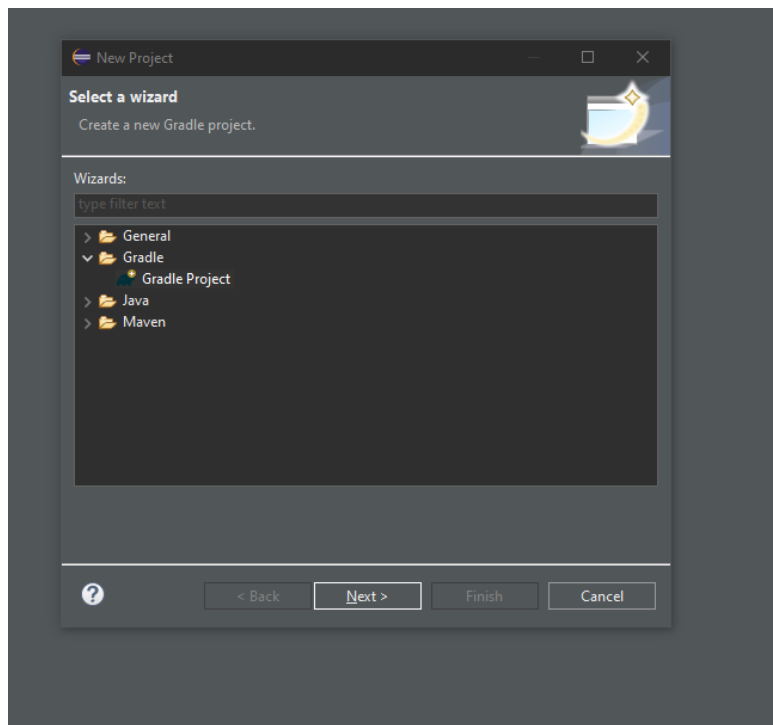


# **Memoria**

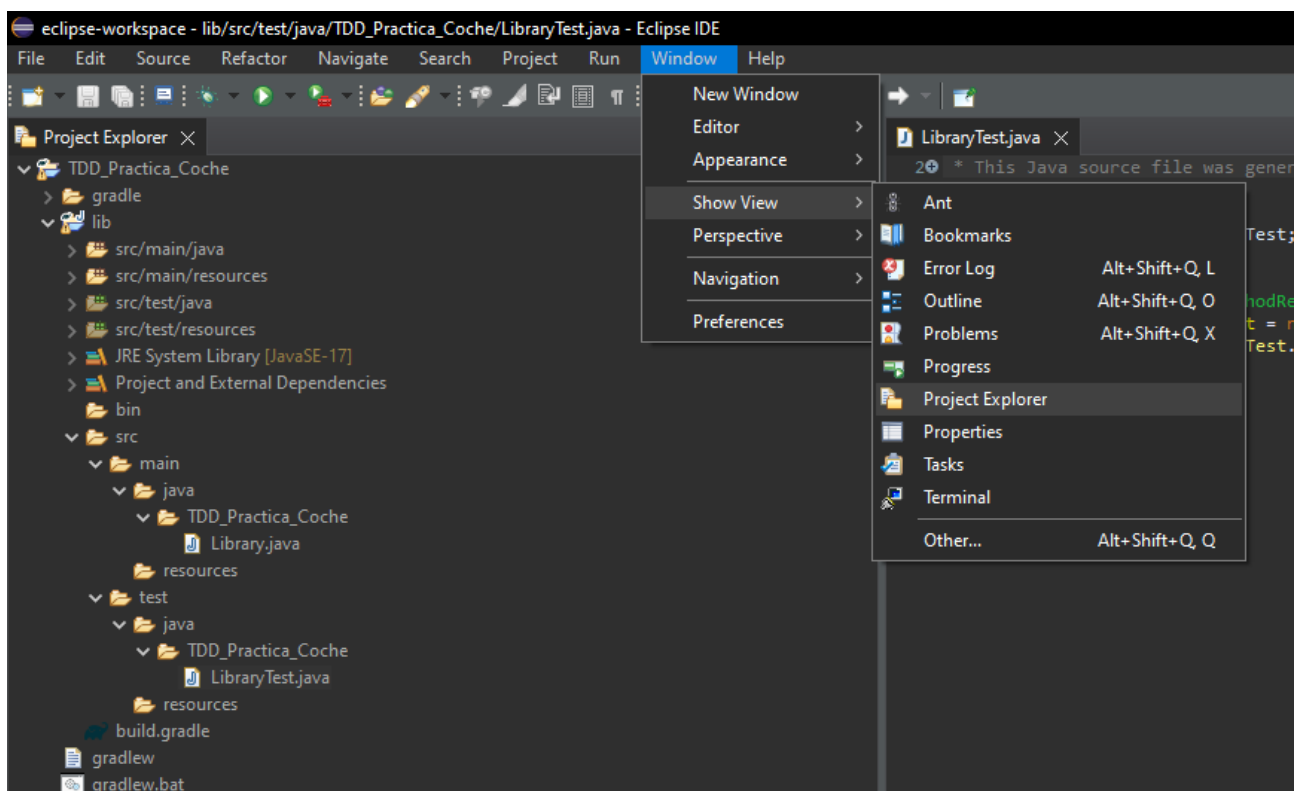
## **Mi primer TDD V2.0**

### **UND 10**

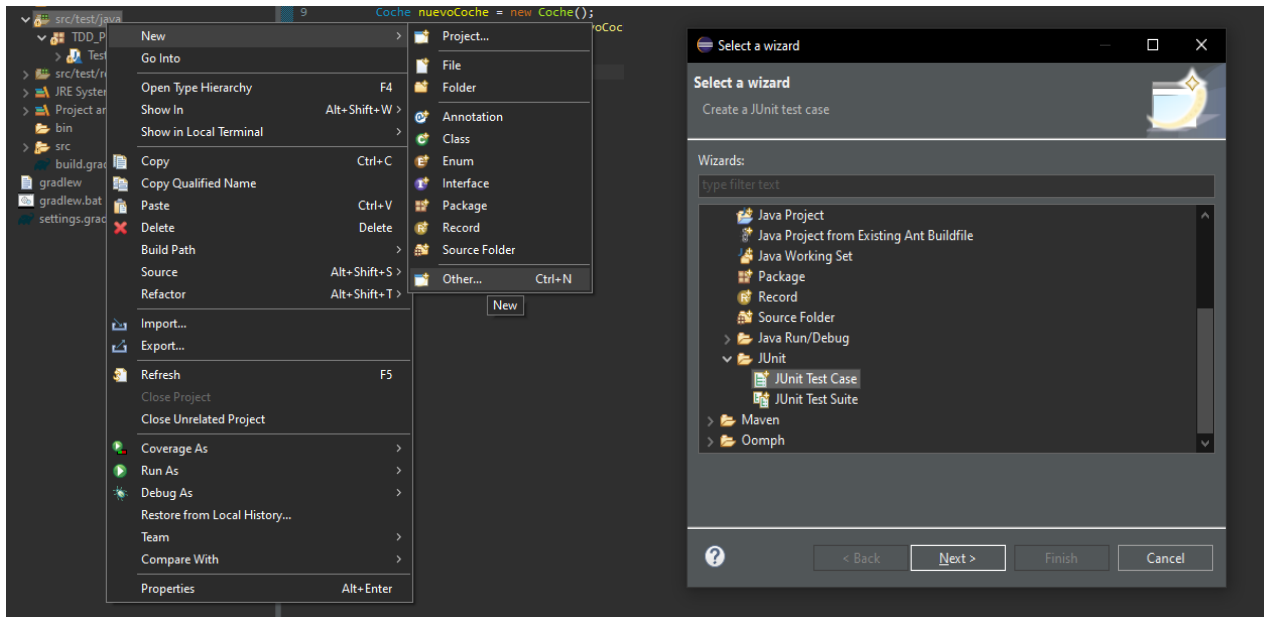
Creación del proyecto en eclipse, seleccionamos 'File', 'New', 'Project', seleccionas lo que quieres utilizar, en este caso Gradle. Le añades el nombre del proyecto y listo.



