Xpaths:

String functions split implementation

Assertions.

**Hard assert:** The java statements following by assertion will not be executed if the assertion fails.

Eg: Assert.*assertEquals*(text, "You are successfull logged in.");

//tagname[@attributeName=’attributeVaalue’]

**Dropdowns:**

**Static dropdown**

**Updated/Latest dropdown**

**Dynamic Dropdown**

**Auto suggestive dropdrown**

Parent – child relationship

Parent xpath //a[@value='HYD']

A=2; b = 3

B= 3 ; a =2

**Handling Alerts:**

{"Cucumber", "Brocolli"} **=======** Brocolli - 1 Kg

Brocolli - 1 Kg

Brocolli – 0

* 1

1 - 2

Kg – 3

Brocolli - 1 Kg

Synchronization: Aligning(matching) the speed of browser with the execution speed

**500 millisecond** is the default polling time in implicit wait and explicit wait in selenium.

Implicit Wait:

Is applicable to each and every line of your code.

Explicit wait:

1. **WebDriver Wait - here polling time is fixed – 500ms**
2. **Fluent Wait -** 
   1. **here you can define the polling time - you can change the polling time 2 sec or any other values.**
   2. **You can skip some exceptions in here.**

Thread.sleep

Code:

Implicit Wait: 10000ms = 10 sec – max liit before throwing no such element exeception

Pros :Code will be simple and readable

Cons: Performance issue

1

2

3

Thread.sleep(8000)

4 - failing

5

Thread.sleep(6000)

6- failing

,

,

Thread.sleep(4000)

9 - failing

,

,

,10

WebDriver Wait: polling time :

// this will also go to next step if the element is located with in less than 10 sec

// polling time - it is the frequency with which selenium monitors an element(target element) - 500ms- half sec

Pros:

Wait is not applied to all elements. It is only applicable to targeted element

Cons: it has multiple line of code. If you want to apply this for 10 elements. You have to write 10 times.

Actions Class:

Mouse hover on an element

Performing keyboard actions and mouse actions

Performing right click(context click)

Double click on an elemnt

Drag and drop

Window Handles:

Frames:

Iframes are containers placed on top of a webpage but not built inside the same page

They are hosted into the wepage from another html source.

Dealing with link in a page:

A = anchor tag, this node contains a href attribute whose value is link address

Window scrolling

Element scrolling /scroll bar

CSS locator:

By id

//input[@id="inputUsername"] -> xpath

input#inputUsername -> css selector

By classname

//div[@class="forgot-pwd-container"] -> xpath

div.forgot-pwd-container -> css path

**Total - 0**

**Amount - 1**

**Collected: - 2**

**296 – 3**

* Invoking multiple windows / **tab** using one driver object]
* Find Dimensions of web elements
* SSL certificates validates

**TestNG:**

1. **Importance if testNG**

* Multiple testcases.
* Parallelly
* Execution time will be reduced
* We can generate proper reports
* We can run selected testcases.
* we can have good control over the execution
* parametrization

1. Installation of testNG
2. TestNG xml files
3. Include and Exclude mechanism
4. TestNG annotations
5. Groups
6. Data provider annotation
7. Helper attributes
8. Parametrization
9. Optional annotation
10. Prioritizing
11. TestNG Listerner
12. Parallel execution

