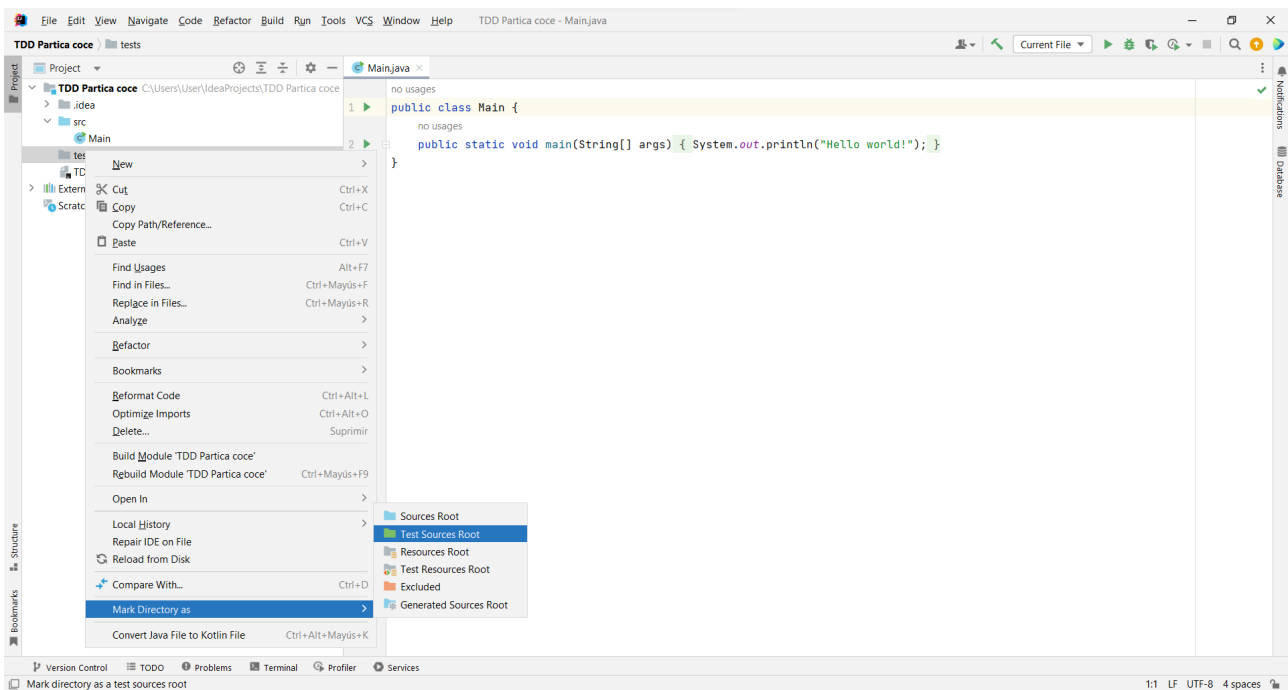
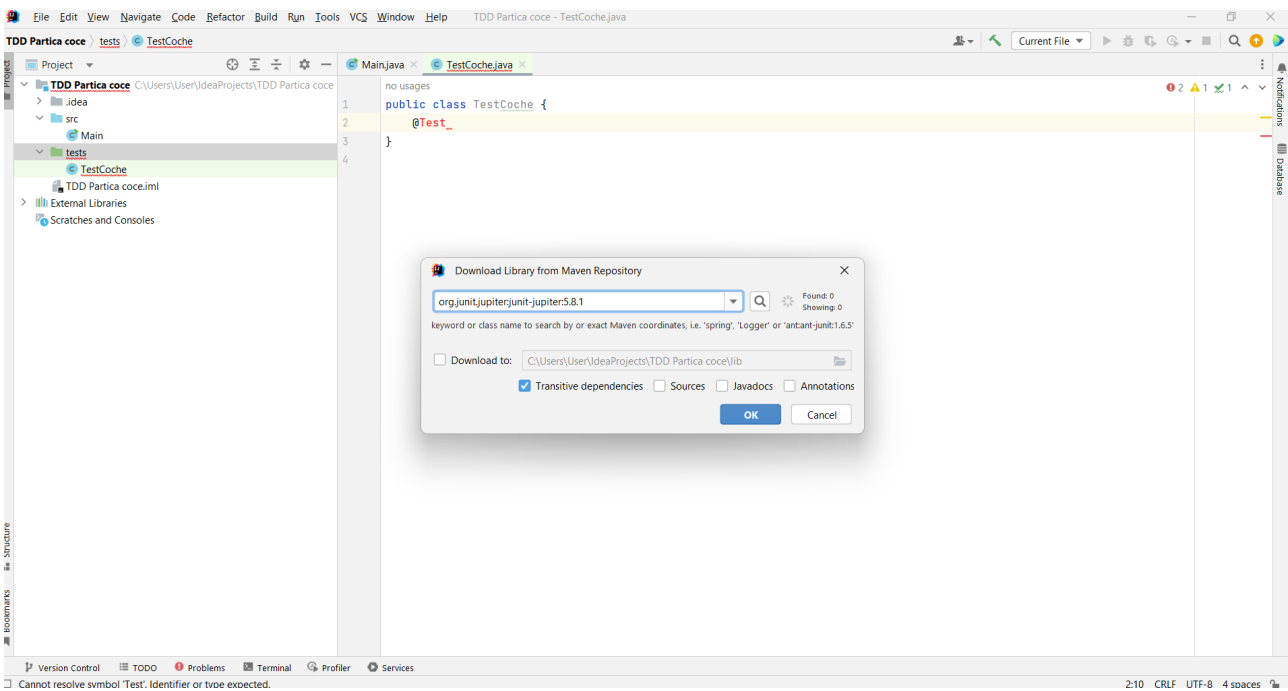


