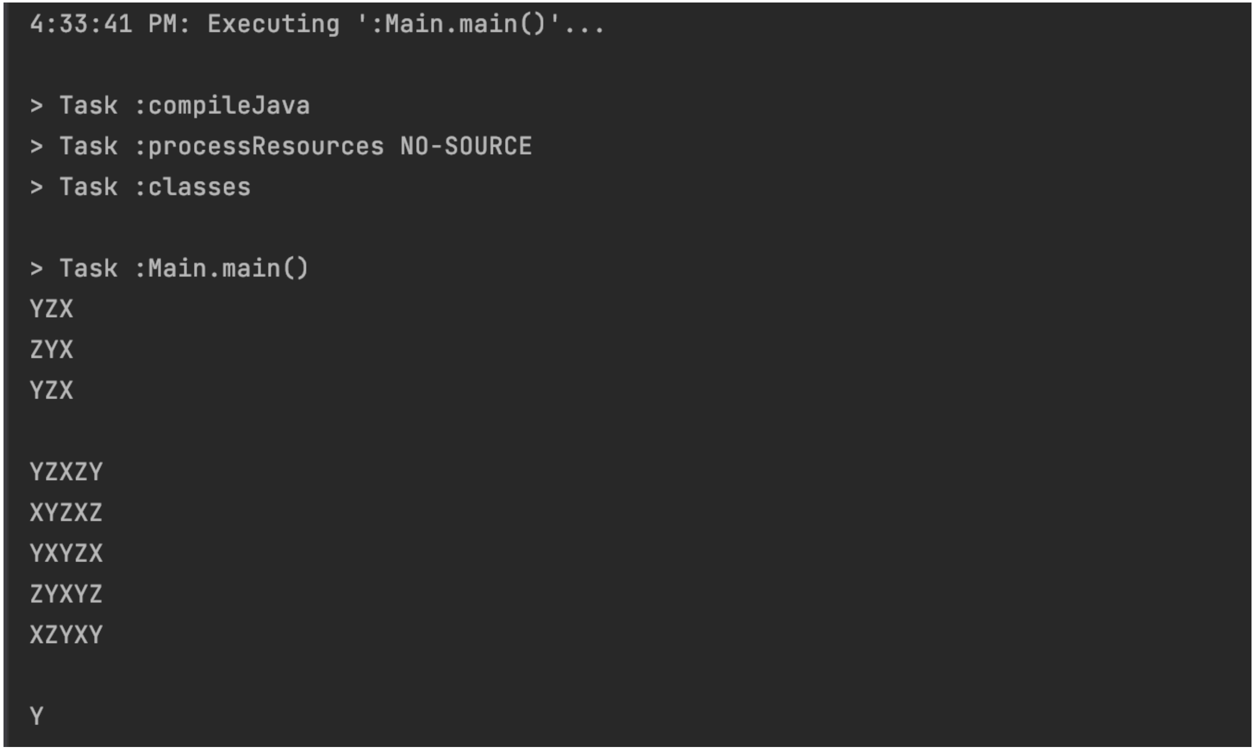
Problem 6. DrawXYZ

Input:



```java
class Main {
    3 usages
    private static void DrawXYZ(int n){
        int height;
        int awalNilai = 1;

        for(int i = 0;i<n;i++) {
            for (int j = 0;j<n;j++) {
                if(awalNilai%3==0) {// kelipatan 3 X
                    System.out.print("X");
                } else if(awalNilai%2==0) {  // bilangan genap Z
                    System.out.print("Z");
                } else { // sisanya pasti ganjil
                    System.out.print("Y");
                }
                awalNilai++;
            }
            // baris baru
            System.out.println("");
        }
        // baris baru
        System.out.println("");
    }
    public static void main(String[] args){
        DrawXYZ( n: 3);
        DrawXYZ( n: 5);
        DrawXYZ( n: 1);
    }
}
```

