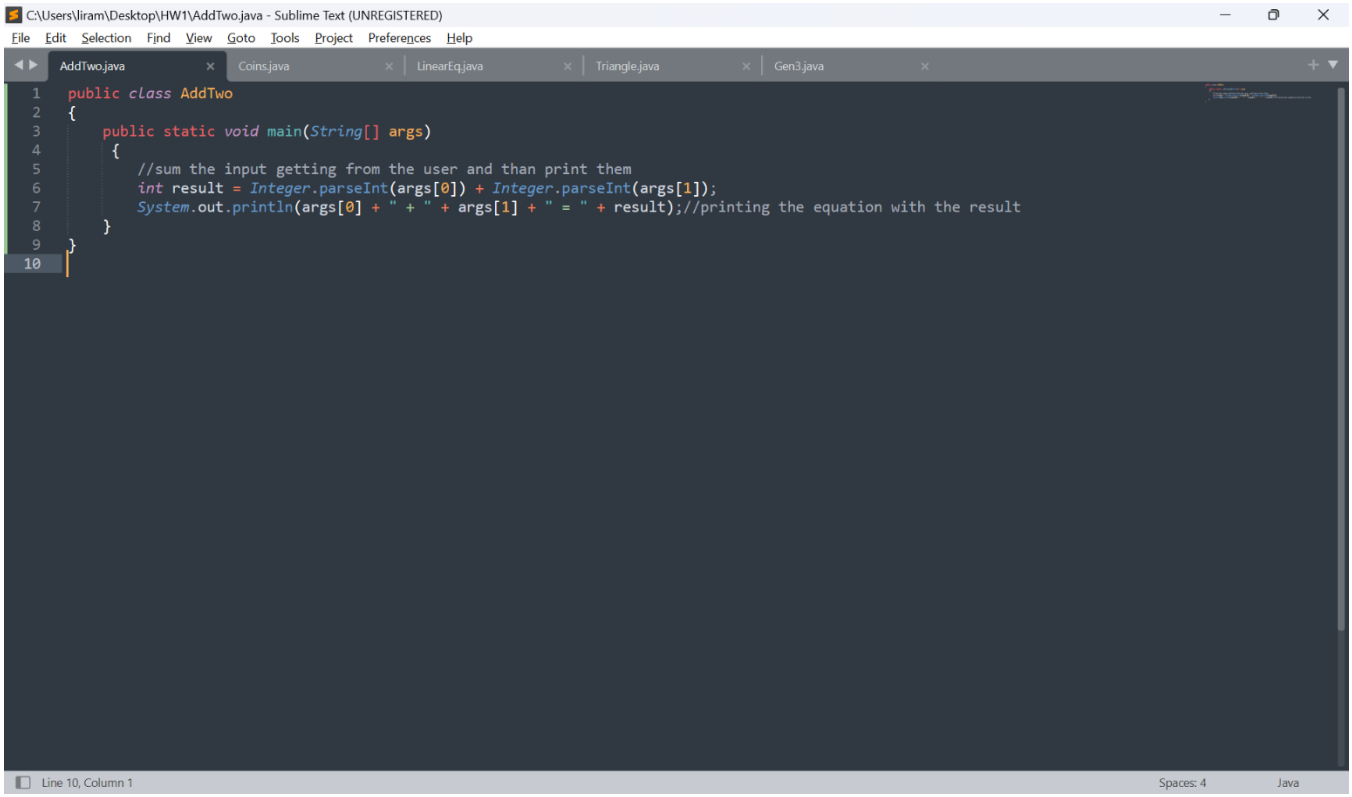
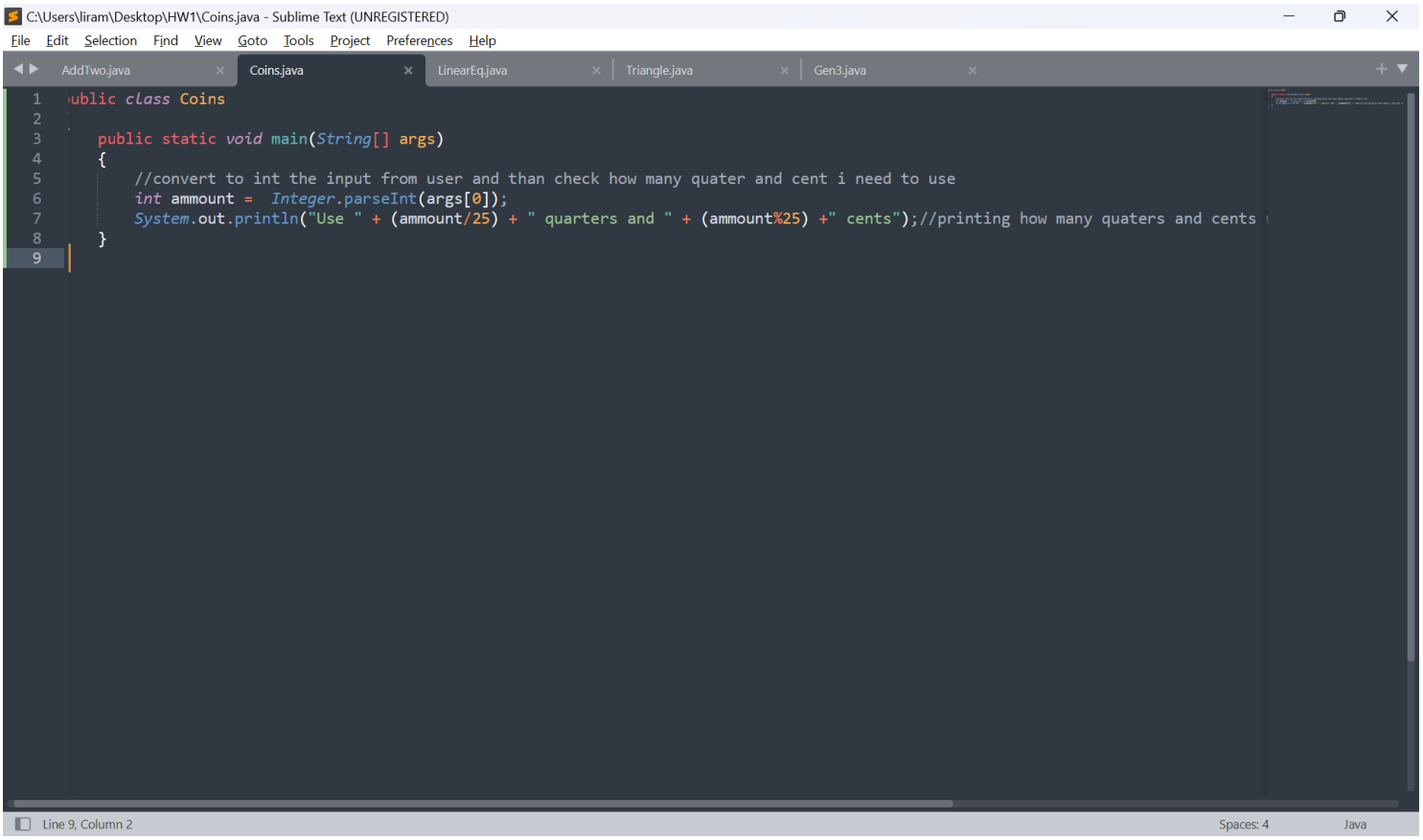
ADDTWO

```
public class AddTwo
{
public static void main(String[] args)
{
sum the input getting from the user and than print them//
int result = Integer.parseInt(args[0]) + Integer.parseInt(args[1])  ;
System.out.println(args[0] + " + " + args[1] + " = " + result);//printing the
equation with the result
}
}
```



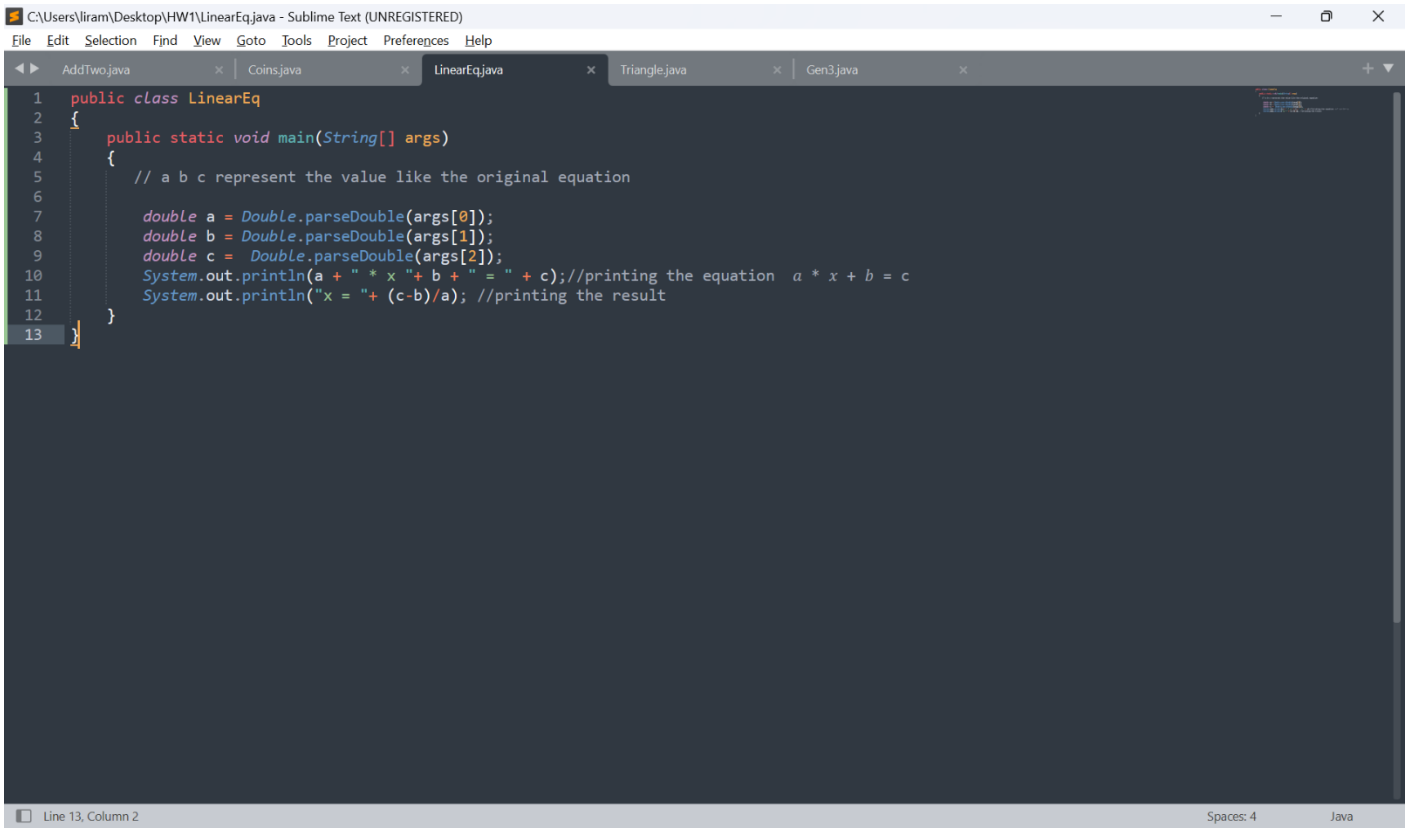
COINS

```
public class Coins

{

public static void main(String[] args)

{

convert to int the input from user and than check how many quater and //
cent i need to use

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

System.out.println("Use " + (ammount/25) + " quarters and " +
(ammount%25) +" cents");//printing how many quaters and cents we can use
