

Selenium

mechanizm wait

Jaki problem chcemy rozwiązać?

Nie wszystkie elementy na stronie załadowane są od razu

- Elementy graficzne
- Skrypty opóźniające ładowanie elementów
- Dynamicznie ładowane treści

Nie wszystkie elementy są w stanie umożliwiającym interakcję

- Nieaktywne elementy dopóki nie zostanie spełniony warunek
 - np. nieaktywny przycisk 'Zatwierdź' dopóki nie zaznaczymy zgody na regulamin


Mechanizm wait

Mechanizm wait w Selenium ma za zadanie ponawianie nieudanej akcji przez zdefiniowany okres czasu.

Dopiero po upływie czasu zgłaszany jest wyjątek.

Wait - wbudowany w język programowania

Java:

```
Thread.sleep(1000);  // value in ms
```

Python:

```
time.sleep(1) # value in s
```

“Pauzuje” program na zadany okres czasu.

- może się przydać do zidentyfikowania problemu
- absolutnie nie polecany do stosowania w finalnych wersjach skryptów

Wait - Implicit wait

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

Powoduje oczekiwanie określonego czasu zanim zostanie rzucony wyjątek. W tym czasie dana operacja jest powtarzana.

- Ustawiony globalnie dla całego drivera

Explicit wait

Java:

```
driver.get("https://google.com/ncr");
driver.findElement(By.name("q")).sendKeys("cheese" + Keys.ENTER);
// Initialize and wait till element(link) became clickable - timeout in 10 seconds
WebElement firstResult = new WebDriverWait(driver, Duration.ofSeconds(10))
    .until(ExpectedConditions.elementToBeClickable(By.xpath("//a/h3")));
```

Python:

```
WebDriverWait(driver).until(document_initialised)
el = driver.find_element(By.TAG_NAME, "p")
```

Explicit wait - Expected conditions

- alert is present
- element exists
- element is visible
- title contains
- title is
- element staleness
- visible text

Dla poszczególnych języków programowania warunki mogą być różne

Wait - Fluent wait

Podobny do explicit wait, tylko zamiast ExpectedConditions określamy jakie wyjątki mają być ignorowane.

Java:

```
// Waiting 30 seconds for an element to be present on the page, checking  
// for its presence once every 5 seconds.  
Wait<WebDriver> wait = new FluentWait<WebDriver>(driver)  
    .withTimeout(Duration.ofSeconds(30))  
    .pollingEvery(Duration.ofSeconds(5))  
    .ignoring(NoSuchElementException.class);  
  
WebElement foo = wait.until(new Function<WebDriver, WebElement>() {  
    public WebElement apply(WebDriver driver) {  
        return driver.findElement(By.id("foo"));  
    }  
});
```


Python:

```
driver = Firefox()  
driver.get("http://somedomain/url_that_delays_loading")  
wait = WebDriverWait(driver, 10, poll_frequency=1,  
    ignored_exceptions=[ElementNotVisibleException,  
        ElementNotSelectableException])  
element = wait.until(EC.element_to_be_clickable((By.XPATH, "//div")))
```

Wait - Kiedy stosować jaki wait

Implicit Wait	Explicit Wait / Fluent Wait
Ustawiany globalnie dla wszystkich elementów	Ustawiany dla konkretnych elementów
Nie specyfikujemy warunków na które należy czekać	Za pomocą ExpectedConditions definiujemy na co WebDriver powinien poczekać
Rekomendowany, jeżeli strona jest stosunkowo prosta i elementy są ładowane w zakładanym czasie	Rekomendowany dla dłużej ładujących się elementów oraz sprawdzenia określonych warunków

Wait - podsumowanie

- `sleep()` w języku programowania stosujemy tylko doraźnie i tymczasowo
- Nie należy łączyć Implicit i Explicit wait, bo efektem może być nieprzewidywalny czas i wydłużenie działania skryptu. Jeżeli jest ustawiony Implicit Wait, to przed zastosowaniem Explicit Wait należy go wyzerować i potem przywrócić