

Practical: Handling Dynamic Content with Selenium

SCS4209/IS4108/CS4113 Natural Language Processing
University of Colombo School of Computing

Objective

To practice web scraping from a dynamic webpage using Selenium WebDriver and understand the use of explicit waits to extract content that loads with JavaScript.

Instructions

1. Setup Selenium and WebDriver on your system.
2. Write and run the given Python script.
3. Capture a screenshot of the output.
4. Submit your Python file and the screenshot.


Task: Extract Dynamic Quote from `http://quotes.toscrape.com/js/`

This website loads quotes dynamically with JavaScript. Your task is to extract the first quote and its author using Selenium.

Sample Python Code

```
from selenium import webdriver
from selenium.webdriver.chrome.service import Service
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
import time

# Set up driver (Update path to your own chromedriver)
driver_path = "chromedriver"
service = Service(driver_path)
driver = webdriver.Chrome(service=service)
```

 chromedriver acts as a bridge between Selenium and the Chrome browser.

```

try:
    driver.get("http://quotes.toscrape.com/js/")

    # Wait until the first quote is loaded
    wait = WebDriverWait(driver, 10)
    quote = wait.until(EC.presence_of_element_located((By.
        ↪ CLASS_NAME, "text"))))
    author = driver.find_element(By.CLASS_NAME, "author")

    print("Quote:", quote.text)
    print("Author:", author.text) if all

    # Save screenshot
    driver.save_screenshot("output.png")

finally:
    driver.quit()

```

Specifically, it waits for an element like: `"Some quote..."`.

By.CLASS_NAME, "text" means: "look for a tag with class text".

Expected Output

The console should display a quote and author, e.g.:

Quote: "The world as we have created it is a process of our thinking. It cannot be changed without changing our thinking."
 Author: Albert Einstein

Submission Checklist

- Python script (.py file)
- Screenshot showing your output (output.png)

What Selenium Does (Step-by-Step simply):

1. Opens a real Chrome browser (you'll see it pop up).
2. Goes to the page you asked for.
3. Waits until JavaScript finishes running (usually a few seconds).
4. Reads the content after it's been loaded into the page.

Advanced Task (Optional)

Modify the script to:

- Extract and print all quotes on the first page.
- Save them to a text file.