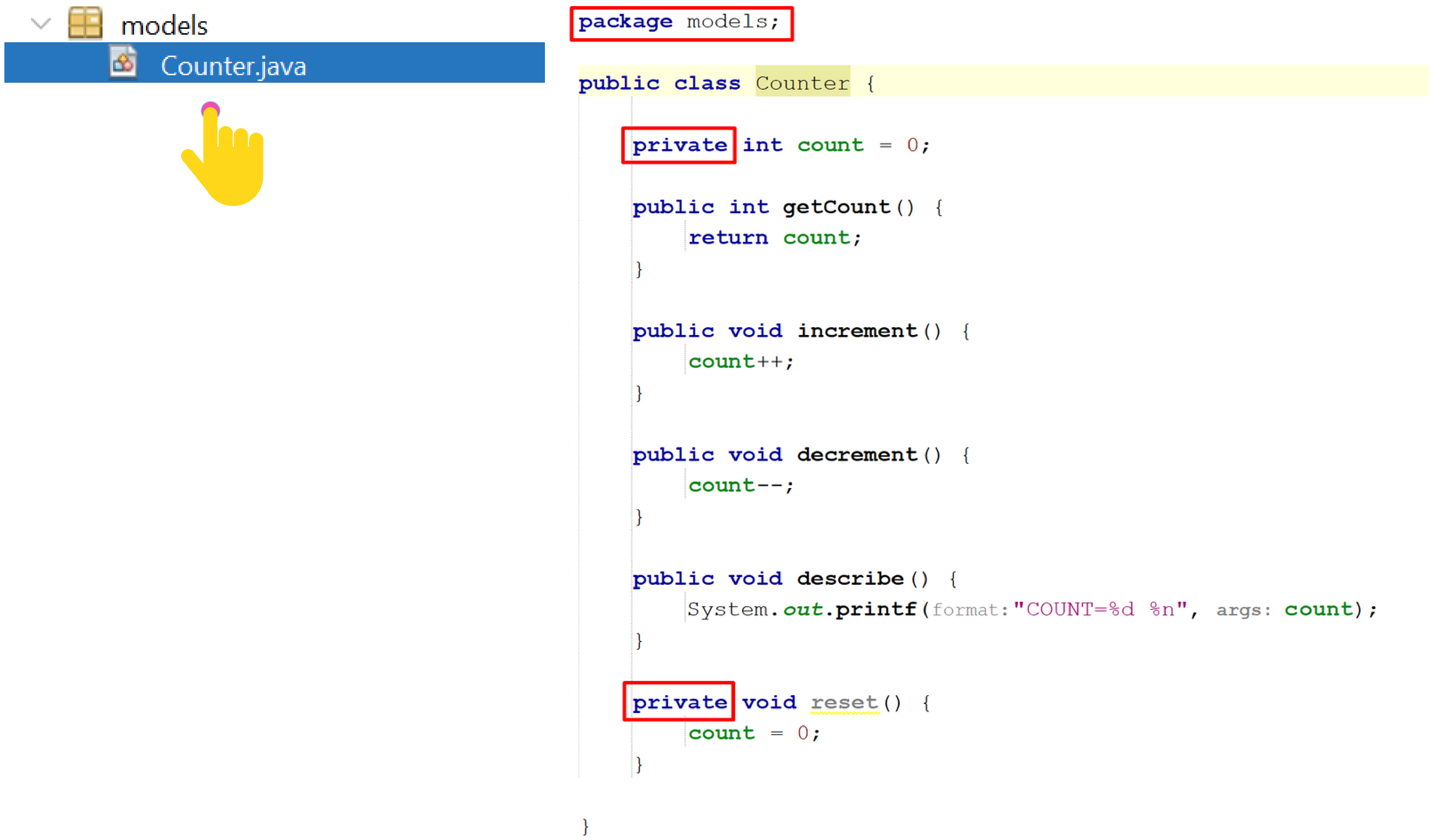
Fundamentos de Java

Parte 7

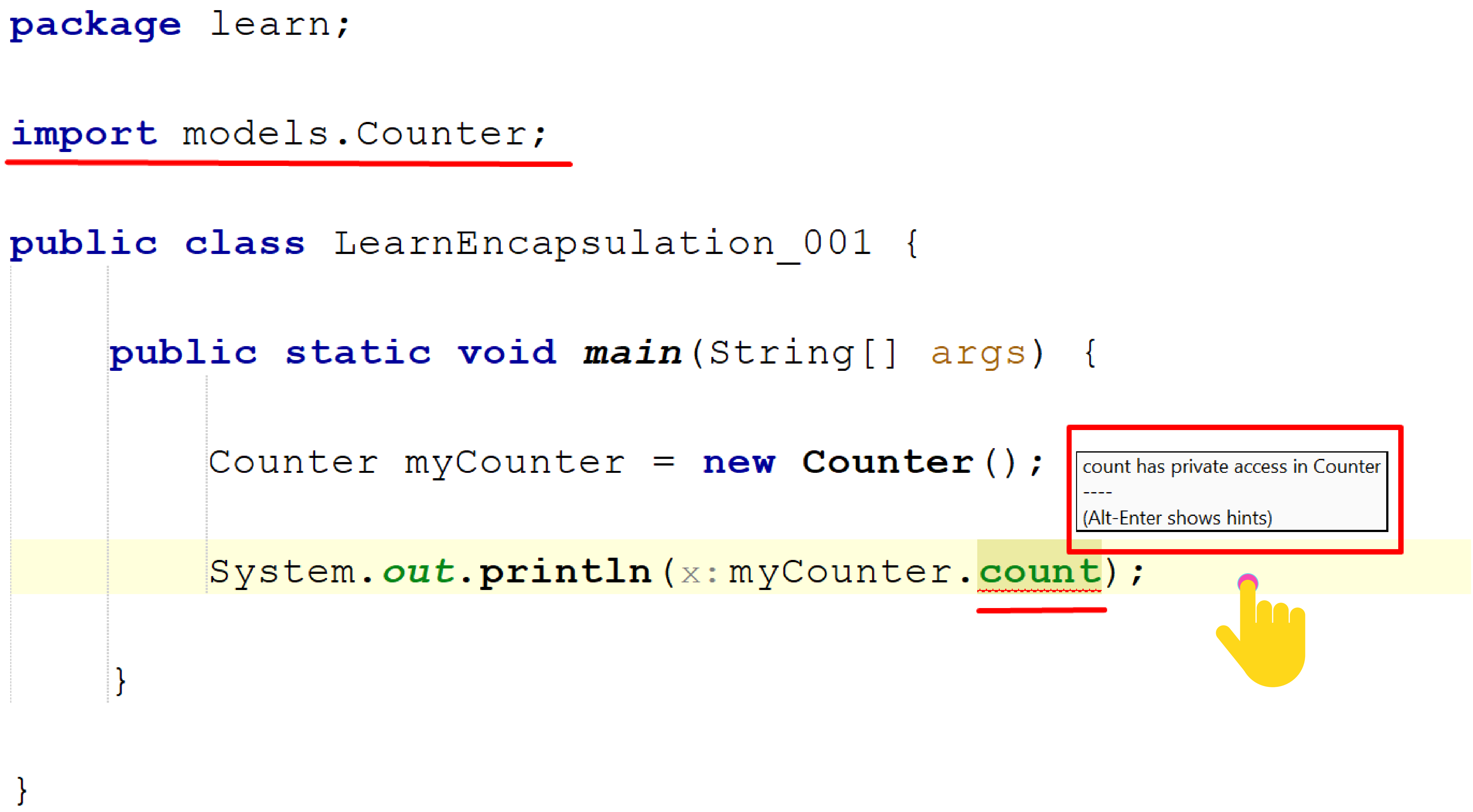
Presenta

Alan Badillo Salas

Marzo 2023



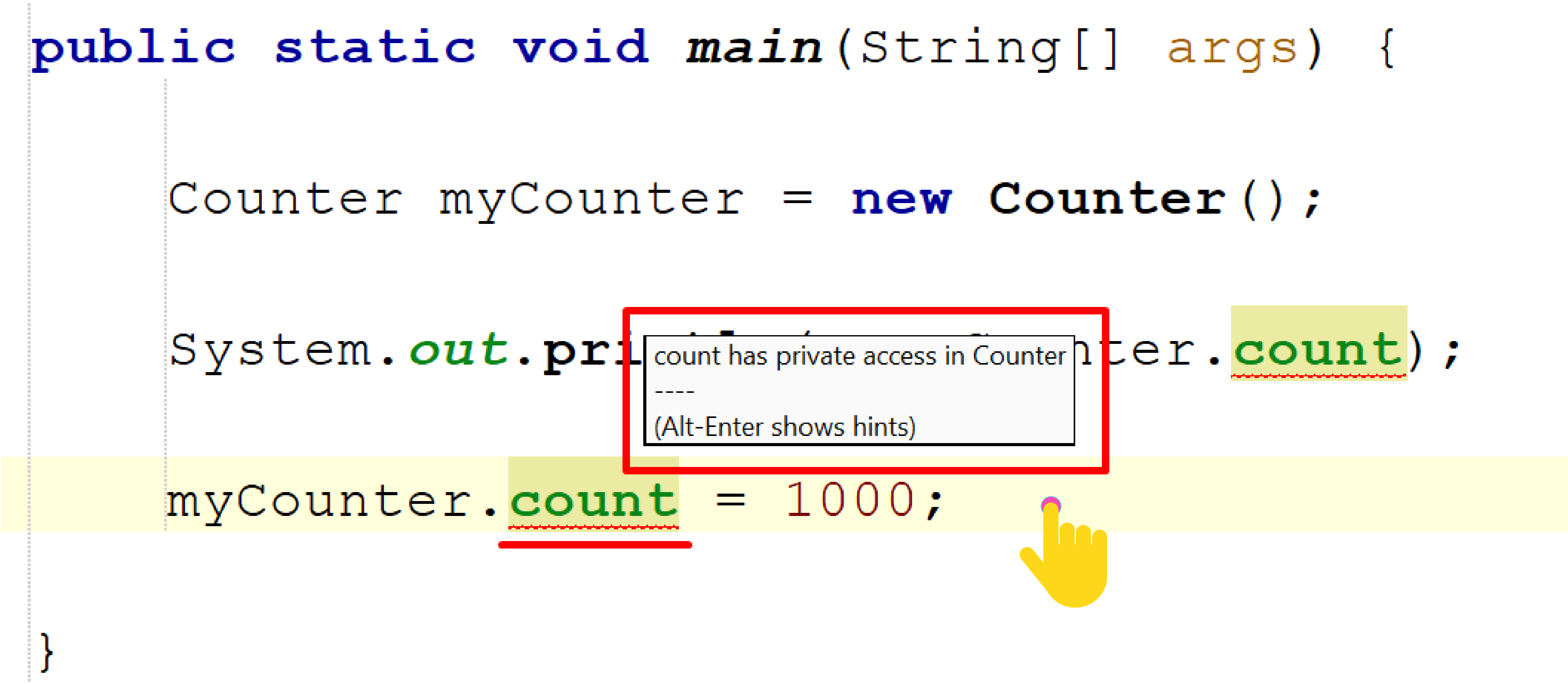
```
package models;

public class Counter {

    private int count = 0;

    public int getCount() {
        return count;
    }

    public void increment() {
        count++;
    }

    public void decrement() {
        count--;
    }

    public void describe() {
