

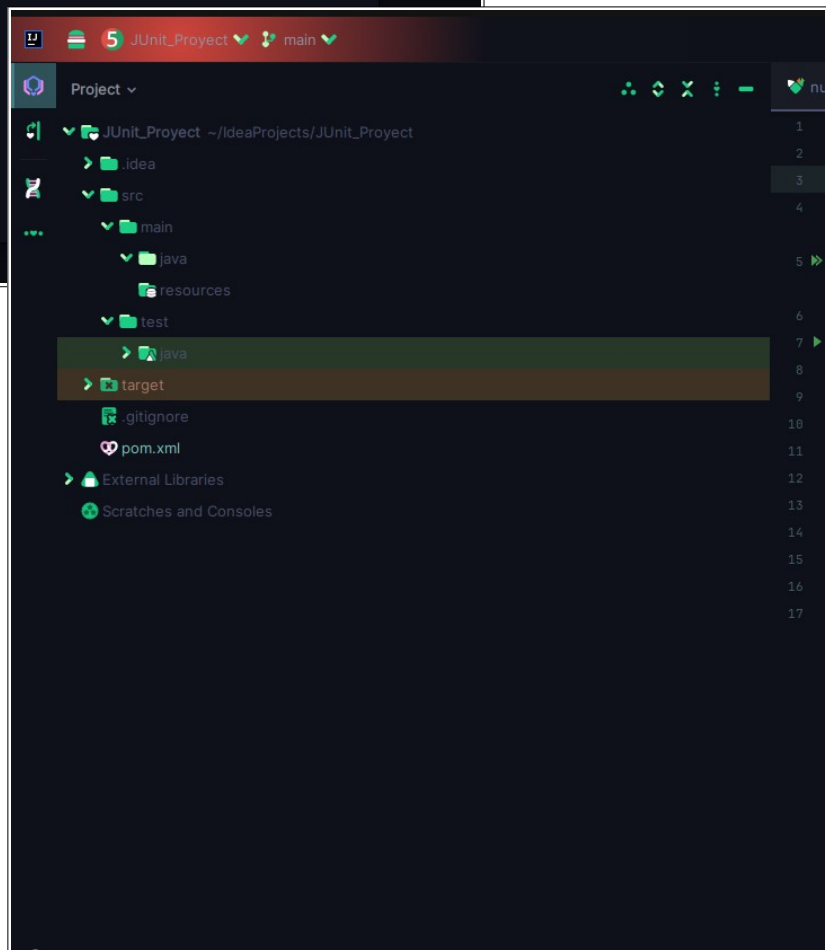
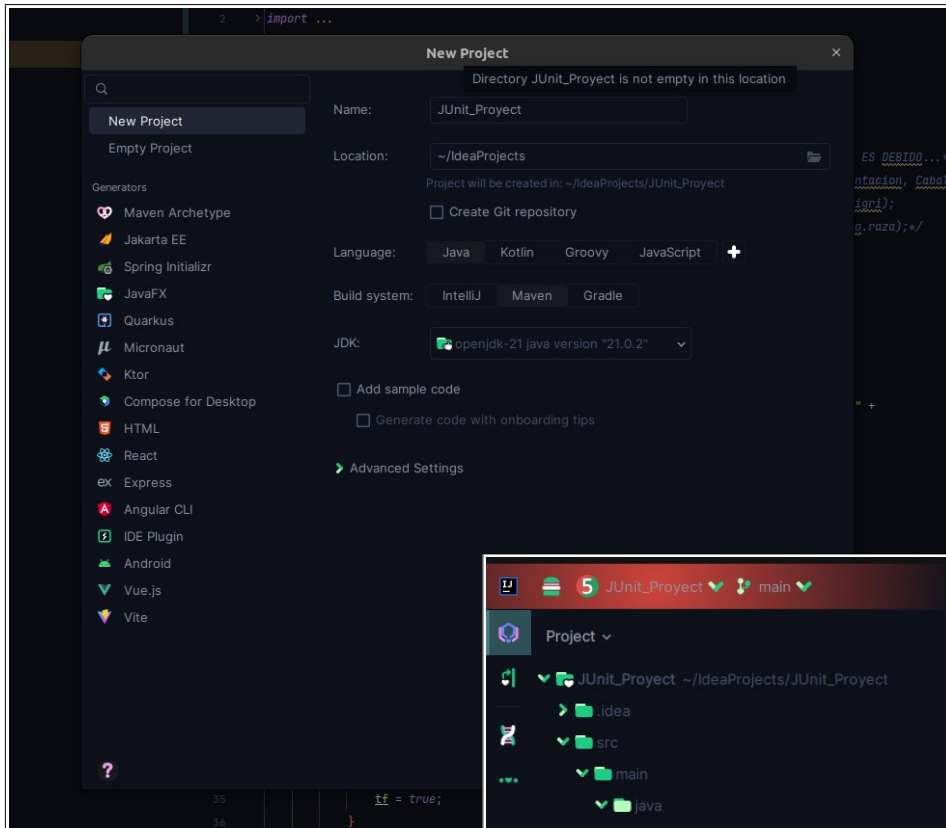
Proyecto JUnit