Una vez creado, en el caso de no ver nada en el IDE desplegaremos la barra de directorios, siguiendo estos pasos. Window → Show View → Project Explorer.



Ahora creare la clase para hacer los Test,



Y creare el primer Test, creara un objeto Coche con velocidad 0 y comprobara que es correcto.

```

1 package TDD_Practica_Coche;
2
3 import org.junit.jupiter.api.Assertions;
4 import org.junit.jupiter.api.Test;
5
6 class TestCoche {
7     @Test
8     public void test_crear_un_coche_su_velocidad_es_cero(){
9         Coche nuevoCoche = new Coche();
10        Assertions.assertEquals(0, nuevoCoche.velocidad);
11    }
12 }
13

```

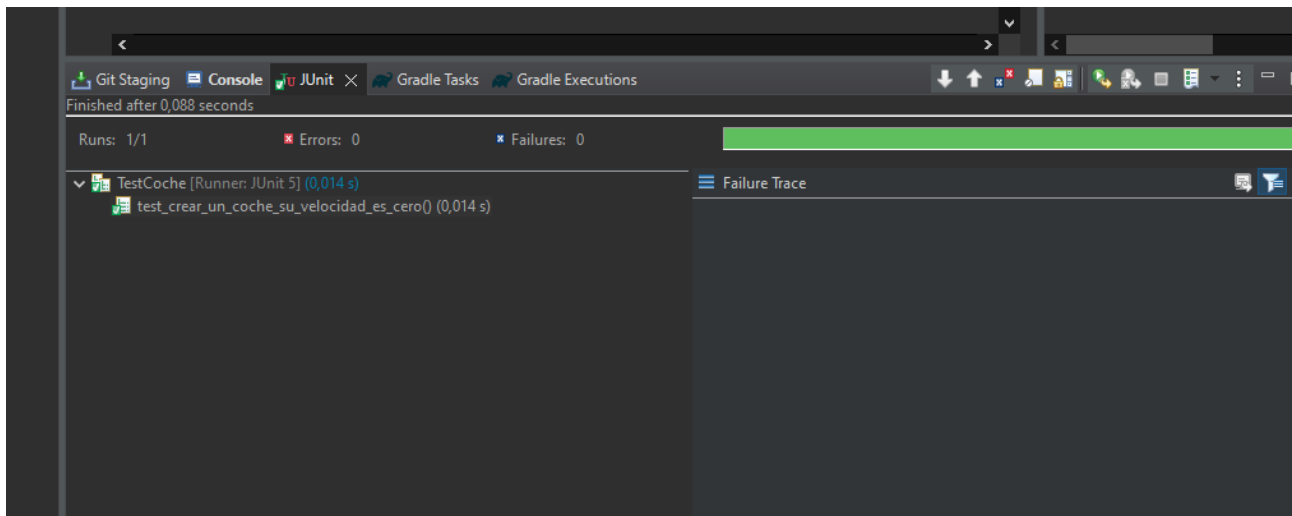
La clase de coche, en eclipse hay que indicarle 0 ya que Null no lo considera nada y sale que hay un fallo.

```

1 package TDD_Practica_Coche;
2
3 public class Coche {
4     public Integer velocidad = 0;
5 }
6