        System.out.printf(format: "COUNT=%d %n", args: count);
    }

    private void reset() {
        count = 0;
    }

}
```



```java
package learn;

import models.Counter;

public class LearnEncapsulation_001 {

    public static void main(String[] args) {

        Counter myCounter = new Counter();

        System.out.println(x: myCounter.count);

    }

}
```

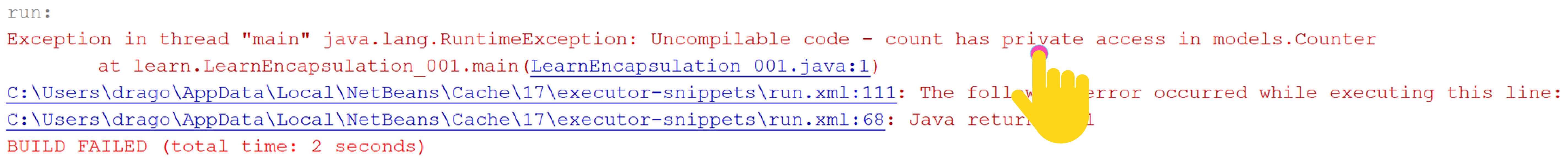
count has private access in Counter

(Alt-Enter shows hints)

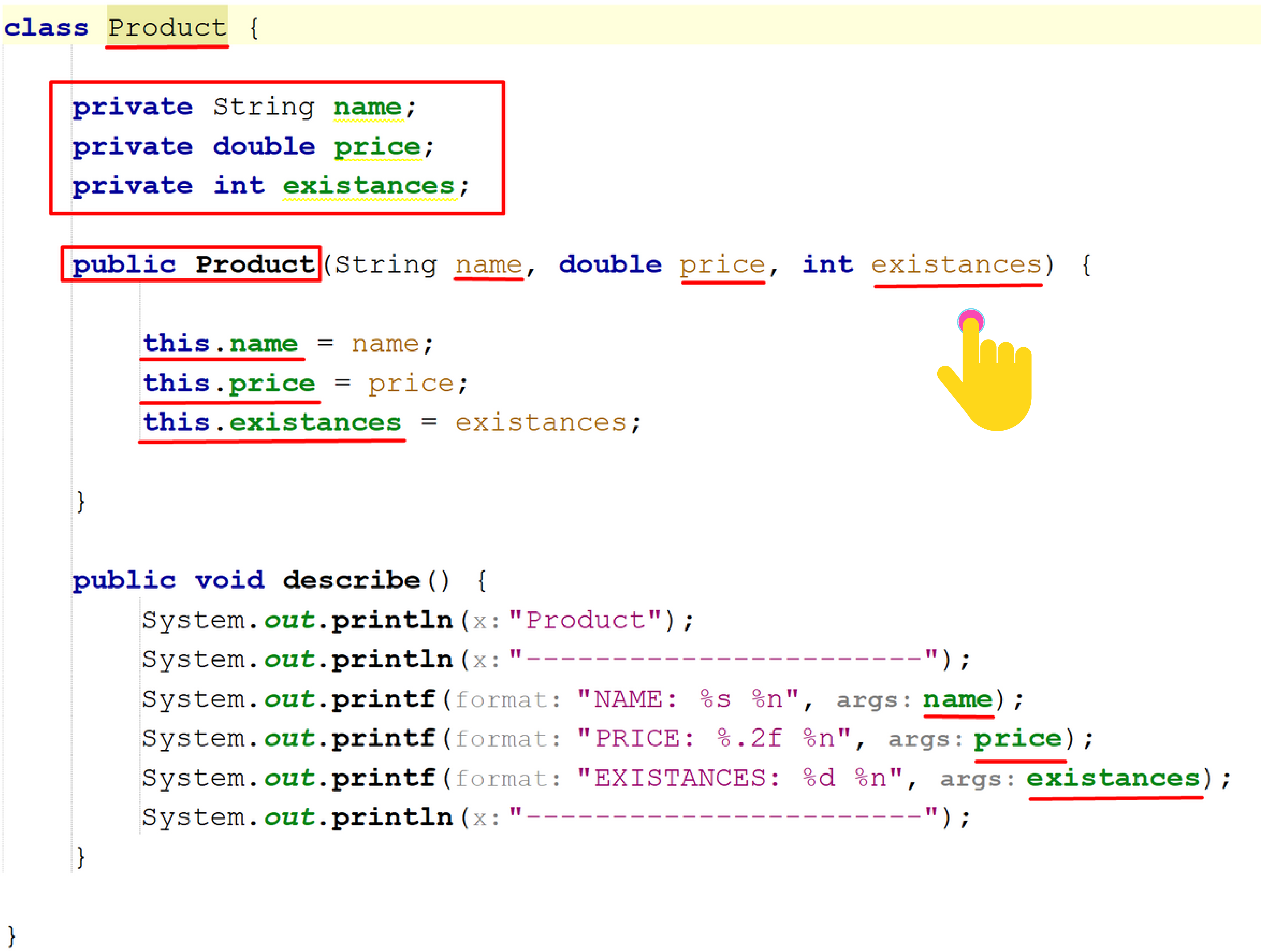


