

*ENTORNOS DE DESARROLLO*

*Mi Primer*

**TDD**

**Test-Driven-Development**

**{ ERROR!!!!!!! }**

DAW 1W 22/23

ALBERT PÉREZ BALEYTO

# **INDEX**

## **1. CREACIÓN DEL PROYECTO**

## **2. CREACIÓN DE TESTS**

### **A.PRIMER TEST**

### **B.SEGUNDO TEST**

### **C.TERCER TEST**

### **D.CUARTO TEST**

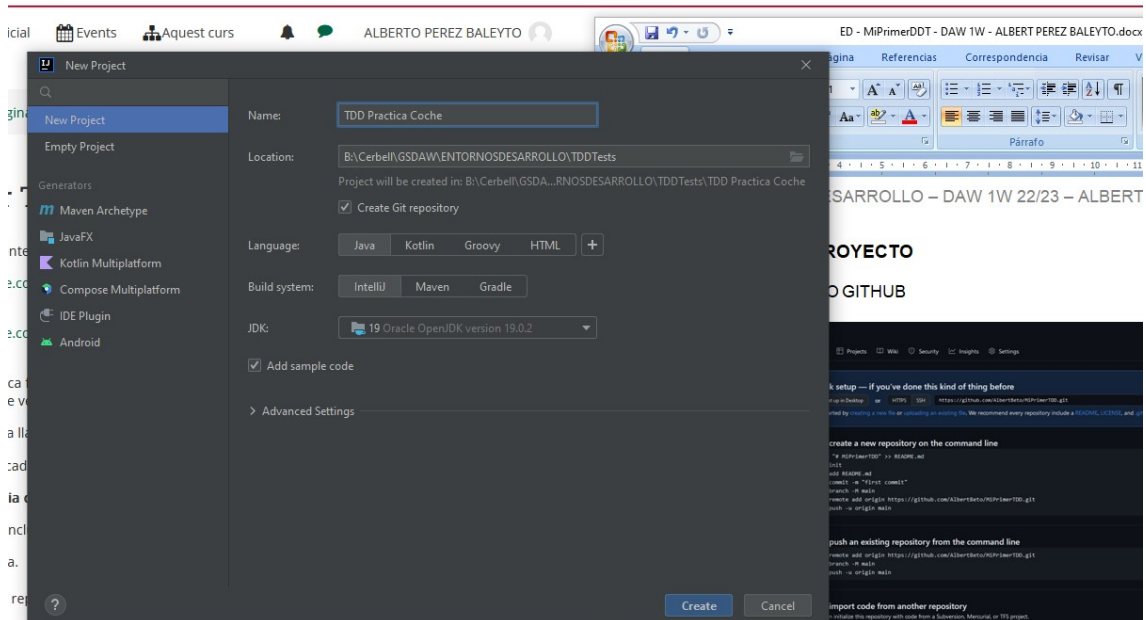
