

Parameterized Tests

Parameterized Test with @ValueSource

```
package com.Simplilearn.JUnit;

import static org.junit.Assert.assertTrue;

import org.junit.jupiter.params.ParameterizedTest;
import org.junit.jupiter.params.provider.ValueSource;

public class Demo8 {
    @ParameterizedTest
    @ValueSource(ints = { 1, 2, 3 })
    public void test1(int i) {
        System.out.println(i);
    }

    @ParameterizedTest
    @ValueSource(strings = { "4", "5", "6" })
    public void test2(String s) {
        assertTrue(Integer.parseInt(s) < 6);
    }
}
```

@ParameterizedTest with @CsvSource

```
package com.Simplilearn.JUnit;

import static org.junit.Assert.assertEquals;

import org.junit.jupiter.params.ParameterizedTest;
import org.junit.jupiter.params.provider.CsvSource;

public class Demo9 {

    @ParameterizedTest
    @CsvSource({ "test", "TEST", "monDAY", "MONday", "July", "july" })
    public void test1(String actual, String expected) {
        String actualValue = actual.toLowerCase();
        String expectedValue = expected.toLowerCase();
        assertEquals(actualValue, expectedValue);
    }
}
```

@ParameterizedTest with @EnumSource

```
package com.Simplilearn.JUnit;

import static org.junit.Assert.assertNotNull;

import java.util.concurrent.TimeUnit;

import org.junit.jupiter.params.ParameterizedTest;
import org.junit.jupiter.params.converter.ArgumentConverter;
import org.junit.jupiter.params.converter.ConvertWith;
import org.junit.jupiter.params.provider.EnumSource;

public class Demo11 {
    @ParameterizedTest
    @EnumSource(TimeUnit.class)
    public void test2(@ConvertWith(ArgumentConverter.class) String arg)
    {
        assertNotNull(TimeUnit.valueOf(arg));
    }
}
```