

## 2.1 In this challenge, we validate the error message for Forgot Your Password?

## 2.2 Code with Comments

### LoginPage

```
package com.ibm.magentopages;

import java.util.List;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindAll;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.How;
import org.openqa.selenium.support.PageFactory;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;

public class LoginPage {

    //Intializing webelement using page factory
    @FindBy(id="email")
    WebElement emailEle;

    @FindBy(how=How.ID,using="pass")
    WebElement passEle;

    @FindBy(how=How.XPATH,using="//button[@type='submit']")
    WebElement loginEle;

    @FindBy(how=How.LINK_TEXT,using="Forgot Your Password?")WebElement fPwd;
    By errorMsgLoc =By.className("error-msg");

    //Intializing driver at global
    WebDriverWait wait;
```

```
WebDriver driver;
```

```
public LoginPage(WebDriver driver,WebDriverWait wait) {  
    PageFactory.initElements(driver, this);  
    this.driver=driver;  
    this.wait=wait;  
}
```

```
//Creating method for each action
```

```
public LoginPage enterEmailAddress(String userName)  
{  
    emailEle.sendKeys(userName);  
    return this;  
}
```

```
public LoginPage enterPassword(String password)  
{  
    passEle.sendKeys(password);  
    return this;  
}
```

```
public void clickOnLogin()  
{  
    loginEle.click();  
}
```

```
public ForgetPwdPage clickForgetPwd()  
{  
    fPwd.click();  
    return new ForgetPwdPage(driver, wait);  
}
```

```
public String getPageSourceForInvalidErrorMessage()  
{  
    wait.until(ExpectedConditions.presenceOfElementLocated(errorMsgLoc));
```

```

        return driver.getPageSource();
    }
}

```

## ForgetPwd Page

```

package com.ibm.magentopages;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.How;
import org.openqa.selenium.support.PageFactory;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;

public class ForgetPwdPage {
    WebDriverWait wait;
    WebDriver driver;

    By errorMsg= By.className("validation-advice");
    By validEmailAddr=By.className("validation-advice");

    public ForgetPwdPage(WebDriver driver, WebDriverWait wait)
    {
        PageFactory.initElements(driver, this);
        this.driver=driver;
        this.wait=wait;
    }
    //Intializing webelement using page factory
    @FindBy(how=How.XPATH,using="//span[text()='Submit']")WebElement submit;
    @FindBy(how=How.NAME,using="email")WebElement email;

    //Creating method for each action
    public ForgetPwdPage clickSubmitButton()
    {
        submit.click();
        return this;
    }
    public String getPageSourceVerifyErrorMessage()
    {
        wait.until(ExpectedConditions.presenceOfElementLocated(errorMsg));

        return driver.getPageSource();
    }
    public void enterEmail(String uname)
    {
        email.sendKeys(uname);
    }
    public String getPageSourceValidEmailAddr()
    {
        wait.until(ExpectedConditions.presenceOfElementLocated(validEmailAddr));

        return driver.getPageSource();
    }
}

```

## MagentoTest

```

package com.ibm.magentotest;

import java.io.FileInputStream;
import java.io.FileNotFoundException;

```

```

import java.io.FileOutputStream;
import java.io.IOException;
import java.util.HashMap;
import java.util.Properties;
import java.util.Set;
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.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.testng.Assert;
import org.testng.Reporter;
import org.testng.annotations.AfterMethod;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeMethod;
import org.testng.annotations.BeforeSuite;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;

import com.ibm.magentopages.ForgetPwdPage;
import com.ibm.magentopages.HomePage;
import com.ibm.magentopages.LoginPage;
import com.ibm.magentopages.MyAccountPage;
import com.ibm.utilities.PropertiesFileHandler;

import jdk.nashorn.internal.runtime.PrototypeObject;

public class MagentoTest {
    WebDriver driver;
    WebDriverWait wait;
    PropertiesFileHandler propFileHandler;
    HashMap<String, String> data;

    @BeforeSuite
    public void preSetForTest() throws IOException {
        String file = "./TestData/magentodata.properties";
        propFileHandler = new PropertiesFileHandler();
        data = propFileHandler.getPropertiesAsMap(file);
    }

    @BeforeMethod
    public void Initialization() {
        System.setProperty("webdriver.chrome.driver", "./drivers/chromedriver.exe");
        driver = new ChromeDriver();
        wait = new WebDriverWait(driver, 60);
        driver.manage().window().maximize();
        driver.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);
    }

    @AfterMethod
    public void closeBrowser() {
        driver.quit();
    }

    /* @Test(priority = 1, testName = "CheckValidCredentials", timeOut = 5000)
    public void positiveCredential() throws IOException {

        String url = data.get("url");
        String userName = data.get("username");
        String password = data.get("password");
        String expectedTitle = data.get("expectedtitle");
        driver.get(url);
    }
    */

