

Hi,

PFB the test cases which needs to be done and submitted by the candidate in a single PDF file.

Process	Steps To Execute	Expected Results
Edit Shipping Pincode	1. Navigate to Systems section, shipping Option 2. Choose Shipping Pincode 3. Edit a new details and Save without giving pincode	1. Click the save icon and verify it is on the same page 2. Try to get the tooltip message "Please fill out this fields." and assert the message Negative test case validation
Edit Order Status	1. Navigate to Systems section, returns Option 2. Choose Order Status 3. Remove the new Order Status name and Save	1. Click the save icon and verify it is on the same page 2. Try to get the tooltip message "Please fill out this fields." and assert the message Negative test case validation

Solution:

Edited one java file as "ShippingPage.java".

```
package com.ibm.pages;

import org.junit.Assert;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;

public class ShippingPage {

    WebDriver driver;
    WebDriverWait wait;

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

    //WebElement for link System
    @FindBy(xpath="//a[contains(text(),'System')]")
    WebElement systemEle;

    //WebElement for link Shipping
    @FindBy(xpath="//a[contains(text(),'Shipping')])[1]")
    WebElement shipEle;
```

```

//WebElement for link Shipping Pincode
@FindBy(xpath="//a[contains(text(),'Shipping Pincode')]")
WebElement spicodeEle;

//WebElement for button Action
@FindBy(xpath="//button[contains(text(),'Action')][1]")
WebElement actionEle;

//WebElement for Edit
@FindBy(xpath="//a[@title='Edit']")
WebElement editEle;

//WebElement for field Pincode
@FindBy(xpath="//input[@name='pincode']")
WebElement pincodeEle;

//WebElement for icon Save
@FindBy(xpath="//button[@title='Save']")
WebElement saveEle;

//WebElement for Header text
@FindBy(xpath="//div[@class='panel-title']")
WebElement hmessageEle;

public void clickOnLinkSystem()
{
    systemEle.click();
}

public void clickOnLinkShipping()
{
    shipEle.click();
}

//
wait.until(ExpectedConditions.presenceOfElementLocated(By.xpath("//a[contains(
text(),'Shipping Pincode')])));
}

public void clickOnLinkShippingPincode()
{
    spicodeEle.click();
}

public void clickOnAction()
{
    actionEle.click();
}

public void clickOnEdit()
{
    editEle.click();
}

public void clearPincode()
{

```

```

        pincodeEle.clear();
    }

    public void clickOnIconSave()
    {
        saveEle.click();
    }

    public void getHeaderMessage()
    {
        String header = hmessageEle.getText();
        if (header.contains("Edit Shipping Pincode"))
        {
            System.out.println("The validation on Edit Shipping Pincode page
after clicking Save...");
            System.out.println("The header of this page: " +header);
            Assert.assertTrue(true);
        }
        else
        {
            Assert.fail();
        }
    }

    public String validationOnMissingField()
    {
        JavascriptExecutor js=(JavascriptExecutor) driver;
        String message = (String) js.executeScript("return
document.getElementsByName('pincode')[0].validationMessage;");
        System.out.println("The validation on error message without
entering data for field Pincode...");
        System.out.println("The error message as: " +message);
        return(message);
    }
}

```

Edited one java file as "ReturnPage.java".

```

package com.ibm.pages;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import java.io.IOException;

import java.util.Properties;

