

# جاءوا في للاختبار عمل إطار: JUnit

JUnit هو إطار عمل للاختبار، الاختبارات وتكرار بكتابة للمطورين يسمح. JUnit في للاختبار شرائية عمل إطار هو JUnit للاختبار: استخدام كفي في أعلى أساسية دليل إليك متوقع. هو كما يعمل الكود أن للاختبار:

## 1. مشروعك في JUnit إعداد

بك. الخاص pom.xml ملف إلى JUnit التبعية أضف:

```
<dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <version>4.13.2</version>
  <scope>test</scope>
</dependency>
```

بك. الخاص build.gradle ملف إلى JUnit التبعية أضف:

```
testImplementation 'junit:junit:4.13.2'
```

## 2. اختبارات فئة كتابة

إضافة مع الاختبار، عليها تجري التي فئة بعد الاختبار فئة تسمية الأفضل من للاختبار. جديدة JUnit فئة إنشاء للاختبار. للاختبار.

```
import org.junit.Test;
import static org.junit.Assert.assertEquals;

public class MyClassTest {

    @Test
    public void testAddition() {
        MyClass myClass = new MyClass();
        int result = myClass.add(2, 3);
        assertEquals(5, result);
    }
}
```

## 3. التعليقات

اختبار. طريقة هي الطريقة أن إلى يشير @Test:

- @Before: مفيدي اختبار. طريقة لكل قبل يجري
- @After: مفيدي اختبار. طريقة لكل بعد يجري
- @BeforeClass: في الاختبار طرق من أي قبل واحدة مرة يجري
- @AfterClass: في الاختبار طرق جميع بعد واحدة مرة يجري

#### 4. التأكيدات

متوقع. هو كما يعمل الكود أن لتأكيدات التأكيدات تستخدم

- assertEquals(expected, actual): قيمي متين أن يحدد
- assertTrue(condition): صحيحًا. شرطًا أن يحدد
- assertFalse(condition): خاطئًا. شرطًا أن يحدد
- assertNull(object): فارغًا. كائنًا أن يحدد
- assertNotNull(object): فارغ. غير كائنًا أن يحدد

#### 5. اختباراتك تشغيل

أو `mvn test` باستخدام الأمر أو `gradle test` باستخدام أدوات الاختبار الخاصة `JUnit` أو `TestNG` يمكنك تشغيل الاختبارات تشغيل كما يلي

- `mvn test`
- `gradle test`

#### مثال

لها: اختبار وكفاءة لفئة بسيط مثال هنا

```
// MyClass.java
public class MyClass {
    public int add(int a, int b) {
        return a + b;
    }
}

// MyClassTest.java
import org.junit.Test;
import static org.junit.Assert.assertEquals;

public class MyClassTest {

    @Test
```

```

public void testAddition() {
    MyClass myClass = new MyClass();
    int result = myClass.add(2, 3);
    assertEquals(5, result);
}
}

```

## إضافة نصائح

- الاختبارات. بين الاعتماد تجنب الآخريين. عن مستقل اختبار لكل يكون أن يجب: الاختبارات استقلال
- تغطية نسبة زيادة من بدل ألمعنى ذات الاختبارات على التركيز ولكن عالية، اختبارات تغطية استهدف: الاختبارات غطاء فقط. الاختبارات
- اعتماداتها. من الاختبارعليها تجرى التي الةلإزالة □□□□□□ مثل التقلدي عمل إطار استخدم: التقلدي

□□□□. في تطبيقاتك لأتمتة اختبارات وتشغيل لكتابفة فعال بشكل □□□□ استخدم يمكنك الخطوات، هذه بمتابعة