### **E.QUINTO TEST**

## **3. CONCLUSIONES**

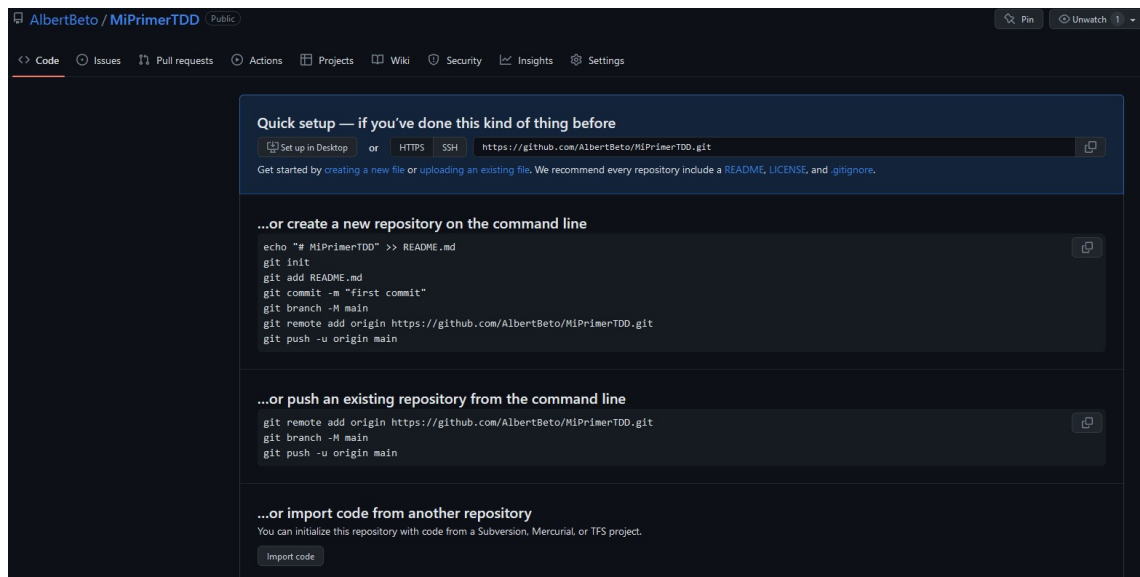
## **4. GIT HUB REPOSITORIO**

# 1-CREACIÓN DEL PROYECTO

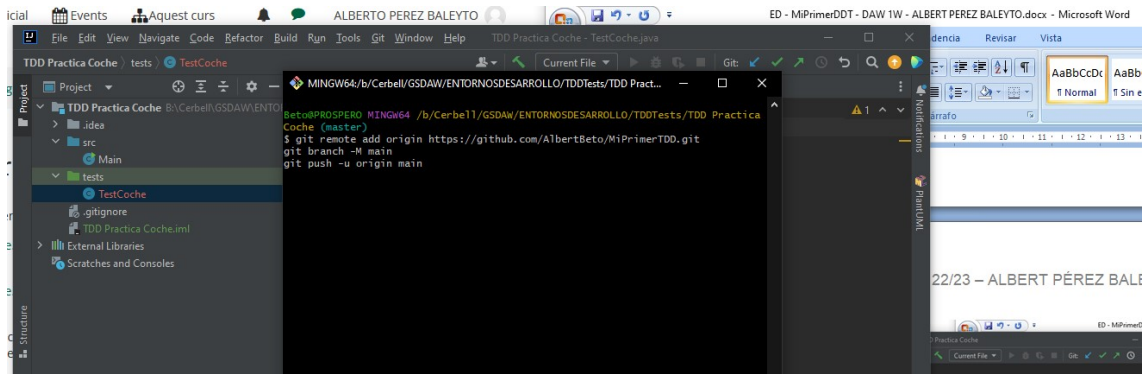
## CREACIÓN PROYECTO EN IntelliJ IDEA



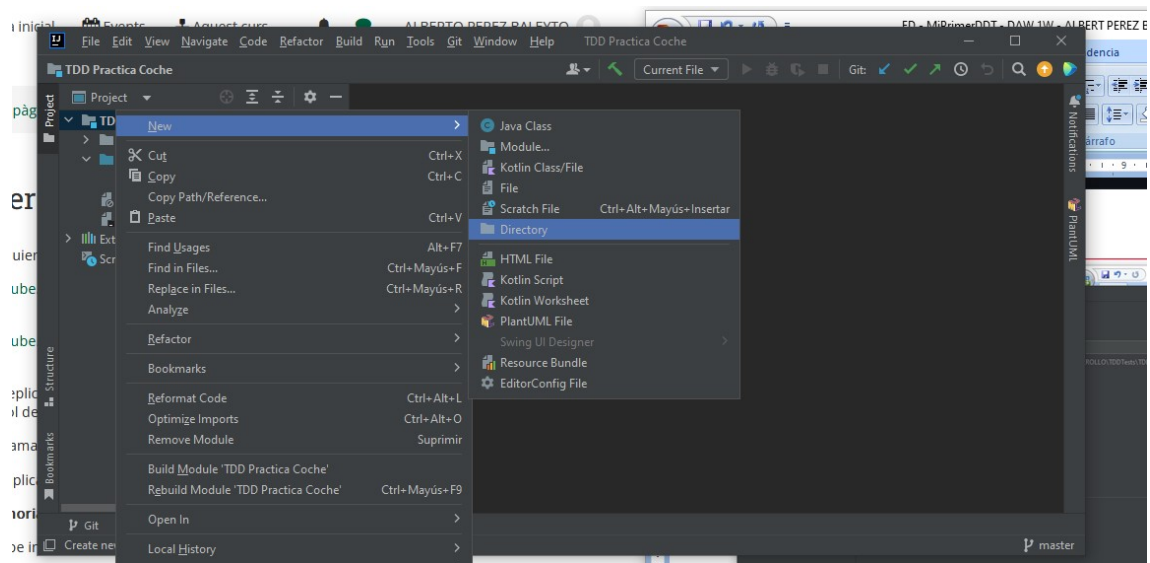
## CREO REPOSITORIO GITHUB



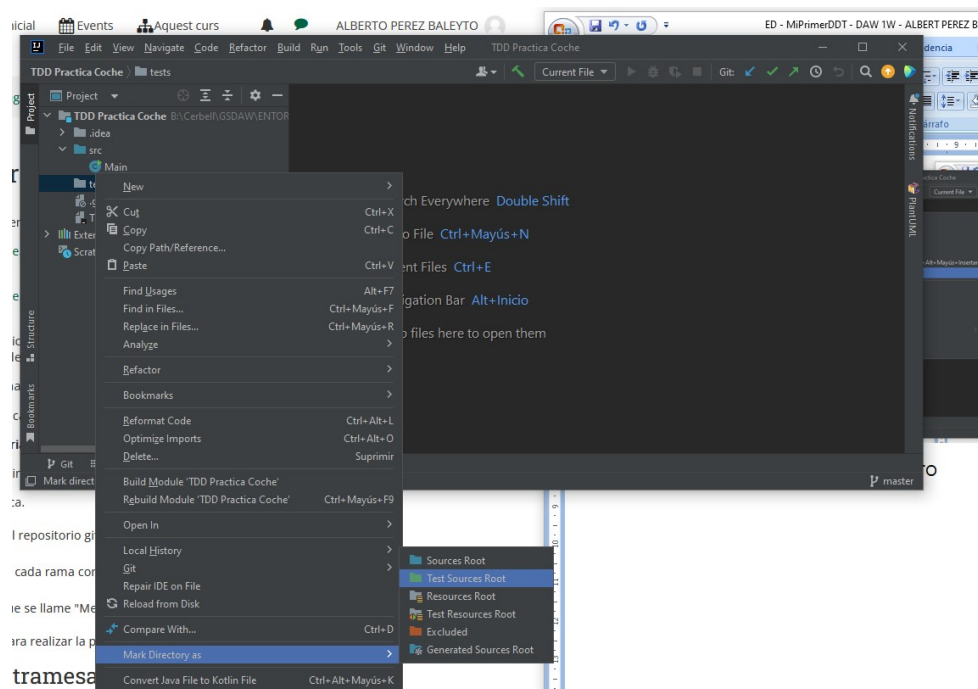
## ENTORNO DE DESARROLLO – DAW 1W 22/23 – ALBERT PÉREZ BALEYTO