```

```
import org.junit.Assert;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.FindingBy;

import org.openqa.selenium.support.PageFactory;

import org.openqa.selenium.support.ui.WebDriverWait;


public class ReturnPage {

    WebDriver driver;

    WebDriverWait wait;


    public ReturnPage(WebDriver driver,WebDriverWait wait)

    {

        this.driver=driver;

        this.wait=wait;

        PageFactory.initElements(driver,this);

    }


    //WebElement for link System

    @FindBy(xpath="//a[contains(text(),'System')]")

    WebElement systemEle;
```

```
//WebElement for link Returns
```

```
@FindBy(xpath="//a[contains(text(),'Returns')][2]")
```

```
WebElement returnEle;
```

```
//WebElement for link Return Statuses
```

```
@FindBy(xpath="//a[contains(text(),'Return Statuses')]")
```

```
WebElement retstatusEle;
```

```
//WebElement for link Return Actions
```

```
@FindBy(xpath="//a[contains(text(),'Return Actions')]")
```

```
WebElement retactionEle;
```

```
//WebElement for button Action
```

```
@FindBy(xpath="//button[contains(text(),'Action')][1]")
```

```
WebElement actionEle;
```

```
//WebElement for Edit
```

```
@FindBy(xpath="//a[@title='Edit']")
```

```
WebElement editEle;
```

```
//WebElement for field Return Status Name
```

```
@FindBy(xpath="//input[@name='name']")
```

```
WebElement retstatusnameEle;
```

```
//WebElement for Header text
```

```
@FindBy(xpath="//div[@class='panel-title']")
```

```
WebElement hmessageEle;
```

```
//WebElement for field Return Action Name
```

```
@FindBy(xpath="//input[@name='name']")
```

```
WebElement retactnameEle;
```

```
//WebElement for icon Save
```

```
@FindBy(xpath="//button[@title='Save']")
```

```
WebElement saveEle;
```

```
//WebElement for link Home
```

```
@FindBy(xpath="//a[contains(text(),'Home')]")
```

```
WebElement homeEle;
```

```
public void clickOnLinkHome()
```

```
{
```

```
    homeEle.click();
```

```
}
```

```
public void clickOnLinkSystem()
```

```
{
```

```
    systemEle.click();
```

```
}
```

```
public void clickOnLinkReturns()
```

```
{
```

```
    returnEle.click();
```

```
}
```

```
public void clickOnLinkRetStatuses()
```

```
{
```

```
    retstatusEle.click();
```

```
}
```

```
public void clickOnLinkRetActions()
```

```
{
```

```
    retactionEle.click();
```

```
}
```

```
public void clickOnAction()
```

```
{
```

```
    actionEle.click();
```

```
}
```

```
public void clickOnEdit()
```

```
{
```

```
    editEle.click();
```

```
}
```

```
public void clearReturnStatus()
```

```
{
```

```
    retstatusnameEle.clear();
```

```
}
```

```
public void enterReturnActionName(String returnactionname)
```

```
{
```

```
    retactnameEle.clear();
```

```
    retactnameEle.sendKeys(returnactionname);
```

```
}
```

```
public void clickOnIconSave()
```

```
{
```

```
    saveEle.click();
```

```
}
```

```
public void getHeaderMessage()
```

```
{
```

```
    String header = hmessageEle.getText();
```

```
    if (header.contains("Edit Return Status"))
```

```
    {
```

```
        System.out.println("The validation on Edit Return Status page after clicking  
Save...");
```

```
        System.out.println("The header of this page: " +header);
```

```
        Assert.assertTrue(true);
```

```
    }
```



```

        else
        {
            Assert.fail();
        }
    }

    public String validationOnMisingField()
    {
        JavascriptExecutor js=(JavascriptExecutor) driver;

        String message = (String) js.executeScript("return
document.getElementsByName('name')[0].validationMessage;");

        System.out.println("The validation on error message without entering data for
field Return Status Name...");

        System.out.println("The error message as: " +message);

        return(message);
    }

    public void validationOnRecord() throws FileNotFoundException, IOException
    {
        Properties p = new Properties();

        p.load(new FileInputStream("./TestData/magentodata.properties"));

        String pagesource = driver.getPageSource();

        //System.out.println(pagesource);

```

```

        if(pagesource.contains(p.getProperty("returnactionname"))) {
            System.out.println("The presence of edited action name is confirmed!");
            Assert.assertTrue(true);
        }
        else {
            System.out.println("The action is not edited on the Return Actions
List");
            Assert.fail();
        }
    }
}

