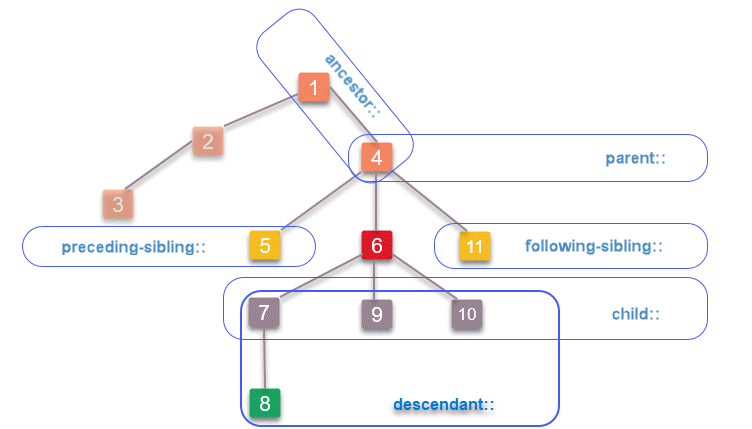
**Dynamic Locator**

**Rest Parameter**

**Constants Class**



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

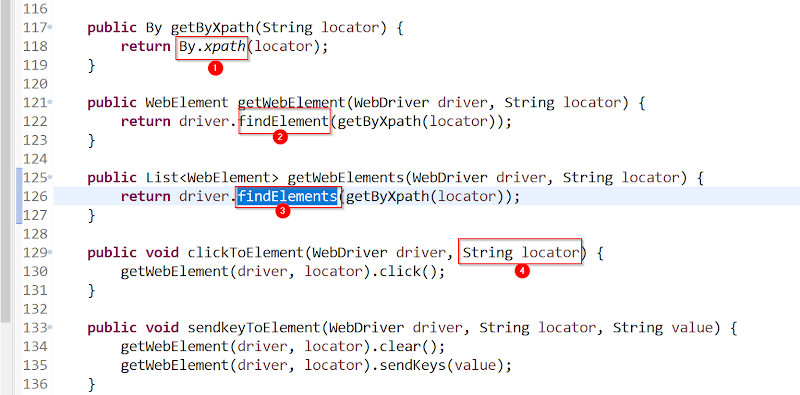
***Skype:****daominhdam (Đào Minh Đảm)*

[***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%3D1628670811103000%26amp;usg%3DAOvVaw0nuxpNMFVoCfqAafl_fl3-&sa=D&source=editors&ust=1628670811170983&usg=AOvVaw3EuVHyLCTv42kzOyG_arR6)*-*[***https://automationfc.com***](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://automationfc.com%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628670811104000%26amp;usg%3DAOvVaw1rIlR_kkk-HSi0o1rO7Egh&sa=D&source=editors&ust=1628670811171081&usg=AOvVaw38tt_DE4lJmdFFCxMFC6j6)

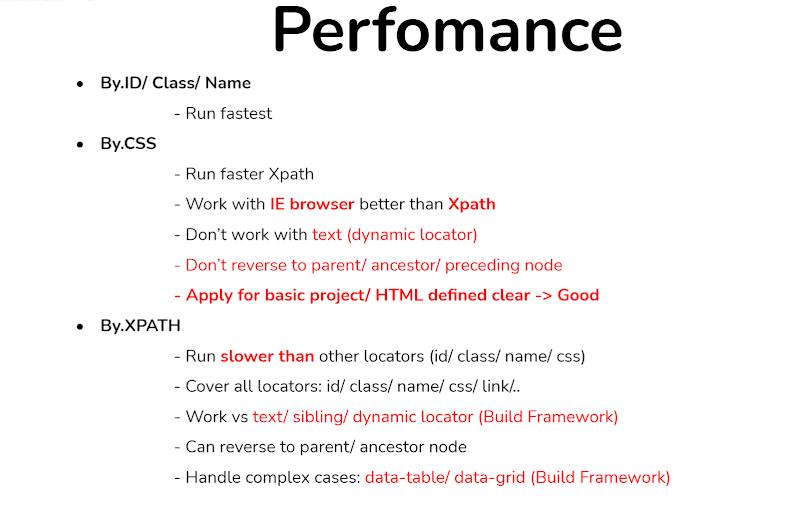
***Tài liệu này chỉ sử dụng trong khóa học - vui lòng không chia sẻ công khai***  
-------------------------------------------------------------------------------------------------------------------------------