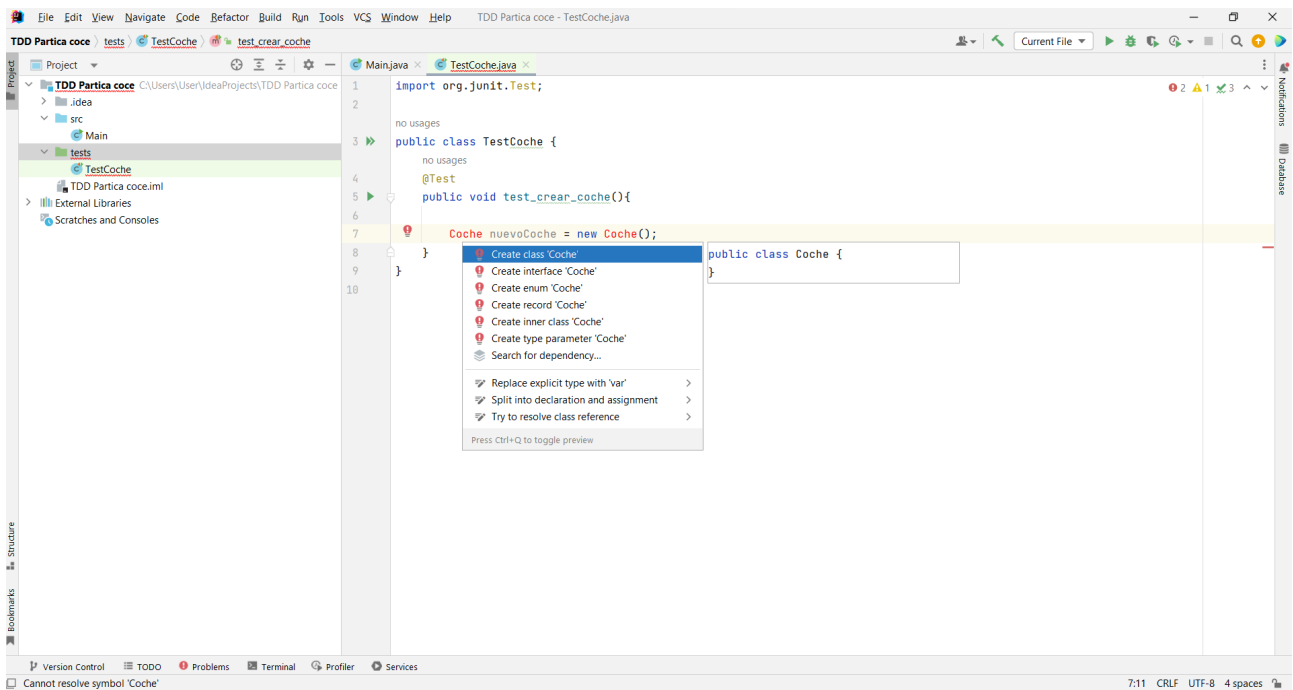
1) Despues de crear el proyecto y crear un nuevo directorio llamado test lo marcamos como test:



