**Writing Basic JUnit Tests**

**1. Create the Java Class to Test**

package com.example;

public class StringUtils {

// Returns true if string is palindrome

public static boolean isPalindrome(String str) {

if (str == null) return false;

String cleaned = str.replaceAll("[^a-zA-Z0-9]", "").toLowerCase();

return cleaned.equals(new StringBuilder(cleaned).reverse().toString());

}

// Counts vowels in a string

public static int countVowels(String str) {

if (str == null) return 0;

return str.replaceAll("[^aeiouAEIOU]", "").length();

}

// Trims and capitalizes first letter

public static String capitalize(String str) {

if (str == null || str.trim().isEmpty()) return str;

String trimmed = str.trim();

return trimmed.substring(0, 1).toUpperCase() + trimmed.substring(1).toLowerCase();

}

}

**2. Create JUnit Test Class**

java

Copy

Download

// src/test/java/com/example/StringUtilsTest.java

package com.example;

import org.junit.Test;

import static org.junit.Assert.\*;

public class StringUtilsTest {

// Test isPalindrome() method

@Test

public void testIsPalindrome() {

assertTrue(StringUtils.isPalindrome("madam"));

assertTrue(StringUtils.isPalindrome("A man, a plan, a canal: Panama"));

assertFalse(StringUtils.isPalindrome("hello"));

assertFalse(StringUtils.isPalindrome(null));

}

// Test countVowels() method

@Test

public void testCountVowels() {

assertEquals(3, StringUtils.countVowels("Hello World"));

assertEquals(5, StringUtils.countVowels("AEIOU"));

assertEquals(0, StringUtils.countVowels("xyz"));

assertEquals(0, StringUtils.countVowels(null));

}

// Test capitalize() method

@Test

public void testCapitalize() {

assertEquals("Hello", StringUtils.capitalize("hello"));

assertEquals("Hello", StringUtils.capitalize("HELLO"));

assertEquals("Hello world", StringUtils.capitalize(" hello WORLD "));

assertNull(StringUtils.capitalize(null));

assertEquals("", StringUtils.capitalize(""));

}

// Edge case testing

@Test

public void testEdgeCases() {

// Empty strings

assertFalse(StringUtils.isPalindrome(""));

assertEquals(0, StringUtils.countVowels(""));

assertEquals("", StringUtils.capitalize(""));

// Single character

assertTrue(StringUtils.isPalindrome("a"));

assertEquals(1, StringUtils.countVowels("a"));

assertEquals("A", StringUtils.capitalize("a"));

}

}

**Common Test Patterns**

// Parameterized testing (JUnit 4)

@RunWith(Parameterized.class)

public class ParameterizedTest {

@Parameters

public static Collection<Object[]> data() {

return Arrays.asList(new Object[][] {

{"test", "Test"}, {"junit", "Junit"}

});

}

// Test cases will run for each parameter set

}

// Exception testing

@Test(expected = IllegalArgumentException.class)

public void testInvalidInput() {

StringUtils.methodThatThrowsException(null);

}

// Timeout testing

@Test(timeout = 1000)

public void testPerformance() {

// Test should complete within 1 second

}