```

Ha pasado el Test



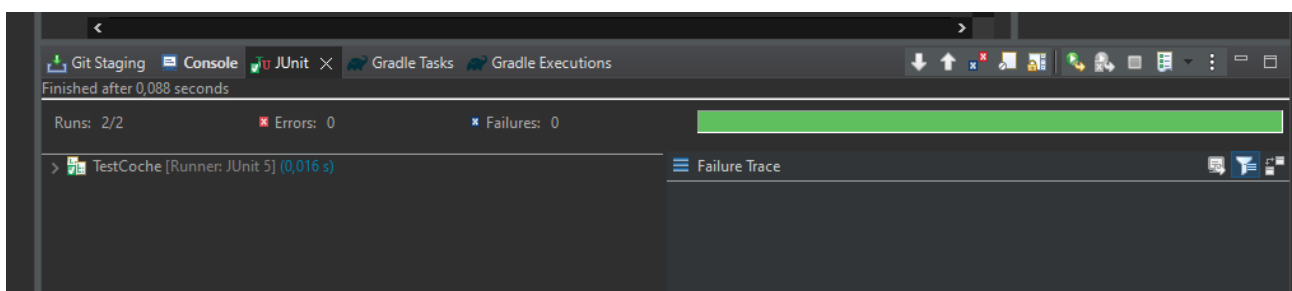
El segundo test, consiste en crear un coche y aumentarle la velocidad en 30, comprobara que la velocidad total sea igual a la acelerada.

```
13 @Test
14 public void test_al_acelerar_un_coche_su_velocidad_aumenta(){
15     Coche nuevoCoche = new Coche();
16     nuevoCoche.acelerar(30);
17     Assertions.assertEquals(30, nuevoCoche.velocidad);
18 }
```

El método utilizado.

```
public void acelerar(int aceleracion) {
    velocidad += aceleracion;
}
```

Ha superado el test.



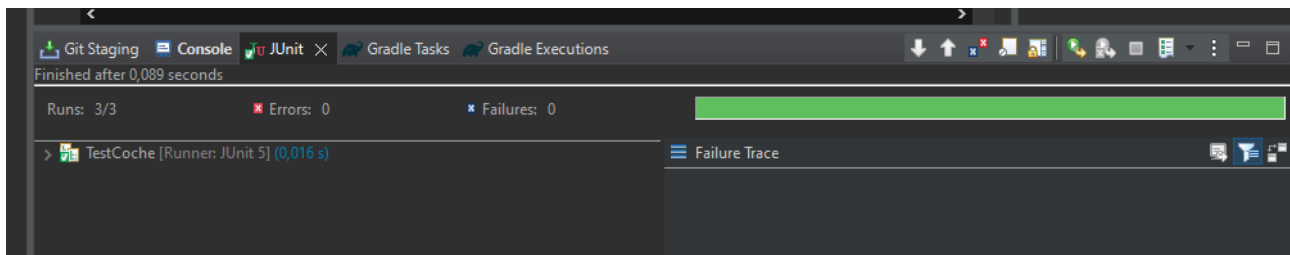
El tercer test restara velocidad al coche, comprobando que si tiene velocidad 50 y le restamos 20 el resultado debe ser 30.

```
@Test
public void test_al_decelerar_un_coche_su_velocidad_disminuye() {
    Coche nuevoCoche = new Coche();
    nuevoCoche.velocidad = 50;
    nuevoCoche.decelerar(20);
    Assertions.assertEquals(30, nuevoCoche.velocidad);
}
```

El método utilizado.

```
public void decelerar(int deceleracion) {
    velocidad -= deceleracion;
}
```

Ha superado el test.



El ultimo test es comprobar que la velocidad disminuida nunca sera inferior a 0.

```
@Test
public void test_al_decelerar_un_coche_su_velocidad_no_puede_ser_menor_que_cero() {
    Coche nuevoCoche = new Coche();
    nuevoCoche.velocidad = 50;
    nuevoCoche.decelerar(80);
    Assertions.assertEquals(0, nuevoCoche.velocidad);
}
```

Para evitar errores se pondrá un If en el metodo que comprobara la velocidad y en el caso de bajar de 0 la colocara a 0.

```
public void decelerar(int deceleracion) {
    velocidad -= deceleracion;
    if (velocidad < 0) velocidad = 0;
}
```

Ha superado la prueba

