

# GARAPEN INGURUMENA

5.3 Mugarria

Taldea 5  
2024/4/29  
Aritz Lekube, Iñigo  
Arrizabalo eta Julen  
Garcia Mata



3 test sortu dira “@Test” jartzen bere gainean. Aurrenekoari **assertNotNull** erabili da, jakiteko zeozer bueltatzen duen, bigarrengean **assertTrue** erabili da, “True” bueltatzen duen jakiteko eta hirugarrengean **assertEquals** erabili da, jakiteko bueltatu duen balorea 3 dela ikusteko. Gero test batzuei Tag batzuk jarri zaie “@Tag(‘izena’)” jartzen @Test-en gainean. Onekin bukatzeko “@DisplayName” jarri zaie testei, izenak aldatu al izateko.

```
@Tag("Sozioak")
@Test
@EnabledOnOs(OS.WINDOWS)
@DisplayName("sozio_Array metodoa ez du itzuliko NULL")
public void test1(){
    Assertions.assertNotNull(sozio_Array());
}

Manex
@Test
@DisplayName("Izena_jarri true itzuliko du.")
public void test2(){
    Assertions.assertTrue(izena_jarri( identifikatzailea: "76390986Y"));
}

Manex
@Test
@DisplayName("id_atera metodoa 3 ida bueltatuko du.")
public void test3(){
    Assertions.assertEquals( expected: 3,id_atera( nan: "85844584D"));
}
```

**BeforeAll** eta **AfterAll** batzuk sortu dira sout batzuekin, jakin al izateko noiz hasten eta bukatzen diren testak.

```
Manex
@BeforeAll
static void beforeAll() { System.out.println("Test hasiera"); }

Manex
@AfterAll
static void afterAll() { System.out.println("Test bukaera"); }
```



Testa windows-en bakarrik egin al izateko “@EnabledOnOs(OS.WINDOWS)” jarri da “@Test”-en azpian.

```
@Tag("Sozioak")
@Test
@EnabledOnOs(OS.WINDOWS)
@DisplayName("sozio_Array metodoa ez du itzuliko NULL")
public void test1(){
    Assertions.assertNotNull(sozio_Array());
}
```

Habiaratutako test bat egin al izateko, aurrena “@Nested” jarri da eta bere azpian klase bat sortu da non bere barruan test bat dagoen.

```
@Nested
class habiaratutakoa{
    new *
    @Tag("4.Panelan")
    @Test
    @DisplayName("CRUD testa")
    void test4(){
        Assertions.assertNotNull(center_4_txertatu());
    }
}
```

Test bat hiru aldiz errepikatu al izateko “@RepeatedTest(value=3)” jarri zaio @Test-en azpian.

```
@Tag("3.Panelan")
@Test
@RepeatedTest(value = 3)
public void test5(){ Assertions.assertNotNull(center_3_txertatu()); }
```

CSV batetik datuak hartu al izateko “@CsvFileSource(resources='fitxategiaren\_kokapena')” erabiliz testa egin izango du.

```
new
@DisplayName("CSV")
@CsvFileSource(resources = "../CSV.csv")
@ParameterizedTest
void test6(int IDa) { Assertions.assertEquals("85844584D", nan_atera(IDa), "NANA aurkitu dut"); }
```



Bakarrik Sozioak, Datuak eta Sozietateak klaseei egin zaie testak. Hauek dira zerbait bueltatu dezaketen metodoak dituzten klase bakarrak, beste denak void motako metodoak dituzte. Sozietateak klasea bi metodo ditu, baina oiletako bat konstruktore bat da.

Element	Class, %	Method... ▾	Line, %
all	18% (8/44)	30% (50/162)	23% (236/994)
Testak	100% (5/5)	100% (24/24)	100% (32/32)
SozioakTest	100% (1/1)	100% (10/10)	100% (15/15)
Testak	100% (2/2)	100% (9/9)	100% (10/10)
DatuakTest	100% (1/1)	100% (4/4)	100% (5/5)
SozietateakTest	100% (1/1)	100% (1/1)	100% (2/2)
erleak	7% (3/39)	18% (26/138)	21% (204/962)
Sozioak	100% (1/1)	95% (19/20)	76% (90/118)
Datuak	100% (1/1)	85% (6/7)	58% (100/170)
Sozietateak	100% (1/1)	50% (1/2)	70% (14/20)
Login	0% (0/13)	0% (0/43)	0% (0/157)
Erregistratu	0% (0/8)	0% (0/28)	0% (0/142)
Index	0% (0/10)	0% (0/24)	0% (0/162)
Sesioa	0% (0/5)	0% (0/14)	0% (0/193)