```
public static void main(String[] args) {
    Counter myCounter = new Counter();
    System.out.pri                    ter.count);
    myCounter.count = 1000;
}
```

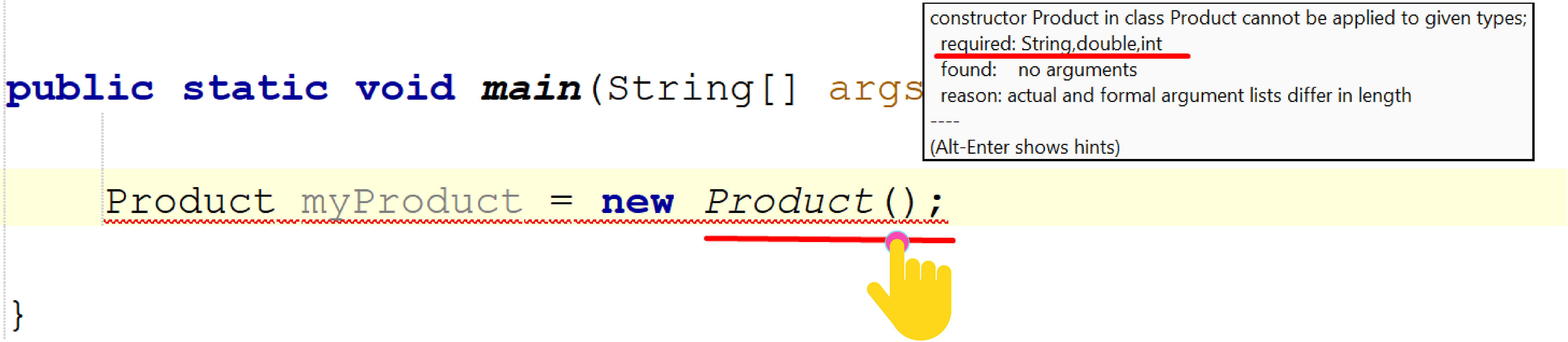
```
count has private access in Counter
----
(Alt-Enter shows hints)
```



```
run:
Exception in thread "main" java.lang.RuntimeException: Uncompilable code - count has private access in models.Counter
	at learn.LearnEncapsulation_001.main(LearnEncapsulation_001.java:1)
