

< Teach
Me
Skills />

Selenium WebDriver. Advanced



О сложном в Selenium



1

Ожидания и фреймы

2

Загрузка файлов

3

Actions

4

Работа с коллекциями

Неявные ожидания (ImplicitlyWait)

Ожидание, указывающее на то какое максимальное количество времени Selenium будет дожидаться появления элемента. В сущности срабатывает каждый раз при вызове функции **driver.findElement()** или **driver.findElements()**. Использовать высокие значения таймаута не рекомендуется. Дополнительно:

<https://www.guru99.com/implicit-explicit-waits-selenium.html>

```
driver.manage().timeouts().implicitlyWait(20,  
TimeUnit.SECONDS);
```



Явные ожидания (WebDriverWait)

Ожидания с указанием определенных условий, например видимость элемента либо его исчезновение. Использовать каждый раз при открытии/закрытие либо переходе на другую страницу. Также в местах, где данные появляются динамически (через какое-то время после загрузки страницы).

```
WebDriverWait wait = new WebDriverWait(driver, 10);
wait.until(ExpectedConditions.visibilityOfElementLocated(By.id("id")));
```



Так когда же вставлять ожидания?



1

Переход на новую страницу

2

Открытие/закрытие модального окна/pop-up

3

Появление/исчезновение элемента после действия

Типы ExpectedConditions

```
WebDriverWait wait = new WebDriverWait(driver, timeOutInSeconds: 10);
wait.until(ExpectedConditions.visibilityOfElementLocated(By lo... ExpectedCondition<WebElement>
wait.until(ExpectedConditions.visibilityOf(webElement element) ExpectedCondition<WebElement>
wait.until(ExpectedConditions.visibilityOfAllElements(We... ExpectedCondition<List<WebElement>>
wait.until(ExpectedConditions.visibilityOfAllElements(Li... ExpectedCondition<List<WebElement>>
wait.until(ExpectedConditions.visibilityOfAllElementsLocate... ExpectedCondition<List<WebElement>>
wait.until(ExpectedConditions.visibilityOfNestedElementsLocate... ExpectedCondition<List<WebElement>>
wait.until(ExpectedConditions.invisibilityOf(webElement element) ExpectedCondition<Boolean>
wait.until(ExpectedConditions.invisibilityOfAllElements(We... ExpectedCondition<Boolean>
wait.until(ExpectedConditions.invisibilityOfElementLocated(By lo... ExpectedCondition<Boolean>
wait.until(ExpectedConditions.invisibilityOfElementWithText(By lo... ExpectedCondition<Boolean>
wait.until(ExpectedConditions.invisibilityOfAllElements(List<WebE... ExpectedCondition<Boolean>
^↓ and ^↑ will move caret down and up in the editor >>
```

- Видимость одного/нескольких элементов по локатору/WebElement
 - Невидимость одного/нескольких элементов по локатору/WebElement



Типы ExpectedConditions

```
WebDriverWait wait = new WebDriverWait(driver, timeOutInSeconds: 10);
wait.until(ExpectedConditions.ffeek);
    m attributeToBeNotEmpty(WebElement el...) ExpectedCondition<Boolean>
    m attributeToBe(By locator, String at...) ExpectedCondition<Boolean>
    m attributeToBe(WebElement element, S... ExpectedCondition<Boolean>
    m attributeContains(By locator, Strin... ExpectedCondition<Boolean>
    m attributeContains(WebElement elemen... ExpectedCondition<Boolean>
    ^↓ and ^↑ will move caret down and up in the editor >>
```

- Ожидания наличия атрибута внутри элемента
(частичное или полное совпадение)