```

Edited java file "BaseTest.java"

```
package com.ibm.test;
```

```
import java.io.FileInputStream;
```

```
import java.io.IOException;
```

```
import java.sql.SQLException;
```

```
import java.util.HashMap;
```

```
import java.util.concurrent.TimeUnit;
```

```
import org.apache.poi.xssf.usermodel.XSSFCell;
```

```
import org.apache.poi.xssf.usermodel.XSSFRow;
```

```
import org.apache.poi.xssf.usermodel.XSSFSheet;
```

```
import org.apache.poi.xssf.usermodel.XSSFWorkbook;
```

```
import org.junit.Assert;
```

```
import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.support.ui.WebDriverWait;

import org.testng.annotations.Test;

import com.ibm.pages.AdminPage;

import com.ibm.pages.BannerPage;

import com.ibm.pages.CouponPage;

import com.ibm.pages.CustomerPage;

import com.ibm.pages.DB;

import com.ibm.pages.LoginPage;

import com.ibm.pages.ProductPage;

import com.ibm.pages.ReturnPage;

import com.ibm.pages.ShippingPage;

import com.ibm.pages.TabPage;

import com.ibm.pages.UserPage;

import com.ibm.utilities.DBUtil;

import com.ibm.utilities.PropertiesFileHandler;
```

```
public class BaseTest {
```

```
    @Test
    public void ExtraTestcase() throws IOException, InterruptedException
    {

        String file = "./TestData/magentodata.properties";

        PropertiesFileHandler propFileHandler = new PropertiesFileHandler();
        HashMap<String, String> data = propFileHandler.getPropertiesAsMap(file);

        String url = data.get("url");
        String email = data.get("email");
```

```

        String password = data.get("password");