**Before suite >**

**Before Test >**

**Before Class >**

**Before Method >**

**Test >**

**After Method >**

**After Class >**

**After Test >**

**After Suite**

**Groups:**

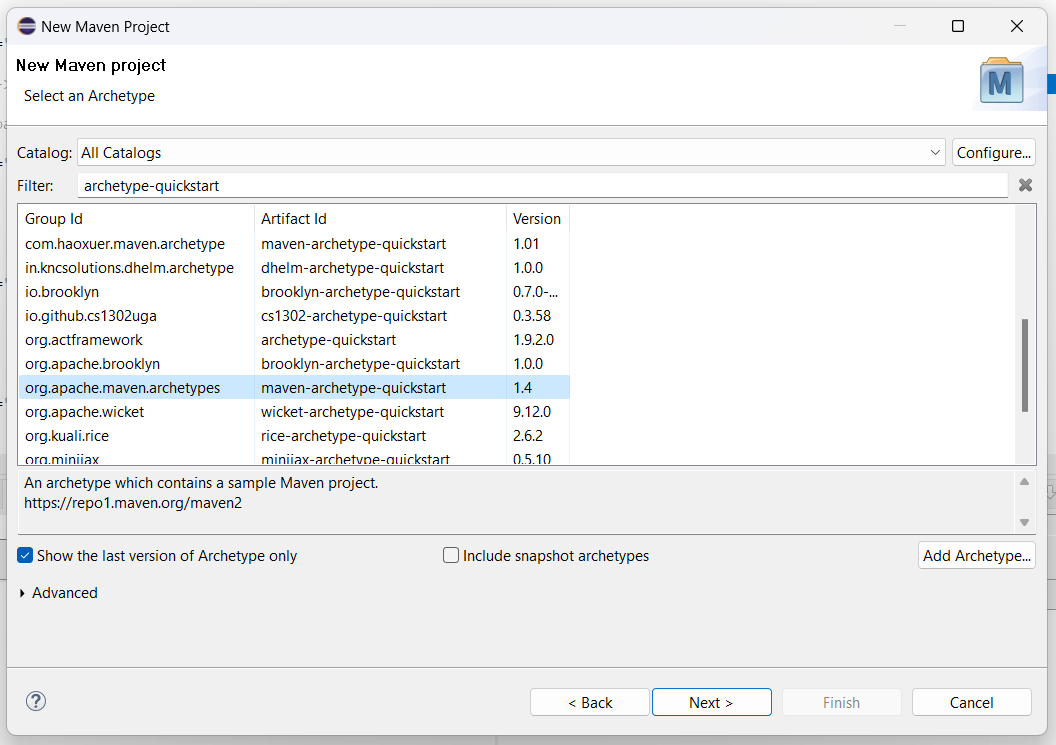
**Prioritizing:**

**TestNG executes testcases in alphabetical order**

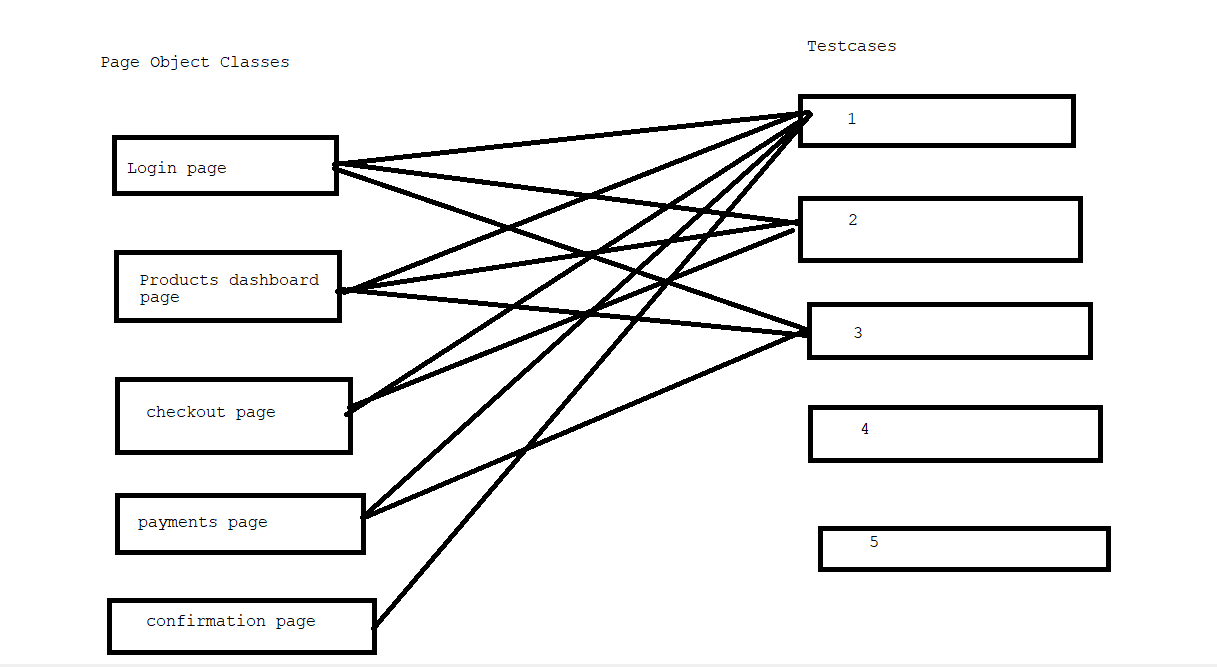
**Alphabetical order> priority tag**

**Parametrization is about passing data to testng tests from xml file.**

**This converts hardcoded testcases to dynamic testcases.**

****

**Page Object Model**

****