```

```

HomePage home = new HomePage(driver);
home.clickOnMyAccountIcon();

LoginPage login = new LoginPage(driver, wait);
login.enterEmailAddress(userName);
login.enterPassword(password);
login.clickOnLogin();

MyAccountPage myAccount = new MyAccountPage(driver, wait);
String actualTitle = myAccount.getCurrentTitle();

Assert.assertEquals(actualTitle, expectedTitle);
myAccount.clickOnLogOut();
}*/

/**
 * @throws IOException
 */
@Test
public void negativeCredential() throws IOException {

    try {
        String url = data.get("url");
        String userName = data.get("invalidusername");
        String password = data.get("password");
        String expectedMessage = data.get("invaliderrormessage");
        String RequiredMessage = data.get("requiredfield");
        String emailAddr=data.get("forgotpwduname");
        driver.get(url);
        String validemailaddr=data.get("enterValidEmailAddress");

        HomePage home = new HomePage(driver);
        home.clickOnMyAccountIcon();

        LoginPage login = new LoginPage(driver, wait);
        login.enterEmailAddress(userName);
        login.enterPassword(password);
        login.clickOnLogin();
        String pageSource = login.getPageSourceForInvalidErrorMessage();

        Assert.assertTrue(pageSource.contains(expectedMessage), "Assertion on
Invalid Login Error Message");

        Reporter.log("Negataive Credentials Passed");

        login.clickForgetPwd();

        //Created a ForgetPwdPage creating object to call all the methods
        ForgetPwdPage forgetPwd=new ForgetPwdPage(driver, wait);
        forgetPwd.clickSubmitButton();

        //Validating error message for Field required email address
        String pageSrc = forgetPwd.getPageSourceVerifyErrorMessage();

        Assert.assertTrue(pageSrc.contains(RequiredMessage), "Assertion on required
field Message");

        Reporter.log("Field required email address");

        //Entering email address and clicking on submit button
        forgetPwd.enterEmail(emailAddr);
        forgetPwd.clickSubmitButton();

        //Validating email address
        String pageSour = forgetPwd.getPageSourceValidEmailAddr();

```

```

        Assert.assertTrue(pageSour.contains(validemailaddr), "Assertion on Valid
Email Address");

        Reporter.log("Enter Valid email address");

    }
    catch (Exception e) {
        // TODO: handle exception
        System.out.println(e.getMessage());
        Reporter.log("Field required email address");
        //screenshot
        //throw e;
        Assert.fail();
    }
}
}
}

```

## TestNg XML

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">
<suite name="Suite" parallel="tests" verbose="5">
    <test thread-count="5" name="Test">
        <classes>
            <class name="com.ibm.magentotest.MagentoTest">
            </class>
        </classes>
    </test> <!-- Test -->
    <!-- <test thread-count="5" name="Test1">
        <classes>
            <class name="com.ibm.magentotest.MagentoTest">
            </class>
        </classes>
    </test> Test -->
</suite> <!-- Suite -->

```

## 2.3 Explanation of the code

1. Created 2 Pages one for login and forgetPwd page using Page Object Model
2. Created method for each action using pagefactory initialized webelement
3. Called all the methods in magentoTest using object
4. Clicking on forgot password and Validated the error message
5. Passing the values in magentodataproperties file.
6. Converted the project into TestNg and in XML file implemented package name and class name and executed.

## 2.4 Output

The screenshot shows an IDE with several tabs: \*MagentoTest.java, HomePage.java, \*LoginPage.java, ForgetPwdPage.java, PropertiesFileHandler.java, testng.xml, and TestNG Report. The testng.xml file is open, showing the following content:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">
3 <suite name="Suite" parallel="tests" verbose="5">
4   <test thread-count="5" name="Test">
5     <classes>
6       <class name="com.ibm.magentotest.MagentoTest">
7       </class>
8     </classes>
9   </test> <!-- Test -->
10  <!-- <test thread-count="5" name="Test1">
11    <classes>
12      <class name="com.ibm.magentotest.MagentoTest">
13      </class>
14    </classes>
15  </test> -->
16 </suite>
```