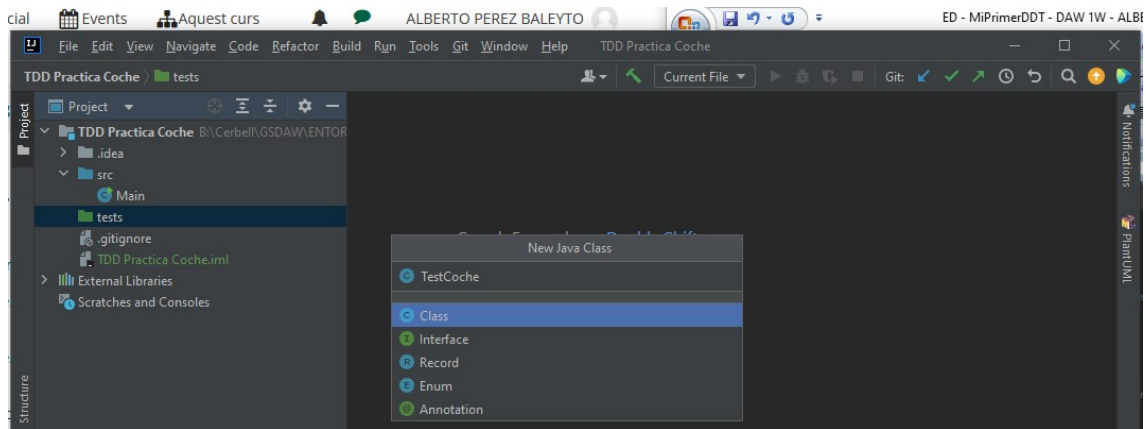
### CREACIÓN DE CARPETA



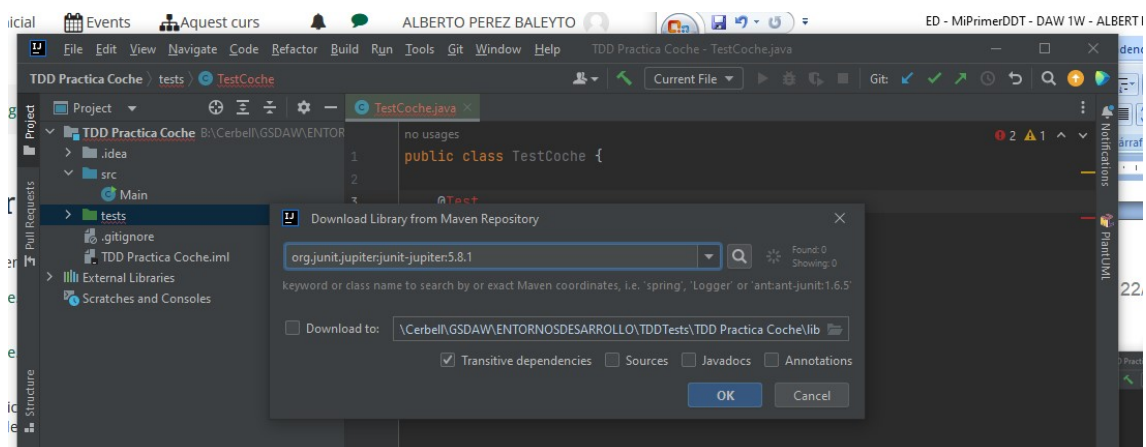
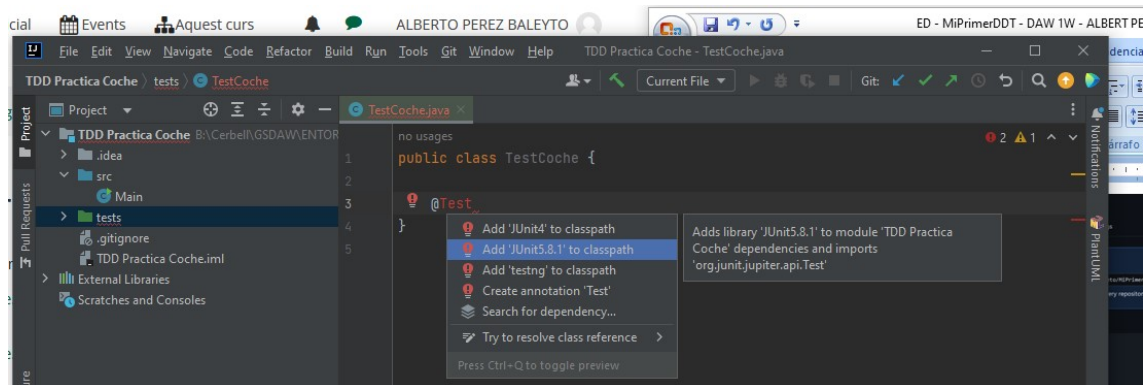
### IDENTIFICAR CARPETA TESTS COMO TEST SOURCES