Типы ExpectedConditions

```
WebDriverWait wait = new WebDriverWait(driver, timeOutInSeconds: 10);
wait.until(ExpectedConditions.number);
    ↗ numberOfElementsToBeMoreThan ExpectedCondition<List<WebElement>>
    ↗ numberOfElementsToBe(By lo... ExpectedCondition<List<WebElement>>
    ↗ numberOfElementsToBeLessThan ExpectedCondition<List<WebElement>>
    ↗ numberOfWorkersToBe(int expectedNum... ExpectedCondition<Boolean>
Press ⌘. to choose the selected (or first) suggestion and insert a dot afterwards >>
```

- Ожидания видимости некоторого количества элементов
 - Ожидание количества открытых окон в браузере



Типы ExpectedConditions

```
wait.until(ExpectedConditions.fra...  
st > setUp()  
    m m frameToBeAvailableAndSwitchToIt(B... ExpectedCondition<WebDriver>...  
    m m frameToBeAvailableAndSwitchToIt(i... ExpectedCondition<WebDriver>...  
    m m frameToBeAvailableAndSwitchToIt(S... ExpectedCondition<WebDriver>...  
    m m frameToBeAvailableAndSwitchToIt(W... ExpectedCondition<WebDriver>...  
    ^↓ and ^↑ will move caret down and up in the editor >>
```

- Ожидания фреймов и переключение на них



И многое-многое другое

```
WebDriverWait wait = new WebDriverWait(driver, timeOutInSeconds: 10);
wait.until(ExpectedConditions.textToBe(By.id("id"), value: "text"));
wait.until(ExpectedConditions.alertIsPresent());
wait.until(ExpectedConditions.elementToBeSelected(By.id("id")));
wait.until(ExpectedConditions.elementToBeClickable(By.id("id")));
wait.until(ExpectedConditions.titleIs(""));
```

- Работа с текстом
- Работа с алертами
- Работа с title
- И многое-многое другое



Fluent Wait

Гибкие ожидания с указанием интервала, условий и некоторых действий между попытками, если необходимо. Сложные, используются редко

```
Wait<WebDriver> fluent = new FluentWait<>(driver)
    .withTimeout(30, SECONDS)
    .pollingEvery(5, SECONDS)
    .ignoring(NoSuchElementException.class);

WebElement foo = fluent.until(driver -> driver.findElement(By.id("foo")));
```



Работа в JS-консоли браузера

- Вкладка ‘Console’ браузера позволяет выполнять JS-код прямо на странице, где вы находитесь.
Примеры полезных запросов ниже:

```
//Получение всех атрибутов элемента
document.getElementById('add-to-cart-sauce-labs-fleece-jacket').click()
//Получение состояния загрузки страницы
document.readyState
```

Ожидание загрузки JS (Ajax)

При использовании AJAX:

- Пользователь заходит на веб-страницу и нажимает на какой-нибудь её элемент.
- Скрипт (на языке JavaScript) определяет, какая информация необходима для обновления страницы.
- Браузер отправляет соответствующий запрос на сервер.
- Сервер возвращает только ту часть документа, на которую пришёл запрос.
- Скрипт вносит изменения с учётом полученной информации (без полной перезагрузки страницы).