that we use minimally ammount of cents that we could

}

}
```



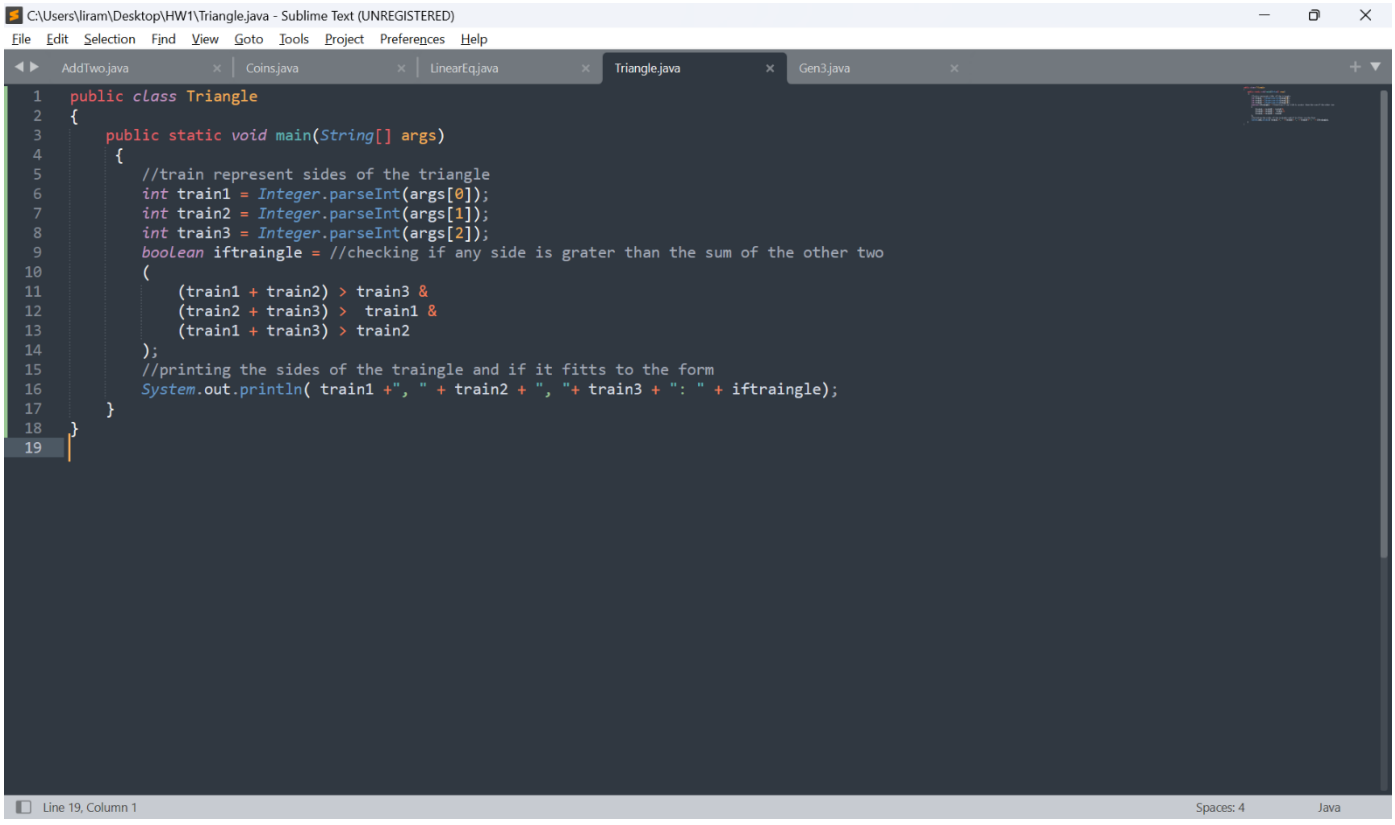
LinearEq

```
public class LinearEq
{
public static void main(String[] args)
{
a b c represent the value like the original equation //


double a = Double.parseDouble(args[0]) ;
double b = Double.parseDouble(args[1]) ;
double c =  Double.parseDouble(args[2]) ;
System.out.println(a + " * x "+ b + " = " + c);//printing the equation  a * x + b = c
System.out.println("x = "+ (c-b)/a); //printing the result
}
}
```



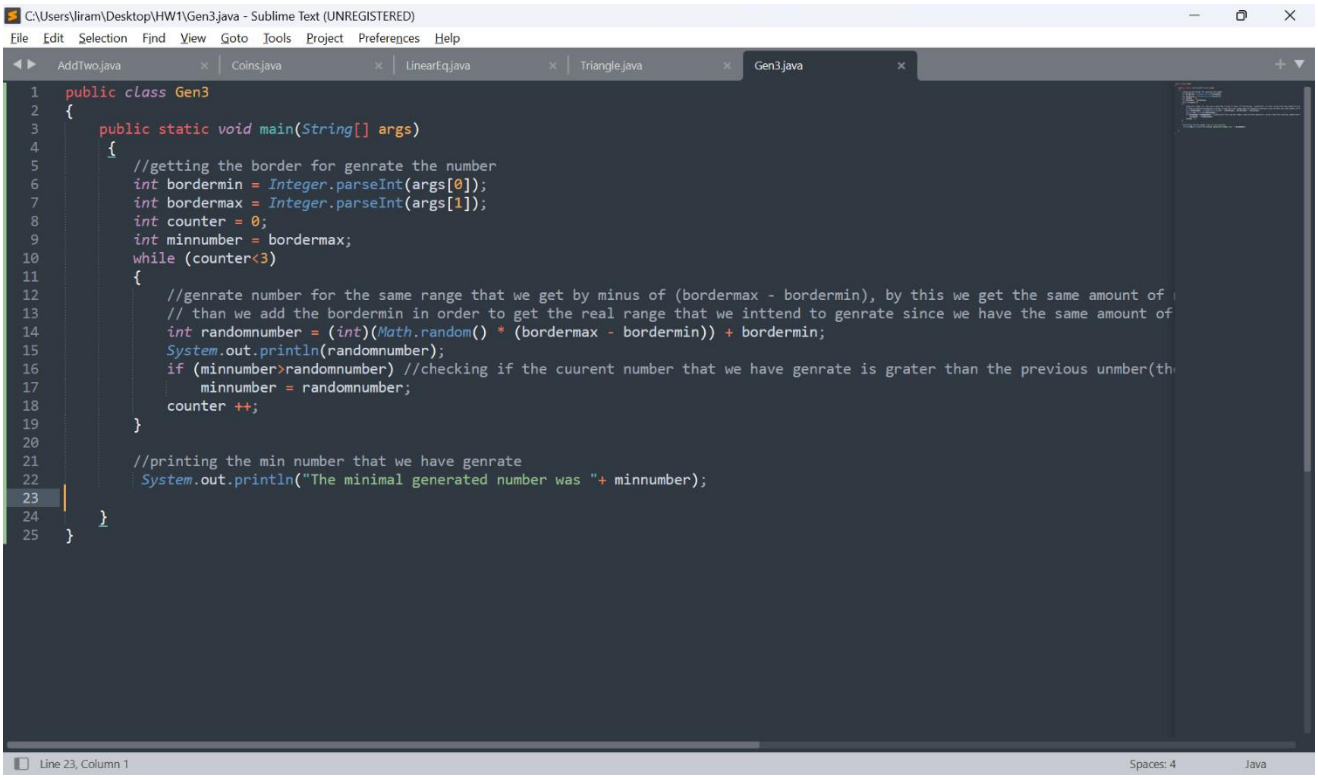
Triangle

```
public class Triangle
{
public static void main(String[] args)
{
train represent sides of the triangle//
int train1 = Integer.parseInt(args[0]) ;
int train2 = Integer.parseInt(args[1]) ;
int train3 = Integer.parseInt(args[2]) ;