        String expectedpincodeessage = data.get("expectedpincodeessage");

        System.setProperty("webdriver.chrome.driver",
"./drivers/chromedriver.exe");
        WebDriver driver = new ChromeDriver();
        WebDriverWait wait=new WebDriverWait(driver, 60);
        driver.manage().window().maximize();
        driver.manage().timeouts().implicitlyWait(30,TimeUnit.SECONDS);
        driver.get(url);

        LoginPage login = new LoginPage(driver);
        login.enterEmailAddress(email);
        login.enterPassword(password);
        login.clickOnLogin();

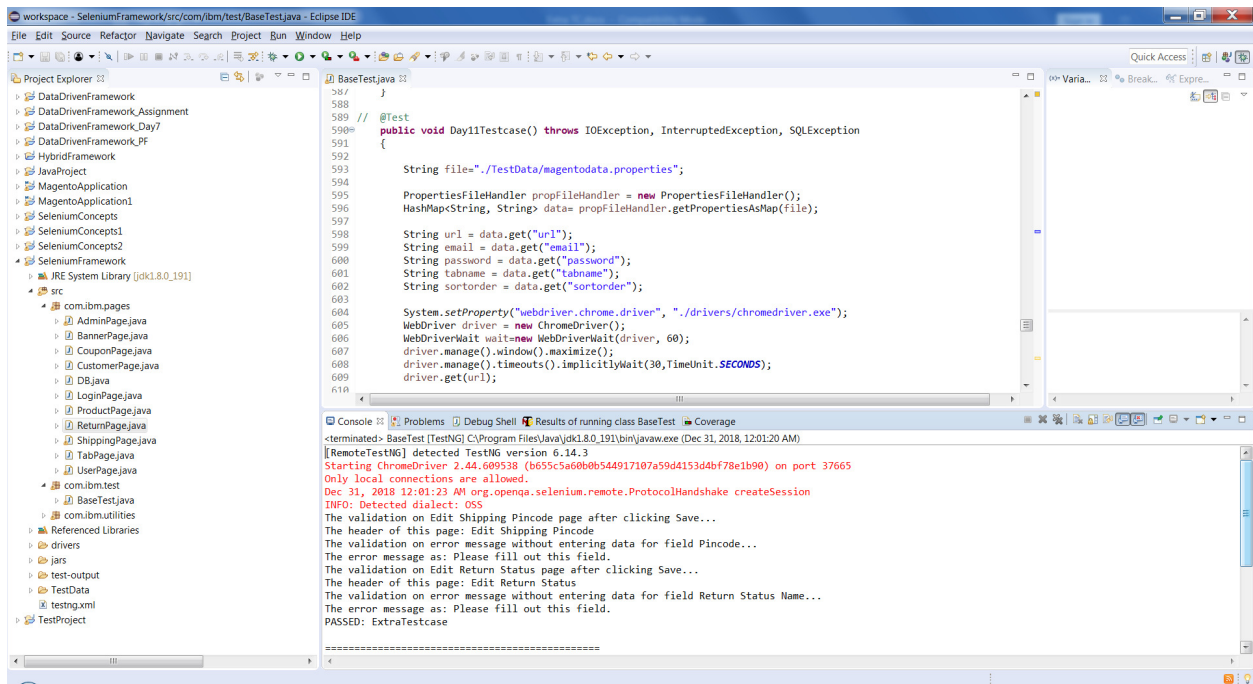
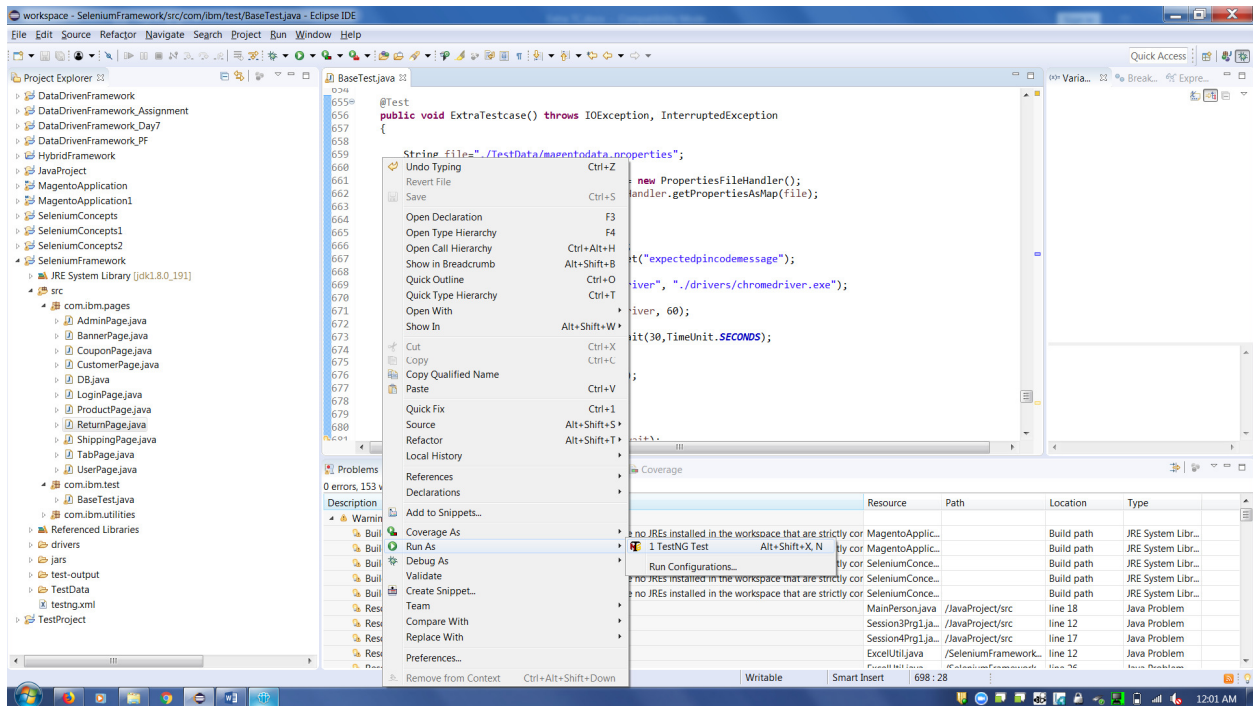
        AdminPage apage=new AdminPage(driver, wait);
        ShippingPage sp = new ShippingPage(driver, wait);
        sp.clickOnLinkSystem();
        Thread.sleep(1000);
        sp.clickOnLinkShipping();
        sp.clickOnLinkShippingPincode();
        sp.clickOnAction();
        sp.clickOnEdit();
        sp.clearPincode();
        sp.clickOnIconSave();
        sp.getHeaderMessage();
        String actualpincodeessage = sp.validationOnMisingField();
        Assert.assertEquals(expectedpincodeessage, actualpincodeessage);

        ReturnPage rp = new ReturnPage(driver, wait);
        rp.clickOnLinkHome();
        rp.clickOnLinkSystem();
        Thread.sleep(1000);
        rp.clickOnLinkReturns();
        Thread.sleep(1000);
        rp.clickOnLinkRetStatuses();
        rp.clickOnAction();
        rp.clickOnEdit();
        rp.clearReturnStatus();
        rp.clickOnIconSave();
        rp.getHeaderMessage();
        rp.validationOnMisingField();

        ap.clickOnLogOut();

        driver.close();
    }
}