Below the XML editor, the 'Results of running class MagentoTest' are displayed. A summary bar shows: Passed: 1, Failed: 0, Skipped: 0, Tests: 1/1, Methods: 1 (28048 ms). The 'All Tests' tab is selected, showing a tree view with 'Default suite (1/0/0/0) (22.529 s)' and 'Default test (22.529 s)'. The 'Failure Exception' column is empty. The 'Console' tab shows the following output:

```
<terminated> MagentoTest [TestNG] C:\Program Files\Java\jre1.8.0_191\bin\javaw.exe (07-Dec-2018, 3:39:28 PM)
[RemoteTestNG] detected TestNG version 6.14.3
Starting ChromeDriver 2.44.609538 (b655c5a60b0b544917107a59d4153d4bf78e1b90) on port 22407
Only local connections are allowed.
Dec 07, 2018 3:39:33 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: OSS
PASSED: negativeCredential
```

```
[RemoteTestNG] detected TestNG version 6.14.3
Starting ChromeDriver 2.44.609538 (b655c5a60b0b544917107a59d4153d4bf78e1b90) on
port 22407
Only local connections are allowed.
Dec 07, 2018 3:39:33 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: OSS
PASSED: negativeCredential
```

```
=====
Default test
Tests run: 1, Failures: 0, Skips: 0
=====
```

```
=====
Default suite
Total tests run: 1, Failures: 0, Skips: 0
=====
```

Test	# Passed	# Skipped	# Failed	Time (ms)	Included Groups	Excluded Groups
Default suite						
<a href="#">Default test</a>	1	0	0	27,851		

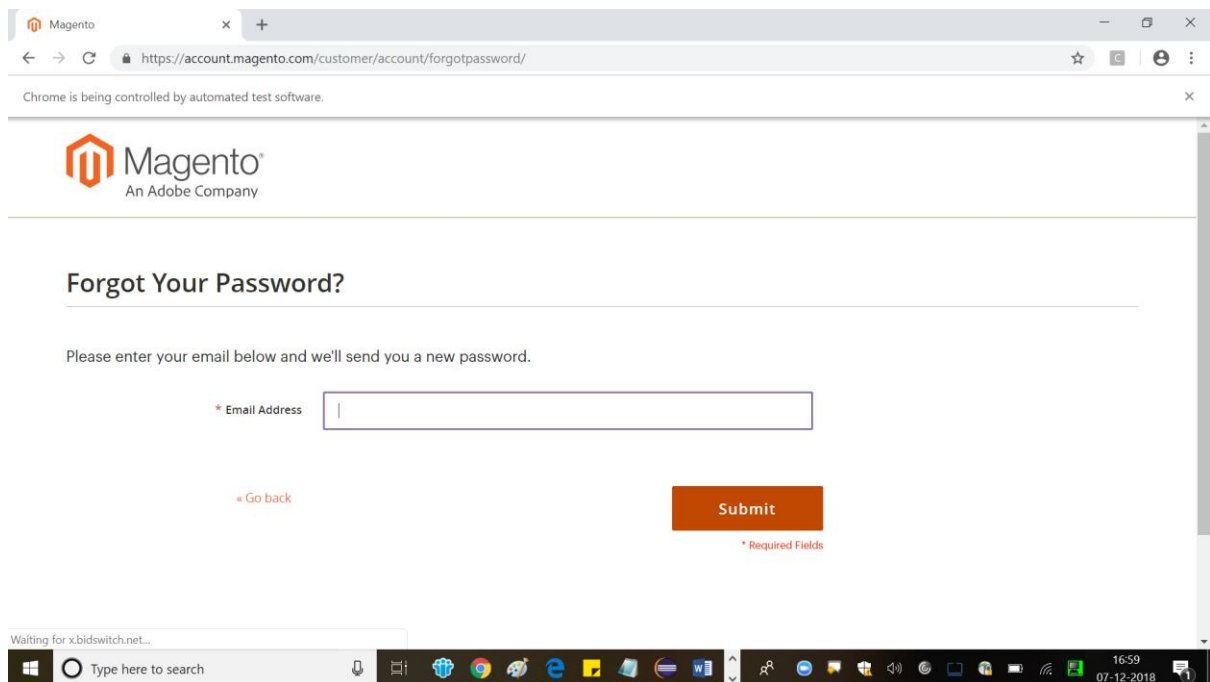
Class	Method	Start	Time (ms)
Default suite			
Default test — passed			
com.ibm.magentotest.MagentoTest	<a href="#">negativeCredential</a>	1544177374310	22529

## Default test

com.ibm.magentotest.MagentoTest#negativeCredential

Messages
Negataive Credentials Passed Field required email address Enter Valid email address

[back to summary](#)







Please enter your email below and we'll send you a new password.

\* Email Address

This is a required field.

« Go back

Submit

\* Required Fields



Please enter your email below and we'll send you a new password.

\* Email Address

Please enter a valid email address. For example johndoe@domain.com.

« Go back

Submit

\* Required Fields