**SYNCHRONIZATION:**

**Matching the speed** of selenium with the speed of application, to avoid synchronisation issue is called “Synchronization”.

**What is synchronization issue:**

Selenium is too fast but application might take time to load, hence this **time mismatch** is called Synchronisation issue.

There are 2 types of waits to match selenium speed

1. Static wait
2. Dynamic wait

**1.Static wait:**

Static wait means **waiting time is fixed**. Here we use **Thread.sleep(3000)** means compulsorily we have to wait for 3 seconds.

**2. Dynamic Wait:**

Dynamic Wait - **waiting period is not fixed**. It is changing according to application speed.

There are 2 Types of Dynamic wait

1. Implicitlywait
2. Explicitlywait

-------------------------------------------- \*\*\*\*\* ---------------------------------------------------------------

**1.Implicitlywait:**

Here no need to give any condition. It will wait for findElement() method and findElements() method.

**Syntax:**

|  |
| --- |
| **driver.manage().timeouts().implicitlyWait(20, TimeUnit.SECONDS);** |

here 20 means maximum time until control should wait before giving **NoSuchElementException**

**Note:**

implicitlywait statement can be declared once at the top of the code. which can work for all the findElement() method/findElements() methods.

Explanation of ImplicitlyWait Flow Diagram:

When the control comes to findElement()/findElements(). It will check whether the element is present or not. If the element is present then findElement() will return the address. But If element is not present then it will check whether implicitlywait statement is mentioned or not If no then we will get **NoSuchElementException/empty List**

If it is mentioned then it will check whether time is over or not If time is over but element not found, **NoSuchElementException**

But if time is not over then control will wait for 1/2 sec or 500 ms and again check whether element is present or not This 500 ms duration is called **polling period.** And it is mentioned in **FluentWait Class.**

-------------------------------------------- \*\*\*\*\* ---------------------------------------------------------------

**2.ExplicitlyWait:**

Here we need to give condition. It will work for any method. create an object of **'WebDriverWait' Class.**

**Syntax:**

|  |
| --- |
| **WebDriverWait wait=new WebDriverWait(reference variable of WebDriver, timeoutInSeconds);** |

ExplicitlyWait will work for any method including findElement() and findElements().

Here we need to create an object of WebDriverWait Class and we use the static methods of

ExpectedConditions Class to give the condition.

WebDriverWait class and ExpectedConditions Class is imported from 'org.openqa.selenium.support.ui' package.

Flow Diagram Explanation:

When the control comes to wait.until() statement, it will check whether the condition is true or false.

If condition is true, then control goes to the next statements and start executing them. If condition is false, then it will check maximum time is over or not. If yes that means condition never became true within the maximum time. it gives **TimeoutException.**

If No, it waits for 1/2 sec or 500 ms and then checks again condition became true or not.

-------------------------------------------- \*\*\*\*\* ---------------------------------------------------------------

**IMPLICITYWAIT:**

1.ImplicitlyWait is used to synchronize findElement() and findElements() only.

2.No need to mention any condition.

3.No need to create Object.

4.TimeUnit can be days,hours,mins,seconds,milliseconds,microSeconds,nanoSeconds.

5.In ImplicitlyWait if element not found within maximum time we get **NoSuchElementException/EmptyList.**

6.Implicitly wait is a kind of global wait, because We can mention implicitly Wait. once at the top of the code which works for every findelement()/findElements() throughout the code"

7.**Syntax: driver.manage().timeouts().implicitlyWait(long arg, TimeUnit)**

8.Example: **driver.manage().timeouts().implicitlyWait(20, TimeUnit.SECONDS)**

-------------------------------------------- \*\*\*\*\* ---------------------------------------------------------------

**EXPILICITYWAIT:**

1.Explicitly Wait is used to synchronize any method

2.We should provide condition according to the situation

3.We create an object of **WebDriverWait class**

4.TimeUnit **can only be Seconds**

5.In Explicitly Wait If condition does not become true within maximum time then we get **TimeoutException**

6.Explicitly Wait is a kind of local wait which has to be mentioned before every **synchronisation issue**

7.**WebDriverWait wait=new WebDriverWait(WebDriver reference, TimeoutInSeconds)**

8.**wait.until(ExpectedConditions.RequiredCondition).**

-------------------------------------------- \*\*\*\*\* ---------------------------------------------------------------