```



Eclipse console:

[RemoteTestNG] detected TestNG version 6.14.3

Starting ChromeDriver 2.44.609538 (b655c5a60b0b544917107a59d4153d4bf78e1b90) on port 37665

Only local connections are allowed.

Dec 31, 2018 12:01:23 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: OSS

The validation on Edit Shipping Pincode page after clicking Save...

The header of this page: Edit Shipping Pincode

The validation on error message without entering data for field Pincode...

The error message as: Please fill out this field.

The validation on Edit Return Status page after clicking Save...

The header of this page: Edit Return Status

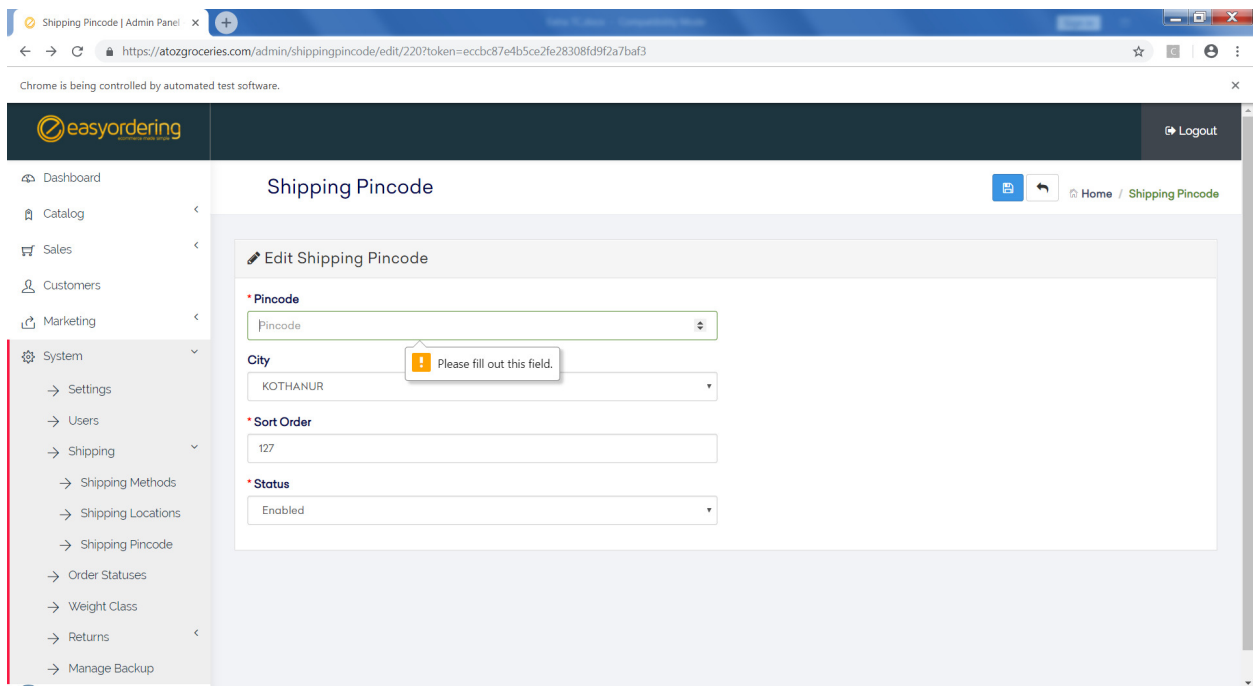
The validation on error message without entering data for field Return Status Name...

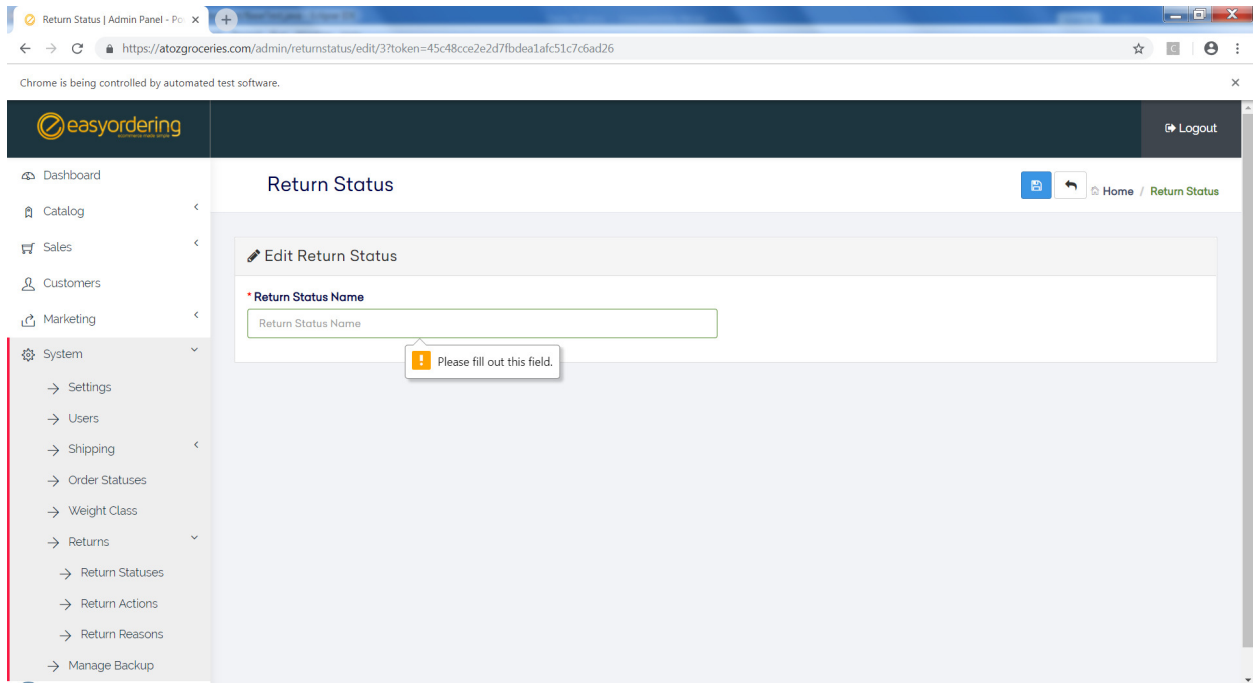
The error message as: Please fill out this field.

