# TestNG Listeners

There are mainly two kinds of Listeners. First is WebDriver Listeners and the second is TestNG Listeners. In this tutorial, we will learn TestNG listeners and their different types of listeners.

## What is TestNG Listeners?

Listener is an interface that is used to modify the default TestNG’s behavior. It helps to generate logs or customize TestNG reports. It can be used in Selenium by implementing the Listeners interface.  
  
There are several types of Listener interfaces that allow us to customize TestNG’s behavior. These listener interfaces are called **TestNG listeners**. The list of TestNG listeners is as follows:  
1. ITestListener  
2. ISuiteListener  
3. IReporter  
4. IAnnotationTransformer  
5. IAnnotationTransformer2  
6. IConfigurable  
7. IConfigurationListener  
8. IExecutionListener  
9. IHookable  
10. IInvokedMethodListener  
11. IInvokedMethodListener2  
12. IMethodInterceptor  
  
In this tutorial, we will learn the most popular and commonly used ITestListener and its several methods.

## **ITestListener Interface and Its Methods**

ITestListener is a listener interface that is used for listening on running tests. It is the most used TestNG listener in Selenium WebDriver. It extends**ITestNGListener** interface.  
  
ITestNGListener is a marker interface for all objects that can be passed as a -listener argument.  
  
ITestListener interface provides the following useful methods to customize the default behavior of TestNG reports and logs. They are as follows:

### 1. onTestStart():

This method will be called only when any test method gets started.  
**Method:**

onTestStart(ITestResult result) : void

onTestStart() method takes an ITestResult interface as a parameter that extends IAttributes, java.lang.Comparable<ITestResult>. This ITestResult class describes the result of a test.

### 2. onTestSuccess():

This method will execute once the test method will be passed successfully. That is it will be called on the success of any test method.  
**Method:**

onTestSuccess(ITestResult result) : void

onTestSuccess() method accepts ITestResult interface as a parameter that contains information about the run test. It returns nothing.

### 3. onTestFailure():

An onTestFailure() method is called only when any test method fails.  
**Method:**

onTestFailure(ITestResult result) : void

This method also takes ITestResult interface as a parameter and returns nothing.

4. **onTestSkipped():** An onTestSkipped() will execute only when any test method has been skipped.  
  
5. **onTestFailedButWithinSuccessPercentage():** This method will be called each time when the test method fails but within success percentage.  
  
6. **onStart():** This method is executed at the start of any test method.  
  
7. **onFinish():** An onFinish() method is called when any test case finishes its execution.