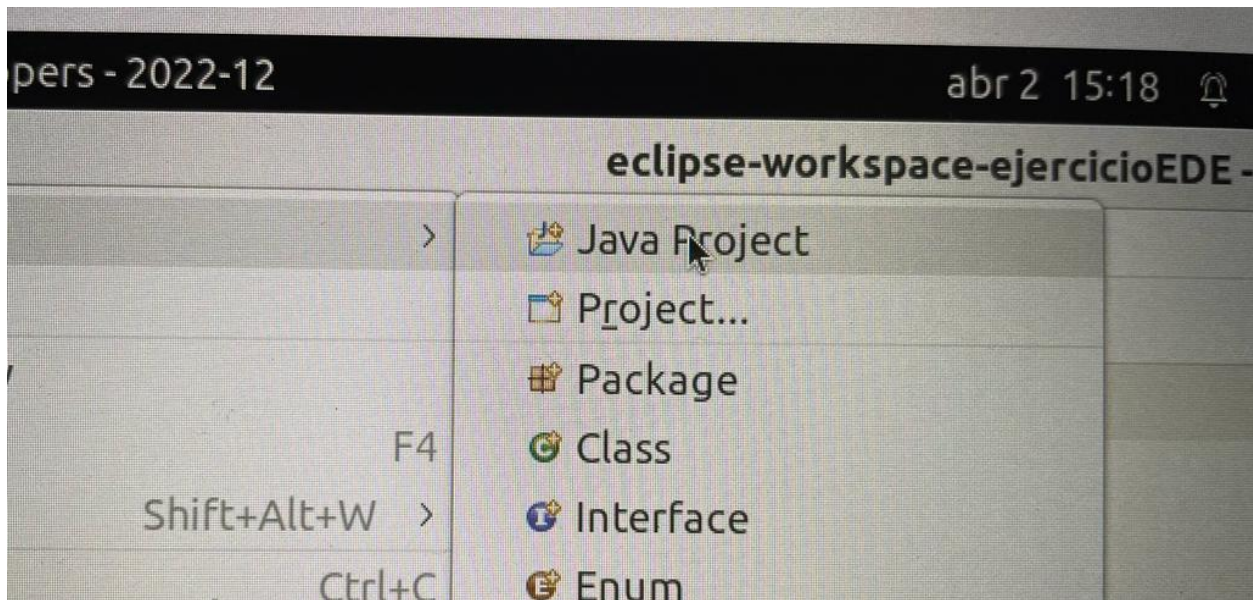


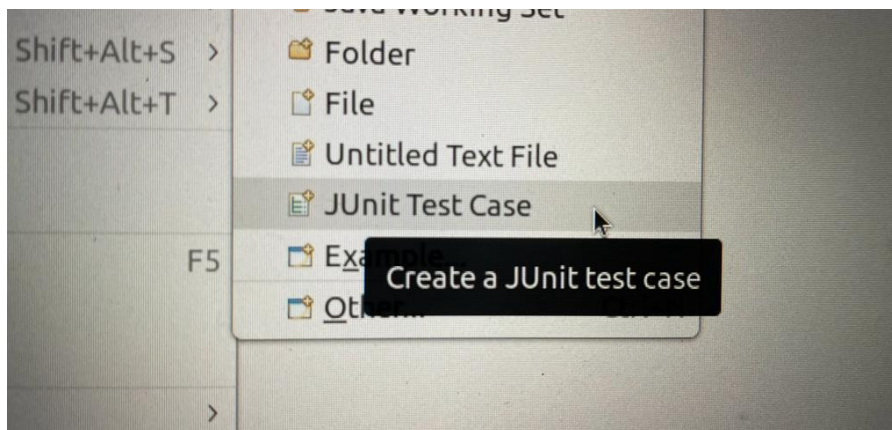
## Memoria

### Mi primer TDD V2.0

1. Primero abrimos el entorno de desarrollo Eclipse, le damos a crear nuevo proyecto + proyecto JAVA.



2. . Una vez creado, se le da click derecho en el nombre del proyecto y pulsamos JUnit Test Case.



3. Agregamos el nombre del test y una vez creado aparece una clase con `@Test` implementado para iniciar las pruebas, en este caso volver hacer el TDD del coche.

```

3 import org.junit.jupiter.api.Assertions;
4 import org.junit.jupiter.api.Test;
5
6 class TestsBasicos {
7 |
8 @Test
9
10 public void test_crear_coche_su_velocidad_es_cero() {
11
12     Coche nuevoCoche = new Coche();
13     Assertions.assertEquals(0,nuevoCoche.velocidad);
14 }
15
16 @Test
17 public void test_acelerar_un_coche_su_velocidad_aumenta() {
18     Coche nuevoCoche = new Coche();
19     nuevoCoche.acelerar(30);
20     Assertions.assertEquals(30,nuevoCoche.velocidad);
21 }
22
23 @Test
24 public void test_decelerar_un_coche_su_velocidad_disminuye() {
25     Coche nuevoCoche = new Coche();
26     nuevoCoche.velocidad = 50;
27     nuevoCoche.decelerar(20);
28     Assertions.assertEquals(30,nuevoCoche.velocidad);
29 }
30
31 @Test
32 public void test_decelerar_un_coche_su_velocidad_no_puede_ser_menor_que_cero() {
33     Coche nuevoCoche = new Coche();
34     nuevoCoche.velocidad = 50;
35     nuevoCoche.decelerar(80);

