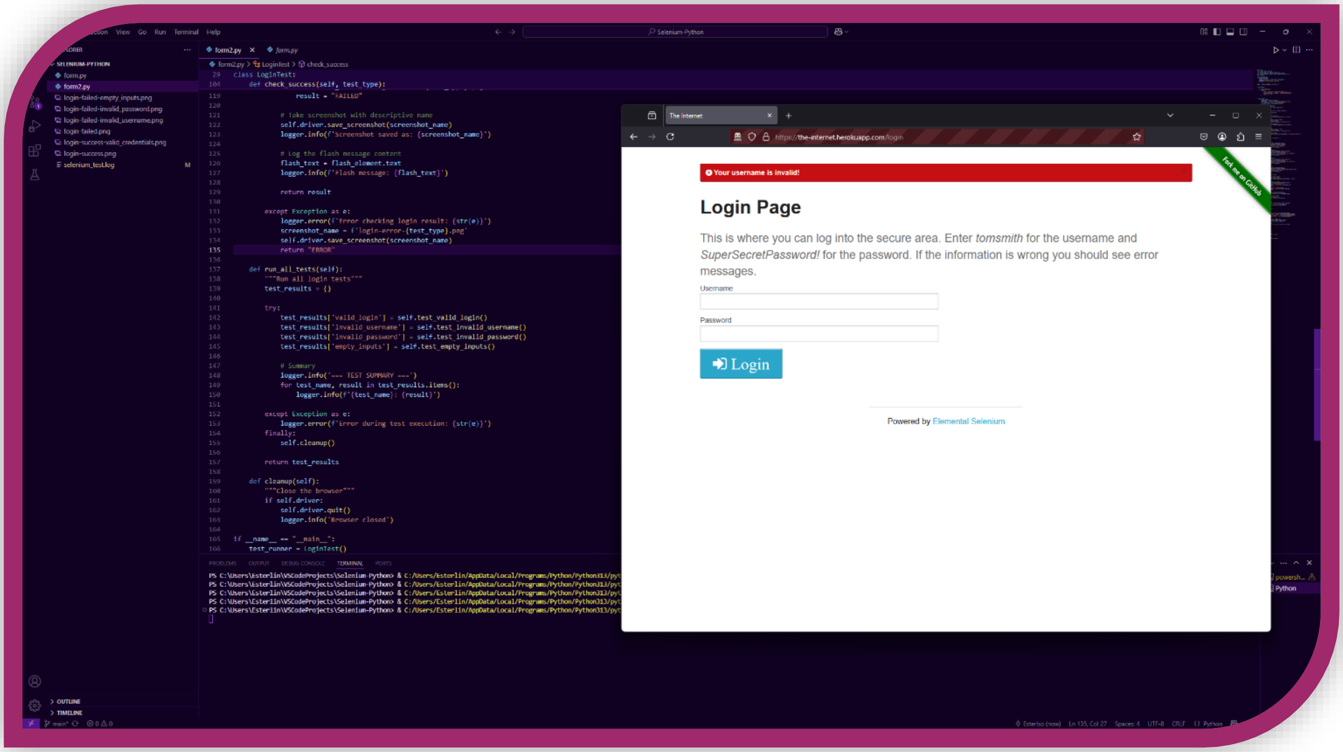


```
class LoginTest:
    def check_success(self, test_type):
                result = "FAILED"

            # Take screenshot with descriptive name
            self.driver.save_screenshot(screenshot_name)
            logger.info(f'Screenshot saved as: {screenshot_name}')

            # Log the flash message content
            flash_text = flash_element.text
            logger.info(f'Flash message: {flash_text}')

            return result

        except Exception as e:
            logger.error(f'Error checking login result: {str(e)}')
            screenshot_name = f'login-error-{test_type}.png'
            self.driver.save_screenshot(screenshot_name)
            return "ERROR"

    def run_all_tests(self):
        """Run all login tests"""
        test_results = {}

        try:
            test_results['valid_login'] = self.test_valid_login()
            test_results['invalid_username'] = self.test_invalid_username()
            test_results['invalid_password'] = self.test_invalid_password()
            test_results['empty_inputs'] = self.test_empty_inputs()

            # Summary
            logger.info('--- TEST SUMMARY ---')
            for test_name, result in test_results.items():
                logger.info(f'{test_name}: {result}')

        except Exception as e:
            logger.error(f'Error during test execution: {str(e)}')
        finally:
            self.cleanup()

        return test_results

    def cleanup(self):
        """Close the browser"""
        if self.driver:
            self.driver.quit()
            logger.info('Browser closed')

if __name__ == "__main__":
    test_runner = LoginTest()
```

```
PS C:\Users\Esterlin\VSCodeProjects\Selenium-Python> & C:/Users/Esterlin/AppData/Local/Programs/Python/Python313/pyt
PS C:\Users\Esterlin\VSCodeProjects\Selenium-Python> & C:/Users/Esterlin/AppData/Local/Programs/Python/Python313/pyt
PS C:\Users\Esterlin\VSCodeProjects\Selenium-Python> & C:/Users/Esterlin/AppData/Local/Programs/Python/Python313/pyt
PS C:\Users\Esterlin\VSCodeProjects\Selenium-Python> & C:/Users/Esterlin/AppData/Local/Programs/Python/Python313/pyt
PS C:\Users\Esterlin\VSCodeProjects\Selenium-Python> & C:/Users/Esterlin/AppData/Local/Programs/Python/Python313/pyt
```