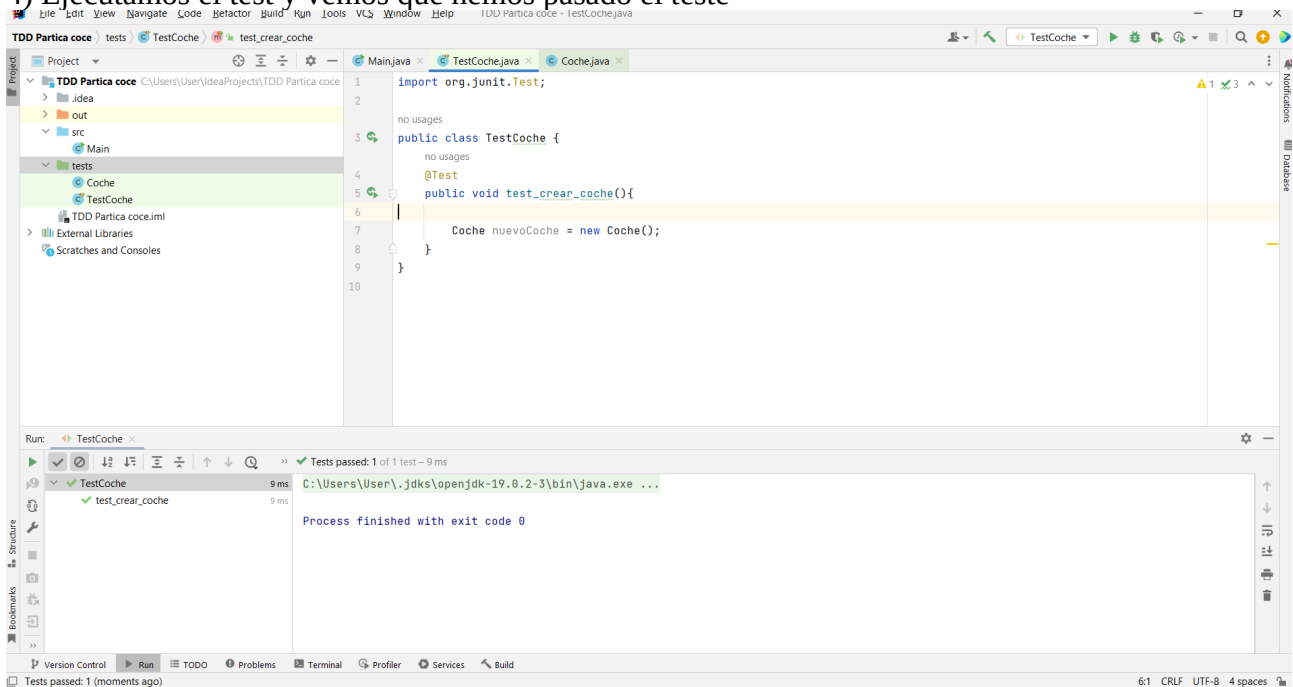
2) Creamos la clase TestCoche al hacer la opicon @Test no existe y tenemos que alt+enter para decirle Uni 5