Índice

Crear proyecto de Maven para JUnit.....	3
Descargar JUnit.....	4
Código para tests.....	5
NumbersTest.java.....	5
booleansTest.java.....	5
arraysTest.java.....	6
stringsTest.java.....	6
Tests.....	7

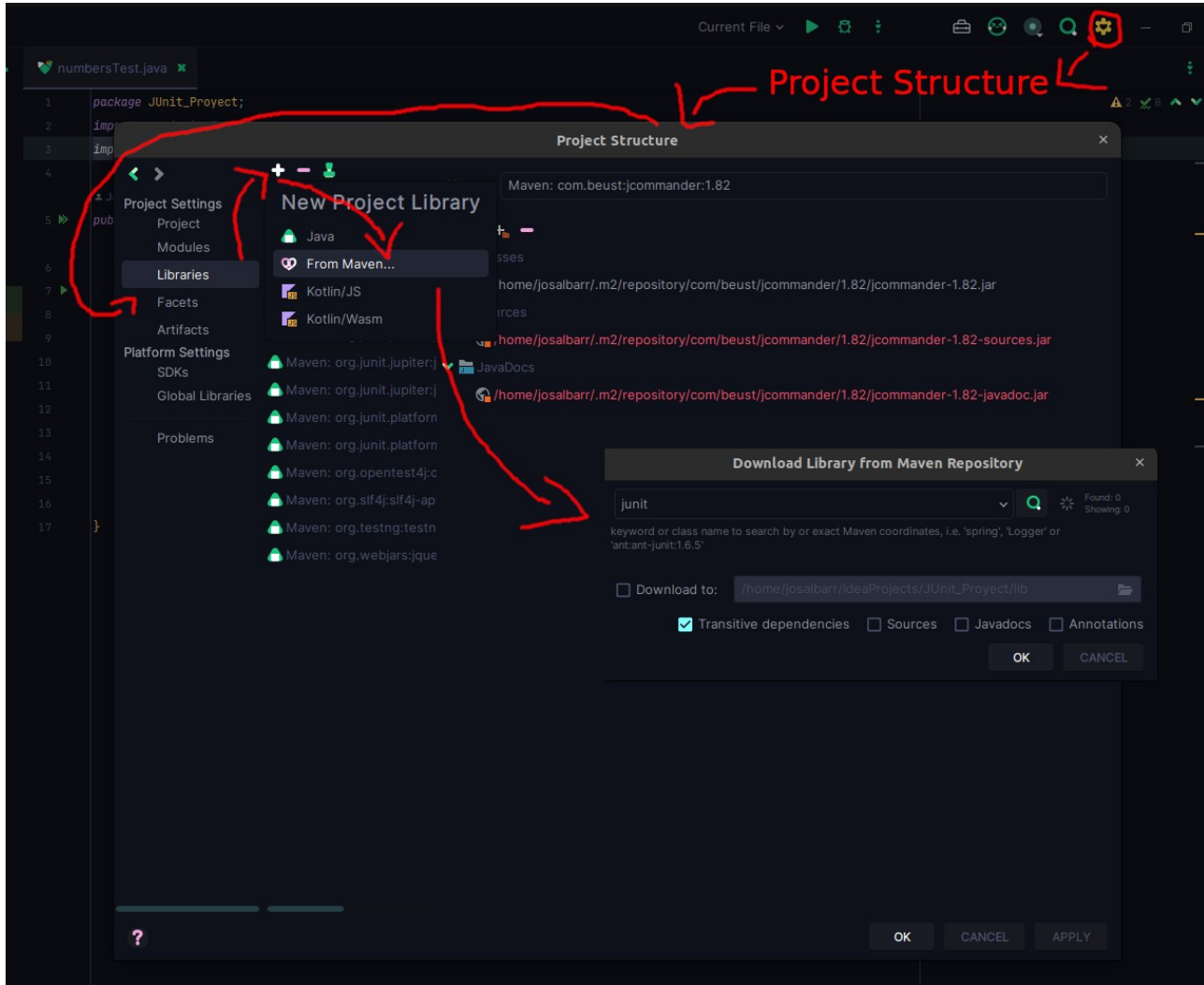
Crear proyecto de Maven para JUnit



Descargar JUnit

En ajustes, vamos a 'Project Structure', y dentro de este, en librerías añadimos junit.