https://the-internet.herokuapp.com/login

Your username is invalid!

Login Page

This is where you can log into the secure area. Enter *tomsmith* for the username and *SuperSecretPassword!* for the password. If the information is wrong you should see error messages.

Username

Password

Login

Powered by Elemental Selenium

Selenium Report

PYTHON AND FIREFOX

Esterlin Jerez Paulino | Introduction to Selenium | 5/16/25

Intro

Initially, I followed the instructor's steps using JavaScript and Chrome. For this project, I chose Python for its readability and Mozilla Firefox to familiarize myself with its features and advantages.

Python Setup

- Go to **python.org** and download current version of Python
- During installation, check **"Add Python to PATH"**
- Verify installation by running python --version in terminal/command prompt

Selenium Setup

- Open terminal/command prompt and run **pip install selenium**
- This installs the Selenium library for Python
- May be useful to create virtual environment to avoid any issues

Set up Firefox WebDriver (GeckoDriver)

- Download GeckoDriver from **GitHub** releases (Mozilla/ GeckoDriver)
- Extract the executable file
- Either add GeckoDriver to your system PATH or place it in your project folder

First Implementation

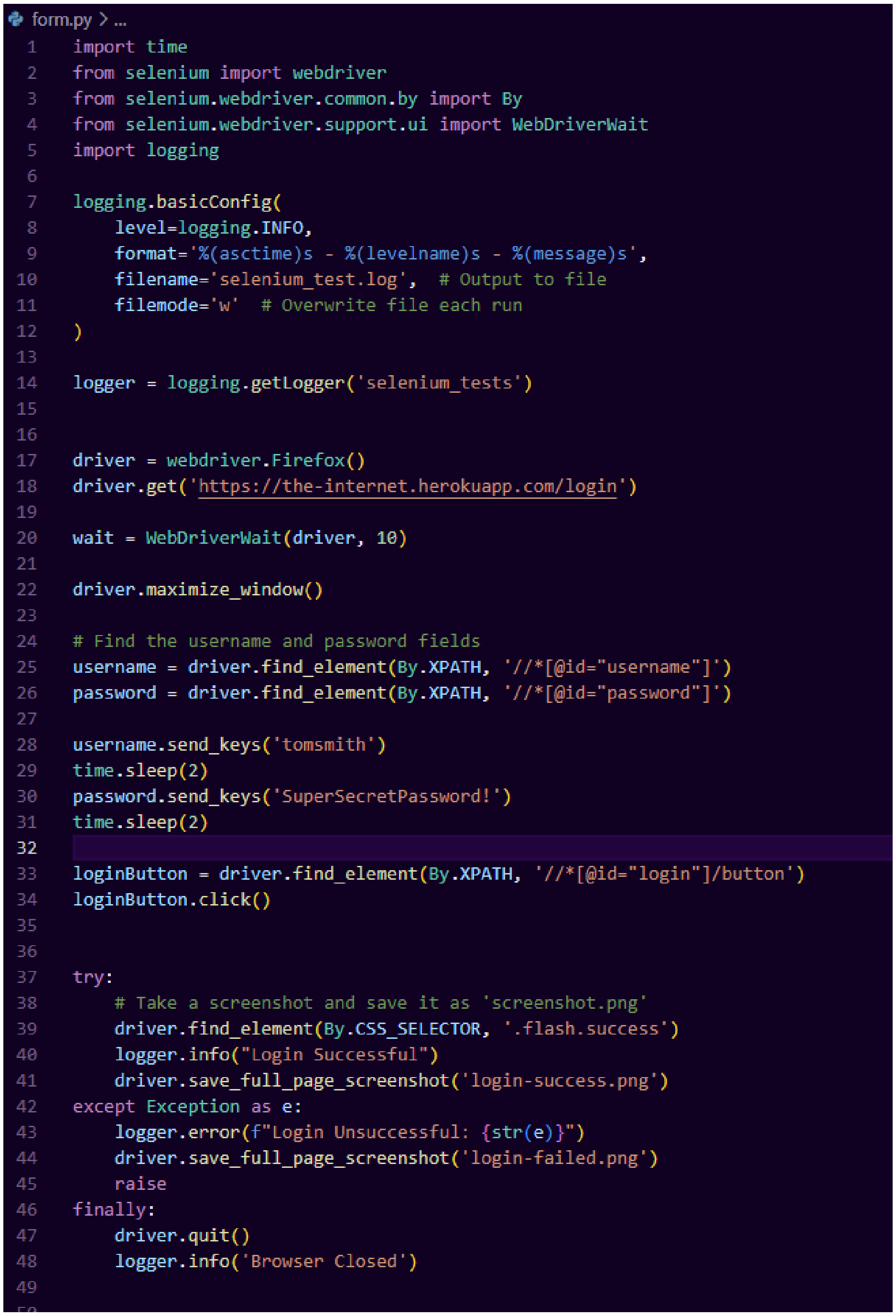
After setting up the environment with Python, Firefox WebDriver, and Selenium I tested that everything worked with a simple script like the example on the right.

```
from selenium import webdriver

driver = webdriver.Firefox()

driver.get("https://www.google.com")

driver.quit()
```