C:\Users\drago\AppData\Local\NetBeans\Cache\17\executor-snippets\run.xml:111: The following error occurred while executing this line:
C:\Users\drago\AppData\Local\NetBeans\Cache\17\executor-snippets\run.xml:68: Java returned: 1
BUILD FAILED (total time: 2 seconds)
```

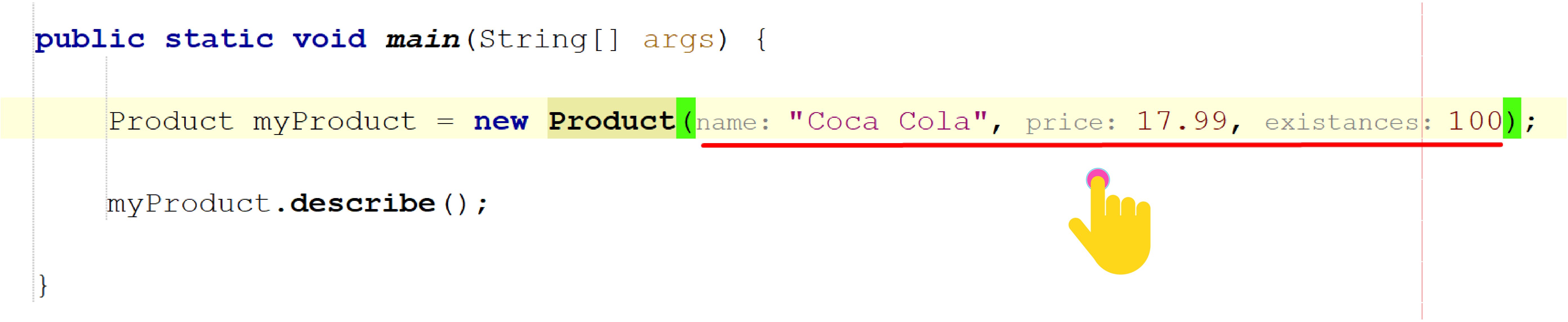


```java
class Product {

    private String name;
    private double price;
    private int existances;

    public Product(String name, double price, int existances) {