PASSED: ExtraTestcase

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

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








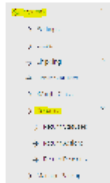
Re: Extra Test Case

Balaji D to: Rajendra K Sahoo

Cc: Natarajan

From: Balaji D <balajidinakaran1@gmail.com>
To:  Rajendra K Sahoo <rajendra.sahoo@in.ibm.com>
Cc: Natarajan <natarajan.ramanathan93@gmail.com>

▼ 2 attachments



17804670.gifgraycol.gif

Please check on order statuses under System.

Regards,
Balaji

Created one new java file as "OrderPage.java"

```
package com.ibm.pages;

import org.junit.Assert;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;
import org.openqa.selenium.support.ui.WebDriverWait;

public class OrderPage {
```



```
WebDriver driver;
```

```
WebDriverWait wait;
```

```
public OrderPage(WebDriver driver,WebDriverWait wait)
```

```
{
```

```
    this.driver=driver;
```

```
    this.wait=wait;
```

```
    PageFactory.initElements(driver,this);
```

```
}
```

```
//WebElement for link System
```

```
    @FindBy(xpath="//a[contains(text(),'System')]")
```

```
    WebElement systemEle;
```

```
//WebElement for link Order Statuses
```

```
    @FindBy(xpath="//a[contains(text(),'Order Statuses')]")
```

```
    WebElement ordstatusEle;
```

```
//WebElement for button Action
```

```
    @FindBy(xpath="(//button[contains(text(), 'Action')])[1]")
```

```
    WebElement actionEle;
```

```
//WebElement for Edit
```

```
    @FindBy(xpath="//a[@title='Edit']")
```

```
    WebElement editEle;
```

```
//WebElement for field Order Status Name
```

```
    @FindBy(xpath="//input[@name='name']")
```

```
    WebElement ordstatusnameEle;
```

```
//WebElement for Header text
```

```
    @FindBy(xpath="//div[@class='panel-title']")
```

```
    WebElement hmessageEle;
```

```
//WebElement for icon Save
```

```
    @FindBy(xpath="//button[@title='Save']")
```

```
    WebElement saveEle;
```

```
//WebElement for link Home
```

```
    @FindBy(xpath="//a[contains(text(),'Home')]")
```

```
    WebElement homeEle;
```

```
    public void clickOnLinkHome()
```

```
    {
```

```
        homeEle.click();
```

```
    }
```

```
    public void clickOnLinkSystem()
```

```
    {
```

```
        systemEle.click();
```

```
}
```

```
public void clickOnLinkOrderStatus()
```

```
{
```

```
    ordstatusEle.click();
```

```
}
```

```
public void clickOnAction()
```

```
{
```

```
    actionEle.click();
```

```
}
```

```
public void clickOnEdit()
```

```
{
```

```
    editEle.click();
```

```
}
```

```
public void clearOrderStatusName()
```

```
{
```

```
    ordstatusnameEle.clear();
```

```
}
```

```
public void clickOnIconSave()
```

```
{
```

```
    saveEle.click();
```

```

    }

    public void getHeaderMessage()
    {
        String header = hmessageEle.getText();
        if (header.contains("Edit Order Status"))
        {
            System.out.println("The validation on Edit Order Status page after clicking
Save...");

            System.out.println("The header of this page: " +header);
            Assert.assertTrue(true);
        }
        else
        {
            Assert.fail();
        }
    }

    public String validationOnMisingField()
    {
        JavascriptExecutor js=(JavascriptExecutor) driver;

        String message = (String) js.executeScript("return
document.getElementsByName('name')[0].validationMessage;");

        System.out.println("The validation on error message without entering data for
field Order Status Name...");

        System.out.println("The error message as: " +message);
    }