Following this, I utilized the previous code from the tutorials as a reference for my next steps. I engaged with online forums and occasionally employed artificial intelligence to learn how to convert some of the code from JavaScript to Python. To maintain a productive learning environment, I initially relied on online resources for learning and attempted implementation based on them. When encountering difficulties, I resorted to AI assistance only if the issues were particularly challenging.

During the first implementation I had an easy experience converting to Python especially due to how much easier it was due to the syntax. The code seemed to make more sense to me especially due to the readability so for future projects I may stick to Python but also keep JavaScript when needed.

IMAGE OF CODE



```python
import time
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
import logging

logging.basicConfig(
    level=logging.INFO,
    format='%(asctime)s - %(levelname)s - %(message)s',
    filename='selenium_test.log',  # Output to file
    filemode='w'  # Overwrite file each run
)

logger = logging.getLogger('selenium_tests')


driver = webdriver.Firefox()
driver.get('https://the-internet.herokuapp.com/login')

wait = WebDriverWait(driver, 10)

driver.maximize_window()

# Find the username and password fields
username = driver.find_element(By.XPATH, '//*[@id="username"]')
password = driver.find_element(By.XPATH, '//*[@id="password"]')

username.send_keys('tomsmith')
time.sleep(2)
password.send_keys('SuperSecretPassword!')
time.sleep(2)

loginButton = driver.find_element(By.XPATH, '//*[@id="login"]/button')
loginButton.click()


try:
    # Take a screenshot and save it as 'screenshot.png'
    driver.find_element(By.CSS_SELECTOR, '.flash.success')
    logger.info("Login Successful")
    driver.save_full_page_screenshot('login-success.png')
except Exception as e:
    logger.error(f"Login Unsuccessful: {str(e)}")
    driver.save_full_page_screenshot('login-failed.png')
    raise
finally:
    driver.quit()
    logger.info('Browser Closed')
```

Second Implementation

In my second implementation, I aimed to improve the code and align it with industry standards. While my code is not perfect, it reflects my quick understanding of these standards. I watched videos and researched business expectations, focusing on script development, project structure, and environment setup.

OBSTACLES ALONG THE WAY

Creating classes and functions in Python helped maintain structure but made it challenging to connect components and manage variables. Despite being easy to learn, I needed a method to log test steps and save them to a file. Logging issues occurred when errors arose from improper setup or attempting to find elements prematurely. I was also coming into issues running all tests at once due to improper setup, so I needed to watch a video to assist me.

