



Bevezetés a JUnit-ba

JUnit előzményei

Java

A Java-ban alapból nincs egységtesztelés támogatva, ezért szükség van keretrendszer alkalmazásához ha egység tesztelni szeretnénk.

Hasonló megvalósítások

TestNG (Eclipse, IntelliJ és NetBeans), Mockito

Verzióváltások

A junit idővel többször is megújult és új verziók jelentek meg. JUnit 1-5



JUnit 5 annotációk és metódusok

@TestFactory
@DisplayName
@Nested
@Tag
@ExtendWith
@BeforeEach
@AfterEach
@BeforeAll
@AfterAll
@Disabled

assertNotEquals(expected, actual)

assertArrayEquals(expected, actual)

assertTrue(condition)

assertFalse(condition)

assertNull(object)

assertNotNull(object)

assertSame(expected, actual)

assertNotSame(expected, actual)

Példák JUnit Tesztekre

```
@Test void lambdaExpressions() {  
    List numbers = Arrays.asList(1, 2, 3);  
    assertTrue(numbers.stream().mapToInt(Integer::intValue).sum() > 5,  
        () -> "Sum should be greater than 5");  
}
```

```
@Test void trueAssumption() {  
    assumeTrue(5 > j);  
    assertEquals(5 + 2, 7);  
}
```

```
@Test void falseAssumption()  
    assumeFalse(5 < j);  
    assertEquals(5 + 2, 7);  
}
```

További példák JUnit Tesztekre

```
@Test void shouldThrowException() {  
    Throwable exception =  
assertThrows(UnsupportedOperationException.class,  
    () -> {  
        throw new UnsupportedOperationException("Not supported");  
    });  
    assertEquals("Not supported", exception.getMessage());  
}
```

```
@Test void assertThrowsException() {  
    String str = null;  
    assertThrows(IllegalArgumentException.class,  
    () -> {  
        Integer.valueOf(str);  
    });  
}
```

Még további példák JUnit Tesztekre

```
@ParameterizedTest
@CsvSource({"awdwd@s@mail.ru,      1234",
           "!smith@gmail.com,      987456",
           "login.mail.com         12345 ",
           ".,/,star@yandex.ru,     123456"})
public void clickLoginTest(String login, String pass) {
    String acc = "Warning: No match for E-Mail Address and/or Password.";
    mainPage
        .goTo()
        .clickLogin();
    loginPage.logIntoAccount(login, pass);
    String textWrongLogIn = loginPage.getTextWrongLogIn();
    Assertions.assertEquals(acc, textWrongLogIn);
}
```

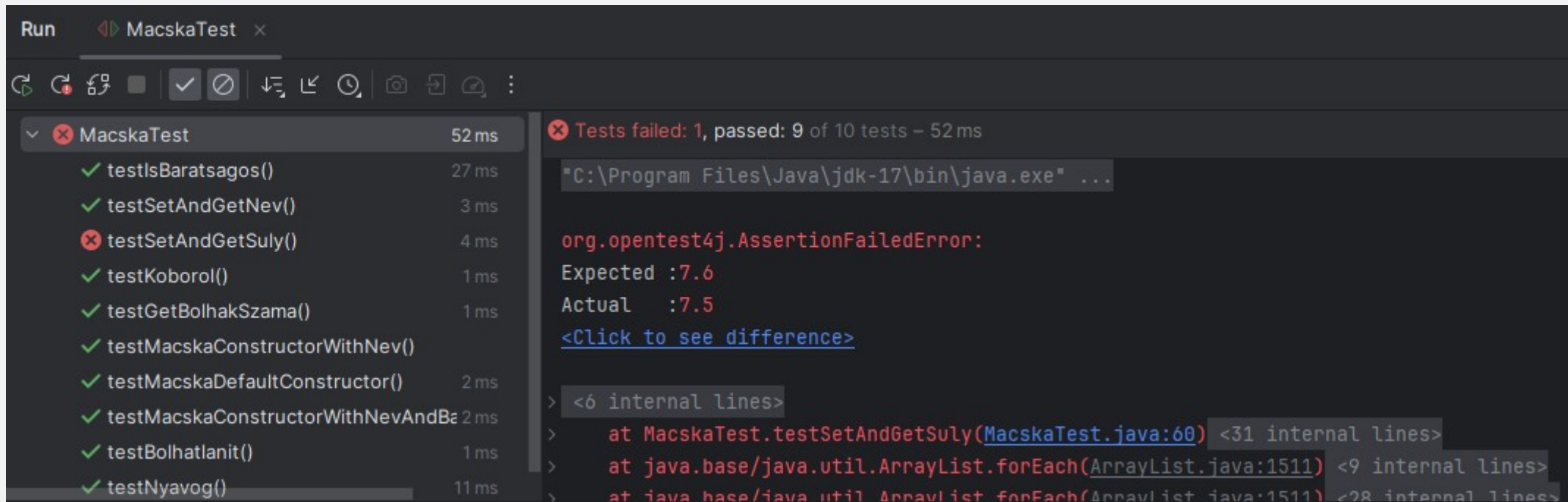
```
import org.junit.jupiter.api.*;

class JUnit5Test {
    @BeforeEach
    void init(TestInfo testInfo) {
        String callingTest = testInfo.getTestMethod().get().getName();
        System.out.println(callingTest);
    }

    @Test
    void firstTest() {
        System.out.println(1);
    }

    @Test
    void secondTest() {
        System.out.println(2);
    }
}
```

Tesztek futtatása, elemzése



The screenshot displays the 'Run' window of an IDE, specifically showing the execution results for a test class named 'MacskaTest'. The window is divided into two main sections. The left section provides a summary of the test results, while the right section shows the detailed output of the failed test.

Test Results Summary:

Test Method	Duration	Status
testIsBaratsagos()	27 ms	Passed
testSetAndGetNev()	3 ms	Passed
testSetAndGetSuly()	4 ms	Failed
testKoborol()	1 ms	Passed
testGetBolhakSzama()	1 ms	Passed
testMacskaConstructorWithNev()	2 ms	Passed
testMacskaDefaultConstructor()	2 ms	Passed
testMacskaConstructorWithNevAndBa	2 ms	Passed
testBolhatlanit()	1 ms	Passed
testNyavog()	11 ms	Passed

Failed Test Details:

Tests failed: 1, passed: 9 of 10 tests – 52 ms

The failed test, `testSetAndGetSuly()`, resulted in an `org.opentest4j.AssertionFailedError`. The error message indicates that the expected value was `7.6` and the actual value was `7.5`. A link is provided to view the difference: [Click to see difference](#).

The stack trace shows the following sequence of calls:

- `<6 internal lines>`
- `at MacskaTest.testSetAndGetSuly(MacskaTest.java:60) <31 internal lines>`
- `at java.base/java.util.ArrayList.forEach(ArrayList.java:1511) <9 internal lines>`
- `at java.base/java.util.ArrayList.forEach(ArrayList.java:1511) <28 internal lines>`