

TUTORIAL 9

9.Perform website testing using Selenium for multiple pages and functionalities.

Objective:

To perform **automated testing on a multi-page website** using **Selenium in Python**, covering functionalities like:

1. Login
2. Navigate to Account Overview
3. Transfer Funds
4. Logout

Tools & Setup

Prerequisites:

- Python installed (python.org)
- Install Selenium:

pip install selenium

- Download ChromeDriver:
<https://sites.google.com/chromium.org/driver>

Test Website:

<http://parabank.parasoft.com/parabank/index.htm>

Step-by-Step Selenium Testing Script

Step 1: Import and Setup

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

# Launch browser
driver = webdriver.Chrome()
driver.maximize_window()
driver.get("http://parabank.parasoft.com/parabank/index.htm")
```

Step 2: Login Test

```
# Login with valid credentials
driver.find_element(By.NAME, "username").send_keys("john") # valid demo user
driver.find_element(By.NAME, "password").send_keys("demo")
driver.find_element(By.XPATH, "//input[@value='Log In']").click()

# Wait for page to load
time.sleep(2)

# Verify Login
if "Accounts Overview" in driver.page_source:
    print("✅ Login Successful")
else:
    print("❌ Login Failed")
```

Step 3: Navigate to Account Overview Page

```
# Click on 'Accounts Overview'
driver.find_element(By.LINK_TEXT, "Accounts Overview").click()
time.sleep(2)

# Verify Account List
if "Account Number" in driver.page_source:
    print("✅ Account Overview Loaded Successfully")
else:
    print("❌ Account Overview Failed")
```

Step 4: Transfer Funds Functionality

```
# Navigate to Transfer Funds
driver.find_element(By.LINK_TEXT, "Transfer Funds").click()
time.sleep(1)

# Input transfer details
driver.find_element(By.ID, "amount").send_keys("100")
driver.find_element(By.ID, "fromAccountId").send_keys("12345") # Use actual account ID if needed
driver.find_element(By.ID, "toAccountId").send_keys("12346") # Use actual account ID if needed
driver.find_element(By.XPATH, "//input[@value='Transfer']").click()
time.sleep(2)

# Verify transfer success
if "Transfer Complete!" in driver.page_source:
    print("✅ Fund Transfer Successful")
else:
    print("❌ Fund Transfer Failed")
```

Step 5: Logout Test

```
# Logout
driver.find_element(By.LINK_TEXT, "Log Out").click()
time.sleep(2)

# Verify logout
if "Customer Login" in driver.page_source:
    print("✅ Logout Successful")
else:
    print("❌ Logout Failed")
```

Step 6: Close Browser

```
driver.quit()
```

Output Summary Example

- ✅ Login Successful
- ✅ Account Overview Loaded Successfully
- ✅ Fund Transfer Successful
- ✅ Logout Successful

Test Report Format (for Students)

Test Case ID	Functionality	Steps Performed	Expected Result	Actual Result	Status
TC_01	Login	Enter valid credentials	Dashboard loads	Dashboard loaded	Pass
TC_02	Account Overview	Click on Account Overview	Account list shown	Account list shown	Pass
TC_03	Transfer Funds	Enter details and transfer	Transfer success message	Message shown	Pass
TC_04	Logout	Click logout	Login page appears	Login page shown	Pass

Assignment for Students

Task	Description
1	Automate login, transfer, and logout
2	Add invalid login test
3	Take screenshots on failure
4	Store test report in Excel
5	Use unittest or pytest to automate multiple tests

Optional Enhancements

Take Screenshot on Failure

```
driver.save_screenshot("failure.png")
```

Add Assertion (using Python assert)

```
assert "Accounts Overview" in driver.page_source
```

