**Assert vs Verify**

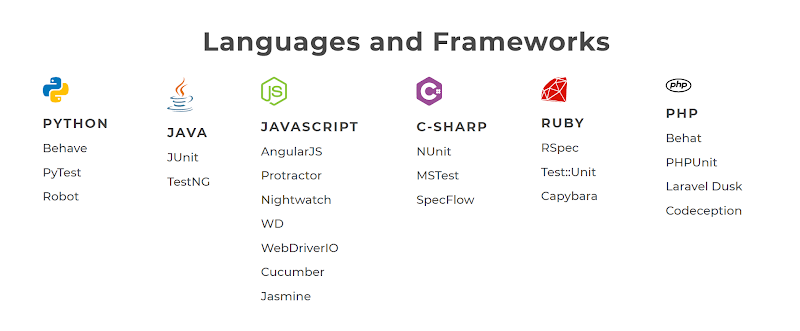
***DAO MINH DAM***[***https://automationfc.vn***](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://automationfc.vn%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339460000%26amp;usg%3DAOvVaw3moh_dwby7IEg0UZxFlCOo&sa=D&source=editors&ust=1628671339494453&usg=AOvVaw3dLkore_hyq5Rkuycyv0pr)

***Email:***[*daominhdam@gmail.com*](mailto:daominhdam@gmail.com)

***Skype:****daominhdam (Đào Minh Đảm)*

***Tài liệu này chỉ sử dụng trong khóa học - vui lòng không chia sẻ công khai***  
-------------------------------------------------------------------------------------------------------------------------------

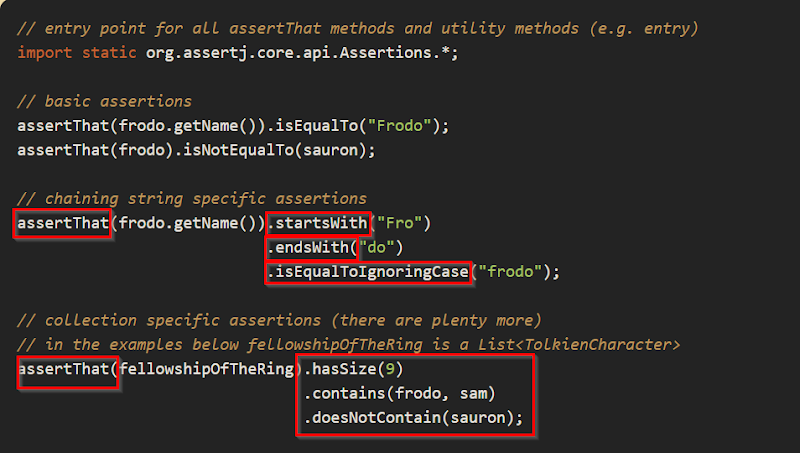
**Language/ Framework/ Tool**



* **Java: JUnit/ TestNG/ AssertJ/ Hamcrest**
* Thị trường chiếm nhiều nhất:
* Open source
* Dễ tích hợp vs nhiều third party
* Build 1 lần -> Chạy trên nhiều nền tảng (MAC/ Linux/ Window/ Mobile)
* Cộng đồng nhiều người sử dụng/ support
* **C#: xUnit/ MSTest/ NUnit/..**
* **PHP: PHPUnit/..**
* **Javascript: Mocha/ Jasmine/..**
* **Python: PyTest**
* **Ruby: RSpec**
* **Typescript: tsUnit (.ts)**
* **Refer**: [https://en.wikipedia.org/wiki/List\_of\_unit\_testing\_frameworks](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://en.wikipedia.org/wiki/List_of_unit_testing_frameworks%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339461000%26amp;usg%3DAOvVaw2W3iGYwkLz61N_gmWeJgnt&sa=D&source=editors&ust=1628671339495229&usg=AOvVaw1hBFCqw5DpAfv3UurabQrH)