## CREAMOS CLASS TestCoche EN LA CARPETA TESTS



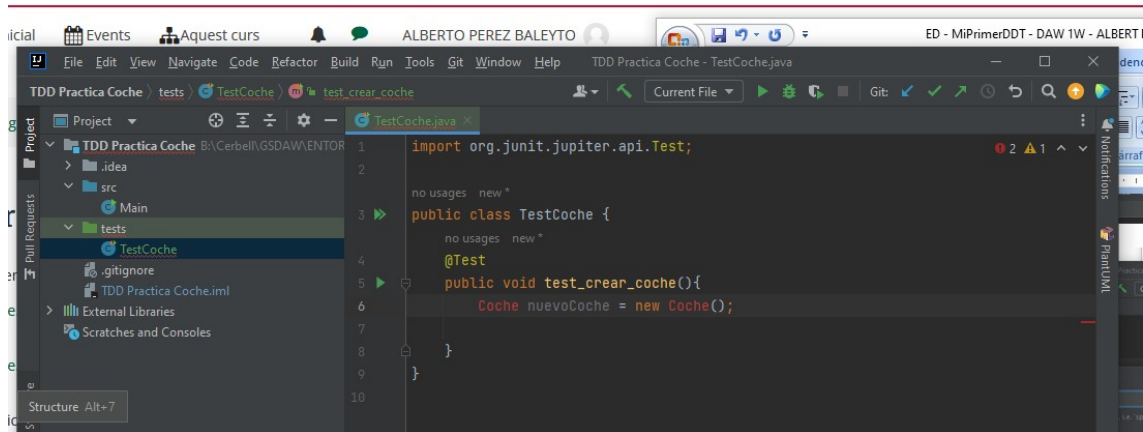
## IDENTIFICO LOS TESTS COMO JUNIT 5



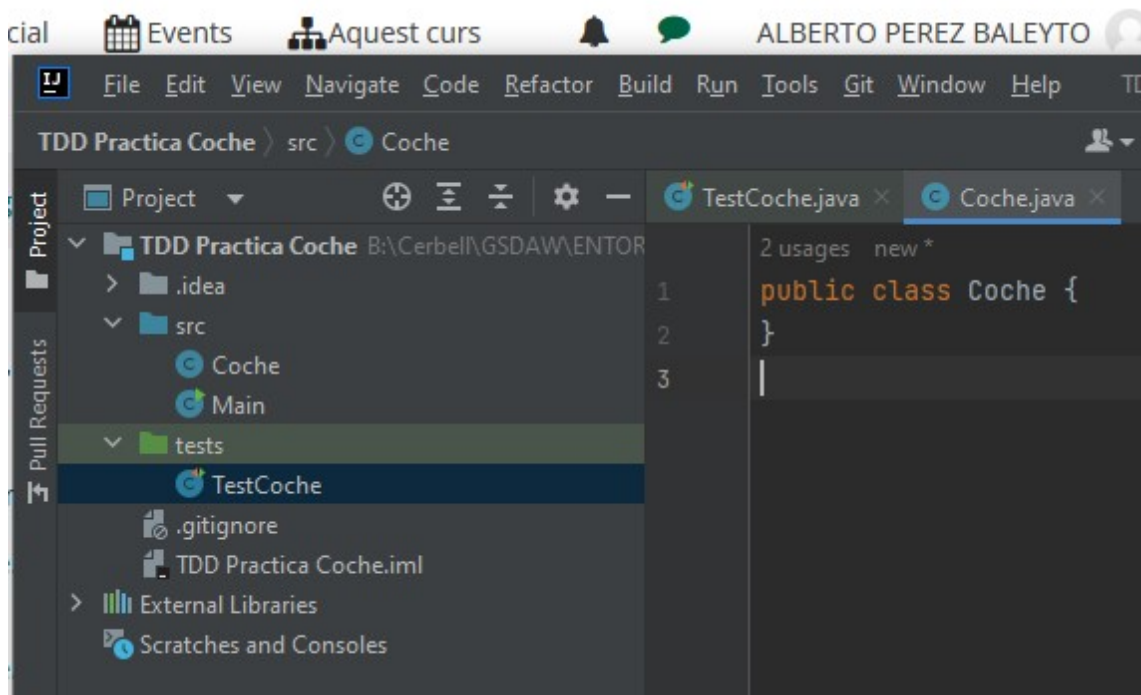
## 2-CREACION DE TESTS

### A) PRIMER TEST

ESCRIBO MI PRIMER TEST PARA CONFIRMAR EL CONSTRUCTOR

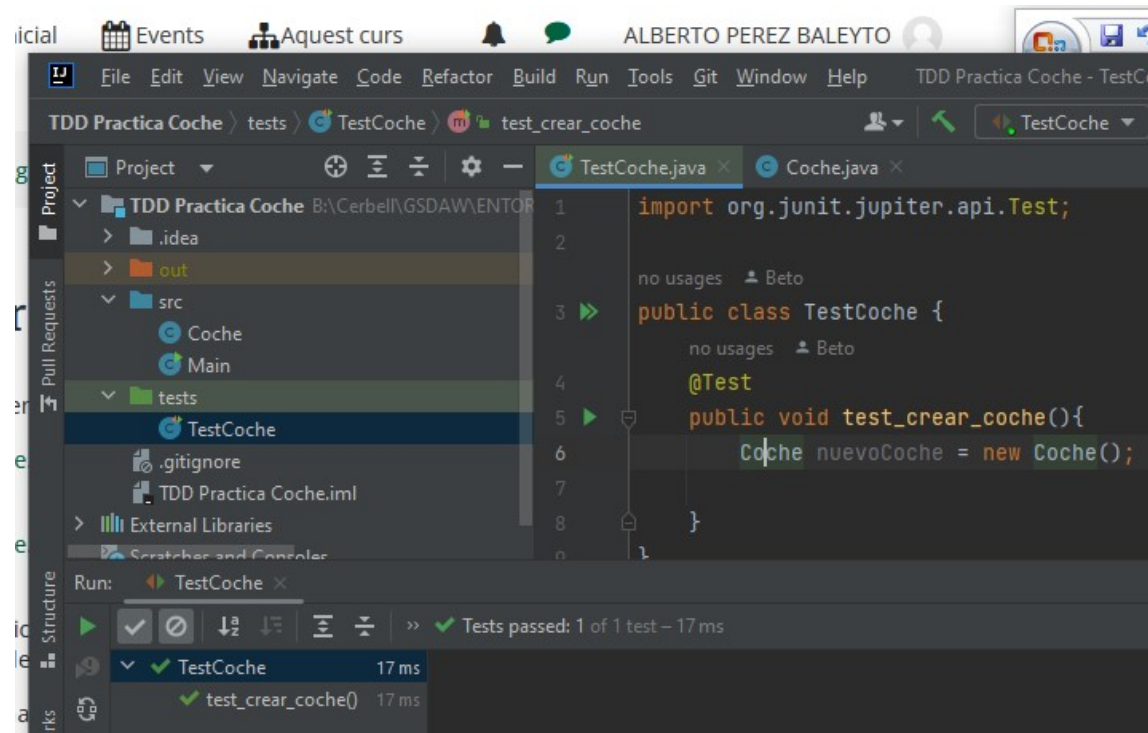


COMO NO TENGO NI CREADA LA CLASS Coche PROCEDO A ELLO



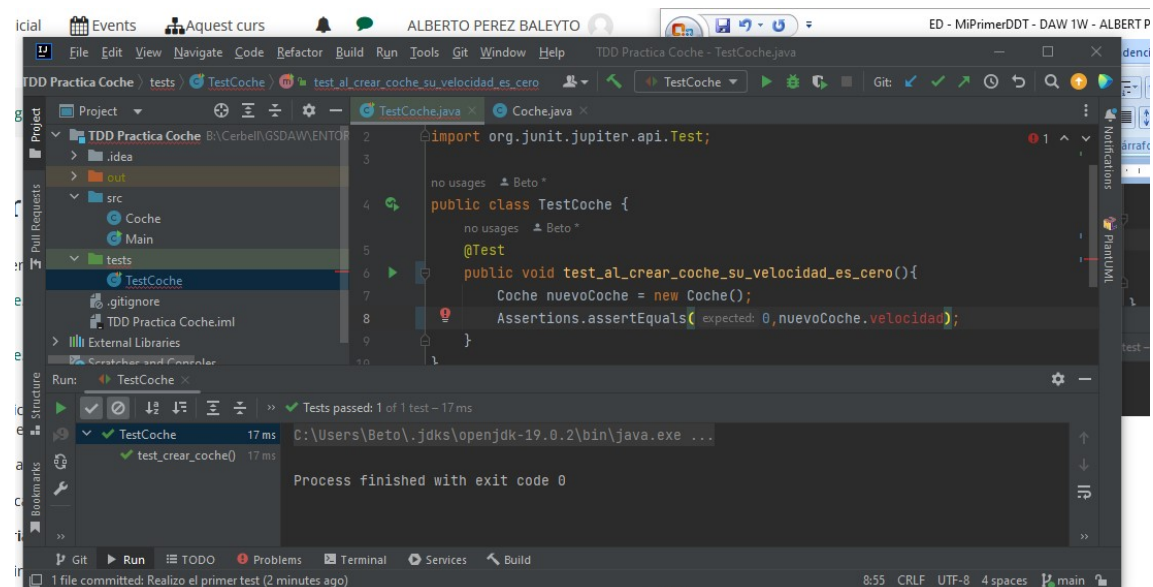


UNA VEZ CREADA LA CLASE PROCEDO A REALIZAR MI PRIMER TEST