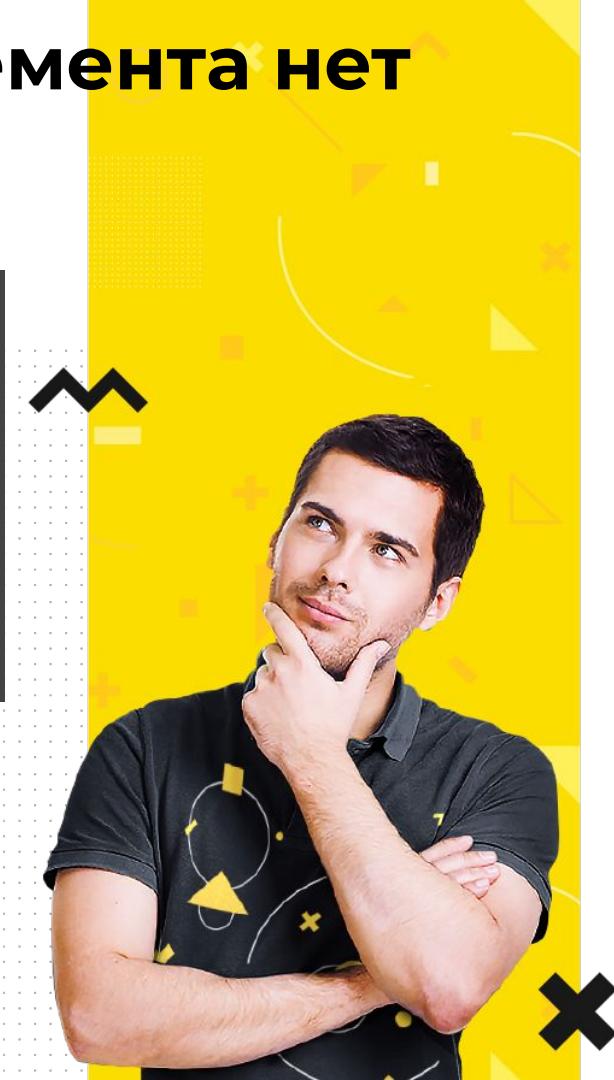
```
public void waitForPageLoaded() {  
    new ExpectedCondition<Boolean>() {  
        public Boolean apply(WebDriver driver) {  
            return ((JavascriptExecutor) driver).executeScript("return  
document.readyState").toString().equals("complete");  
        }  
    };  
}
```

Как проверить, что элемента нет на странице?

```
//Уменьшаем таймаут ожидания
driver.manage().timeouts().implicitlyWait(0, TimeUnit.SECONDS);

//Проверяем, что коллекция пуста
int numberOfElements = driver.findElements(By.tagName("локатор")).size();
assertEquals(numberOfElements, 0, "Элемент присутствует на странице");

//Возвращаем таймаут
driver.manage().timeouts().implicitlyWait(20, TimeUnit.SECONDS);
```



Frames

```
<html>
<body>
  <p id="1">HTML Page 1</p>
  <iframe>
    <html>
      <body>
        <p id="2"> HTML Page 2</p>
      </body>
    </html>
  </iframe>
</body>
</html>
```

- **IFrame** — это кадр внутри кадра. Это компонент **HTML**-элемента, который позволяет встраивать документы, видео и интерактивные медиафайлы на страницу. Делая **это**, вы можете отобразить дополнительную веб-страницу на главной странице. Элемент **iframe** позволяет включать часть содержимого из других источников.
- Пример:
<http://the-internet.herokuapp.com/iframe>



Actions

```
Actions actions = new Actions(driver);
actions. .
    m  perform()                                void
    m  click(WebElement target)                  Actions
    m  click()                                  Actions
    m  sendKeys(CharSequence... keys)           Actions
    m  sendKeys(WebElement target, CharSequence... keys) Actions
    m  build()                                 Action
    m  moveToElement(WebElement target)        Actions
    m  moveToElement(WebElement target, int xOffset, int yOffset) Actions
    m  clickAndHold()                          Actions
    m  clickAndHold(WebElement target)          Actions
    m  contextClick()                          Actions
    m  contextClick(WebElement target)          Actions
    m  doubleClick()                           Actions
    m  doubleClick(WebElement target)           Actions
Press ^ to choose the selected (or first) suggestion and insert a dot afterwards >> π
```

File Upload

```
driver.findElement(By.xpath("//input[@type='file']")).sendKeys("путь к файлу");
driver.findElement(By.cssSelector("Кнопка старта загрузки")).click();
wait.until(ExpectedConditions.invisibilityOf(loader));
```

- Пути только относительные
- Разделитель между папками/файлами в пути
ТОЛЬКО / либо File.separator
- Пример: <http://the-internet.herokuapp.com/upload>