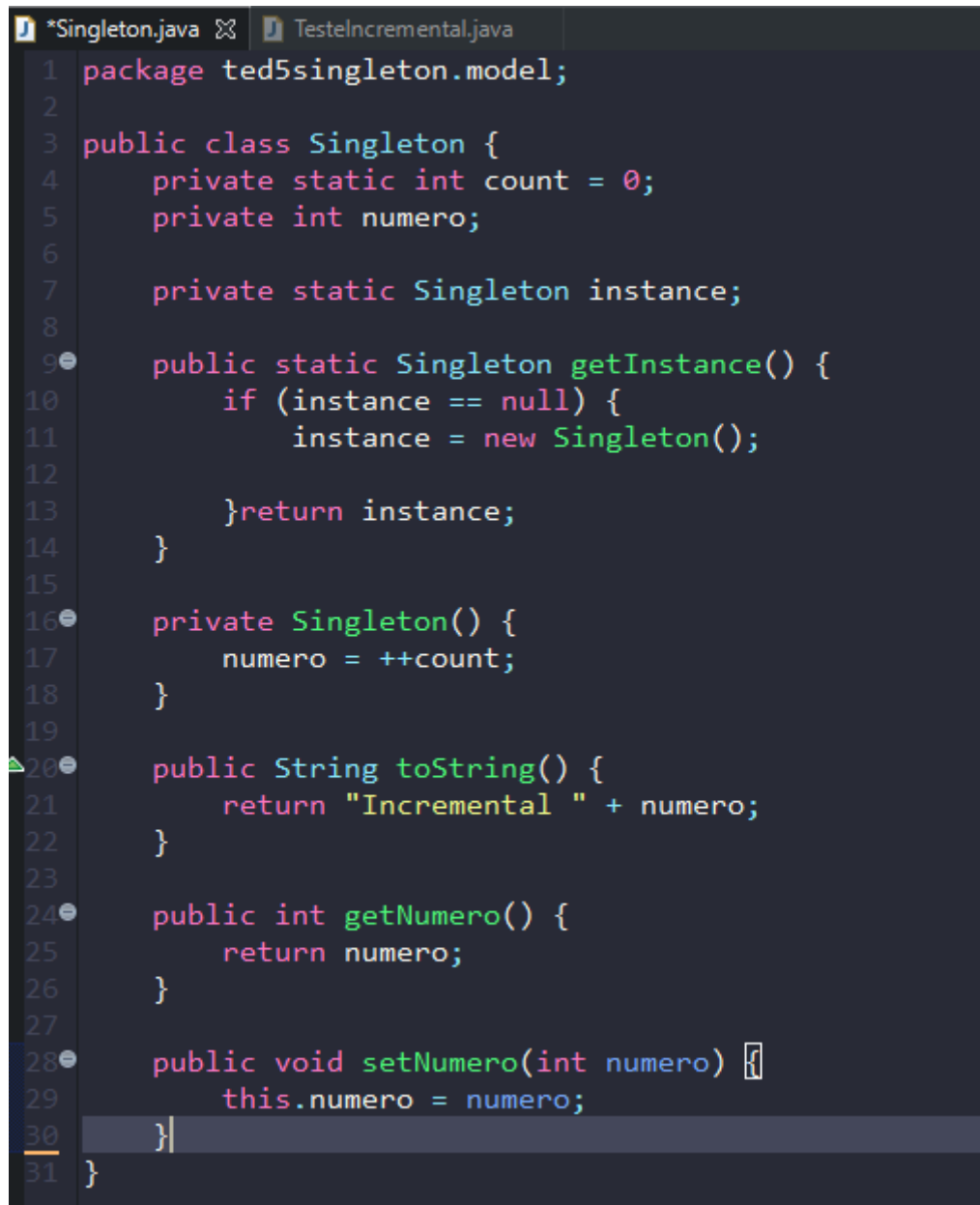


## Usando o Singleton.

Ao meu ver, usando o padrão Singleton para este tipo de instanciamento de objetos, a vantagem que existe é de não ter que ficar criando varias instancias a todo tempo.

Deu pra perceber um código de boa compreensão, apesar da classe extremamente simples.



```
1 package ted5singleton.model;
2
3 public class Singleton {
4     private static int count = 0;
5     private int numero;
6
7     private static Singleton instance;
8
9     public static Singleton getInstance() {
10         if (instance == null) {
11             instance = new Singleton();
12         }
13         return instance;
14     }
15
16     private Singleton() {
17         numero = ++count;
18     }
19
20     public String toString() {
21         return "Incremental " + numero;
22     }
23
24     public int getNumero() {
25         return numero;
26     }
27
28     public void setNumero(int numero) {
29         this.numero = numero;
30     }
31 }
```

Aqui na classe principal da pra notar a simplicidade em não ter que criar algo que já foi anteriormente instanciado.

```
*Singleton.java  TesteIncremental.java ✕
1 package ted5singleton_visão;
2
3 import ted5singleton.model.*;
4
5 public class TesteIncremental {
6     public static void main(String[] args) {
7
8         for (int i = 0; i < 10; i++) {
9             Singleton object = Singleton.getInstance();
10            System.out.println(object.getNumero());
11
12        }
13
14    }
15 }
16
```

```
*Singleton.java  TesteIncremental.java ✕
1 package ted5singleton_visão;
2
3 import ted5singleton.model.*;
4
5 public class TesteIncremental {
6     public static void main(String[] args) {
7
8         for (int i = 0; i < 10; i++) {
9             Singleton object = Singleton.getInstance();
10            System.out.println(object.getNumero());
11
12        }
13
14    }
15 }
16
```