### Ожидание появления элемента

#### **Примеры кода**

##### **Java**

import static org.openqa.selenium.support.ui.ExpectedConditions.presenceOfElementLocated;  
// ...  
WebDriverWait wait = new WebDriverWait(driver, 10/\*seconds\*/);  
WebElement element = wait.until(presenceOfElementLocated(By.name("q")));  
WebElement element2 = wait.until((WebDriver d) -> d.findElement(By.name("q")));

##### **C#**

WebDriverWait wait = new WebDriverWait(driver, TimeSpan.FromSeconds(10));  
IWebElement element = wait.Until(ExpectedConditions.ElementExists(By.Name("q")));  
IWebElement element2 = wait.Until(d => d.FindElement(By.Name("q")));

##### **Python**

from selenium.webdriver.support.wait import WebDriverWait  
from selenium.webdriver.support import expected\_conditions as EC  
from selenium.webdriver.common.by import By  
  
wait = WebDriverWait(driver, 10) # seconds  
# обратите внимание, что локатор передается как tuple!  
element = wait.until(EC.presence\_of\_element\_located((By.NAME, "q"))  
element2 = wait.until(lambda d: d.find\_element\_by\_name("q"))

##### **Ruby**

wait = Selenium::WebDriver::Wait.new(:timeout => 10) # seconds  
element = wait.until { @driver.find\_element(:name => "q") }

##### **JavaScript**

// wait в качестве условия может принимать промис, объект типа Condition или просто функцию, возвращает промис  
driver.wait(until.elementLocated(By.name('q')), 10000/\*ms\*/).then(function(element) {...});

### Ожидание исчезновения элемента

#### **Примеры кода**

##### **Java**

import static org.openqa.selenium.support.ui.ExpectedConditions.stalenessOf;  
// ...  
WebDriverWait wait = new WebDriverWait(driver, 10/\*seconds\*/);  
driver.navigate().refresh();  
wait.until(stalenessOf(element));

##### **C#**

WebDriverWait wait = new WebDriverWait(driver, TimeSpan.FromSeconds(10));  
driver.Navigate().Refresh();  
wait.Until(ExpectedConditions.StalenessOf(element));

##### **Python**

from selenium.webdriver.support.wait import WebDriverWait  
from selenium.webdriver.support import expected\_conditions as EC  
from selenium.webdriver.common.by import By  
  
wait = WebDriverWait(driver, 10) # seconds  
driver.refresh()  
wait.until(EC.staleness\_of(element)

##### **Ruby**

def staleness\_of(element)  
 element.enabled?  
 false  
rescue Selenium::WebDriver::Error::ObsoleteElementError  
 true  
end

wait = Selenium::WebDriver::Wait.new(:timeout => 10) # seconds  
driver.navigate.refresh

@wait.until { staleness\_of(element) }

##### **JavaScript**

driver.navigate().refresh().then(function(){  
 driver.wait(until.stalenessOf(element), 10000/\*ms\*/);  
});

### Ожидание видимости элемента

#### **Примеры кода**

##### **Java**

import static org.openqa.selenium.support.ui.ExpectedConditions.visibilityOf;  
// ...  
WebDriverWait wait = new WebDriverWait(driver, 10/\*seconds\*/);  
wait.until(visibilityOf(element));

##### **C#**

WebDriverWait wait = new WebDriverWait(driver, TimeSpan.FromSeconds(10));  
wait.Until(ExpectedConditions.ElementIsVisible(element));

##### **Python**

from selenium.webdriver.support.wait import WebDriverWait  
from selenium.webdriver.support import expected\_conditions as EC  
  
wait = WebDriverWait(driver, 10) # seconds  
wait.until(EC.visibility\_of(element)

##### **Ruby**

wait = Selenium::WebDriver::Wait.new(:timeout => 10) # seconds  
@wait.until { element.displayed? }

##### **JavaScript**

driver.wait(until.visibilityOf(element), 10000/\*ms\*/).then(function(element){...});

### Другие условия ожидания

#### **Ссылки**

Java -- <http://seleniumhq.github.io/selenium/docs/api/java/org/openqa/selenium/support/ui/ExpectedConditions.html>

C# -- <http://seleniumhq.github.io/selenium/docs/api/dotnet/html/T_OpenQA_Selenium_Support_UI_ExpectedConditions.htm>

Python -- <http://seleniumhq.github.io/selenium/docs/api/py/webdriver_support/selenium.webdriver.support.expected_conditions.html>

JavaScript -- <http://seleniumhq.github.io/selenium/docs/api/javascript/module/selenium-webdriver/lib/until.html>

### Ожидание загрузки страницы

#### **Ссылки**

Что означает "окончание загрузки страницы"? -- <http://barancev.github.io/page-loading-complete/>

Как Selenium ожидает окончания загрузки страницы? -- <http://barancev.github.io/how-selenium-waits-for-page-to-load/>

Что делать, если страница загружается слишком долго? -- <http://barancev.github.io/slow-loading-pages/>

### Особые ситуации

#### **Ссылки**

StaleTolerantWebDriver (Java) -- <https://github.com/barancev/decorated-webdriver/tree/master/src/main/java/ru/stqa/selenium/decorated/staletolerant>

Repeatable Actions (Java) -- <https://github.com/barancev/webdriver-repeatable-actions>

## **Задание 13. Сделайте сценарий работы с корзиной**

Сделайте сценарий для добавления товаров в корзину и удаления товаров из корзины.

Сценарий должен состоять из следующих частей:

1) открыть страницу какого-нибудь товара  
2) добавить его в корзину  
3) подождать, пока счётчик товаров в корзине обновится  
4) вернуться на главную страницу, повторить предыдущие шаги ещё два раза, чтобы в общей сложности в корзине было 3 единицы товара

5) открыть корзину (в правом верхнем углу кликнуть по ссылке Checkout)  
6) удалить все товары из корзины один за другим, после каждого удаления подождать, пока внизу обновится таблица