**TestNG (Next Gen)** : A framework inspired by Junit & Nunit. Used to help us make testing easier & efficient.

**To Download:**

Goto eclipse Marketplace

In search box, type – testNG & click install

Confirm license agreement, Select all (always trust authority)

Yes, accept the risk

*Restart Eclipse.*

**Add to dependency:**

Goto Maven Repository

Search box – testNG

Copy the dependency & paste in POM file

**How testNG is efficient:**

1. **@test** -:By adding *@test* on top of the method, creates individual test. You do not need the ‘Public void method’ to print.
2. ***@test (priority =1,2,3) -:*** you can prioritize test cases***,*** it can also take **-1** as a priority... means it will run first.
3. **(enabled = false) -:** you can enable / disable test cases
4. ***@test (alwaysRun = true) -:*** this can be used to set up driver. It will always run first.
5. ***@test (dependsOnMethods = {“MethodName”}) -:*** Used if one test case depends on the other. For Example: if login page does not work or fail, then we will not need to run the logout test case. That way we can save time. It takes String. It takes multiple methods.
6. **Assertion:** - this line is used to fail test case.
   * ***assertFail();*-*assertTrue();* - *assertFalse();-* *assertEqual();***
   * SoftAssert: SAME ABOVE
7. ***@test (groups = {“smoke”,” regression”}) -:*** Used to group test and run the test cases needed for the day. For Example, you have 20 classes and only needed to run a Smoke test. The fastest way is to group the test cases and execute. 🡪 NOTE: to run as groups, select all classes that are grouped, right click... TestNG...convert to TestNG.
8. **@DataProvider (name = login ) -:** helps with providing data from one method to another with the help of parameters. You do not need to create xml file.
9. **Annotation -:** are used to control which method will be executed in order
   * + ***Test*** before/After , ***Class*** before/After, ***Methods*** before/After, then ***@test*** before/After
10. **@Parameters ({“String”}, {“String“}) -:** helps with passing parameters and we can pass values through xml file.
11. **Emailable report:** TestNG have to the access to send email access.
12. **Parallel Execution:** Multiple Application running at the same time
    * It will execute Method 1 line 1 & method 2 line 1. It will go back and forth -- **XML.File**

* **Multi dimension Array is handy when using Data provider!!**

**Multi Array ( very useful to run multiple tests at once… insert multiple username & password within one click )**

**int**[][] data2 ={ {1,2}, {4,5}, {6,0} }; // Multi Array

System.***out***.println(data2[0][0]); // how to print

**String**[][] loginInfo ={ {“ab@gmail.com”,”pas12”}, {“x@gmail.com”,”pas123”}, {{“y@gmail.com”,”pas123”};

Multi dimension array is like regular array;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 20 | 30 | 40 | 50 |
| 1 | 2 | 3 | 4 | 5 |
| 00 | 01 | 02 | 03 | 04 |

Int [][] data = { {0,0} } = 10

Int [][] data = { {0,1} } = 20