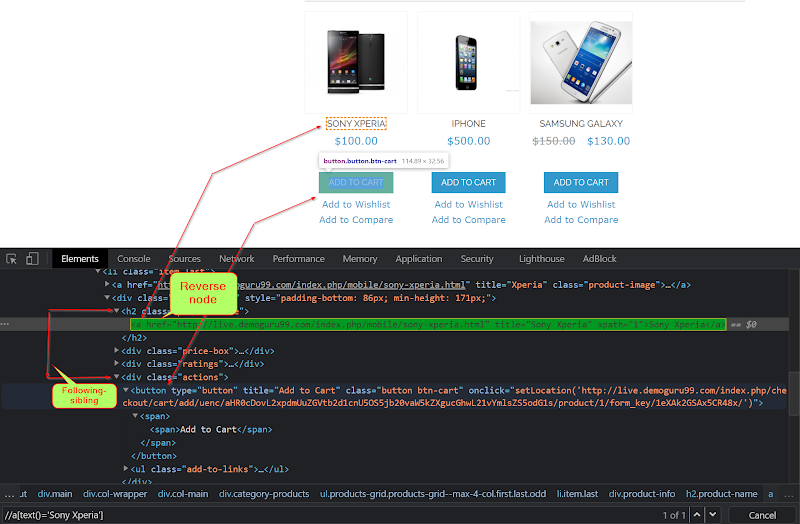
**Xpath vs CSS**

* **findElement**

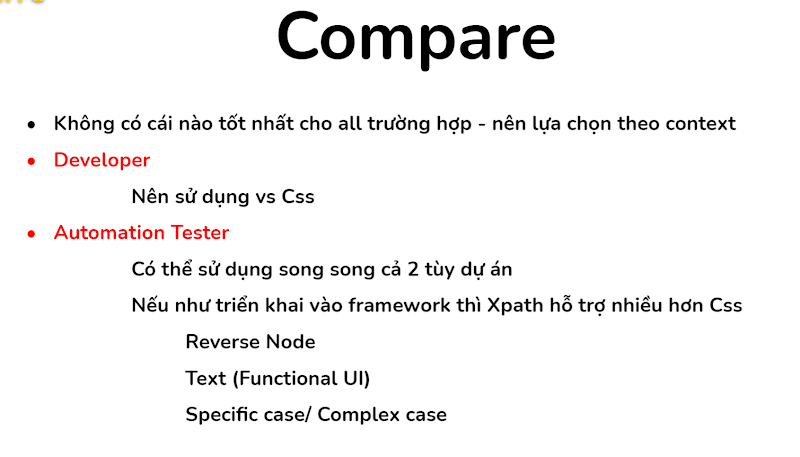


* **Perfomance**



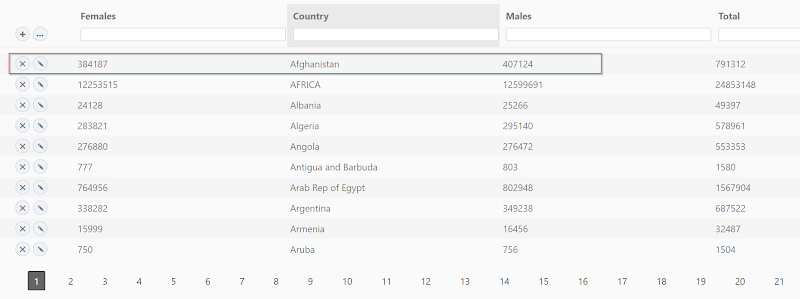


* **Compare**



**Issue**

* **Issues**
* 14 locator đại diện cho 14 page trong AbstractPageUI có thể tối ưu được nữa không?
* Những locator có format giống nhau - chỉ khác nhau bởi value
* //a[text()='Manager']
* //a[text()='New Customer']
* //a[text()='Edit Customer']
* Verify cho những table có nhiều cột/ giá trị
* [https://www.jqueryscript.net/demo/CRUD-Data-Grid-Plugin-jQuery-Quickgrid](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://www.jqueryscript.net/demo/CRUD-Data-Grid-Plugin-jQuery-Quickgrid/%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628670811107000%26amp;usg%3DAOvVaw10cTPiYDC9WAcV_5YjMML_&sa=D&source=editors&ust=1628670811172044&usg=AOvVaw0cwl3y8DLFTkvnhVo8I6XG)
* Click vào icon Delete/ Edit dựa vào các giá trị ở cột Females/ Country/ Males
* [https://www.jqueryscript.net/demo/jQuery-Dynamic-Data-Grid-Plugin-appendGrid](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://www.jqueryscript.net/demo/jQuery-Dynamic-Data-Grid-Plugin-appendGrid/%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628670811107000%26amp;usg%3DAOvVaw0IRXq2ZXelsKbW8SEiWMDx&sa=D&source=editors&ust=1628670811172227&usg=AOvVaw15I_UWKHr0J3dvAOjIvjzh)
* //td[@data-key='females' and text()='384187']/following-sibling::td[@data-key='country' and text()='Afghanistan']/following-sibling::td[@data-key='males' and text()='407124']/preceding-sibling::td[@class='qgrd-actions']/button[@class='qgrd-edit-row-btn']



**String Format**

* Sử dụng String.format và %s

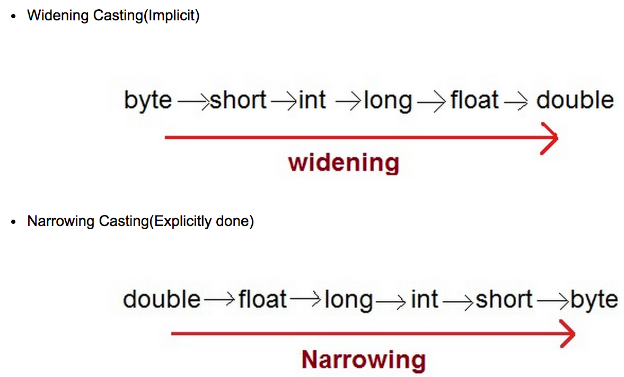
|  |
| --- |
| public class DynamicLocator {          public static void main(String[] args) {                 String NEW\_ACCOUNT\_LINK = "//a[text()='New Account']";                 String DEPOSIT\_LINK = "//a[text()='Deposit']";                 String NEW\_CUSTOMER\_LINK = "//a[text()='New Customer']";                 String HOMEPAGE\_LINK = "//a[text()='Manager']";                 String DYNAMIC\_LINK\_1\_PARAM = "//a[text()='%s']";                 String DYNAMIC\_LINK\_2\_PARAM = "//a[text()='%s']//a[text()='%s']";                 String DYNAMIC\_LINK\_3\_PARAM = "//a[text()='%s']//a[text()='%s']//a[text()='%s']";                 String DYNAMIC\_EDIT\_TABLE = "//td[@data-key='females' and text()='%s']/following-sibling::td[@data-key='country' and text()='%s']/following-sibling::td[@data-key='males' and text()='%s']/preceding-sibling::td[@class='qgrd-actions']/button[@class='qgrd-edit-row-btn']";                 