Le daremos a buscar cuando lo tengamos escrito, le damos a buscar y descargaremos alguna dependencia de JUnit.



Código para tests

NumbersTest.java

```
package JUnit_Proyect;
import org.junit.Test;
import static junit.framework.Assert.assertEquals;

public class numbersTest {
    @Test
    public void test(){
        int num1 = 6;
        int num2 = 6;
        int total = num1 + num2;
        int totalFalso = num1 + num2 - 1;

        assertEquals(total, num1 + num2);
        /*assertEquals(totalFalso, num1 + num2);/*Dará siempre error porque totalFalso resta -1, y da un
        resultado que no se espera que sea num1 + num2*/
    }
}
```

booleansTest.java

```
package JUnit_Proyect;
import org.junit.Test;
import static junit.framework.Assert.assertEquals;

public class booleansTest {
    @Test
    public void test() {
        boolean v = true;
        boolean trueFalse;

        int bin = (int) (Math.random() * 2 + 0);

        if(bin == 1){
            System.out.print("verdadero ijo mio");
            trueFalse = true;
        }else{
            System.out.print("falso como tu");
            trueFalse = false;
        }
        assertEquals(v,trueFalse);
    }
}
```

arraysTest.java

```
package JUnit_Proyect;
import org.junit.Test;
import static junit.framework.Assert.assertEquals;
import java.util.ArrayList;
import java.util.Arrays;

public class arraysTest {

    @Test
    public void test(){
        ArrayList<Integer> arrNums = new ArrayList<>(Arrays.asList(1,2,3,4,5));
        ArrayList<String> arrStrings = new ArrayList<>(Arrays.asList("La'zel", "Astarion", "Shadow Heart",
"Halsin", "Gale", "La'zel"));

        assertEquals(arrNums.get(0), arrNums.get(0));
        /*assertEquals(arrStrings.get(0), arrStrings.get(3));/*Aquí dará error porque se espera que el
resultado sea La'zel*/
    }

}
```

stringsTest.java

```
package JUnit_Proyect;
import org.junit.Test;
import static junit.framework.Assert.assertEquals;

public class stringsTest {

    @Test
    public void test(){
        String name1 = "Manolo";
        String name2 = "Alberto";

        assertEquals(name1, name1);
        /*assertEquals(name1, name2); /*Si ejecutamos esta línea dará error, aquí lo que estamos haciendo
es comparar si name1 y name2 son igual, si no lo son, da false*/

    }

}
```

Tests

En este caso lo haré con el numbersTest.java, nos dará bien a no ser que descomentemos el assertEquals de abajo.

La cosa de assertEquals es que, lo primero que pongas, será el resultado esperado, y tras poner una coma, lo demás será lo que se compara para ver si da el mismo resultado, en esta ocasión, num1 + num2 si da total.

```
13 assertEquals(total, actual: num1 + num2);
14 /*assertEquals(totalFalso, num1 + num2);/*Dará si
15
16 }
17 }
```

```
ms ✓ Tests passed: 1 of 1 test - 2 ms
/home/josalbarr/.jdk/openjdk-21.0.2/bin/java ...

Process finished with exit code 0
```

Por el contrario, si descomento el assertEquals de abajo, si dará error, pues ahí estoy sumando + 1 a la ecuación, por ende el resultado entre num1 y num2 ya no será igual a total.

```
6 @Test
7 public void test(){
8     int num1 = 0;
9     int num2 = 0;
10    int total = num1 + num2;
11    int totalFalso = num1 + num2 + 1;
12
13    assertEquals(total, actual: num1 + num2);
14    assertEquals(totalFalso, actual: num1 + num2);/*Dará si
15
16 }
17 }
```

```
Tests failed: 1 of 1 test - 7 ms
/home/josalbarr/.jdk/openjdk-21.0.2/bin/java ...

junit.framework.AssertionFailedError:
Expected :11
Actual   :12
<Click to see difference>

> <5 internal lines>
at junit.swingtest.swingtest.Test(numbersTest (numbersTest.java:14)) <22 internal lines>
```

Con el resto es igual, pero con booleans y strings.