```

```
        return(message);  
    }  
}
```

Edited the existing file "BaseTest.java"

```
package com.ibm.test;  
  
import java.io.FileInputStream;  
import java.io.IOException;  
import java.sql.SQLException;  
import java.util.HashMap;  
import java.util.concurrent.TimeUnit;  
  
import org.apache.poi.xssf.usermodel.XSSFCell;  
import org.apache.poi.xssf.usermodel.XSSFRow;  
import org.apache.poi.xssf.usermodel.XSSFSheet;  
import org.apache.poi.xssf.usermodel.XSSFWorkbook;  
import org.junit.Assert;  
import org.openqa.selenium.WebDriver;  
import org.openqa.selenium.chrome.ChromeDriver;  
import org.openqa.selenium.support.ui.WebDriverWait;  
import org.testng.annotations.Test;  
import com.ibm.pages.AdminPage;  
import com.ibm.pages.BannerPage;  
import com.ibm.pages.CouponPage;
```

```
import com.ibm.pages.CustomerPage;

import com.ibm.pages.DB;

import com.ibm.pages.LoginPage;

import com.ibm.pages.OrderPage;

import com.ibm.pages.ProductPage;

import com.ibm.pages.ReturnPage;

import com.ibm.pages.ShippingPage;

import com.ibm.pages.TabPage;

import com.ibm.pages.UserPage;

import com.ibm.utilities.DBUtil;

import com.ibm.utilities.PropertiesFileHandler;


public class BaseTest {

    @Test
    public void ExtraTestcase() throws IOException, InterruptedException
    {

        String file = "./TestData/magentodata.properties";

        PropertiesFileHandler propFileHandler = new PropertiesFileHandler();
        HashMap<String, String> data = propFileHandler.getPropertiesAsMap(file);

        String url = data.get("url");
        String email = data.get("email");
        String password = data.get("password");
        String expectedpincode = data.get("expectedpincode");
        String expectedmissingordstatusmessage =
data.get("expectedmissingordstatusmessage");

        System.setProperty("webdriver.chrome.driver",
"./drivers/chromedriver.exe");
        WebDriver driver = new ChromeDriver();
        WebDriverWait wait = new WebDriverWait(driver, 60);
        driver.manage().window().maximize();
        driver.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);
        driver.get(url);

        LoginPage login = new LoginPage(driver);
        login.enterEmailAddress(email);
        login.enterPassword(password);
```

```

        login.clickOnLogin();

        AdminPage apage=new AdminPage(driver, wait);
        ShippingPage sp = new ShippingPage(driver, wait);
        sp.clickOnLinkSystem();
        Thread.sleep(1000);
        sp.clickOnLinkShipping();
        sp.clickOnLinkShippingPincode();
        sp.clickOnAction();
        sp.clickOnEdit();
        sp.clearPincode();
        sp.clickOnIconSave();
        sp.getHeaderMessage();
        String actualpincodemessage = sp.validationOnMissingField();
        Assert.assertEquals(expectedpincodemessage, actualpincodemessage);

        /*ReturnPage rp = new ReturnPage(driver, wait);
        rp.clickOnLinkHome();
        rp.clickOnLinkSystem();
        Thread.sleep(1000);
        rp.clickOnLinkReturns();
        Thread.sleep(1000);
        rp.clickOnLinkRetStatuses();
        rp.clickOnAction();
        rp.clickOnEdit();
        rp.clearReturnStatus();
        rp.clickOnIconSave();
        rp.getHeaderMessage();
        rp.validationOnMissingField();*/

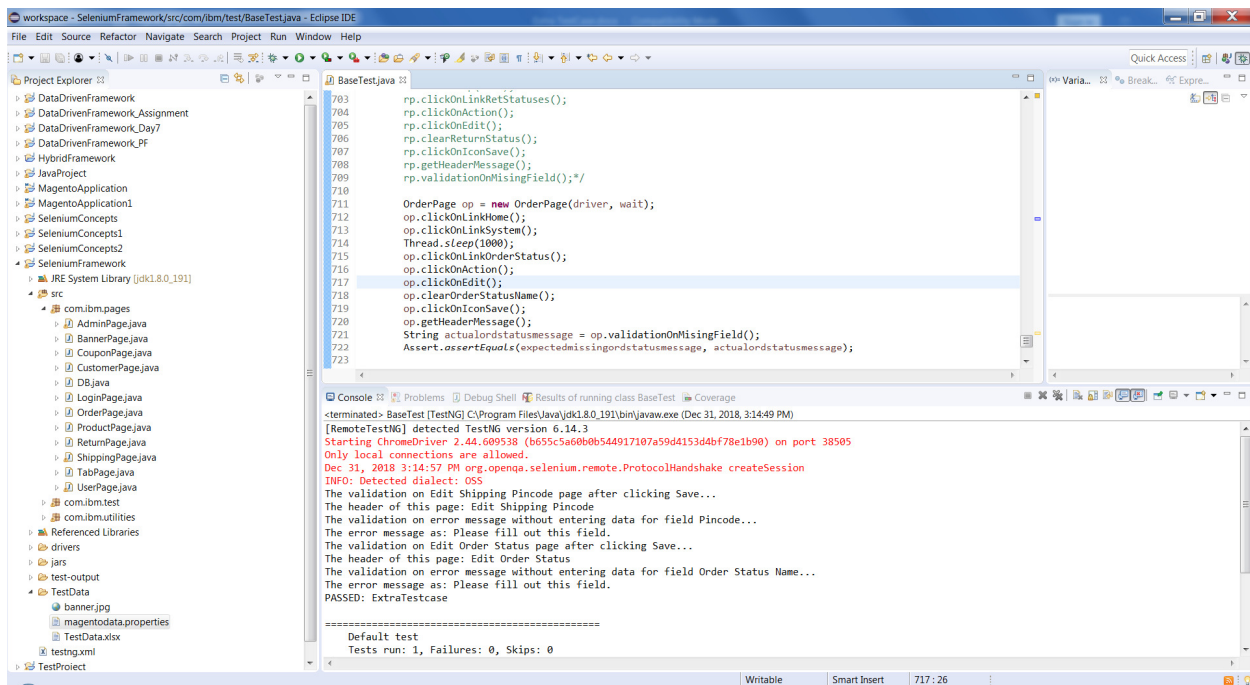
        OrderPage op = new OrderPage(driver, wait);
        op.clickOnLinkHome();
        op.clickOnLinkSystem();
        Thread.sleep(1000);
        op.clickOnLinkOrderStatus();
        op.clickOnAction();
        op.clickOnEdit();
        op.clearOrderStatusName();
        op.clickOnIconSave();
        op.getHeaderMessage();
        String actualordstatusmessage = op.validationOnMissingField();
        Assert.assertEquals(expectedmissingordstatusmessage,
actualordstatusmessage);

        apage.clickOnLogOut();

        driver.close();

    }
}