String DYNAMIC\_DELETE\_TABLE = "//td[@data-key='females' and text()='%s']/following-sibling::td[@data-key='country' and text()='%s']/following-sibling::td[@data-key='males' and text()='%s']/preceding-sibling::td[@class='qgrd-actions']/button[@class='qgrd-remove-row-btn']";                  clickToElement(NEW\_CUSTOMER\_LINK);                 clickToElement(NEW\_ACCOUNT\_LINK);                 clickToElement(DEPOSIT\_LINK);                 clickToElement(HOMEPAGE\_LINK);                  clickToElement(DYNAMIC\_LINK\_1\_PARAM, "New Account");                 clickToElement(DYNAMIC\_LINK\_1\_PARAM, "New Customer");                 clickToElement(DYNAMIC\_LINK\_1\_PARAM, "Manager");                 clickToElement(DYNAMIC\_LINK\_1\_PARAM, "Deposit");                                  clickToElement(DYNAMIC\_LINK\_2\_PARAM, "Female", "Afghanistan");                 clickToElement(DYNAMIC\_LINK\_2\_PARAM, "Male", "Albania");                                  clickToElement(DYNAMIC\_LINK\_3\_PARAM, "Female", "Afghanistan", "Male");                 clickToElement(DYNAMIC\_LINK\_3\_PARAM, "Male", "Albania", "Female");                                  clickToElement(DYNAMIC\_DELETE\_TABLE, "12253515", "AFRICA", "12599691");                 clickToElement(DYNAMIC\_DELETE\_TABLE, "384187", "Afghanistan", "407124");                                  clickToElement(DYNAMIC\_EDIT\_TABLE, "12253515", "AFRICA", "12599691");                 clickToElement(DYNAMIC\_EDIT\_TABLE, "384187", "Afghanistan", "407124");         }          public static void clickToElement(String pageName) {                 System.out.println(pageName);         }                  public static void clickToElement(String pageName, String dynamicValue) {                 System.out.println(String.format(pageName, dynamicValue));         }                  public static void clickToElement(String pageName, String dynamicValue\_01, String dynamicValue\_02) {                 System.out.println(String.format(pageName, dynamicValue\_01, dynamicValue\_02));         }                  public static void clickToElement(String pageName, String dynamicValue\_01, String dynamicValue\_02, String dynamicValue\_03) {                 System.out.println(String.format(pageName, dynamicValue\_01, dynamicValue\_02, dynamicValue\_03));         }                  public static void clickToElement(String locator, String... dynamicValue) {                 System.out.println("Click to Element: " + String.format(locator, (Object[]) dynamicValue));         } } |