boolean iftraingle = //checking if any side is grater than the sum of the
other two
(
& (train1 + train2) > train3
& (train2 + train3) >  train1
(train1 + train3) > train2
);
printing the sides of the traingle and if it fitts to the form//
;System.out.println( train1 +", " + train2 + ", "+ train3 + ": " + iftraingle)
}
}
```



Gen3

```
public class Gen3
{
public static void main(String[] args)
{
getting the border for genrate the number//
int bordermin = Integer.parseInt(args[0])  ;
int bordermax = Integer.parseInt(args[1]) ;
int counter = 0 ;
int minnumber = bordermax ;
while (counter<3)
{
genrate number for the same range that we get by minus of //
(bordermax - bordermin), by this we get the same amount of number that
coule possibly can been genrate
than we add the bordermin in order to get the real range that we  //
inttend to genrate since we have the same amount of opption we can get
int randomnumber = (int)(Math.random() * (bordermax - bordermin)) +
bordermin ;
System.out.println(randomnumber) ;
if (minnumber>randomnumber) //checking if the cuurent number that
we have genrate is grater than the previous unmber(the bordermax is the first
min number but it allways change since this number is not in the range)
minnumber = randomnumber ;
counter ++ ;
}


printing the min number that we have genrate//
System.out.println("The minimal generated number was "+ minnumber) ;


}
}
```


C:\Users\liram\Desktop\HW1\Gen3.java - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
AddTwo.java
Coins.java
LinearEq.java
Triangle.java
Gen3.java
public class Gen3
{
public static void main(String[] args)
{
//getting the border for genrate the number
int bordermin = Integer.parseInt(args[0]);
int bordermax = Integer.parseInt(args[1]);
int counter = 0;
int minnumber = bordermax;
while (counter<3)
{
//genrate number for the same range that we get by minus of (bordermax - bordermin), by this we get the same amount of
// than we add the bordermin in order to get the real range that we inttend to genrate since we have the same amount of
int randomnumber = (int)(Math.random() * (bordermax - bordermin)) + bordermin;
System.out.println(randomnumber);
if (minnumber>randomnumber) //checking if the cuurent number that we have genrate is grater than the previous unmber(th
minnumber = randomnumber;
counter ++;
}
//printing the min number that we have genrate
System.out.println("The minimal generated number was "+ minnumber);
}
}
Line 23, Column 1
Spaces: 4
Java