        this.name = name;
        this.price = price;
        this.existances = existances;

    }

    public void describe() {
        System.out.println(x: "Product");
        System.out.println(x: "-----------------------");
        System.out.printf(format: "NAME: %s %n", args: name);
        System.out.printf(format: "PRICE: %.2f %n", args: price);
        System.out.printf(format: "EXISTANCES: %d %n", args: existances);
        System.out.println(x: "-----------------------");
    }

}
```

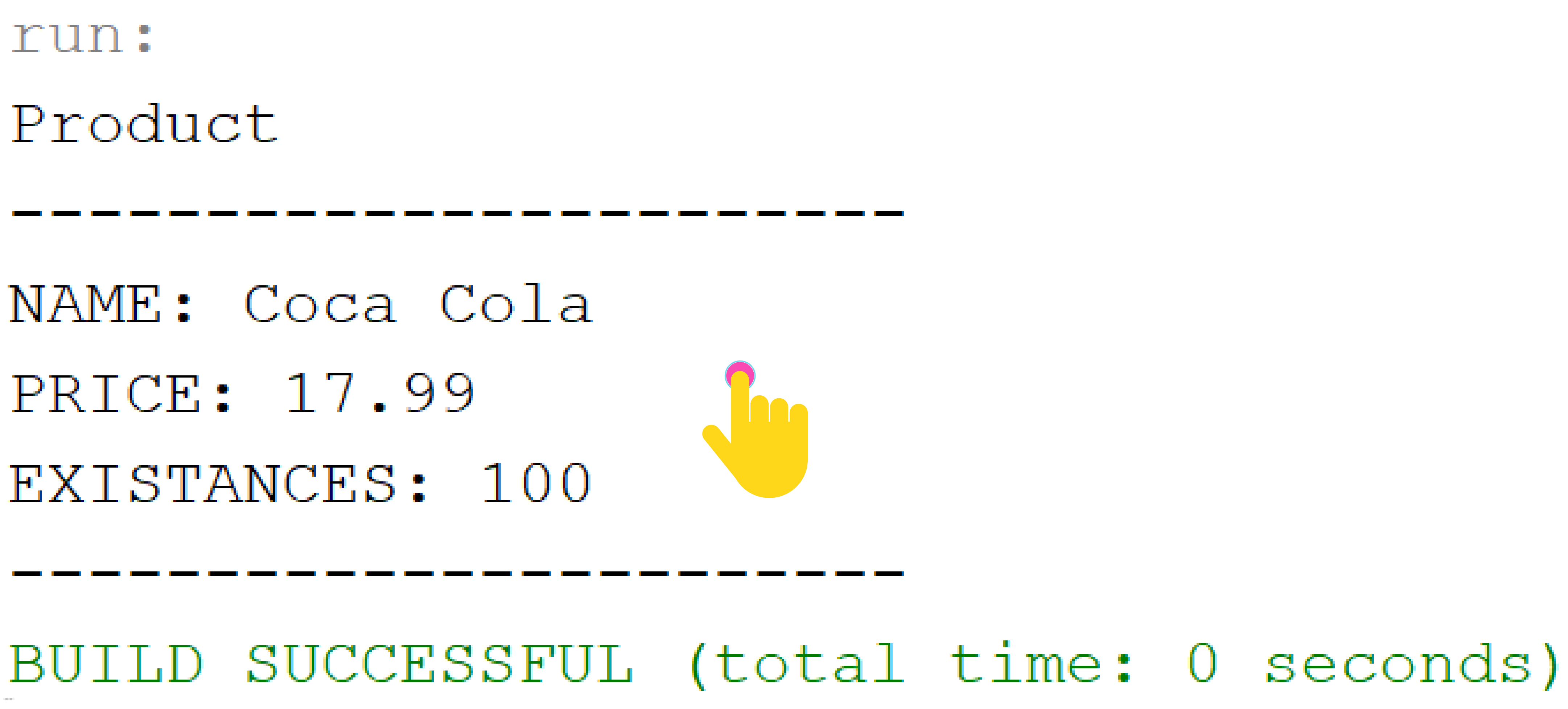


```
public static void main(String[] args

    Product myProduct = new Product();

}
```

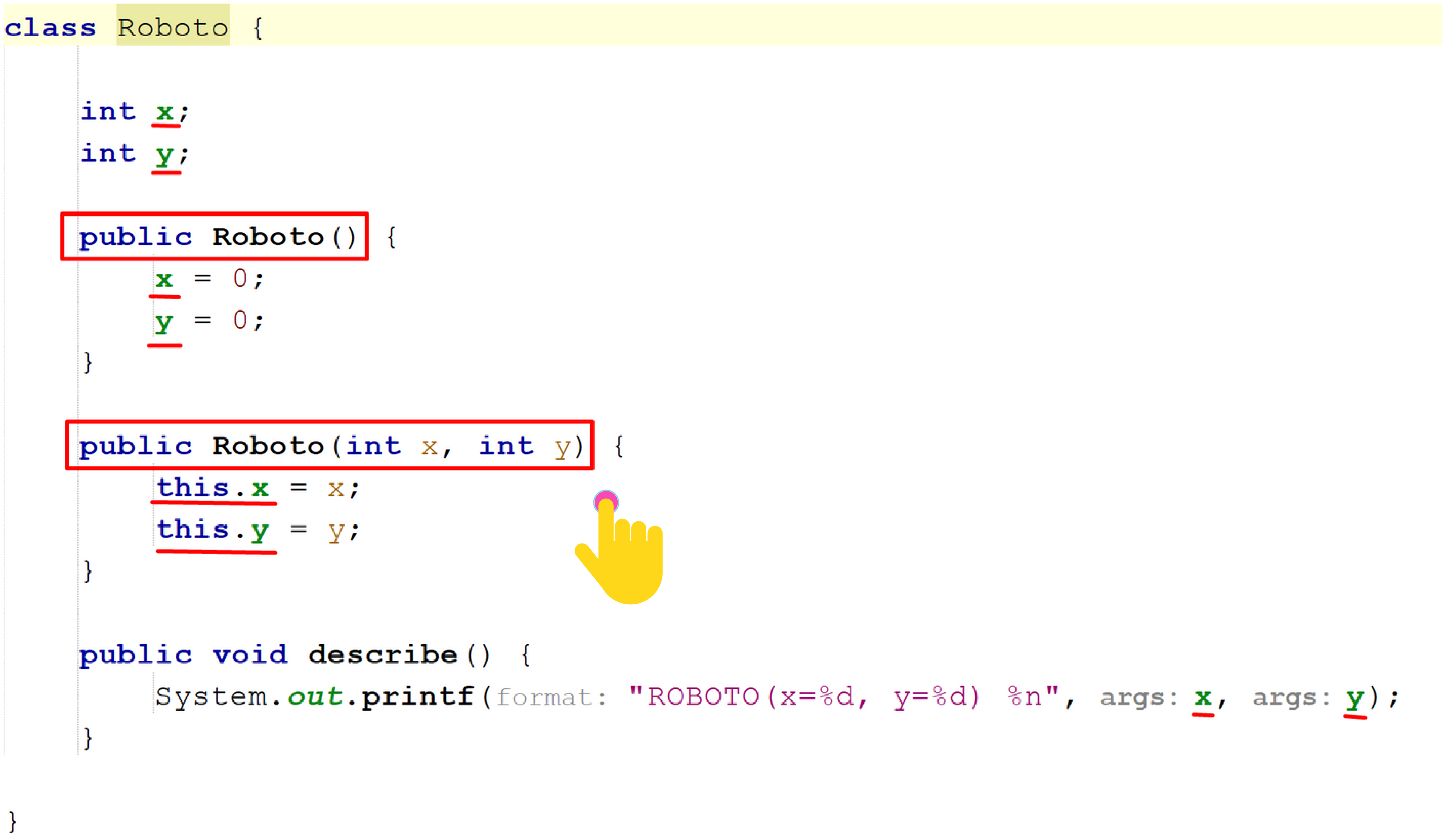
```
constructor Product in class Product cannot be applied to given types;
  required: String,double,int
  found:    no arguments
  reason: actual and formal argument lists differ in length
----
(Alt-Enter shows hints)
```



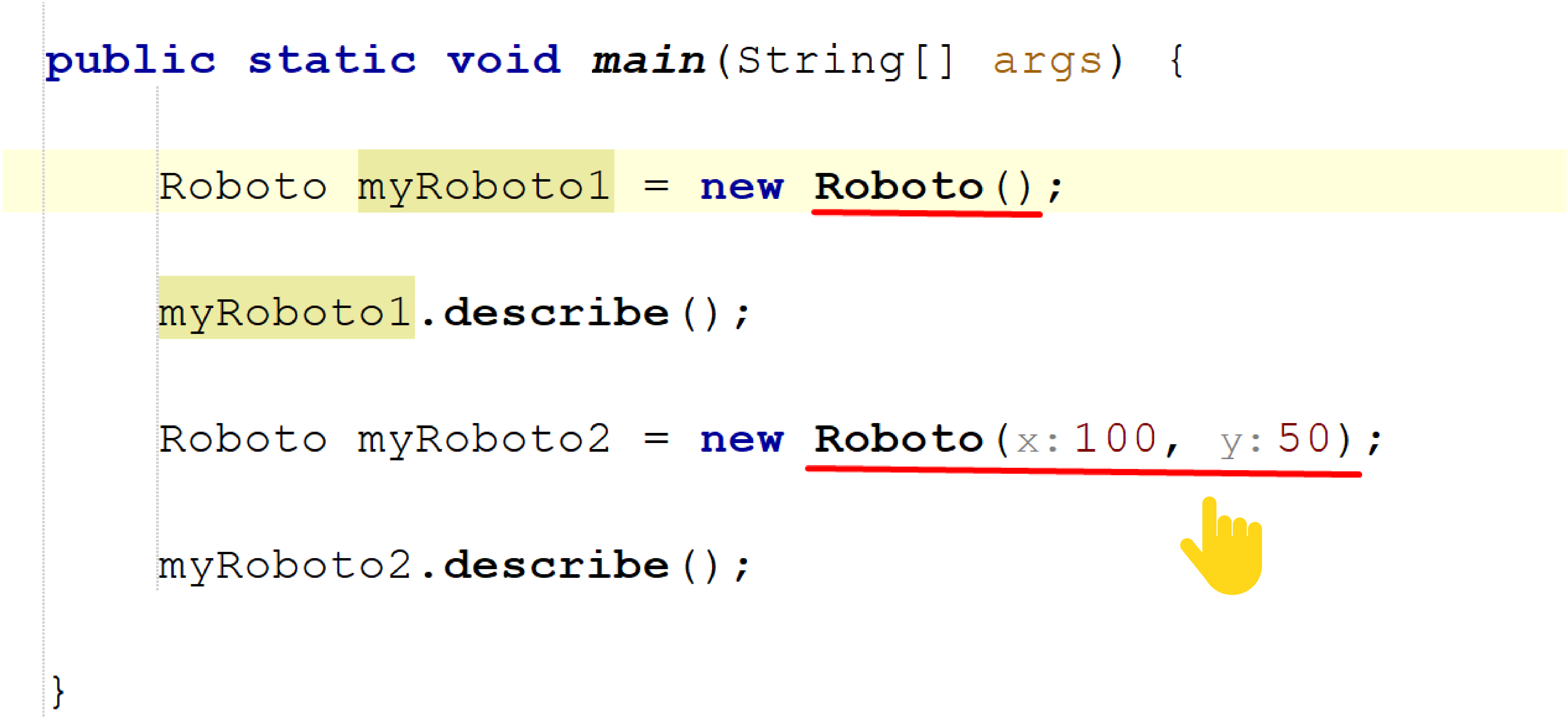