* **String.format -**[**Link**](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://gpcoder.com/2352-huong-dan-su-dung-string-format-trong-java/%2523Dinh_dang_kieu_chuoi_String%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628670811118000%26amp;usg%3DAOvVaw2IJQ9GQ6lFk8QNdeejJa9e&sa=D&source=editors&ust=1628670811174614&usg=AOvVaw2vw37AAbUlnCtixt9CIMrE)



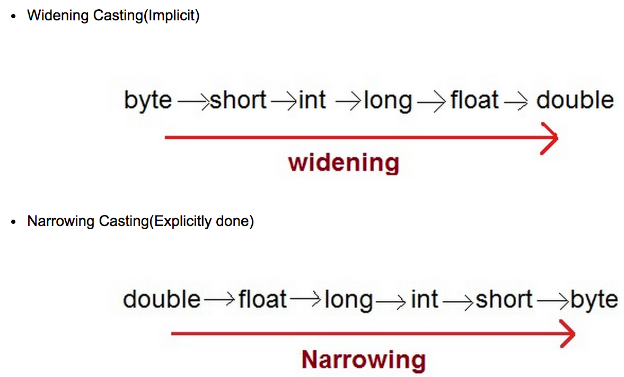
**Casting Type (Ép kiểu)**

* Ép kiểu ngầm định (Từ kiểu dữ liệu có range nhỏ đến lớn) -> không bị mất dữ liệu
* byte có thể ép kiểu sang short, int, long, float, double
* short có thể ép kiểu sang int, long, float, double
* int có thể ép kiểu sang long, float, double
* long có thể ép kiểu sang float, double
* float có thể ép kiểu sang double



|  |
| --- |
| byte i = 50; short j = i; int k = j; long l = k; float m = l; double n = m; |

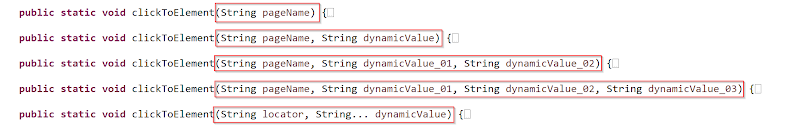
* Ép kiểu tường minh (Từ kiểu dữ liệu có range lớn về nhỏ) -> mất dữ liệu nếu phạm vi vùng nhớ ko đủ



|  |
| --- |
| int i = 50; short j = (short) i;  double d = 7.5; int i = (int) d;  public String getDynamicLocator(String xpathValue, String... values) {         xpathValue = String.format(xpathValue, (Object[]) values);         return xpathValue; }  public void clickToElement(WebDriver driver, String xpathValue, String... values) {         find(driver, getDynamicLocator(xpathValue, values)).click(); } |

