**1. Why do we disable test cases?**

During the development, we may want to disable or skip some test cases from execution because of there is know issue in the functionality or feature. To **ignore a test** in **JUnit** you can either comment a method we can remove @Test annotation above the class or method or we can comment @Test annotation, in that case disabled test case record wont be there in generated report or runner results. Its always good to include record for every test case in report whether its failed or skipped or passed and the reason.

In**Junit 5** Jupiter we can disable tests using @Disabled annotation. We use @Ignore annotation to disable or ignore the tests in **JUnit 4**.

**2. Junit 5 disable tests using @Disabled**

Junit 5 @Disabled accepts only **one optional parameter**, which indicates the reason this test is disabled. Its good practice that developers provide a short explanation for why a test class or test method has been disabled.

For example, @Disabled("Disabled until issue #319 has been resolved")

2.1. Disabling entire test class :

@Disabled("Disable until issue #319 got fixed")

**public** **class** Junit5DisableClassLevelTest {

@Test

**void** test\_Add() {

assertEquals(5, MathUtil.add(3, 2));

}

@Test

**void** test\_Multiply() {

assertEquals(15, MathUtil.multiple(3, 5));

}

@Test

**void** test\_Devide() {

assertEquals(5, MathUtil.devide(25, 5));

}

@Test

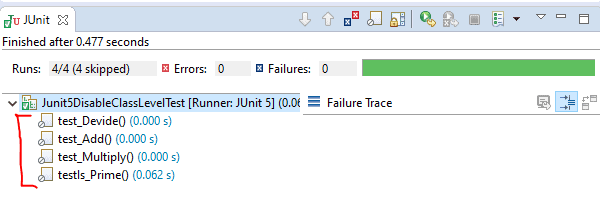
**void** testIs\_Prime() {

assertTrue(MathUtil.isPrime(13));

}

}

**Runner Results**

[](https://javabydeveloper.com/wp-content/uploads/2020/01/Junit5-disable-tests-results-class-level.png)

2.2. Disabling test method :

Use @Disabled at method level annotation to disable particular test method.

**public** **class** Junit5DisableMethodLevelTest {

@Test

**void** test\_Add() {

assertEquals(5, MathUtil.add(3, 2));

}

@Test

**void** test\_Multiply() {

assertEquals(15, MathUtil.multiple(3, 5));

}

@Test

**void** test\_Devide() {

assertEquals(5, MathUtil.devide(25, 5));

}

@Disabled("Disable until issue #112 got fixed")

@Test

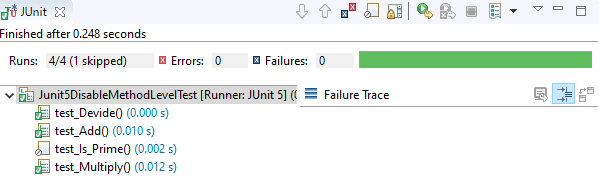
**void** testIs\_Prime() {

assertTrue(MathUtil.isPrime(13));

}

}

**Runner Results**

[](https://javabydeveloper.com/wp-content/uploads/2020/01/Junit-5-disable-perticular-test-method.png)