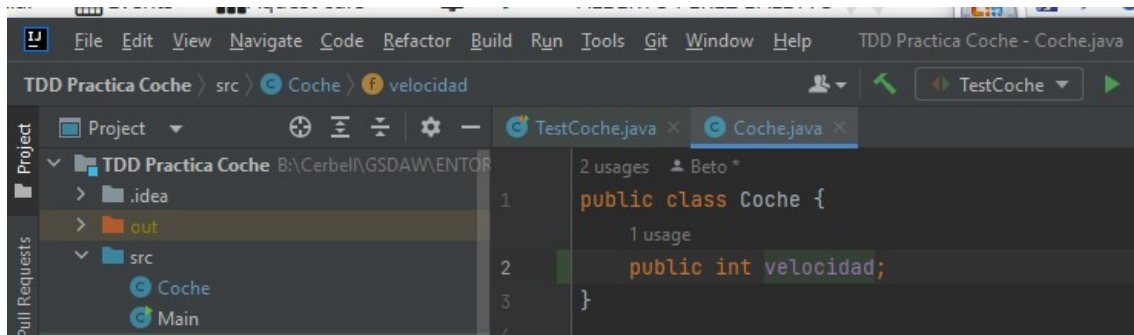


## B) SEGUNDO TEST

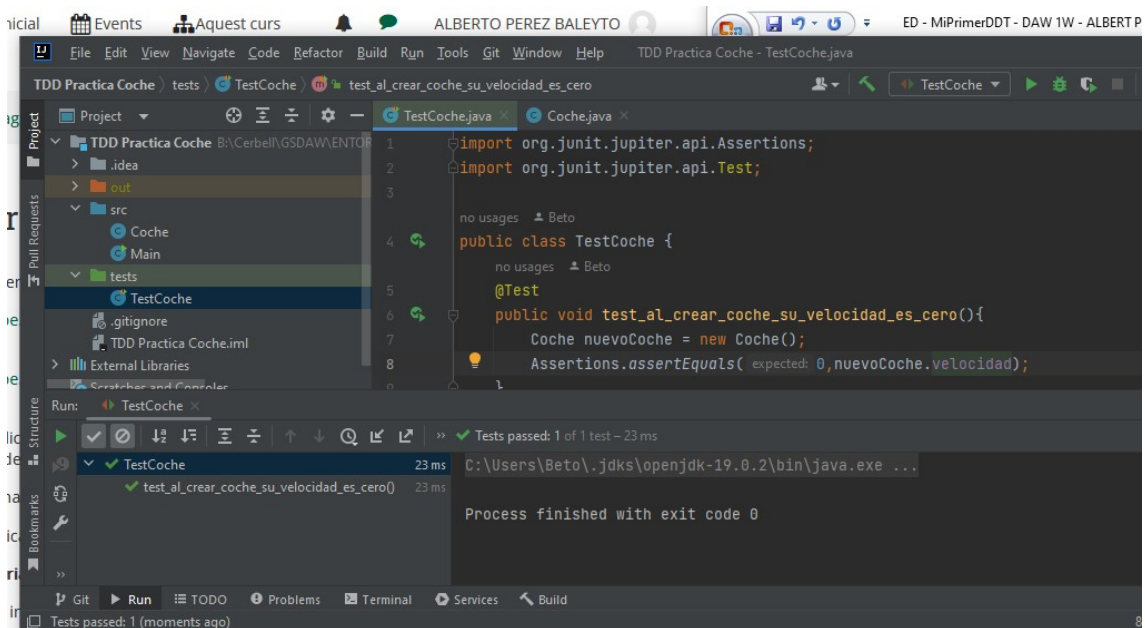
MODIFICO EL TEST PARA TESTEAR SI AL CREAR COCHE LA VELOCIDAD INICIAL ES 0



## MODIFICO CLASS Coche CREANDO LA VARIABLE VELOCIDAD



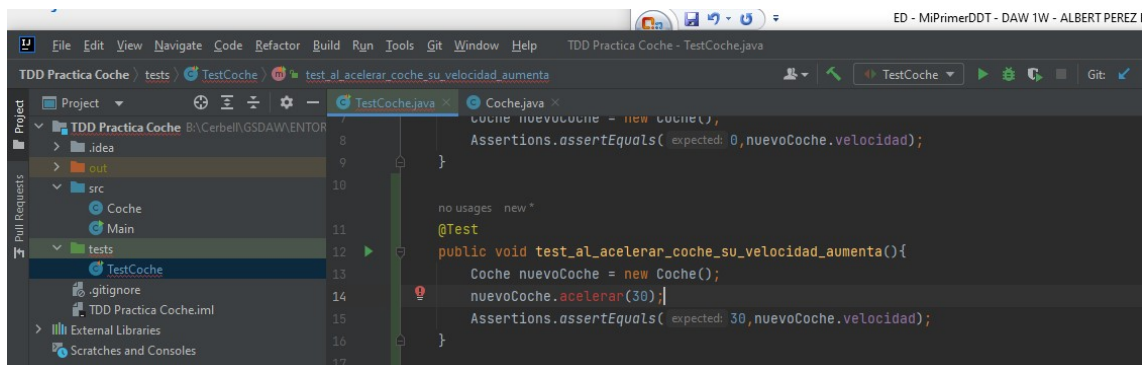
## REALIZO TEST Y CONFIRMO QUE FUNCIONA CORRECTAMENTE



VIDEO1 END

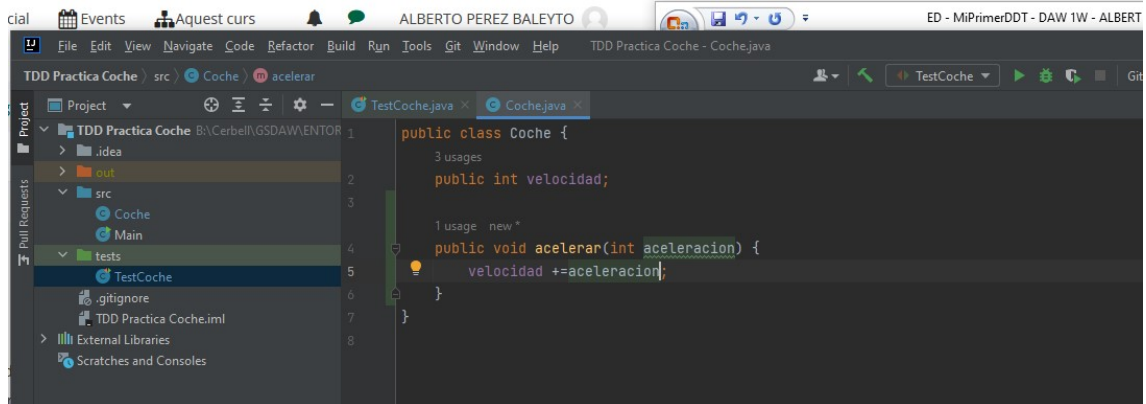
## C) TERCER TEST

CREAMOS UNA NUEVO TEST PARA REVISAR QUE VELOCIDAD AUMENTA CON EL METODO ACELERAR Y LE DAMOS UN VALOR.