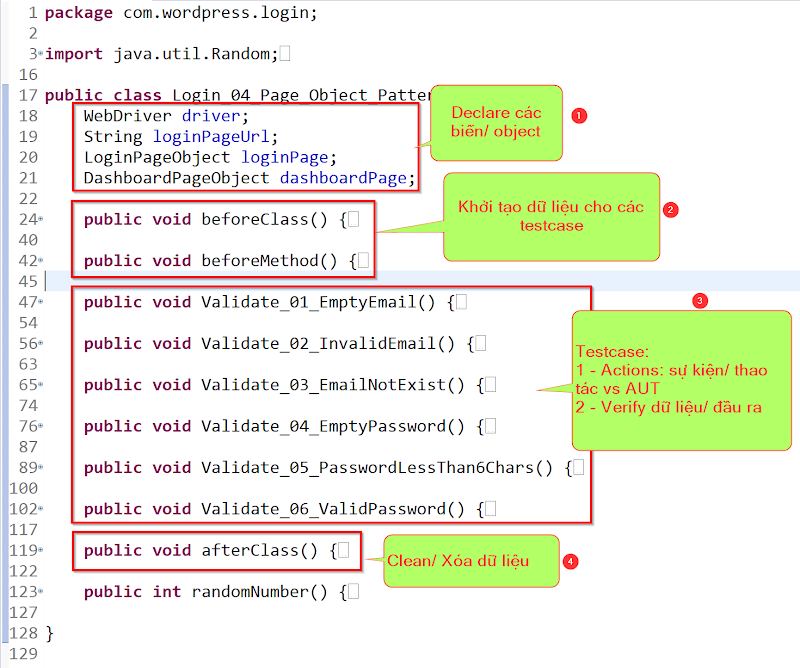
**Java Testing Framework**

* **Assert libraries**
* JUnit (4/ 5)
* [TestNG (Next Generation)](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://docs.google.com/document/d/1Icx24E9tRuK0K_KDp96ebPIiGihYNBA_qyAavus3yq0/edit%2523%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339462000%26amp;usg%3DAOvVaw3TCR89_kCMdiBLTaMruY4V&sa=D&source=editors&ust=1628671339495415&usg=AOvVaw3F9ysmIg4EkXwtCOHokwQ0)
* Support for Unit/ Intergration/ E2E testing: parameter/ multi-browser/ loop/ listener/...
* Hamcrest - [http://hamcrest.org/JavaHamcrest/tutorial](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttp://hamcrest.org/JavaHamcrest/tutorial%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339462000%26amp;usg%3DAOvVaw2dCMJDipAEcziQSJMBqG9c&sa=D&source=editors&ust=1628671339495574&usg=AOvVaw2Nh9vgusTvK0avMkWuirz6)
* AssertJ
* [https://joel-costigliola.github.io/assertj/](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://joel-costigliola.github.io/assertj/%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339462000%26amp;usg%3DAOvVaw0DcRQoEb6UmjjLiku_ClqD&sa=D&source=editors&ust=1628671339495703&usg=AOvVaw0zQMGJKm8Eo6tX0FMTXSz6)
* [https://assertj.github.io/doc/](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://assertj.github.io/doc/%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339462000%26amp;usg%3DAOvVaw3UH6iIdhuQao0VYOFdXouR&sa=D&source=editors&ust=1628671339495795&usg=AOvVaw3ymcbCJjA_mz7AVlgSoVhI)

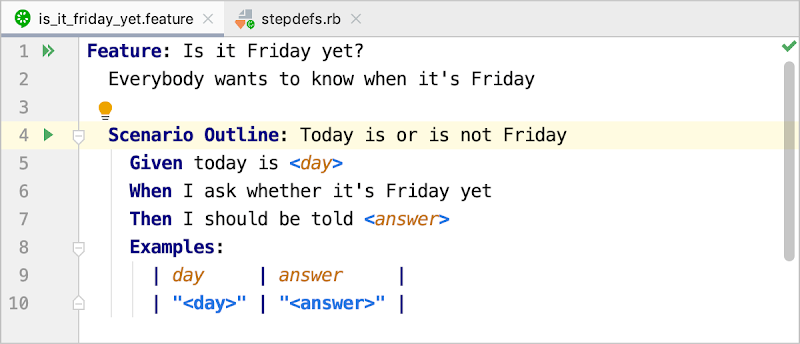


**Arrange Action Assert**

* *Quản lí testcase/ testclass như thế nào cho hiệu quả?*
* Bố trí/ sắp xếp testclass/ testcase hiệu quả và khoa học hơn
* Arrange/ Act/ Assert (AAA) is a pattern for arranging and formatting code - great way to structure test cases



