Annotations Attributes:

In TestNG, attributes are used to provide additional information for test methods,

There are some common TestNG attributes used with the @Test annotation:

**1. priority:**

-------------------------------------------------------------

.Defines the order in which test methods should run. Lower priority values execute first.

code example:

package tut5\_Annotations\_Attributes;

import org.testng.annotations.Test;

public class PriorityDemo

{

@Test(priority = 1)

public void testCase1()

{

System.out.println("Mobile Login TestCase");

}

@Test(priority = 2)

public void testCase2() throws InterruptedException

{

Thread.sleep(400);

System.out.println("Web Login TestCase");

}

@Test(priority = 3)

// @Test() if there is no priority means it has 0 priority and execute first

public void testCase3()

{

System.out.println("API Login TestCase");

}

//

}

**2. enabled**

-------------------------------------------------------------

Controls whether a test method should be executed. By default, it is true.

If set to false, the test will be skipped.

code example:

package tut5\_Annotations\_Attributes;

import org.testng.annotations.Test;

public class EnableTestAttribute

{

@Test(enabled = false)

public void skippedTest() {

System.out.println("This will be skipped");

}

@Test

public void executeMethod()

{

System.out.println("This is executable method");

}

}

**3. dependsOnMethods**

-------------------------------------------------------------

Specifies that a test method should be executed only after certain other methods have successfully run.

code example:

package tut5\_Annotations\_Attributes;

import org.testng.annotations.Test;

public class EnableTestAttribute

{

@Test(enabled = false)

public void skippedTest() {

System.out.println("This will be skipped");

}

@Test

public void exectueMethod()

{

System.out.println("This is executable method");

}

}

**4. groups**

-------------------------------------------------------------

it allows us to execute a specific group of tests.

*Note: we have to use xml file*

code example:

package tut5\_Annotations\_Attributes;

import org.testng.annotations.Test;

public class GroupsAttribute

{

@Test(groups = "software company")

public void infoSys()

{

System.out.println("Infosys");

}

@Test(groups = "software company")

public void wipro()

{

System.out.println("Wipro");

}

@Test(groups = "automobile")

public void maruti()

{

System.out.println("Maruti");

}

@Test(groups = "automobile")

public void tata()

{

System.out.println("Tata");

}

}

*XML file:*

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">

<suite name="Suite">

<test thread-count="5" name="Test">

*<groups>*

*<run>*

*<include name="automobile"></include>*

*</run>*

*</groups>*

<classes>

<class name="tut5\_Annotations\_Attributes.GroupsAttribute" />

</classes>

</test> <!-- Test -->

</suite> <!-- Suite -->

**5. timeout**

-------------------------------------------------------------

Specifies the maximum amount of time (in milliseconds) that a test can run before it is marked as failed.

code example:

package tut5\_Annotations\_Attributes;

import org.testng.annotations.Test;

public class TimeOut\_Attribute

{

@Test(description = "This is test case 1")

public void testCase1()

{

System.out.println("Mobile Login TestCase");

}

@Test(timeOut = 200)

public void testCase2() throws InterruptedException

{

Thread.sleep(400);

System.out.println("Web Login TestCase");

}

@Test(description = "This is case 3")

public void testCase3()

{

System.out.println("API Login TestCase");

}

}

**6. description**

-------------------------------------------------------------

The description attribute is used to provide a brief explanation or description of a test method.

code example:

package tut5\_Annotations\_Attributes;

import org.testng.annotations.Test;

public class Description\_Attributes

{

@Test(description = "This is test case 1")

public void testCase1()

{

System.out.println("Mobile Login TestCase");

}

@Test(description = "This is case 2")

public void testCase2()

{

System.out.println("Web Login TestCase");

}

@Test(description = "This is case 3")

public void testCase3()

{

System.out.println("API Login TestCase");

}

}

**7. invocationCount**

-------------------------------------------------------------

Specifies how many times a test method should be invoked.

code example:

package tut5\_Annotations\_Attributes;

import org.testng.annotations.Test;

public class InvocationCount

{

@Test(invocationCount = 5)

public void repeatTest()

{

System.out.println("invocation count ");

// This test will run 5 times

}

}