Listeners

As the name suggests Listeners "listen" to the event defined in the selenium script and behave accordingly. It is used in selenium by implementing Listeners Interface. It allows customizing TestNG reports or logs. There are many types of TestNG listeners available.

Types of Listeners in TestNG

There are many types of listeners which allows you to change the TestNG's behavior.

Below are the few TestNG listeners:

1. IAnnotationTransformer ,
2. IAnnotationTransformer2 ,
3. IConfigurable ,
4. IConfigurationListener ,
5. IExecutionListener,
6. IHookable ,
7. IInvokedMethodListener ,
8. IInvokedMethodListener2 ,
9. IMethodInterceptor ,
10. IReporter,
11. ISuiteListener,
12. ITestListener.

Above Interface are called TestNG Listeners. These interfaces are used in selenium to generate logs or customize the[Testing](https://www.guru99.com/software-testing.html)reports.

ITestListener has following methods

* **onStart-** OnStart method is called when any Test starts.
* **onTestSuccess-** onTestSuccess method is called on the success of any Test.
* **onTestFailure-** onTestFailure method is called on the failure of any Test.
* **onTestSkipped-**onTestSkippedmethod is called on skipped of any Test.
* **onTestFailedButWithinSuccessPercentage-**method is called each time Test fails but is within success percentage.
* **onFinish-**onFinish method is called after all Tests are executed.

## Steps to create a TestNG Listener

Create a Class with and implements 'ITestListener '. Click on “Add unimplemented methods” option from the quick fixes. All the unimplemented methods of this interface will added to the code.

Write the customized code in this methods as per the requirement.

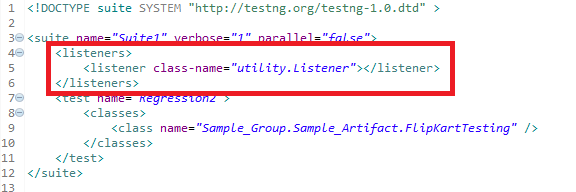
1. onTestFailure: write code to take the screen shot and place it in the project screen shot folder with the name of the file as method name.
2. onTestSkipped, onTestStart, onTestSuccess, we just print the name of the Test. Logs are created in the console. It is easy for the user to understand which test is a pass, fail, and skip status.

 Next, implement this listener in our regular project class i.e. "TestCases". There are two different ways to connect to the class and interface.

The first way is to use Listeners annotation (@Listeners) as shown below:

@Listeners(Listener\_Demo.ListenerTest.class)

The second way is we can create a testng.xml and add listeners tag in XML.



This listener is implemented throughout the test suite irrespective of the number of classes you have. When you run this XML file, listeners will work on all classes mentioned. You can also declare any number of listener class.