**Lab 4: Automated Form Submissions**

Objective:*Automate the process of submitting web forms using Selenium.*

Tasks:

1. Write a Selenium script to fill and submit a sample web form.

2. Store submission results or responses on AWS RDS.

Documentation:

- Automating form inputs with Selenium.

- Introduction to AWS RDS.

- Storing results on RDS.

Prerequisites:

1- An AWS account with administrative access.

2- Python Automation Course

3- Python Selenium Course

4- Bash Script Deep Dive Course

5- Previous Lab completed

Implementation Documentation:

**Automating Form Inputs with Selenium**

Selenium can be used to interact with web forms by locating form elements, filling them out, and submitting them. This is particularly useful for automating tasks like registration or data entry.

**Introduction to AWS RDS**

Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while managing time-consuming database administration tasks.

**Step 1: Setting Up an AWS RDS Instance**

Log in to your AWS account.

Go to the RDS dashboard.

Click on "Create database".

Choose a database engine, configuration, and instance details.

Set up the master username and password.

Configure advanced settings (optional).

You can configure RDS like following

*standard*

*mysql*

*free tier*

*my-test-db*

*admin*

*password*

*choose existing 21*

*my\_test\_db*

Review and create the RDS instance.

**Step 2: Writing the Selenium Script in PyCharm for form submission and saving in MySQL RDS instance.**

Make sure to install webdriver manager in PyCharm

| # Import necessary modules from selenium import webdriver from selenium.webdriver.chrome.service import Service as ChromeService from webdriver\_manager.chrome import ChromeDriverManager from selenium.webdriver.common.by import By import time import pymysql  # Set up Selenium WebDriver driver = webdriver.Chrome()  # Navigate to the target website driver.get("https://practicetestautomation.com/practice-test-login/")  # Wait for page elements to load (in this case, waiting for 5 seconds) time.sleep(5)  # Find and input username input\_name = driver.find\_element(By.ID, "username") input\_name.send\_keys("student")  # Find and input password input\_password = driver.find\_element(By.ID, "password") input\_password.send\_keys("Password123")  # Find and click the submit button submit\_button = driver.find\_element(By.ID, "submit") submit\_button.click()  # Wait for a short period (3 seconds) to allow the page to load time.sleep(3)  # Close the browser driver.quit()  # Set up connection to MySQL database conn = pymysql.connect(  host='my-test-db.c3ogfbfieeet.us-east-2.rds.amazonaws.com',  user='admin',  password='password',  database='my\_test\_db' )  cursor = conn.cursor()  # Create the table if it doesn't already exist create\_table\_query = """  CREATE TABLE IF NOT EXISTS form\_submissions (  id INT AUTO\_INCREMENT PRIMARY KEY,  username VARCHAR(255),  password VARCHAR(255)  ) """ cursor.execute(create\_table\_query)  # Define SQL query for inserting data into the table sql = "INSERT INTO form\_submissions (username, password) VALUES (%s, %s)" values = ("student", "Password123")  # Execute the SQL query with specified values cursor.execute(sql, values)  # Commit the changes to the database conn.commit()  # Close the database connection conn.close() |
| --- |

**Explanation:**

- The code begins by importing necessary modules for Selenium, ChromeDriver, and MySQL interaction.

- It sets up a Selenium WebDriver using the Chrome browser.

- It navigates to a specific webpage.

- After waiting for 5 seconds to allow page elements to load, it finds and inputs a username and password.

- It then clicks the submit button.

- After a 3-second delay (to allow the page to load), it closes the browser.

- Next, it establishes a connection to a MySQL database hosted on AWS RDS.

- It creates a table named `form\_submissions` if it doesn't already exist.

- It defines an SQL query for inserting data into the table and executes it with specified values.

- The changes are committed to the database, and the connection is closed.

Remember to customize the database credentials, URLs, and table names according to your specific setup. Additionally, ensure that the necessary modules are installed in your Python environment.

**Confirmation**

To confirm the data is placed in mysql RDS instance, open terminal in PyCharm and use following commands.



| sudo apt-get install mysql-client mysql -h my-test-db.c3ogfbfieeet.us-east-2.rds.amazonaws.com -u admin -p my\_test\_db #Now enter the password you have configured previously.  #to display the data stored in table SELECT \* FROM Table\_Name |
| --- |

**Conclusion**

By following these steps, you have successfully automated the process of submitting web forms using Selenium and stored the submission results on Amazon RDS. This lab demonstrates how to combine web automation with database operations.