# **Selenium Wait – Implicit, Explicit and Fluent Waits**

## **Why Do We Need Waits In Selenium?**

Most of the web applications are developed using [Ajax](https://www.guru99.com/php-ajax.html) and [Javascript](https://www.guru99.com/interactive-javascript-tutorials.html). When a page is loaded by the browser the elements which we want to interact with may load at different time intervals.

Not only it makes this difficult to identify the element but also if the element is not located it will throw an “**ElementNotVisibleException**” exception. Using Selenium Waits, we can resolve this problem.

## **Implicit Wait in Selenium**

The **Implicit Wait in Selenium** is used to tell the web driver to wait for a certain amount of time before it throws a “No Such Element Exception”. The default setting is 0. Once we set the time, the web driver will wait for the element for that time before throwing an exception.

Selenium Web Driver has borrowed the idea of implicit waits from Watir.

### **Implicit Wait syntax:**

driver.manage().timeouts().implicitlyWait(TimeOut, TimeUnit.SECONDS);

## **Explicit Wait in Selenium**

The **Explicit Wait in Selenium** is used to tell the Web Driver to wait for certain conditions (Expected Conditions) or maximum time exceeded before throwing “ElementNotVisibleException” exception. It is an intelligent kind of wait, but it can be applied only for specified elements. It gives better options than implicit wait as it waits for dynamically loaded Ajax elements.

Once we declare explicit wait we have to use “**ExpectedConditions**” or we can configure how frequently we want to check the condition using **Fluent Wait**. These days while implementing we are using **Thread.Sleep()** generally it is not recommended to use

### **Explicit Wait syntax:**

WebDriverWait wait = new WebDriverWait(WebDriverRefrence,TimeOut);

The following are the Expected Conditions that can be used in Selenium Explicit Wait

1. alertIsPresent()
2. elementToBeClickable()
3. elementToBeSelected()
4. frameToBeAvaliableAndSwitchToIt()
5. invisibilityOfTheElementLocated()
6. invisibilityOfElementWithText()
7. presenceOfElementLocated()
8. textToBePresentInElement()
9. titleContains()
10. visibilityOf()
11. visibilityOfAllElements()
12. visibilityOfAllElementsLocatedBy()
13. visibilityOfElementLocated()

## **Fluent Wait in Selenium**

The **Fluent Wait in Selenium** is used to define maximum time for the web driver to wait for a condition, as well as the frequency with which we want to check the condition before throwing an “ElementNotVisibleException” exception. It checks for the web element at regular intervals until the object is found or timeout happens.

**Frequency:** Setting up a repeat cycle with the time frame to verify/check the condition at the regular interval of time

Let’s consider a scenario where an element is loaded at different intervals of time. The element might load within 10 seconds, 20 seconds or even more then that if we declare an explicit wait of 20 seconds. It will wait till the specified time before throwing an exception. In such scenarios, the fluent wait is the ideal wait to use as this will try to find the element at different frequency until it finds it or the final timer runs out.

### **Fluent Wait syntax:**

Wait wait = new FluentWait(WebDriver reference)

.withTimeout(Duration.ofSeconds(SECONDS))

.pollingEvery(Duration.ofSeconds(SECONDS))

.ignoring(Exception.class);