* Arrange (Pre-Condition) – setup the testing objects and prepare the prerequisites for your test
* Arrange inputs and targets
* Arrange steps should set up the test case
* Does the test require any objects or special settings?
* Does it need to prep a database?
* Does it need to log into a web app?
* Handle all of these operations at the start of the testcases
* Action – perform the actual work of the test
* Act steps should cover the main thing to be tested
* This could be calling a function or method, calling a REST API, or interacting with a web page
* Keep actions focused on the target behavior
* Assert – verify the result
* Act steps should elicit some sort of response
* Assert steps verify the goodness or badness of that response
* Assert are as simple as checking numeric or string values
* Assertions will ultimately determine if the test passes or fails
* Behavior Driven Development follows the Arrange/ Act/ Assert pattern by another name
* **Given (Arrange) - When (Action) - Then (Verify)**



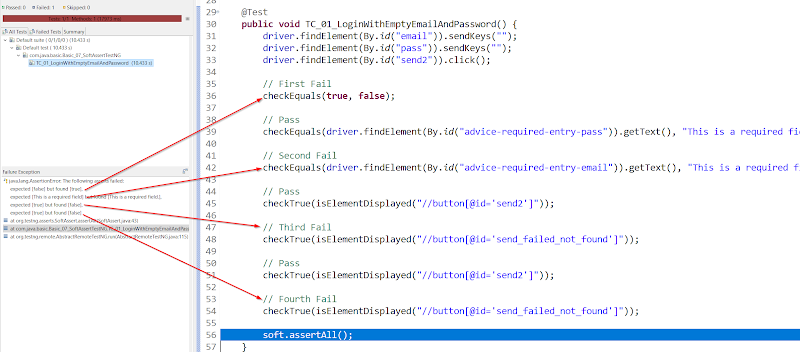
* **Pre-Condition (Arrange)**
* Init Browser/ Driver
* Connect to DB (Database)
* Init Data Test
* Init Page Object
* ...
* **Testcase(s)**
* **Actions**
* Open page link
* Click to button/ checkbox/.. (elements)
* Select dropdown
* ...
* **Verify**
* Check point (Input = Output)
* assertEquals(expected, actual)
* Condition (true/ false)
* assertTrue/ False
* Condition (null/ not null/..)
* Condition (instance/ data type/..)
* **Post-Condition**
* Clean/ Clear Browser/ Driver (executable)
* Disconnect to DB
* **References:**
* [https://automationpanda.com/2020/07/07/arrange-act-assert-a-pattern-for-writing-good-tests/](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://automationpanda.com/2020/07/07/arrange-act-assert-a-pattern-for-writing-good-tests/%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339465000%26amp;usg%3DAOvVaw2A1aT2I6A6LqpuvHOHJNId&sa=D&source=editors&ust=1628671339497418&usg=AOvVaw12y1frF0am04yrJpzE2a4q)
* [https://docs.telerik.com/devtools/justmock/basic-usage/arrange-act-assert](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://docs.telerik.com/devtools/justmock/basic-usage/arrange-act-assert%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339465000%26amp;usg%3DAOvVaw2ay5zVJGy_1QwWa3Kuj6WK&sa=D&source=editors&ust=1628671339497518&usg=AOvVaw0PDdd3QHGrLVc88pVf3eo5)

Hard Assert (T**estNG)**

* **Hàm để Assert:**
* Assert.assertTrue(conditionTrue)
* Assert.assertFalse(conditionFalse)
* Assert.assertEquals(actualType, expectedType)
* int - int
* String - String
* boolean - boolean
* Array - Array
* Collection (ArrayList/ LinkedList/ Set/...)
* import static org.testng.Assert.assertFalse;
* assertFalse(...);
* assertTrue(...);
* assertEquals(...);
* **Best Practice**
* **Hàm nó trả về là true/ false thì mình dùng assertTrue/ False**
* isDisplayed/ isEnabled/ isSelected/ isMultiple/.. -> true/ false
* **Hàm mà trả về text thì nên dùng assertEqual:**
* getCurrentUrl/ getTitle/ getText/ getAttribute/ getSize/...
* [http://www.eliasnogueira.com/dont-use-asserttrue-to-verify-a-text/](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttp://www.eliasnogueira.com/dont-use-asserttrue-to-verify-a-text/%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339466000%26amp;usg%3DAOvVaw3wjtiCTlnbZJUDO-tDFkNE&sa=D&source=editors&ust=1628671339498292&usg=AOvVaw2BTSJea09KU9xJiLVPSLFQ)
* **Assert: Stop các step sau nếu có 1 step bị fail**
* **Ưu điểm:**
* Có kết quả nhanh
* Phù hợp vs các level: Unit/ API testing
* **Nhược điểm:**
* Phải run lại nhiều lần để chạy qua hết all test cases/ steps hoặc phải manual test lại nhiều step/ case
* Không phù hợp vs UI testing
* 1 testcase có 200 steps:
* Tại step 70 - 106 - 180 (assert) -> Chạy 3 lần
* **Reference:**
* [https://gist.github.com/daominhdam/a3016309b5da46f0856a35cb930deb2d](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://gist.github.com/daominhdam/a3016309b5da46f0856a35cb930deb2d%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339467000%26amp;usg%3DAOvVaw3yguGSVrAInV6iOeidbu7w&sa=D&source=editors&ust=1628671339498743&usg=AOvVaw0KVhhWAMnmNH8vnPhBsRQL)