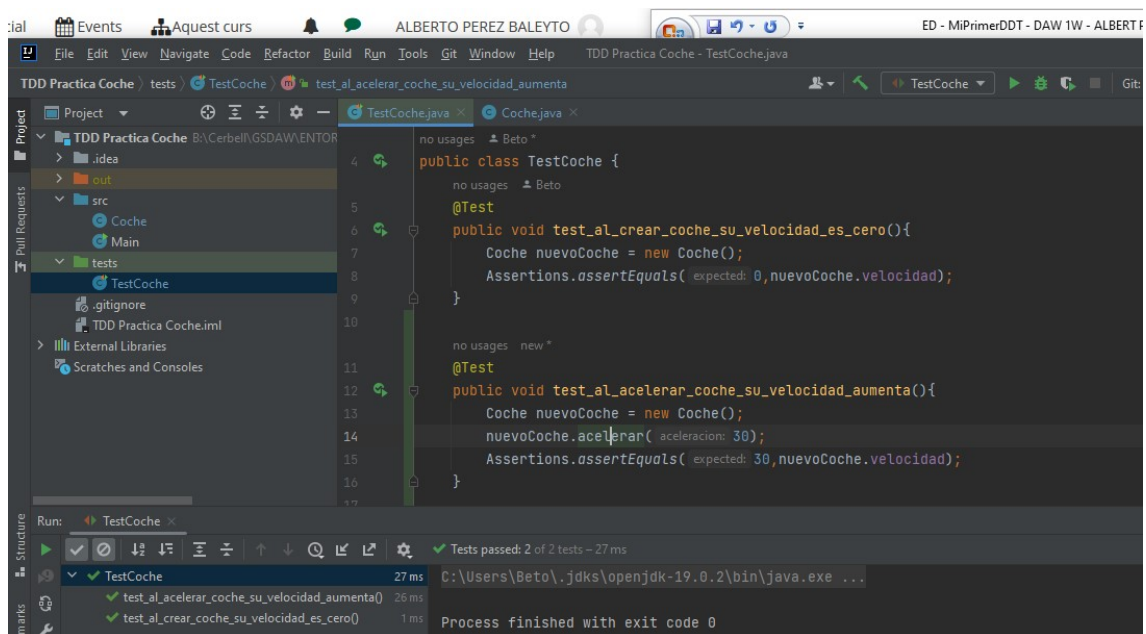




## PROCEDO A CREAR EL METODO ACELERAR EN COCHE QUE MODIFIQUE VELOCIDAD

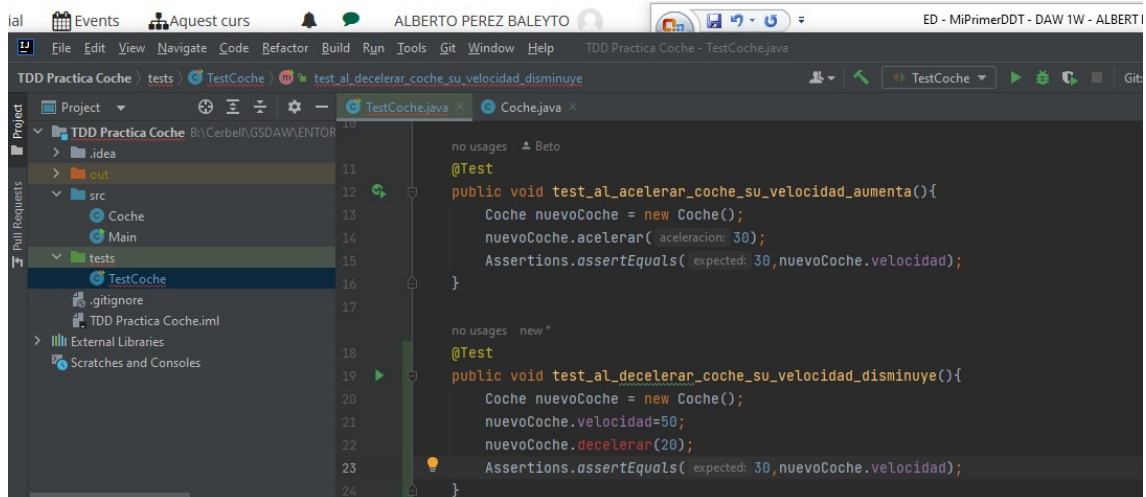


## PROCEDO A NUEVO TEST



## D) CUARTO TEST

CREO NUEVO TEST PARA TESTEAR QUE VELOCIDAD DECRECE AL DECELERAR

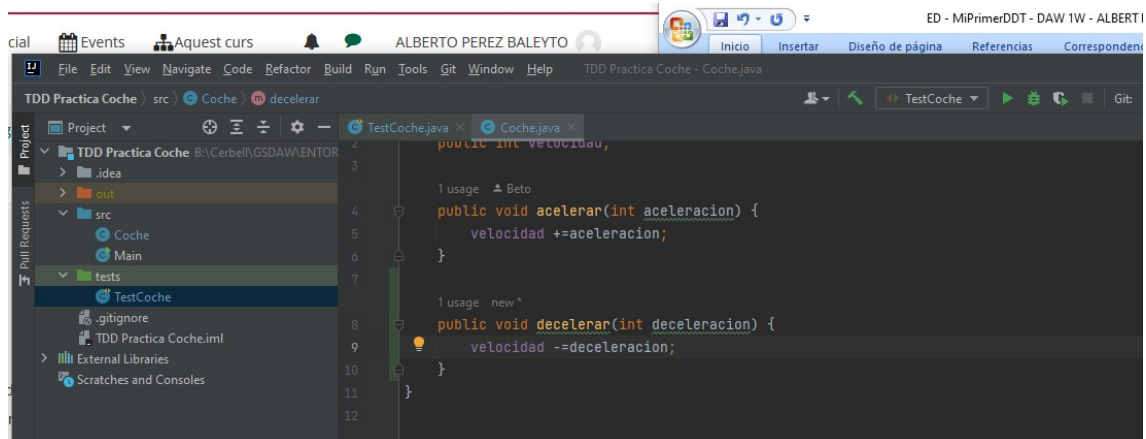


```

10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
no usages 1 Beto
@Test
public void test_al_acelerar_coche_su_velocidad_aumenta(){
    Coche nuevoCoche = new Coche();
    nuevoCoche.acelerar( aceleracion: 30);
    Assertions.assertEquals( expected: 30,nuevoCoche.velocidad);
}

no usages new *
@Test
public void test_al_decelerar_coche_su_velocidad_disminuye(){
    Coche nuevoCoche = new Coche();
    nuevoCoche.velocidad=50;
    nuevoCoche.decelerar(20);
    Assertions.assertEquals( expected: 30,nuevoCoche.velocidad);
}
    
```