```java
public static void main(String[] args) {
    Product myProduct = new Product(name: "Coca Cola", price: 17.99, existances: 100);
    myProduct.describe();
}
```



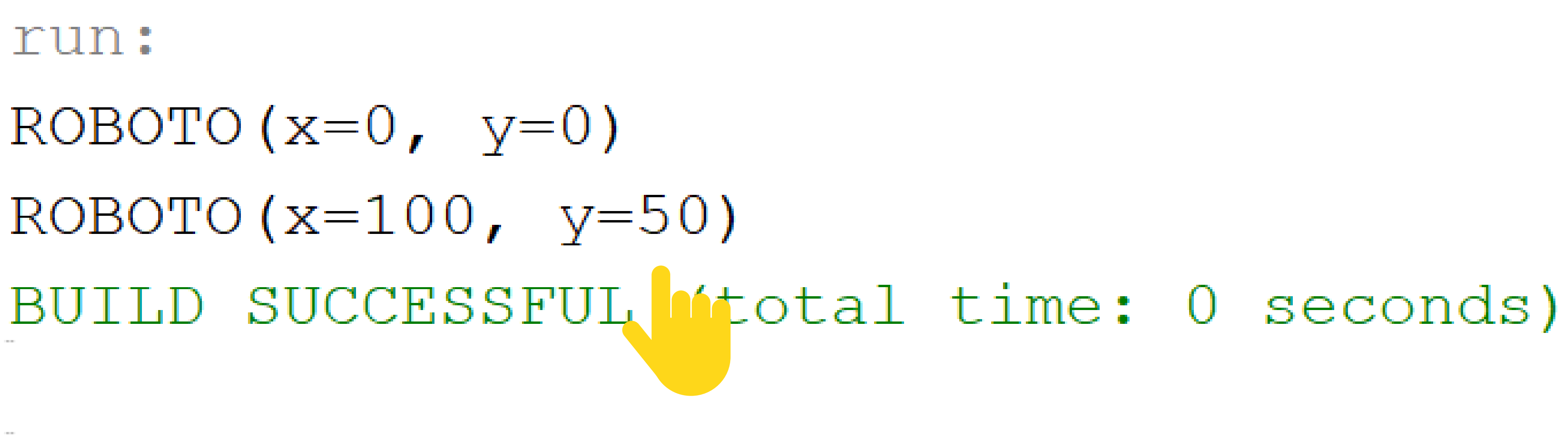
```
run:
Product
-----------------------
NAME: Coca Cola
PRICE: 17.99
EXISTANCES: 100
-----------------------
BUILD SUCCESSFUL (total time: 0 seconds)
```



```
class Roboto {

    int x;
    int y;

    public Roboto() {
        x = 0;
        y = 0;
    }

    public Roboto(int x, int y) {
        this.x = x;
        this.y = y;
    }

    public void describe() {
        System.out.printf(format: "ROBOTO(x=%d, y=%d) %n", args: x, args: y);
    }

}
```



```
public static void main(String[] args) {
    Roboto myRoboto1 = new Roboto();
    myRoboto1.describe();
    Roboto myRoboto2 = new Roboto(x: 100, y: 50);
    myRoboto2.describe();
}
```



```
run:
ROBOTO(x=0, y=0)
ROBOTO(x=100, y=50)
BUILD SUCCESSFUL (total time: 0 seconds)
```