Soft Assert **(TestNG)**

* Thỏa mãn điều kiện nếu 1 step fail vẫn chạy các step còn lại cho đến hết testcase
* Sử dụng **assertAll**khó cho việc investigate/ dò lỗi - chỉ reference tới step assertAll - không reference tới từng step bị lỗi



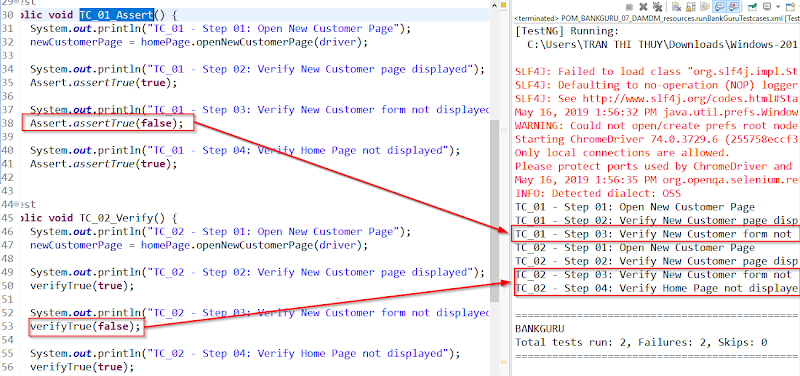
* **Refer:** [https://gist.github.com/daominhdam/5b23dcf3285147387e3cd3a565dcad28](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://gist.github.com/daominhdam/5b23dcf3285147387e3cd3a565dcad28%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339467000%26amp;usg%3DAOvVaw0-0Dsdu_mCvj_sbAu2T2O3&sa=D&source=editors&ust=1628671339499016&usg=AOvVaw2lnrzsnj8sfFApKFM6TxNj)

Verify (C**ustom Hard Assert)**

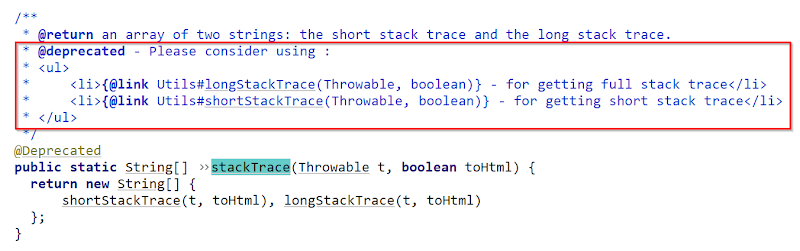
* **Verify: Chạy tiếp nếu có 1 step bất kì bị fail**
* **Ưu điểm:**
* Chạy qua hết all test cases/ steps - Khi có kết quả - biết được có bao nhiêu step bị fail trong 1 testcase
* Phù hợp vs UI/ E2E testing
* **Nhược điểm:**
* Có kết quả chậm
* 1 testcase có 200 steps:
* Tại step 70 - 106 - 180 -> Chỉ chạy 1 lần

|  |
| --- |
| @Test public void TC\_01\_Assert() {                 System.out.println("TC\_01 - Step 01: Open New Customer Page");                 newCustomerPage = homePage.openNewCustomerPage(driver);                  System.out.println("TC\_01 - Step 02: Verify New Customer page displayed");                 Assert.assertTrue(true);                  System.out.println("TC\_01 - Step 03: Verify New Customer form not displayed");                 Assert.assertTrue(false);                  System.out.println("TC\_01 - Step 04: Verify Home Page not displayed");                 Assert.assertTrue(true); }  @Test public void TC\_02\_Verify() {                 System.out.println("TC\_02 - Step 01: Open New Customer Page");                 newCustomerPage = homePage.openNewCustomerPage(driver);                  System.out.println("TC\_02 - Step 02: Verify New Customer page displayed");                 verifyTrue(true);                  System.out.println("TC\_02 - Step 03: Verify New Customer form not displayed");                 verifyTrue(false);                  System.out.println("TC\_02 - Step 04: Verify Home Page not displayed");                 verifyTrue(true); } |