**Polymorphism**

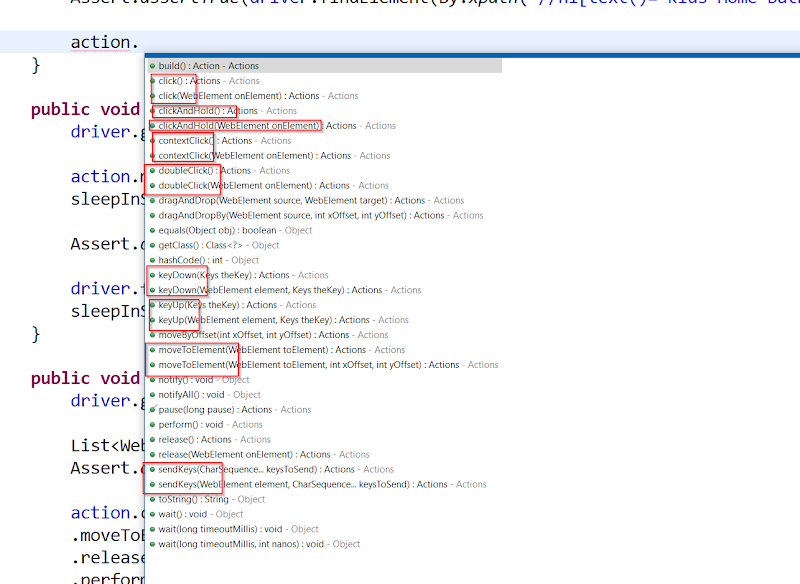
* **Tính đa hình (Polymorphism) trong OOP (Apply trong automation testing) => Overloading**
* Chồng hàm là trong phạm vi một class có thể định nghĩa nhiều phương thức (hàm)
* Cùng số lượng tham số nhưng phải khác kiểu dữ liệu
* Khác số lượng tham số nhưng không cần khác kiểu dữ liệu



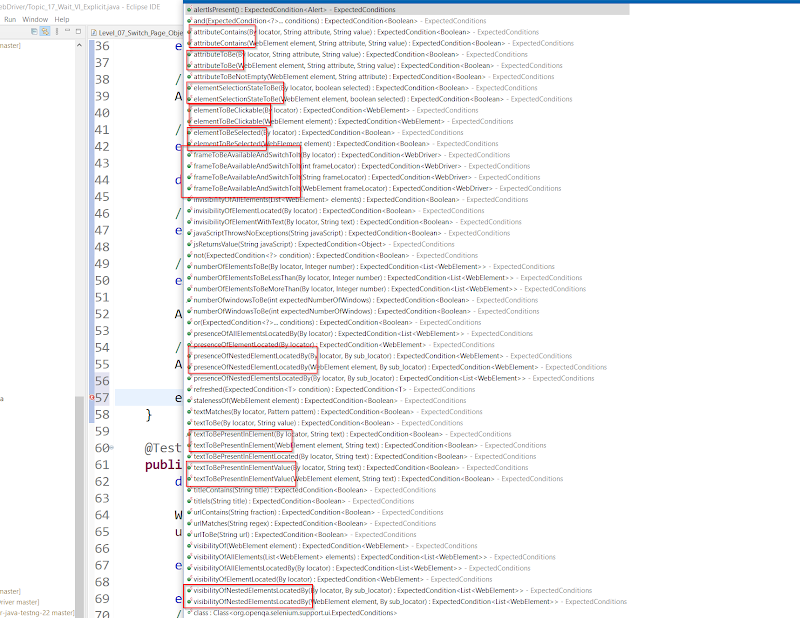
|  |
| --- |
| public boolean isControlDisplayed(WebDriver driver, String locator) {..} public boolean isControlDisplayed(WebDriver driver, String locator, String value) {..} public boolean isControlDisplayed(WebDriver driver, String locator, String... value) {..} |

|  |
| --- |
| // Constructor in PageObject  public HomePage(WebDriver driver) { ... } public HomePage(WebDriver driver, WebElement element) { ... }  public HomePage(WebDriver driver, WebDriverWait wait) { ... } |