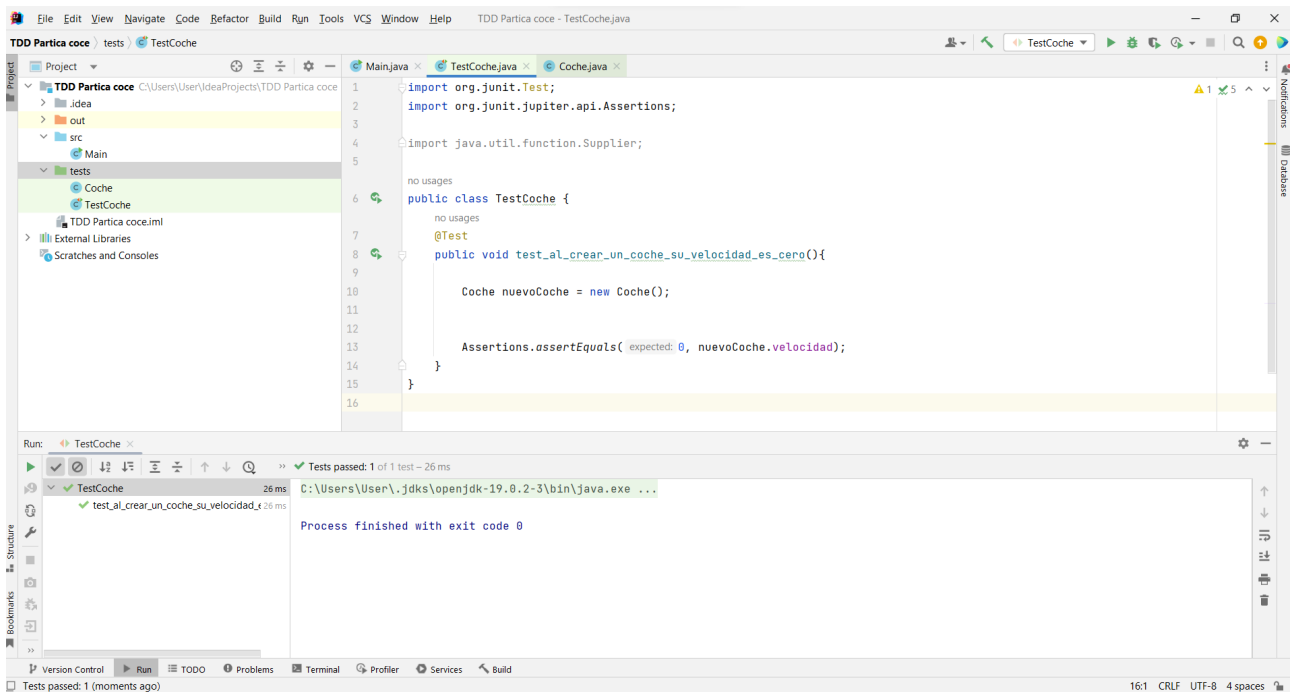
3) Creamos un test y al no tener la clase coche creada la creamos tambien automaticamente mediante alt+enter



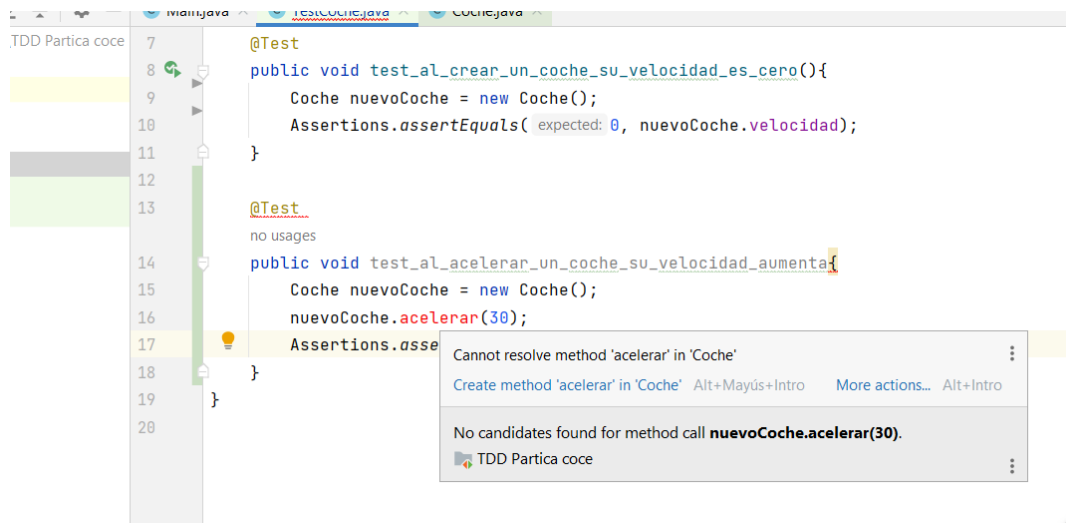
4) Ejecutamos el test y vemos que hemos pasado el teste



5) Cambiamos el Test y realizamos el nuevo test y nos vuelve a salir correcto



6) Creamos el nuevo test para acelerar y creamos el metodo acelerar de forma atomica