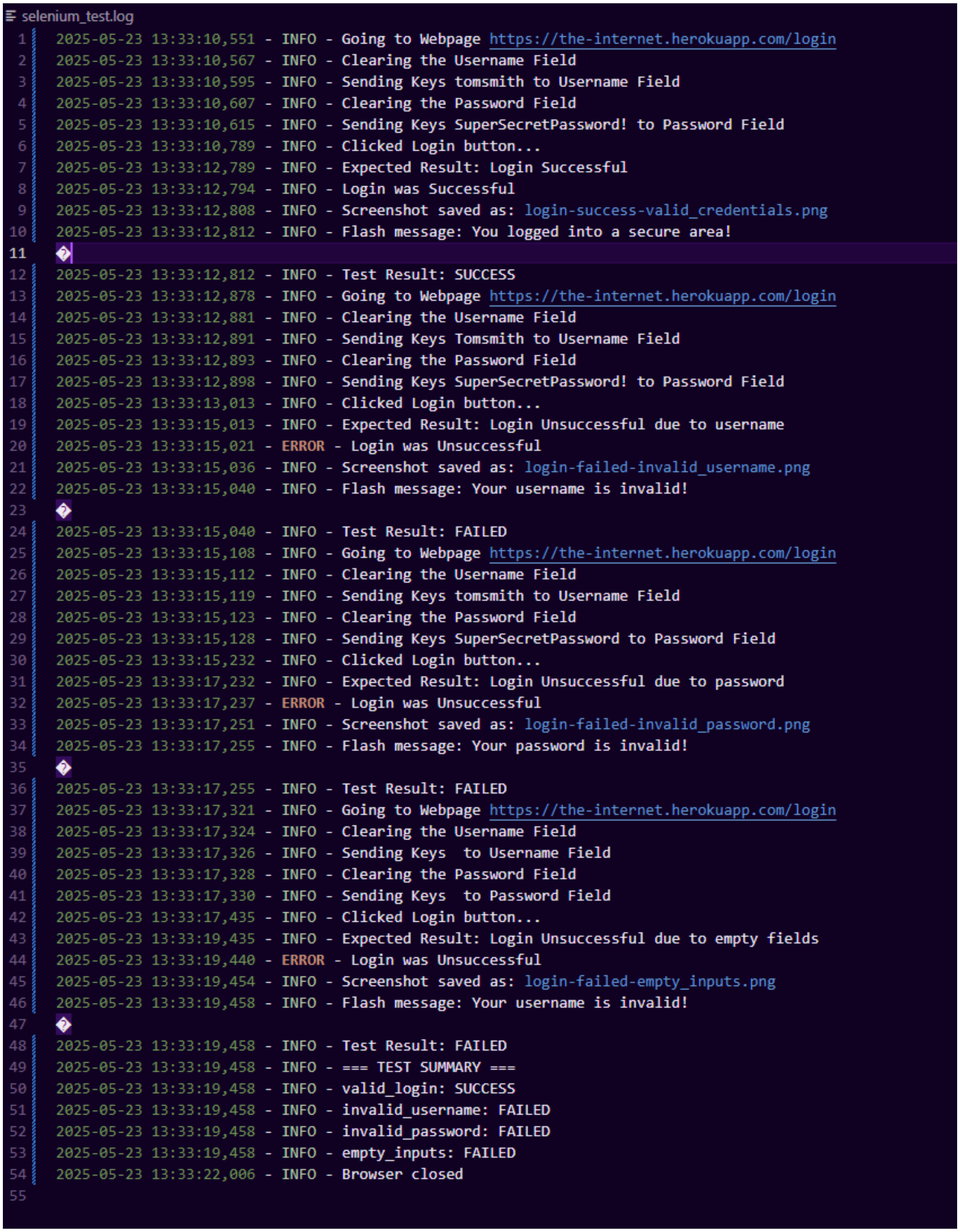
DIFFERENCE BETWEEN ORIGINAL AND NEW

The main difference between the original implementation and the new one is that the original tests one possibility at a time, whereas the new implementation runs multiple tests simultaneously to improve efficiency and save time. Another difference is how it is setup; I have multiple classes and functions in Python distributing the actions and making it easier to read.

The test being conducted on the new approach extends beyond simply verifying correct credentials with an incorrect password. It also includes scenarios such as incorrect usernames and empty input fields, which are additional instances of invalid logins. The code captures screenshots for both successful and unsuccessful login attempts, naming each screenshot according to the specific test case. The logging mechanism rigorously monitors nearly all actions, guaranteeing the proper execution of tests. In the event of an error at any stage, this comprehensive logging allows pinpointing the exact location of the issue(s).