* **Selenium WebDriver**
* **UserInteraction (Actions)**

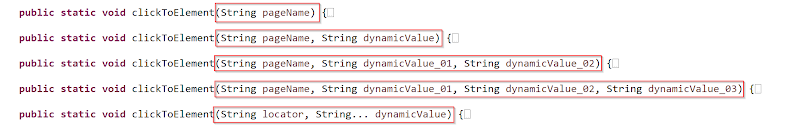


* **WebDriverWait**



**Rest Parameter (Variable Arguments)**

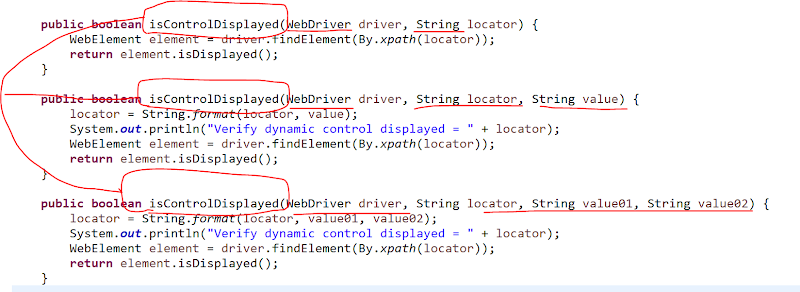
* **Tham số cuối cùng**
* [**https://www.geeksforgeeks.org/variable-arguments-varargs-in-java/**](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://www.geeksforgeeks.org/variable-arguments-varargs-in-java/%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628670811126000%26amp;usg%3DAOvVaw1xTDMZPLGX0Zw8Nlvi2tBY&sa=D&source=editors&ust=1628670811176781&usg=AOvVaw0EIJMLGc_3l_6MyJyF3zsL)
* Thay vì có thêm 1 tham số phải define thêm 1 hàm trong AbstractPage - sử dụng rest parameter để nhóm các **tham số có kiểu dữ liệu giống nhau** thành 1
* Không biết chắc chắn số lượng tham số cần có của một hàm
* Phân biệt với các parameter thông thường thông qua 3 dấu chấm **...**được đặt phía trước tên param
* Chỉ sử dụng khi các tham số là cuối cùng - như hình dưới chỉ cần hàm clickToElement dưới cùng để xử lí cho 3 hàm bên trên với các tham số: dynamicValue, value\_01, value\_02, value\_03,...



* **Add thêm các hàm vào AbstractPage/ BasePage**

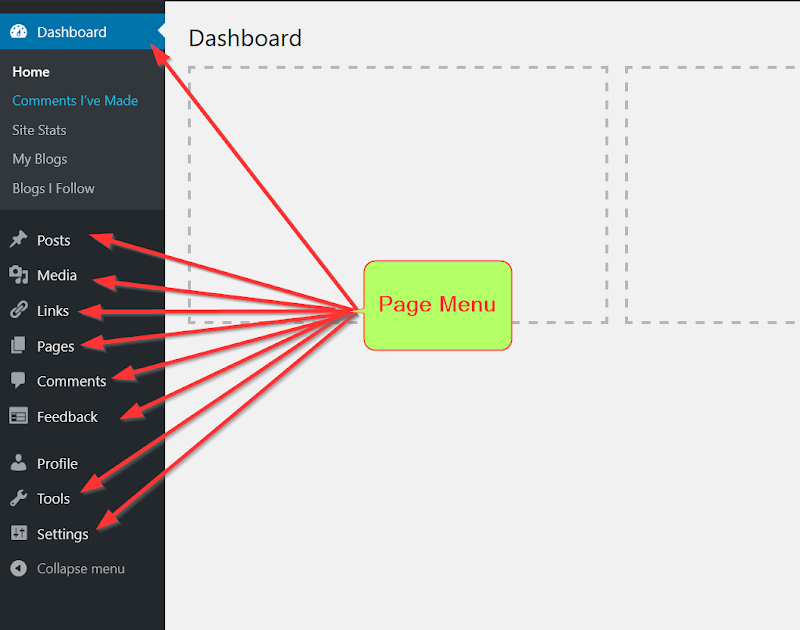
|  |
| --- |
| public String castRestParamter(String locator, String... values) {         locator = String.format(locator, (Object[]) values);         return locator; } |

