

# Selenium Class 36 - TestNG Testing Framework for Selenium

- 1) Introduction to TestNG Testing Framework
- 2) Install TestNG and write first TestNG program
- 3) Create multiple Test cases and run
- 4) TestNG Annotations
- 5) Execute multiple Programs / Classes using XML
- 6) Grouping Test Cases
- 7) Data driven testing using @DataProvider Annotation
- 8) Parallel Test Execution

**In Selenium using Java, there are two Testing Frameworks available,**

- i) JUnit
- ii) TestNG,

**Note:**

Java - JUnit, TestNG

C#.NET - NUnit

PHP - Behat+Mink

Ruby - Repect, Test:Unit,

Python - PyUnit, Py:Test

## 1) Introduction to TestNG Testing Framework

- TestNG Testing Framework was designed to simplify a broad range of testing needs from Unit Testing to System Testing....

- Initially developed for Unit Testing, now used for all levels of Testing
- TestNG is open source software, where NG stands for next Generation
- TestNG inspired by JUnit and NUnit and introduced some new functionalities

### **Advantages of TestNG for Selenium:**

- TestNG Annotations are easy to create Test Cases
  - Test cases can be grouped and prioritized more easily
  - Executes multiple programs/Classes using XML
  - Parallel Testing
  - Generates Test Reports
- Etc...

## **2) Install TestNG and write first TestNG program**

Navigation to install TestNG on Eclipse IDE,

In Eclipse IDE,

- Help Menu,
- Select "install New Software"
- Click "Add"
- Enter name as "TestNG"
- Enter URL as "http://beust.com/eclipse/

- Click OK
- Then "TestNG" will be populated,
- Select "TestNG"
  - Next
  - Next
  - accept the agreement
  - Finish

Write TestNG program/Test Case

**a) Manual Test case**

Test Case name: Verify page Title (Google)

Test Steps:

- i) Launch the Browser
- ii) Navigate to Google Home page (<https://www.google.com/>)

Expected Result:

"Google"

Verification Point:

Capture the Page Title and compare with expected

Test Data/Input:

NA

**b) TestNG Test Case:**

- i) main method is not used in TestNG programs
- ii) TestNG programs contain methods only that contain Annotations
- iii) If we don't write @Test Annotation then the method won't be executed

```
public class VerifyTitle {
    WebDriver driver;

    @Test
    public void verifyTitle(){
        System.setProperty("webdriver.chrome.driver", "E:/chromedriver.exe");
        driver = new ChromeDriver();
        driver.manage().window().maximize();
        driver.get("https://www.google.com/");
        String pageTitle = driver.getTitle();

        /*if (pageTitle.equals("Google2")){
            System.out.println("Google Application was Launched - Passed");
        }
        else {
            System.out.println("Google Application was Not Launched - Failed");
        }*/
        Assert.assertEquals("Google", pageTitle);
    }

    @AfterMethod
    public void closeBrowser(){
        driver.close();
    }
}
```

}

}

.....