7) Volvemos a ejecutar y el test pasa correctamente



The screenshot shows an IDE with a Java class `TestCoche` and a test method `test_al_acelerar_un_coche_su_velocidad_aumenta()`. The test method creates a `Coche` object, calls `acelerar(30)`, and asserts that the velocity is 30. The test is highlighted in yellow. Below the code, the Run window shows the test passing successfully.

```
10     Assertions.assertEquals( expected: 0, nuevoCoche.velocidad);
11 }
12
13 no usages new *
14 @Test
15 public void test_al_acelerar_un_coche_su_velocidad_aumenta(){
16     Coche nuevoCoche = new Coche();
17     nuevoCoche.acelerar( aceleracion: 30);
18     Assertions.assertEquals( expected: 30, nuevoCoche.velocidad);
19 }
```

Run: TestCoche.test_al_acelerar_un_coche_su_velocidad_aumenta

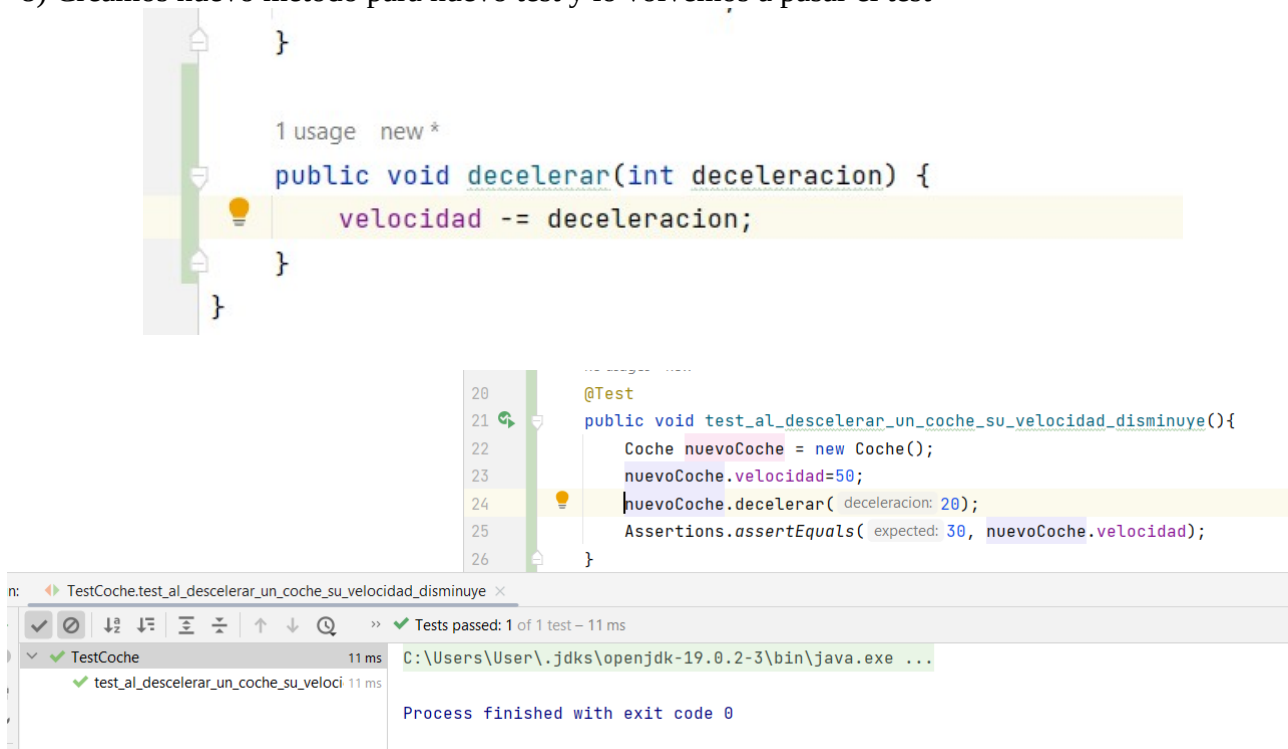
Tests passed: 1 of 1 test - 0 ms

C:\Users\User\.jdk\openjdk-19.0.2-3\bin\java.exe ...

test_al_acelerar_un_coche_su_velocidad_aumenta 0 ms

Process finished with exit code 0

8) Creamos nuevo metodo para nuevo test y lo volvemos a pasar el test



The screenshot shows the `Coche` class with a new method `decelerar(int deceleracion)` added. The method subtracts the deceleration from the velocity. Below the code, the Run window shows a new test `test_al_descelerar_un_coche_su_velocidad_disminuye()` passing successfully.