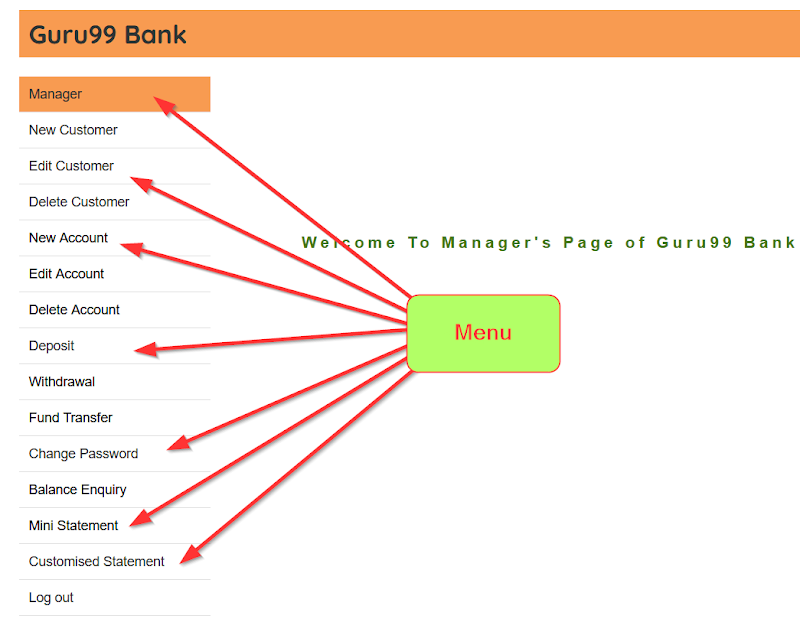
* clickToElement(WebDriver driver, String locator, String... values) {..}
* sendkeyToElement(WebDriver driver, String locator, String valueToSendkey, String... dynamicValue) {..}
* getTextInElement(WebDriver driver, String locator, String... values) {..}
* isControlDisplayed(WebDriver driver, String locator, String... values) {..}
* waitToElementVisbile(WebDriver driver, String locator, String... values) {..}
* waitForElementInvisible(WebDriver driver, String locator, String... values) {..}
* waitToElementClickable(WebDriver driver, String locator, String... values) {..}
* selectItemInDefaultDropdown….
* getFisrtSelectedItemInDefaultDropdown....
* getAttributeValue….
* getElementSize…
* highlightElement…
* clickToElementByJS...



**Handle Dynamic Page**

* **Apply cho Menu/ Footer/ Header**





* **Số lượng page ít (10-20 pages)**
* Dùng if-else hoặc switch-case
* Sử dụng tính năng cast type (ép kiểu của hàm - downcasting) ở tầng testcase
* Downcasting là dạng chuyển kiểu chuyển 1 đối tượng là một thể hiện của lớp cha xuống thành đối tượng là thể hiện của lớp con trong quan hệ kế thừa
* Downcasting 1 page object cụ thể có kiểu AbstractPage mà không xảy ra vấn đề trong quá trình biên dịch (Compile) và thực thi (Runtime)

|  |
| --- |
| public AbstractPage openDynamicPage(WebDriver driver, String pageName) {         waitToElementVisbile(driver, AbstractPageUI.DYNAMIC\_LINK, pageName);         clickToElement(driver, AbstractPageUI.DYNAMIC\_LINK, pageName);          switch (pageName) {         case "New Customer":                 return PageFactoryManager.getNewCustomerPage(driver);         case "Edit Customer":                 return PageFactoryManager.getNewCustomerPage(driver);         case "Deposit":                 return PageFactoryManager.getDepositPage(driver);         case "New Account":                 return PageFactoryManager.getNewAccountPage(driver);         default:                 return PageFactoryManager.getHomePage(driver);         } } |

|  |
| --- |
| // HomePage > NewCustomer newCustomerPage = (NewCustomerPageObject) homePage.openDynamicPage(driver, "New Customer"); Assert.assertTrue(newCustomerPage.isNewCustomerPageDisplayed());  // NewCustomer > NewAccount newAccountPage = (NewAccountPageObject) newCustomerPage.openDynamicPage(driver, "New Account"); Assert.assertTrue(newAccountPage.isNewAccountPageDisplayed());  // NewAccount > Deposit depositPage = (DepositPageObject) newAccountPage.openDynamicPage(driver, "Deposit"); Assert.assertTrue(depositPage.isDepositPageDisplayed());  // Deposit > HomePage homePage = (HomePageObject) depositPage.openDynamicPage(driver, "Manager"); Assert.assertTrue(homePage.isHomePageDisplayed());  // HomePage > NewAccount newAccountPage = (NewAccountPageObject) homePage.openDynamicPage(driver, "New Account"); Assert.assertTrue(newAccountPage.isNewAccountPageDisplayed()); |

* **Số lượng page nhiều (vài chục/ trăm pages)**

|  |
| --- |
| public void openDynamicMorePage(WebDriver driver, String pageName) {         waitToElementVisbile(driver, AbstractPageUI.DYNAMIC\_LINK, pageName);         clickToElement(driver, AbstractPageUI.DYNAMIC\_LINK, pageName); } |