```

4. Creamos los métodos siguientes:

```

5. @Test
6. public void test_crear_coche_su_velocidad_es_cero_Elisabet_Agullo() {
7.     Coche nuevoCoche = new Coche();
8.     Assertions.assertEquals(0,nuevoCoche.velocidad);
9. }
10.     @Test
11.     public void
12.     test_acelerar_un_coche_su_velocidad_aumenta_Elisabet_Agullo() {
13.         Coche nuevoCoche = new Coche();
14.         nuevoCoche.acelerar(30);
15.         Assertions.assertEquals(30,nuevoCoche.velocidad);
16.     }
17.     @Test

```

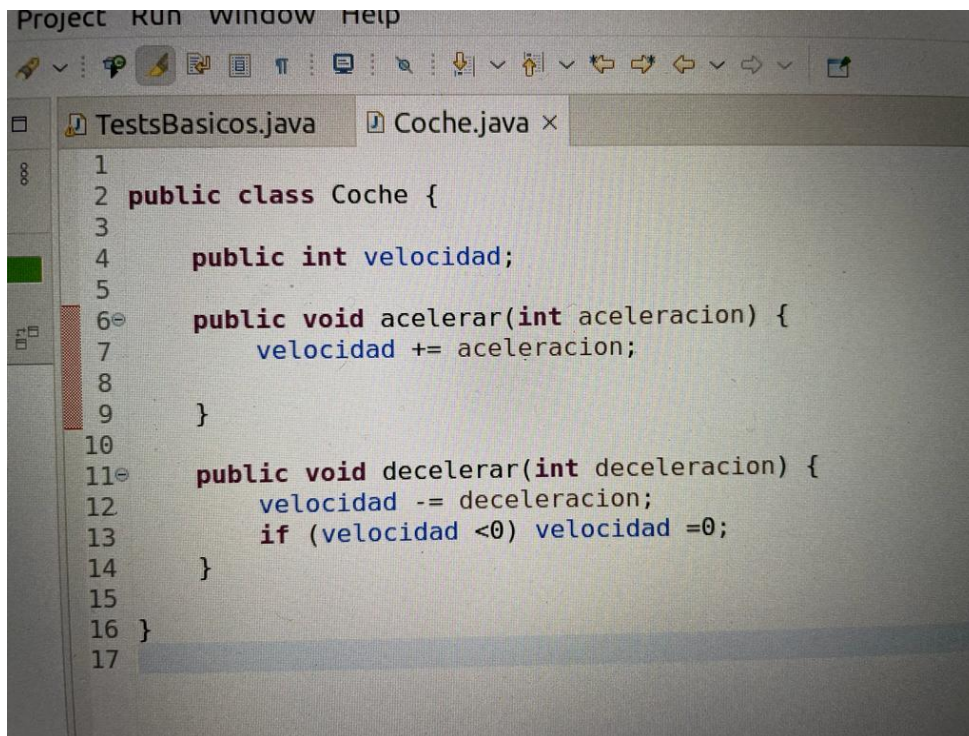


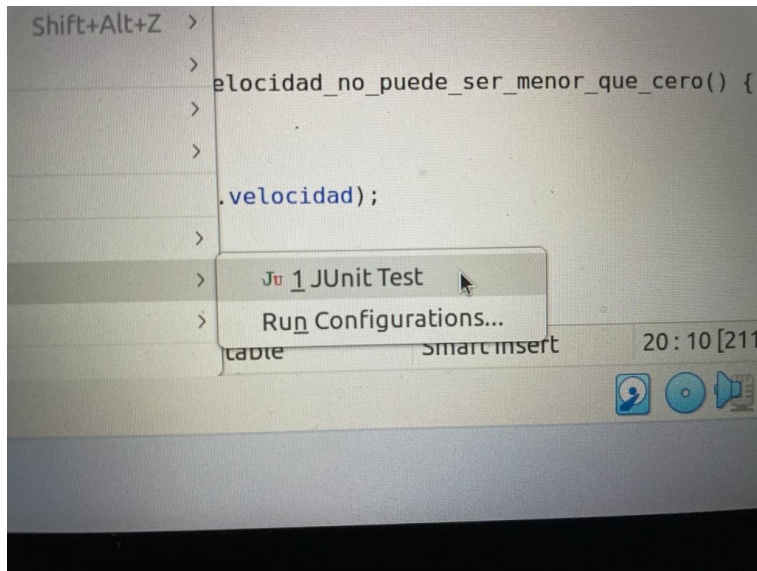
```

17.     public void
    test_decelerar_un_coche_su_velocidad_disminuye_Elisabet_Agullo() {
18.         Coche nuevoCoche = new Coche();
19.         nuevoCoche.velocidad = 50;
20.         nuevoCoche.decelerar(20);
21.         Assertions.assertEquals(30,nuevoCoche.velocidad);
22.     }
23.     @Test
24.     public void
    test_decelerar_un_coche_su_velocidad_no_puede_ser_menor_que_cero_Elisab
    et_Agullo() {
25.         Coche nuevoCoche = new Coche();
26.         nuevoCoche.velocidad = 50;
27.         nuevoCoche.decelerar(80);
28.         Assertions.assertEquals(0,nuevoCoche.velocidad);
29.     }

```

30. Luego creamos los métodos en la clase coche y hacemos test si funcionan correctamente.





31. Creamos la rama Refactorizado y cambiamos el nombre de los métodos con mi nombre.

32. Por último, creamos otra rama llamada memoria y añadimos un pdf de la memoria.