

Session 3.2

Introduction to Script based Testing

AN INITIATIVE BY

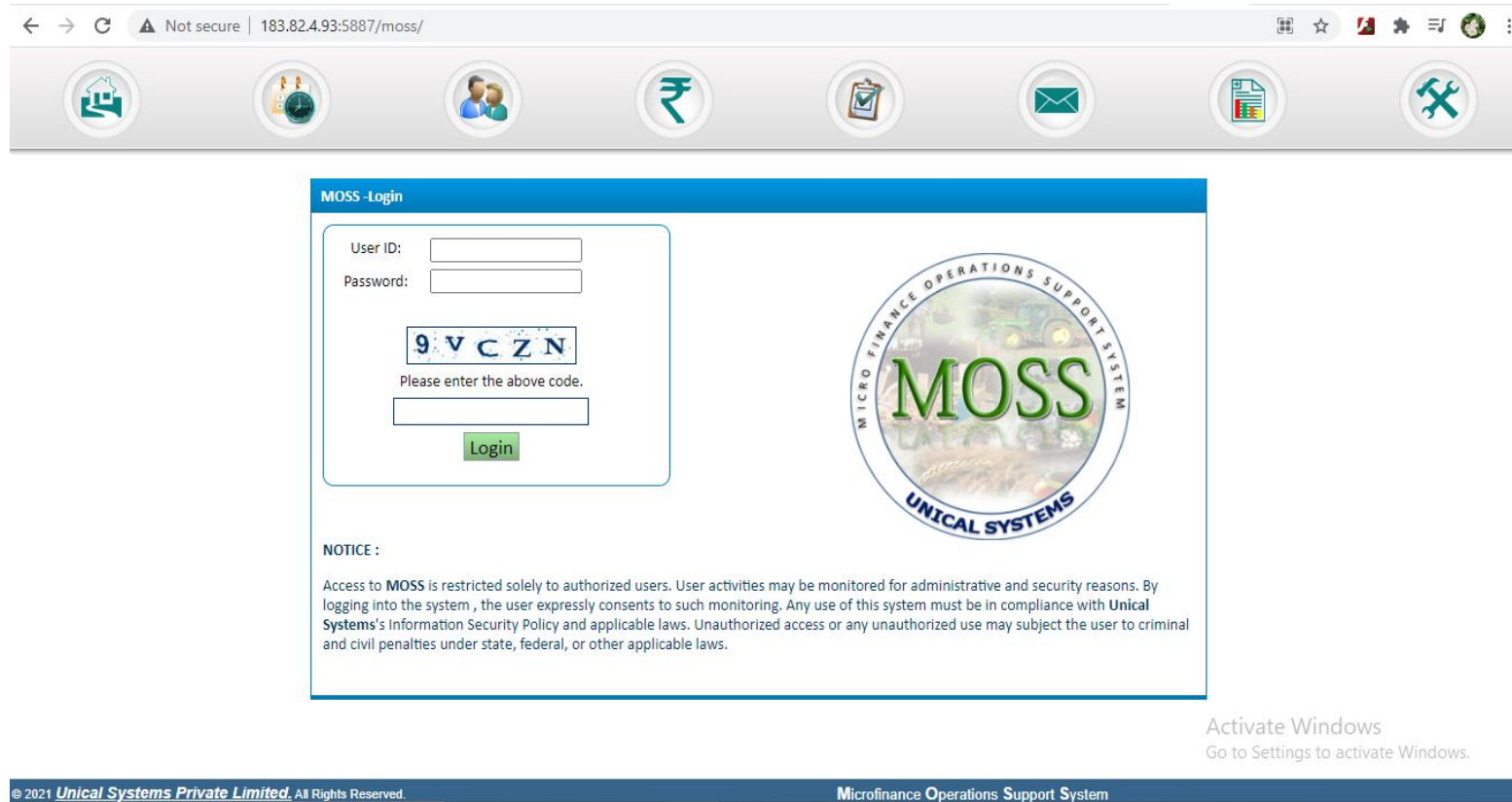
UNICAL ACADEMY



Let's go!!!