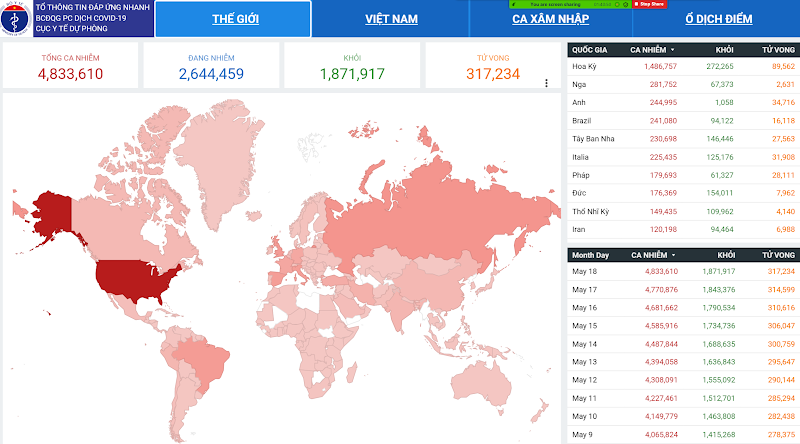
* Khởi tạo page trực tiếp trên tầng testcase

|  |
| --- |
| // New Account > New Customer newAccountPage.openDynamicMorePage(driver, "New Customer"); newCustomerPage = PageFactoryManager.getNewCustomerPage(driver); Assert.assertTrue(newCustomerPage.isNewCustomerPageDisplayed());                  // New Customer > Deposit newCustomerPage.openDynamicMorePage(driver, "Deposit"); depositPage = PageFactoryManager.getDepositPage(driver); Assert.assertTrue(depositPage.isDepositPageDisplayed()); |

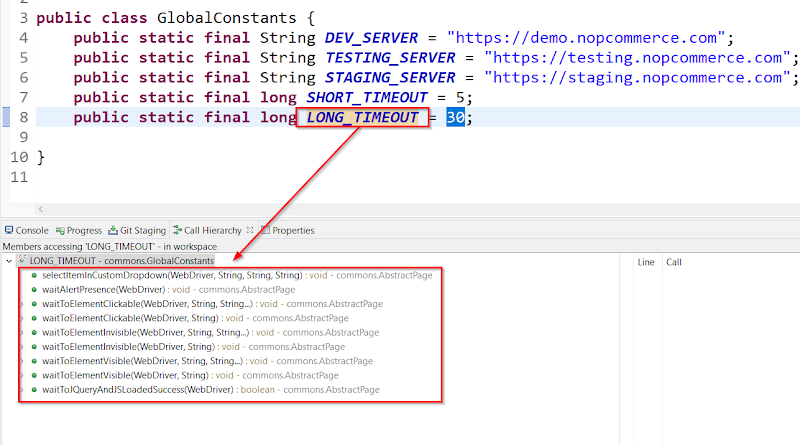
* Reference:

[**https://gpcoder.com/2406-co-che-upcasting-va-downcasting-trong-java/**](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://gpcoder.com/2406-co-che-upcasting-va-downcasting-trong-java/%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1628670811139000%26amp;usg%3DAOvVaw2prR8RXrryONc-Hmpp2mpF&sa=D&source=editors&ust=1628670811180196&usg=AOvVaw0YdPWJ_qaz_zFOV85m0aLM)

**Global Constants Class**



* **Class chứa những dữ liệu dùng chung cho toàn bộ system (project)**



* Quản lí tập trung thông tin của System
* Tên các server/ link: Dev - Test - Staging - PreProduct - Production - Live
* User/ Pass truy cập vào các AUT (Application Under Test)
* Tên Database/ User/ Pass theo từng server
* Biến timeout: Short timeout/ Long timeout
* Số lần cần retry lại test failed
* Đường dẫn tương đối đến thư mục download/ file data (xml/ excel/ json/..)/ file upload



|  |
| --- |
| public class GlobalConstants {         public static String ROOT\_FOLDER = System.getProperty("user.dir");         public static String BROWSER\_LOG\_FOLDER = ROOT\_FOLDER + "\\browserLog";         public static String DOWNLOAD\_FOLDER = ROOT\_FOLDER + "\\downloadFiles";         public static String UPLOAD\_FOLDER = ROOT\_FOLDER + "\\uploadFiles";         public static String WORDPRESS\_URL = "https://automationfc.wordpress.com/wp-admin";         public static String USER\_NAME = "automationeditor";         public static String PASSWORD = "automationfc";         public static long LONG\_TIMEOUT = 30;         public static long SHORT\_TIMEOUT = 10; } |