CREO EL METODO DECELERAR



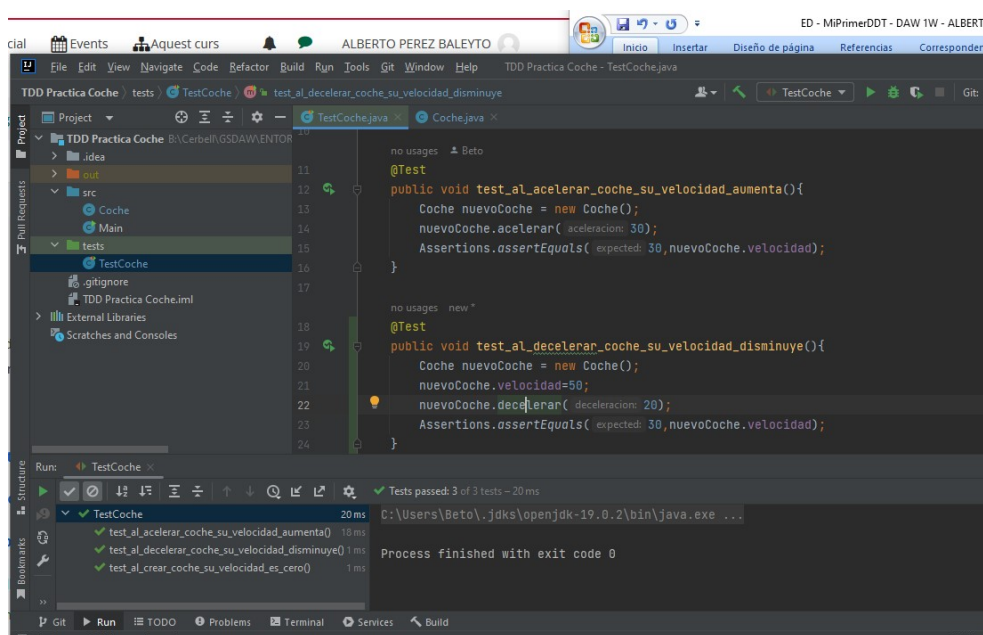
```

1
2
3
4
5
6
7
8
9
10
11
12
no usages 1 Beto
public int velocidad;

1 usage 1 Beto
public void acelerar(int aceleracion) {
    velocidad +=aceleracion;
}

1 usage new *
public void decelerar(int deceleracion) {
    velocidad -=deceleracion;
}
    
```

TESTEO QUE ESTE TODO BIEN



```

11
12
13
14
15
16
17
18
19
20
21
22
23
24
no usages 1 Beto
@Test
public void test_al_acelerar_coche_su_velocidad_aumenta(){
    Coche nuevoCoche = new Coche();
    nuevoCoche.acelerar( aceleracion: 30);
    Assertions.assertEquals( expected: 30,nuevoCoche.velocidad);
}

no usages new *
@Test
public void test_al_decelerar_coche_su_velocidad_disminuye(){
    Coche nuevoCoche = new Coche();
    nuevoCoche.velocidad=50;
    nuevoCoche.decelerar( deceleracion: 20);
    Assertions.assertEquals( expected: 30,nuevoCoche.velocidad);
}
    
```

Run: TestCoche

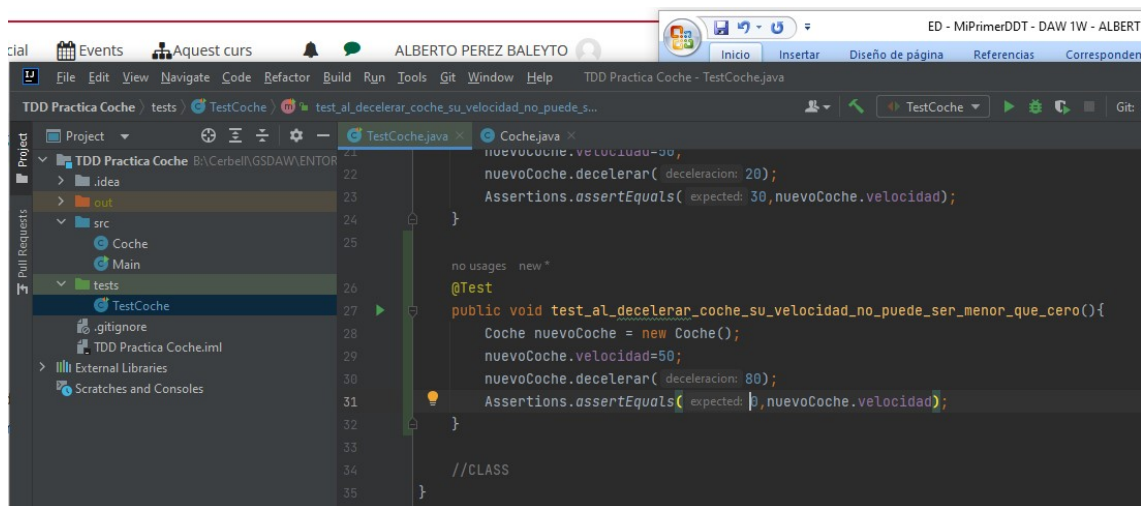
Tests passed: 3 of 3 tests – 20 ms

C:\Users\Beto\.jdk\openjdk-19.0.2\bin\java.exe ...

Process finished with exit code 0

## E) QUINTO TEST

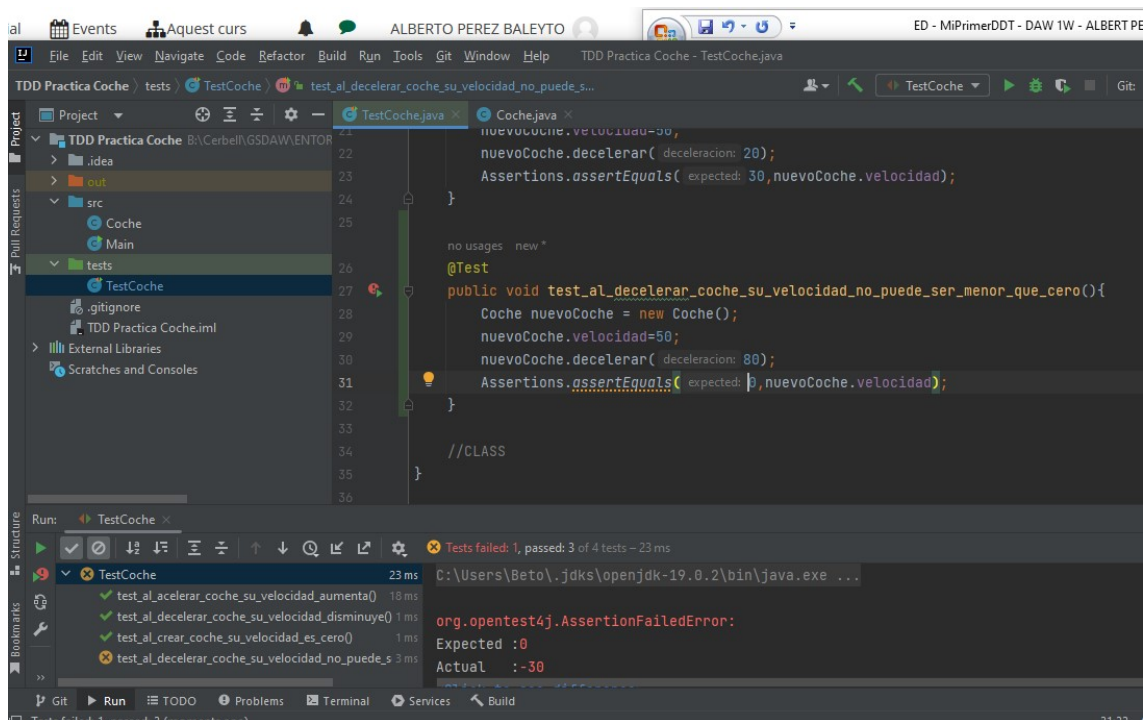
CREO TEST PARA CHECKEAR QUE AL DESACELERAR NO BAJA VELOCIDAD POR DEBAJO DE CERO



```

21 nuevoCoche.velocidad=30;
22 nuevoCoche.decelerar( deceleracion: 20);
23 Assertions.assertEquals( expected: 30,nuevoCoche.velocidad);
24 }
25
26 no usages new *
27 @Test
28 public void test_al_decelerar_coche_su_velocidad_no_puede_ser_menor_que_cero(){
29     Coche nuevoCoche = new Coche();
30     nuevoCoche.velocidad=50;
31     nuevoCoche.decelerar( deceleracion: 80);
32     Assertions.assertEquals( expected: 0,nuevoCoche.velocidad);
33 }
34
35 //CLASS
36 }
    
```

