
Python
Browser Automation
by Selenium

Selenium 소개

- Selenium?
  - 브라우저를 자동화하는 도구.
  - 일반적으로 웹 애플리케이션의 테스팅 목적으로 사용됨.
  - 동적인 웹사이트, 비동기 통신에 대한 크롤링 등에 사용함.

- Selenium 설치
  - (커맨드 창에서) `python -m pip install selenium`

- WebDriver 설치
  - 자동화할 브라우저에 맞는 웹드라이버를 요구함.
  - https://pypi.org/project/selenium/

WebDriver 객체 생성

- WebDriver 객체 생성
  - 브라우저 자동화를 위한 WebDriver 객체 생성.

```
from selenium import webdriver

driver = webdriver.Chrome(드라이버 디렉토리 문자열)
```

암묵적인 기다림

- implicitly_wait() 함수
  - WebDriver 객체 생성 후, Selenium은 기본적으로 모든 웹 자원이 로드되는 걸 기다려주지만 보통 이를 암묵적으로 표현해줌.
  - WebDriver 객체의 implicitly_wait() 함수를 사용.

```
from selenium import webdriver

driver = webdriver.Chrome(드라이버 디렉토리 문자열)
driver. implicitly_wait(3)
```

URL 접속

- get() 함수
  - 로딩이 완료된 WebDriver 객체에서 원하는 URL에 접속하는 방법.

```
from selenium import webdriver

driver = webdriver.Chrome(드라이버 디렉토리 문자열)
driver. implicitly_wait(3)
driver.get(접속할 URL 문자열)
```

요소 찾기

- find_element_by_?() 함수
  - 정해진 기준에 따라 요소 하나를 찾는 함수.

함수	설명
find_element_by_id()	id 속성값으로 요소를 찾는다.
find_element_by_name()	name 속성값으로 요소를 찾는다.
find_element_by_xpath()	xpath 값으로 요소를 찾는다.
find_element_by_tag_name()	태그명으로 요소를 찾는다.
find_element_by_class_name()	class 속성값으로 요소를 찾는다.
find_element_by_css_selector()	CSS 선택자로 요소를 찾는다.

구글 검색창 선택 예제

- find_element_by_?() 함수 사용 예제

```
from selenium import webdriver

driver = webdriver.Chrome()
driver.implicitly_wait(3)
driver.get("https://www.google.co.kr/")

driver.find_element_by_name("q")
driver.find_element_by_xpath('//*[@id="tsf"]/div[2]/div/div[1]/div/div[1]/input')
driver.find_element_by_class_name('gLFyf')
driver.find_element_by_css_selector("div.vdLsw.gsfi+input")
```

요소 찾기

- find_elements_by_?() 함수
  - 정해진 기준에 따라 요소들을 찾는 함수(리스트 형태로 리턴).

함수	설명
find_elements_by_name()	name 속성값으로 요소를 찾는다.
find_elements_by_xpath()	xpath 값으로 요소를 찾는다.
find_elements_by_tag_name()	태그명으로 요소를 찾는다.
find_elements_by_class_name()	class 속성값으로 요소를 찾는다.
find_elements_by_css_selector()	CSS 선택자로 요소를 찾는다.

구글 검색창 선택 예제

- find_elements_by_?() 함수 사용 예제

```
from selenium import webdriver

driver = webdriver.Chrome()
driver.implicitly_wait(3)
driver.get("https://www.google.co.kr/")

driver.find_elements_by_name("q")[0]
driver.find_elements_by_xpath('//*[@id="tsf"]/div[2]/div/div[1]/div/div[1]/input')[0]
driver.find_elements_by_class_name('gLFyf')[0]
driver.find_elements_by_css_selector("div.vdLsw.gsfi+input")[0]
```

요소 찾기

- find_element(), find_elements() 함수
  - 기준을 직접 전달하여 요소를 찾는 함수.

```
from selenium.webdriver.common.by import By
```

함수	설명
By.ID	"id"
By.XPATH	"xpath"
By.NAME	"name"
By.TAG_NAME	"tag name"
By.CLASS_NAME	"class name"
By.CSS_SELECTOR	"css selector"

구글 검색창 선택 예제

- find_elements_by_?() 함수 사용 예제

```
from selenium import webdriver

driver = webdriver.Chrome()
driver.implicitly_wait(3)
driver.get("https://www.google.co.kr/")

driver.find_element(By.NAME, "q")
driver.find_element ('xpath', '//*[@id="tsf"]/div[2]/div/div[1]/div/div[1]/input')
driver.find_elements(By.CLASS_NAME, 'gLFyf')[0]
driver.find_elements("css selector", "div.vdLsw.gsfi+input")[0]
```

문자열 보내기

- send_keys()
  - 선택한 요소에 문자열을 보낸다.
- submit()
  - form을 제출한다.
  - form 안에 있는 아무 요소를 선택한 뒤에 실행할 수 있다.
- click()
  - 선택한 요소에 대해 클릭을 실행한다.
  - 주로 버튼 요소에 동작한다.
- clear()
  - input 태그 등의 내용을 지운다.