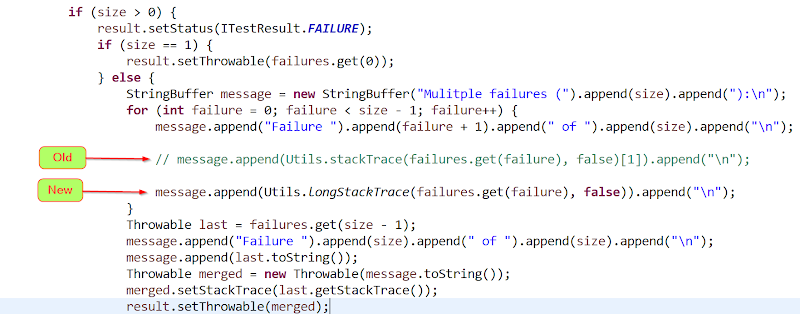
* TC\_01\_Assert
* Step 03 failed - Step 04 ko chạy
* TC\_02\_Verify
* Step 03 failed - Step 04 vẫn chạy tiếp



* verifyTrue/ False/ Equal
* Custom assert function of TestNG
* [https://gist.github.com/daominhdam/61f6a7a816959216a4f9accda2337dad](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://gist.github.com/daominhdam/61f6a7a816959216a4f9accda2337dad%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339470000%26amp;usg%3DAOvVaw0uPLj5DN9r-9e1eCMg3kqC&sa=D&source=editors&ust=1628671339500283&usg=AOvVaw1rb3GTdMxWSdW-Nk_T8JFZ)
* Thư viện Logging **-**[**Download**](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://drive.google.com/file/d/1ORcM8uojFhhA_y8QSnoHNwSS99-4OIPy/view?usp%253Dsharing%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339470000%26amp;usg%3DAOvVaw3hAENANxzhtSGWBF8M40sI&sa=D&source=editors&ust=1628671339500421&usg=AOvVaw1T0ENU6mAePui-iU6fKT12)
* VerificationFailures
* Get/ set failed status for testcase
* Set failed message to Report HTML
* [https://gist.github.com/daominhdam/c7cff49748e9b55d650f0a8eb724e2fe](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://gist.github.com/daominhdam/c7cff49748e9b55d650f0a8eb724e2fe%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339471000%26amp;usg%3DAOvVaw3h0G49Zw48wvu39T1_YRcb&sa=D&source=editors&ust=1628671339500624&usg=AOvVaw0lWiM4vV0qu97B9jk7wtI_)
* MethodListener
* Show all failed (number/ error message) to Console/ Log/ Report/ TestNG Result
* [https://gist.github.com/daominhdam/9b648a953028293fbd44f9205df92788](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://gist.github.com/daominhdam/9b648a953028293fbd44f9205df92788%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339471000%26amp;usg%3DAOvVaw29mDAG1aXYp6hd7Rgt_znm&sa=D&source=editors&ust=1628671339500800&usg=AOvVaw3lHeeDA0Ja1a-fK-Q7dEXc)
* Update: Nếu dùng cách cũ với hàm stackTrace bị deprecated (lỗi thời) - thay đổi bằng shortStackTrace hoặc longStackTrace



* [https://java-browser.yawk.at/org.testng/testng/6.14.3/org/testng/internal/Utils.java](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://java-browser.yawk.at/org.testng/testng/6.14.3/org/testng/internal/Utils.java%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628671339471000%26amp;usg%3DAOvVaw0jTjhOYgetiXGLxMGGUl_l&sa=D&source=editors&ust=1628671339501004&usg=AOvVaw3Z3MiQ4MIFZdl5IDdZBhhY)



* Add MethodListener class to testng.xml file

|  |
| --- |
| <listeners>         <listener class-name="commons.MethodListener" /> </listeners> |

