TestNG

TestNG plays a major role in Test Automation Frameworks Development.

- **Installing TestNG Plug-in in Eclipse IDE**
  - Checking that TestNG is not available in Eclipse IDE
  - Go to Testng.org > Eclipse > Plug-in Installation > Use the latest URL referred at this location
  - Check whether TestNG is installed in Eclipse IDE
- **Configure the Project with TestNG JAR file**
  - Create a new Maven Project and execute a sample Selenium Script
  - Configure the Project with TestNG (By adding Dependency tags)
- **TestNG Annotations**
  - TestNG is an API similar to Java and Selenium WebDriver
  - TestNG has a huge list of Annotations and below are few annotations we need to learn for Selenium:
    - Check the annotations list in TestNG API
      - @Test
      - @BeforeMethod
      - @AfterMethod
      - @BeforeTest
      - @AfterTest
      - @BeforeSuite
      - @AfterSuite
      - @BeforeClass
      - @AfterClass
      - And many more
  - **@Test**
    - The purpose of this annotation is to represent the methods inside Java class as Tests.
    - This annotation replaces main() method in traditional Java programs - [Demonstrate]
    - Run as 'TestNG Test' and view the execution results in both Eclipse IDE Console and TestNG Results tab
    - TestNG Reports
      - Under test-output folder > index.html
      - Refresh the Project and view the TestNG report generated at 'test-output' > old > index.html
      - Passing a Test
      - **Failing @Test annotated methods**

- If any @Test annotated method fails, it will be displayed as failed test in Eclipse IDE Console, TestNG and default TestNG reports

    - Demonstrate a program in which the @Test annotated method is failing - [Demonstrate here](#)

  - **Skipping a test in TestNG**

    - **throw**

      - If we want to manually throw any exception based on some condition, we have to use **throw** - [Demonstrate here](#)

        - Syntax: **throw new Exception();**
        - Exception is the predefined class of Java

    - If we want to manually skip any test method in TestNG, we have to use throw - [Demonstrate here](#)

      - Syntax: **throw new SkipException();**
      - SkipException is the predefined class of TestNG

- **TestNG Assertions**

  - We perform testing to verify whether a particular test is passed or failed.

  - In the similar way, TestNG provides a predefined Class '**Assert**' and its predefined methods assertEquals(), assertNotEquals(), assertTrue(), assertFalse() and fail() to verify whether @Test annotated methods are passed or failed.

  - **assertEquals()**

    - Demonstrate a program which uses assertEquals() to verify a failing test - Demonstrate [here](#)
    - Demonstrate a program which uses assertEquals() to verify a passing test - Demonstrate [here](#)

  - **assertTrue()**

    - Demonstrate a program which uses assertTrue() to verify a failing test - Demonstrate [here](#)
    - Demonstrate a program which uses assertTrue() to verify a passing test - Demonstrate [here](#)

  - **assertFalse()**

    - Demonstrate a program which uses assertFalse() to verify a failing test - Demonstrate [here](#)
    - Demonstrate a program which uses assertFalse() to verify a passing test - Demonstrate [here](#)

  - **fail()**

    - Demonstrate a program which fails a test directly - Demonstrate [here](#)

- Demonstrate a program which has multiple **@Test** testNG annotated tests

- Tests will be executed in alphabetical order
  - Demonstrate a program which executes the @Test annotated methods according to their priority - Demonstrate [here](#)
    - **priority** attribute of @Test annotation is used to prioritize the tests
      - @Test(priority=1)
    - Check the priority attribute in TestNG API

By,
Arun Motoori

Terms of Service   Privacy Policy   Report Spam