

PFA day 7 test case,

Process	Steps To Execute	Expected Results
Sign Up and validate fill address	1. Sign up on User page (https://atozgroceries.com/) 2. Goto My Account 3. Goto My Address 4. Fill the Address Details except email 5. Update the details	1. Click the update 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

Code: User sign in page :

```
package com.ibm.pages;

import org.openqa.selenium.By;
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 UserSigninPage {

    WebDriver driver;
    WebDriverWait wait;
    JavascriptExecutor js;

    @FindBy(xpath="/html/body/header/div[1]/div/div[2]/div[3]/li/a/i[2]")
    WebElement myaccountEle;

    @FindBy(xpath="//a[text()='My Address']")
    WebElement myaddressEle;

    //constructor

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

    // method to perform the signin process by filling all the details.
```

```
    public void signupProcess(String username, String phonenum, String password,
String confirmpassword ) throws InterruptedException {
```

```
        driver.findElement(By.xpath("//a[@class='register']")).click();
        driver.findElement(By.id("name")).sendKeys(username);
        driver.findElement(By.id("pnum")).sendKeys(phonenum);
        driver.findElement(By.id("password")).sendKeys(password);
        driver.findElement(By.id("cpassword")).sendKeys(confirmpassword);
        driver.findElement(By.id("tccheckbox")).click();
        driver.findElement(By.id("mem_signup")).click();
        Thread.sleep(5000);
        wait.until(ExpectedConditions.alertIsPresent());
        driver.switchTo().alert().accept();
```

```
    }
```

```
    // method to click on my account
```

```
    public void clickOnMyaccount() throws InterruptedException{
```

```
        Thread.sleep(5000);
        js.executeScript("arguments[0].click()",myaccountEle);
```

```
    }
```

```
    // method to click on My address and update the address field leaving the
email field blank
```

```
    public void clickOnMyaddress() throws InterruptedException {
```

```
        myaddressEle.click();
        driver.findElement(By.id("address")).sendKeys("test");
        driver.findElement(By.xpath("//select[@id='city']/option[2]")).click();
        driver.findElement(By.id("pincode")).sendKeys("700156");
        Thread.sleep(5000);
        driver.findElement(By.xpath("//button[@type='submit']")).click();
```

```
    }
```

```
    // method to verify the tooltip msg for negative validation
```

```
    public String verifyToolTipMsg() {
```

```
        String tooltpmsg= js.executeScript(" return
document.getElementsByName('email')[0].validationMessage").toString().trim();
```

```
        return tooltpmsg;
```

```
    }
```

```
}
```

Main method :

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

```
import static org.testng.Assert.assertTrue;
```

```
import java.io.File;
```

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

```
import java.util.Date;
```

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

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

```
import org.apache.commons.io.FileUtils;
```

```
import org.openqa.selenium.JavascriptExecutor;
```

```
import org.openqa.selenium.OutputType;
```

```
import org.openqa.selenium.TakesScreenshot;
```

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

```
import org.openqa.selenium.chrome.ChromeDriver;
```

```
import org.openqa.selenium.firefox.FirefoxDriver;
```

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

```
import org.testng.Assert;
```

```
import org.testng.annotations.AfterMethod;
```

```
import org.testng.annotations.BeforeMethod;
```

```
import org.testng.annotations.BeforeSuite;
import org.testng.annotations.DataProvider;
import org.testng.annotations.Parameters;
import org.testng.annotations.Test;
import com.ibm.pages.AdminPage;
import com.ibm.pages.LoginPage;
import com.ibm.pages.MarketingPage;
import com.ibm.pages.ProductPage;
import com.ibm.pages.ReturnSatusPage;
import com.ibm.pages.UserPage;
import com.ibm.pages.UserSigninPage;
import com.ibm.utilities.ExcelUtil;
import com.ibm.utilities.PropertiesFileHandler;
```

```
public class BaseTest {
```

```
    WebDriver driver;
```

```
    WebDriverWait wait;
```

```
    PropertiesFileHandler propFileHandler;
```

```
    HashMap<String, String> data;
```

```
    // beforesuit method is used to declare the statement that are going to execute
```

```
    // before the running of testcase
```

```
    @BeforeSuite
```

```
    public void preSetForTest() throws IOException {
```

```
        String file = "./TestData/data.properties";
```

```

        propFileHandler = new PropertiesFileHandler();
        data = propFileHandler.getPropertiesAsMap(file);

    }

    // beforemethod is used to declare the statement that are going to run at the
    // starting of all test cases
    @BeforeMethod

    public void Initialization() {

        System.setProperty("webdriver.chrome.driver", "./driver/chromedriver.exe");
        System.setProperty("webdriver.gecko.driver", "./driver/geckodriver.exe");
        driver = new ChromeDriver();// opening the chrome browser

        //driver=new FirefoxDriver();
        wait = new WebDriverWait(driver, 120);// giving explicit wait
        driver.manage().window().maximize();// maximizing the window
        driver.manage().deleteAllCookies();
        driver.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS); // declaring implicit
wait

    }

    // aftermethod is used to run once all the test cases are run

    @AfterMethod

    public void closeBrowser() {

        driver.quit();// close the browser
    }

```

```
}
```

```
@Test
```

```
public void testcase7() throws InterruptedException, IOException {
```

```
    // for the sign in process its taking all the data from the properties file
    String urlnew = data.get("urlnew");
    String username = data.get("fullname");
    String phonenum = data.get("Phonenum");
    String password = data.get("signinpassword");
    String confirmpassword = data.get("signinconfirmpassword");
    String userpageexpected = data.get("userpageexpectedmsg");

    // opening the url :https://atozgroceries.com
    driver.get(urlnew);

    // calling signin page on mainmethod toperform the signin process and update the
    address tab after signing in

    UserSigninPage signin = new UserSigninPage(driver, wait);

    signin.signupProcess(username, phonenum, password, confirmpassword);
    signin.clickOnMyaccount();
    signin.clickOnMyaddress();
    Thread.sleep(5000);
    String actualuserpagemsg = signin.verifyToolTipMsg();

    // Assertion to verify the negative field validation
    Assert.assertTrue(actualuserpagemsg.contains(userpageexpected), "Assertion on
    negativefield validation");

    TakesScreenshot ts = (TakesScreenshot) driver;
    File file = ts.getScreenshotAs(OutputType.FILE);
    Date date = new Date();
    String currentDate = date.toString().replace(":", "-");
    FileUtils.copyFile(file, new File("./screenshots/Error_" +
    currentDate + ".png"));
```

```
}
```

```
}
```

Screen Shot while running the code :