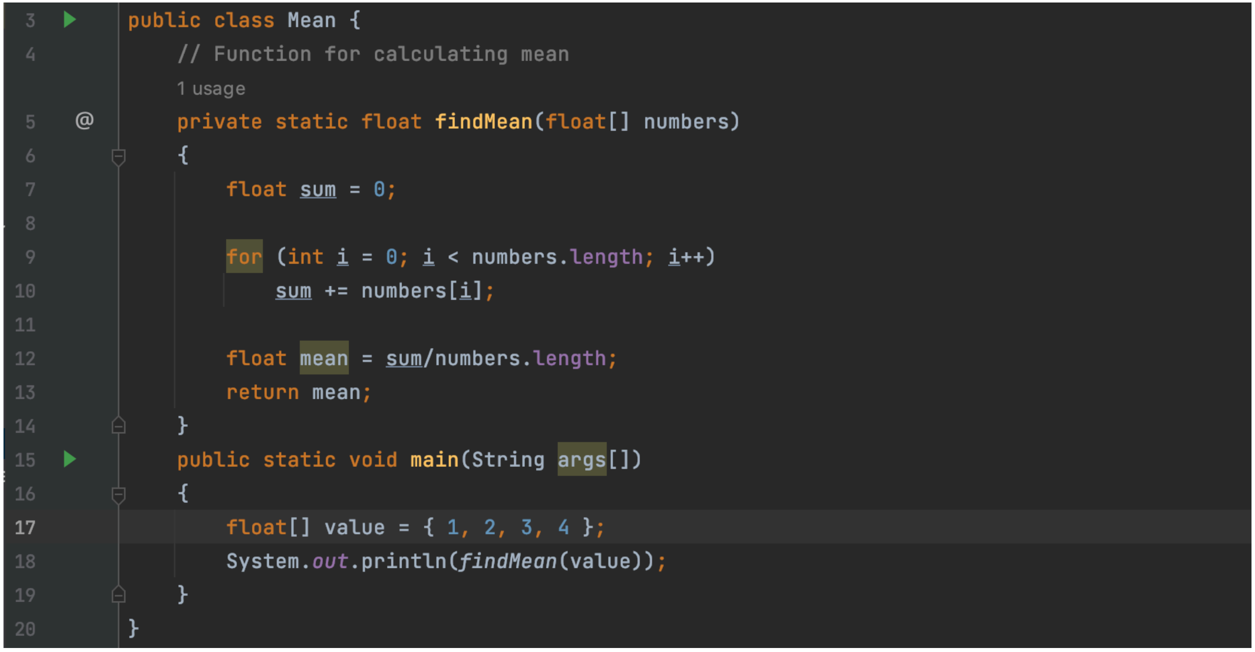
Output:



```
4:33:41 PM: Executing ':Main.main()'...

> Task :compileJava
> Task :processResources NO-SOURCE
> Task :classes

> Task :Main.main()
YZX
ZYX
YZX

YZXZY
XYZXZ
YXYZX
ZYXYZ
XZYXY

Y
```

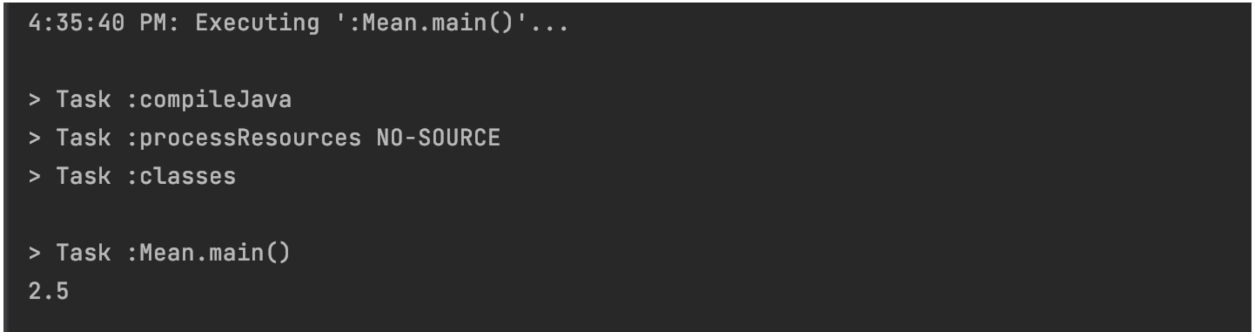
Problem 7. Mean

Input:



```java
public class Mean {
    // Function for calculating mean
    1 usage
    private static float findMean(float[] numbers)
    {
        float sum = 0;

        for (int i = 0; i < numbers.length; i++)
            sum += numbers[i];

        float mean = sum/numbers.length;
        return mean;
    }
    public static void main(String args[])
    {
        float[] value = { 1, 2, 3, 4 };
        System.out.println(findMean(value));
    }
}
```

Output:



```
4:35:40 PM: Executing ':Mean.main()'...

> Task :compileJava
> Task :processResources NO-SOURCE
> Task :classes

> Task :Mean.main()
2.5
```