```



Eclipse console:

[RemoteTestNG] detected TestNG version 6.14.3

Starting ChromeDriver 2.44.609538 (b655c5a60b0b544917107a59d4153d4bf78e1b90) on port 38505

Only local connections are allowed.

Dec 31, 2018 3:14:57 PM org.openqa.selenium.remote.ProtocolHandshake createSession

INFO: Detected dialect: OSS

The validation on Edit Shipping Pincode page after clicking Save...

The header of this page: Edit Shipping Pincode

The validation on error message without entering data for field Pincode...

The error message as: Please fill out this field.

The validation on Edit Order Status page after clicking Save...

The header of this page: Edit Order Status

The validation on error message without entering data for field Order Status Name...

The error message as: Please fill out this field.

PASSED: ExtraTestcase

```

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

```

```

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

```


Shipping Pincode | Admin Panel

https://atozgroceries.com/admin/shippingpincode/edit/220?token=45c48cce2e2d71bdea1afc51c7c6ad26

Chrome is being controlled by automated test software.

easyordering [Logout](#)

Dashboard
Catalog
Sales
Customers
Marketing
System
→ Settings
→ Users
→ Shipping
→ Shipping Methods
→ Shipping Locations
→ Shipping Pincode
→ Order Statuses
→ Weight Class
→ Returns
→ Manage Backup

Shipping Pincode

[Home](#) / [Shipping Pincode](#)

Edit Shipping Pincode

*** Pincode**
Pincode

City
KOTHANUR

*** Sort Order**
127

*** Status**
Enabled

Please fill out this field.

Order Status | Admin Panel

https://atozgroceries.com/admin/orderstatus/edit/1?token=a87ff679a2f3e71d9181a67b7542122c

Chrome is being controlled by automated test software.

easyordering [Logout](#)

Dashboard
Catalog
Sales
Customers
Marketing
System
→ Settings
→ Users
→ Shipping
→ Order Statuses
→ Weight Class
→ Returns
→ Manage Backup
Reports

Order Status

[Home](#) / [Order Status](#)

Edit Order Status

*** Order Status Name**
Order Status Name

*** Status**
Enabled

Please fill out this field.