

In Selenium Java, there are three primary types of waits that are commonly used to handle synchronization between the script and the web page elements:

1. Implicit Wait

- **Definition:** Implicit wait is applied globally and tells the WebDriver to poll the DOM for a certain amount of time when trying to find an element, before throwing a NoSuchElementException.
- **How it works:** Once set, the implicit wait is applied for the lifetime of the WebDriver instance.

```
driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
```

- **Use case:** Useful when the elements are available after a reasonable delay, but you do not want to wait for each element explicitly.

2. Explicit Wait

- **Definition:** Explicit wait allows you to define conditions for specific elements and wait until those conditions are met. This type of wait is more precise compared to implicit wait.
- **How it works:** You can wait for a certain condition to occur (e.g., element to be visible, clickable, etc.) before proceeding with the next step.

```
WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(10));
```

```
WebElement element = wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("exampleId")));
```

- **Use case:** When waiting for a specific condition on a specific element (e.g., the element becomes clickable or visible).

3. Fluent Wait

- **Definition:** Fluent wait is a more advanced form of wait that allows you to define not only the maximum wait time but also the polling frequency and exceptions to ignore while waiting.
- **How it works:** It polls the DOM at regular intervals to check for a condition to be met.

```
Wait<WebDriver> fluentWait = new FluentWait<WebDriver>(driver)
```

```
.withTimeout(Duration.ofSeconds(30))
```

```
.pollingEvery(Duration.ofSeconds(5))
```

```
.ignoring(NoSuchElementException.class);
```

```
WebElement element =  
fluentWait.until(ExpectedConditions.visibilityOfElementLocated(By.id("exampleId")));
```

- **Use case:** Useful when you need to frequently poll for an element's presence or state, with a custom polling interval.

Each of these waits serves different use cases depending on how your web elements behave and when they are rendered.