REALIZO EL TEST COMPLETO Y SALTA UN ERROR YA QUE NO SE HA MODIFICADO EL METODO PARA EVITAR QUE BAJE DE CERO. (REPLICANDO VIDEO)



```

21 nuevoCoche.velocidad=30;
22 nuevoCoche.decelerar( deceleracion: 20);
23 Assertions.assertEquals( expected: 30,nuevoCoche.velocidad);
24 }
25
26 no usages new *
27 @Test
28 public void test_al_decelerar_coche_su_velocidad_no_puede_ser_menor_que_cero(){
29     Coche nuevoCoche = new Coche();
30     nuevoCoche.velocidad=50;
31     nuevoCoche.decelerar( deceleracion: 80);
32     Assertions.assertEquals( expected: 0,nuevoCoche.velocidad);
33 }
34
35 //CLASS
36 }
    
```

Run: TestCoche

Tests failed: 1, passed: 3 of 4 tests – 23 ms

TestCoche 23 ms C:\Users\Beto\.jdk\openjdk-19.0.2\bin\java.exe ...

test\_al\_acelerar\_coche\_su\_velocidad\_aumenta() 16 ms

test\_al\_decelerar\_coche\_su\_velocidad\_disminuye() 1 ms

test\_al\_crear\_coche\_su\_velocidad\_es\_cero() 1 ms

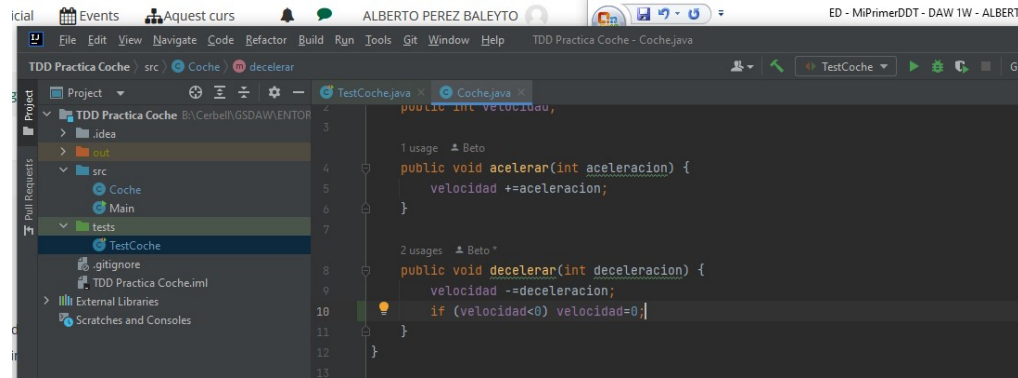
test\_al\_decelerar\_coche\_su\_velocidad\_no\_puede\_s 3 ms

org.opentest4j.AssertionFailedError:

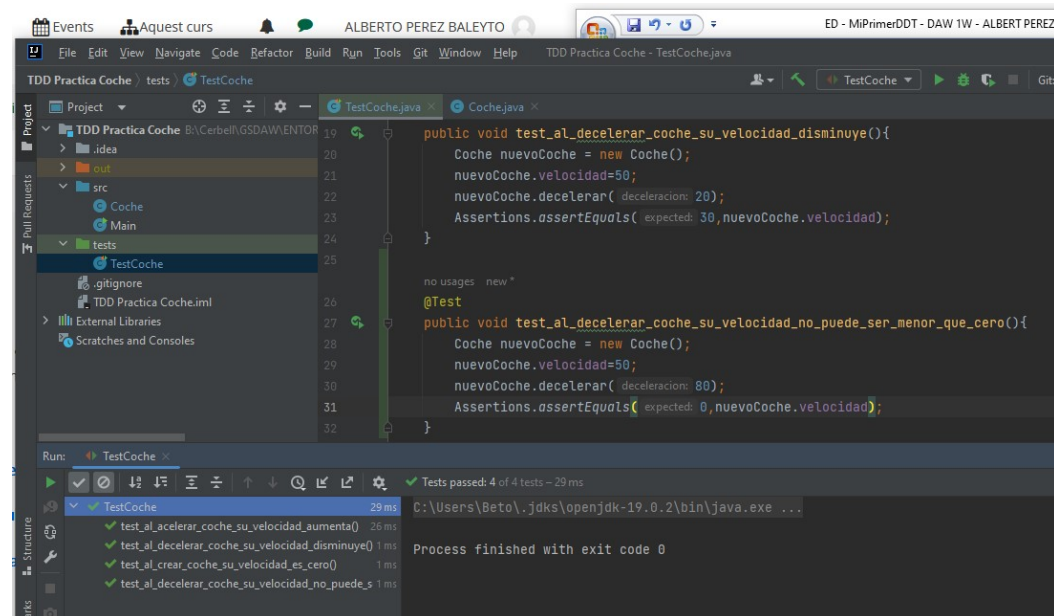
Expected :0

Actual :-30

## PROCEDO A MODIFICAR EL METODO DECELERACIÓN PARA QUE NO BAJE DE 0



## REALIZO TEST PARA CONFIRMAR QUE FUNCIONA BIEN



**TODO OK!!**

## 3-CONCLUSIONES

Una vez realizado este ejercicio vemos claramente cómo funciona el sistema TDD.

Creamos el test con el resultado que debe dar y con los valores de entrada determinados.

Confirmamos que da error para asegurar que el test está bien escrito y así también evitar falsos positivos.

Escribimos el código, lo testeamos y lo trabajamos hasta que consigamos el OK.

Y volvemos a empezar con un nuevo test.

## 4-GITHUB REPOSITORIO

<https://github.com/AlbertBeto/MiPrimerTDD.git>