GITHUB CONTAINING CODE AND OUTPUTS

IMAGE OF LOGS



```
2025-05-23 13:33:10,551 - INFO - Going to Webpage https://the-internet.herokuapp.com/login
2025-05-23 13:33:10,567 - INFO - Clearing the Username Field
2025-05-23 13:33:10,595 - INFO - Sending Keys tomsmith to Username Field
2025-05-23 13:33:10,607 - INFO - Clearing the Password Field
2025-05-23 13:33:10,615 - INFO - Sending Keys SuperSecretPassword! to Password Field
2025-05-23 13:33:10,789 - INFO - Clicked Login button...
2025-05-23 13:33:12,789 - INFO - Expected Result: Login Successful
2025-05-23 13:33:12,794 - INFO - Login was Successful
2025-05-23 13:33:12,808 - INFO - Screenshot saved as: login-success-valid_credentials.png
2025-05-23 13:33:12,812 - INFO - Flash message: You logged into a secure area!
�
2025-05-23 13:33:12,812 - INFO - Test Result: SUCCESS
2025-05-23 13:33:12,878 - INFO - Going to Webpage https://the-internet.herokuapp.com/login
2025-05-23 13:33:12,881 - INFO - Clearing the Username Field
2025-05-23 13:33:12,891 - INFO - Sending Keys Tomsmith to Username Field
2025-05-23 13:33:12,893 - INFO - Clearing the Password Field
2025-05-23 13:33:12,898 - INFO - Sending Keys SuperSecretPassword! to Password Field
2025-05-23 13:33:13,013 - INFO - Clicked Login button...
2025-05-23 13:33:15,013 - INFO - Expected Result: Login Unsuccessful due to username
2025-05-23 13:33:15,021 - ERROR - Login was Unsuccessful
2025-05-23 13:33:15,036 - INFO - Screenshot saved as: login-failed-invalid_username.png
2025-05-23 13:33:15,040 - INFO - Flash message: Your username is invalid!
�
2025-05-23 13:33:15,040 - INFO - Test Result: FAILED
2025-05-23 13:33:15,108 - INFO - Going to Webpage https://the-internet.herokuapp.com/login
2025-05-23 13:33:15,112 - INFO - Clearing the Username Field
2025-05-23 13:33:15,119 - INFO - Sending Keys tomsmith to Username Field
2025-05-23 13:33:15,123 - INFO - Clearing the Password Field
2025-05-23 13:33:15,128 - INFO - Sending Keys SuperSecretPassword to Password Field
2025-05-23 13:33:15,232 - INFO - Clicked Login button...
2025-05-23 13:33:17,232 - INFO - Expected Result: Login Unsuccessful due to password
2025-05-23 13:33:17,237 - ERROR - Login was Unsuccessful
2025-05-23 13:33:17,251 - INFO - Screenshot saved as: login-failed-invalid_password.png
2025-05-23 13:33:17,255 - INFO - Flash message: Your password is invalid!
�
2025-05-23 13:33:17,255 - INFO - Test Result: FAILED
2025-05-23 13:33:17,321 - INFO - Going to Webpage https://the-internet.herokuapp.com/login
2025-05-23 13:33:17,324 - INFO - Clearing the Username Field
2025-05-23 13:33:17,326 - INFO - Sending Keys  to Username Field
2025-05-23 13:33:17,328 - INFO - Clearing the Password Field
2025-05-23 13:33:17,330 - INFO - Sending Keys  to Password Field
2025-05-23 13:33:17,435 - INFO - Clicked Login button...
2025-05-23 13:33:19,435 - INFO - Expected Result: Login Unsuccessful due to empty fields
2025-05-23 13:33:19,440 - ERROR - Login was Unsuccessful
2025-05-23 13:33:19,454 - INFO - Screenshot saved as: login-failed-empty_inputs.png
2025-05-23 13:33:19,458 - INFO - Flash message: Your username is invalid!
�
2025-05-23 13:33:19,458 - INFO - Test Result: FAILED
2025-05-23 13:33:19,458 - INFO - === TEST SUMMARY ===
2025-05-23 13:33:19,458 - INFO - valid_login: SUCCESS
2025-05-23 13:33:19,458 - INFO - invalid_username: FAILED
2025-05-23 13:33:19,458 - INFO - invalid_password: FAILED
2025-05-23 13:33:19,458 - INFO - empty_inputs: FAILED
2025-05-23 13:33:22,006 - INFO - Browser closed
```