Here, we demonstrate how to automate the login screen (without captcha) using java scripting in Selenium webdriver. As a first step, we need to enter URL <http://183.82.4.93:5887/moss/> in any browser to reach the landing page of the Loan Management System web application



Steps in detail to perform automation testing of "login screen "using Selenium with Java:

Step 1: Create A Selenium WebDriver Instance

- Launch the application in desired Browser (Chrome/ Firefox/IE/safari)
`Webdriver driver=new ChromeDriver();`
- Set the system properties to the path of the driver for the required browser. We will use Chromedriver for demonstrating login screen with Java
`System.setProperty("webdriver.chrome.driver", "File path for the Exe");`

Step 2: Navigate To The Required URL

- Configure your Browser if required
`driver.manage().window().maximize();`
- Open the browser with the URL
`driver.get("http://183.82.4.93:5887/moss/");`

Step 3: Locate The HTML Element

- Use ID locator/ Name locator/ TagName locator/ CSS selector/ Xpath locator to find the HTML element
- **DOM structure for the UserID input box of login screen of Loan Management System:**
- locate it via ID locator in Selenium WebDriver as below:
`driver.findElement(By.id("username"));`
`WebElement username=driver.findElement(By.id("username"));`
`driver.findElement(By.id("password"));`
`WebElement password=driver.findElement(By.id("password"));`
`driver.findElement(By.xpath("//button[text()='Sign in']"));`
`WebElement login= driver.findElement(By.xpath("//button[text()='Sign in']"));`

Steps in detail to perform automation testing of “login screen “using Selenium with Java:

Step 4: Perform Action On The Located HTML Element

- Perform the sending text action to user ID and password fields and click on the login button using sendKeys and click methods.

```
username.sendKeys("0001");  
password.sendKeys("1234567");  
login.click();
```

Step 5: Verify & Validate The Action

- Validate the results using assertion method.

```
Assert.assertEquals(String actual, String expected);
```

- Save actual URL post login into a string value which is String

```
actualUrl="http://183.82.4.93:5887/moss/HierarchyLocationChange";
```

- Expected URL can be found from the below method

```
String expectedUrl= driver.getCurrentUrl();
```

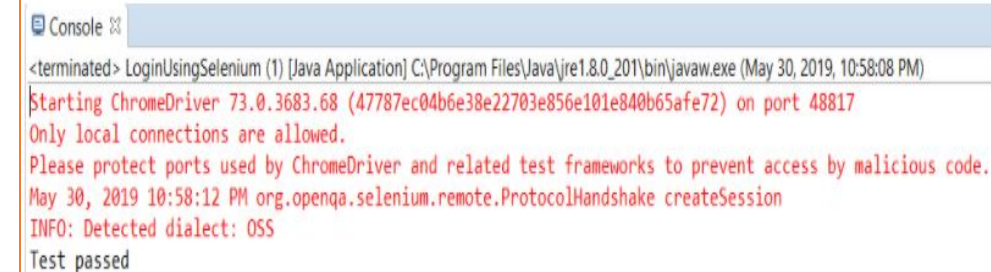
- Final assertion would become as

```
Assert.assertEquals(actualUrl, expectedUrl);
```

Executed automation testing of "login screen "using below Java Script:

```
import java.util.concurrent.TimeUnit;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;
public class LoginUsingSelenium {
    @Test public static void main(String[] args) { // TODO Auto-generated method
stub System.setProperty("webdriver.chrome.driver", " path of driver ");
WebDriver driver=new ChromeDriver();
driver.manage().window().maximize();
driver.get("https://183.82.4.93:5887/moss/");
WebElement username=driver.findElement(By.id("username"));
WebElement password=driver.findElement(By.id("password"));
WebElement login=driver.findElement(By.xpath("//button[text()='Sign in']"));
username.sendKeys("0001");
password.sendKeys("1234567");
login.click();
String actualUrl="http://183.82.4.93:5887/moss/HierarchyLocationChange/";
String expectedUrl= driver.getCurrentUrl();
if(actualUrl.equalsIgnoreCase(expectedUrl))
{ System.out.println("Test passed");}
else { System.out.println("Test failed"); } }
```

<Output should be Loan Management system login screen output >

Console Output:

The screenshot shows a console window titled 'Console' with a list of log messages. The messages include the termination of the 'LoginUsingSelenium' application, the successful start of 'ChromeDriver' on port 48817, a warning about local connections, a security warning about port protection, session creation details for May 30, 2019, and a final 'Test passed' message.

```
<terminated> LoginUsingSelenium (1) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (May 30, 2019, 10:58:08 PM)
Starting ChromeDriver 73.0.3683.68 (47787ec04b6e38e22703e856e101e840b65afe72) on port 48817
Only local connections are allowed.
Please protect ports used by ChromeDriver and related test frameworks to prevent access by malicious code.
May 30, 2019 10:58:12 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: OSS
Test passed
```

Session Recap