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

```
20 @Test
21 public void test_al_descelerar_un_coche_su_velocidad_disminuye(){
22     Coche nuevoCoche = new Coche();
23     nuevoCoche.velocidad=50;
24     nuevoCoche.decelerar( deceleracion: 20);
25     Assertions.assertEquals( expected: 30, nuevoCoche.velocidad);
26 }
```

Run: TestCoche.test_al_descelerar_un_coche_su_velocidad_disminuye

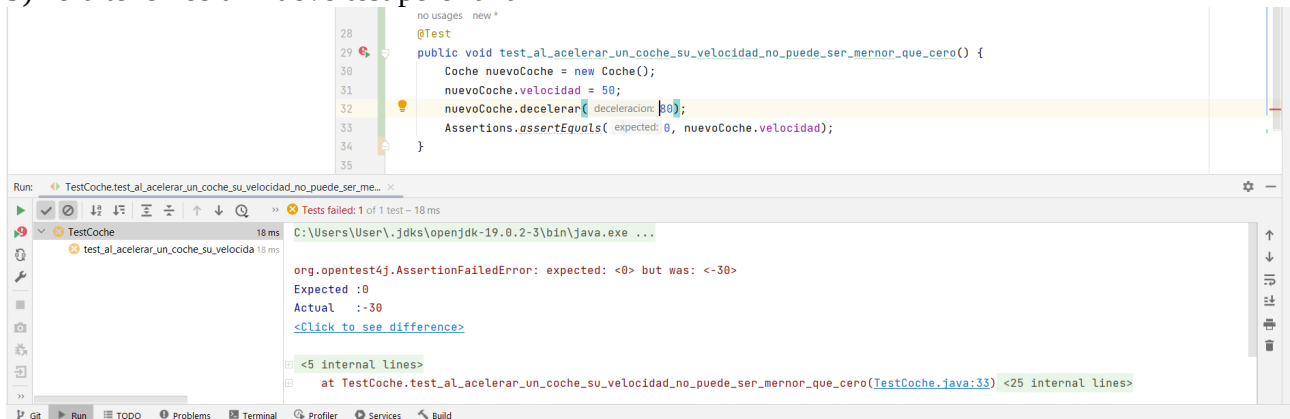
Tests passed: 1 of 1 test - 11 ms

C:\Users\User\.jdk\openjdk-19.0.2-3\bin\java.exe ...

test_al_descelerar_un_coche_su_velocidad_disminuye 11 ms

Process finished with exit code 0

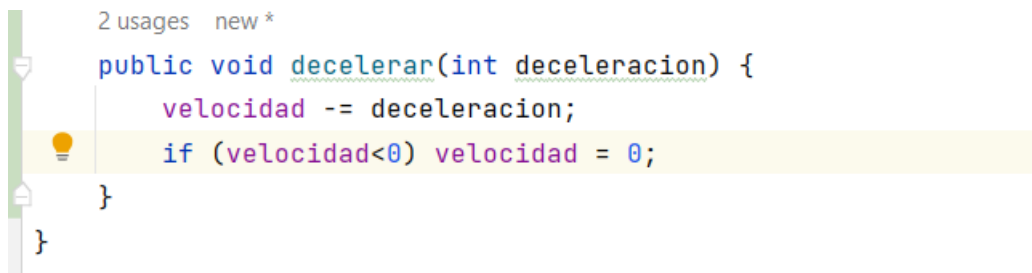
9) Hora tenemos un nuevo test pero falla



miramos para ver cual es el problema del test y vemos que espera 0 y el valor real fue -30

```
org.opentest4j.AssertionFailedError: expected: <0> but was: <-30>
Expected :0
Actual :-30
<Click to see difference>
```

para arreglar esto lo que debemos hacer es que la velocidad no pueda ser menor que 0 para eso arreglamos el metodo en la classe coche



vemos que todos los test funcionan correctamente

