Day 9 test case, which includes DB validations

Process	Steps To Execute	Expected Results
Add Users	1. Navigate to Systems section, shipping Option 2. Choose Users 3. Edit a users and Save.	1. Validate the presence of the modified users. 2. Verify the presence of modified users the on the database.

Practitioners needs to wait for Monday's session in order to understand DB details.

Solution

Base Class

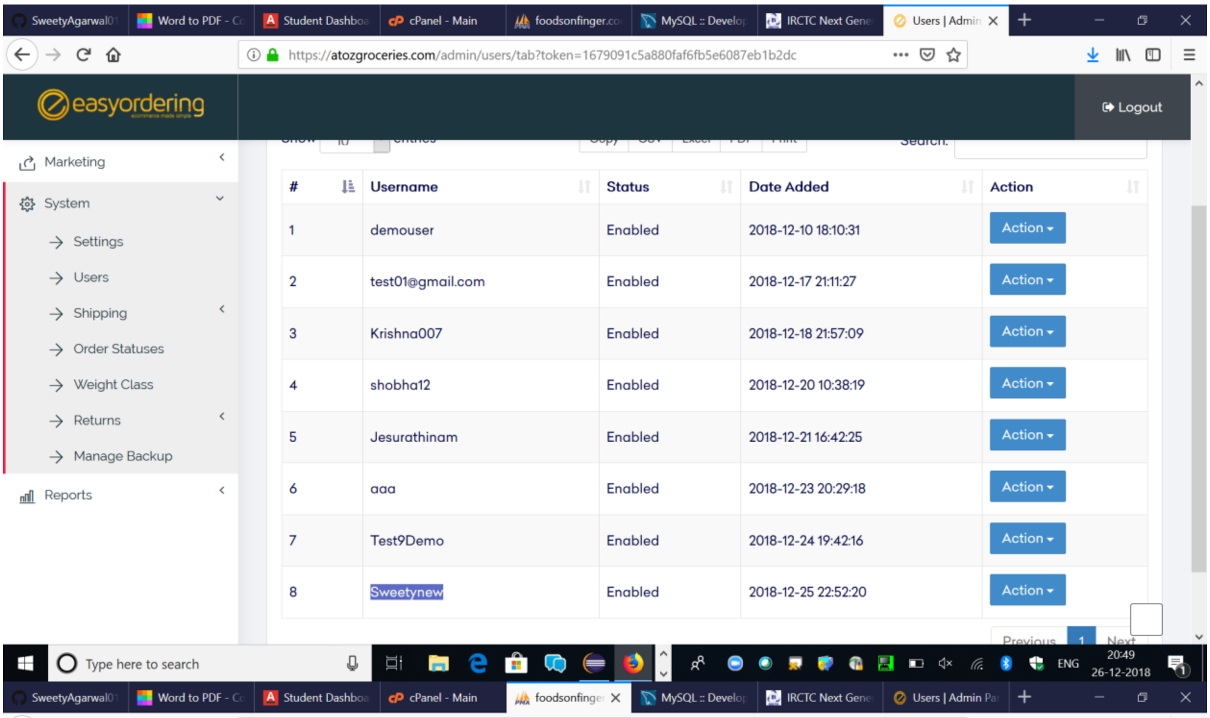
```
package com.ibm.test;

import static org.junit.Assert.assertTrue;
import static org.testng.Assert.assertEquals;
import static org.testng.Assert.assertFalse;
import static org.testng.Assert.assertNotEquals;
import static org.testng.Assert.assertTrue;

import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Date;
import java.util.HashMap;
import java.util.List;
import java.util.Properties;
import java.util.concurrent.TimeUnit;

import org.apache.commons.io.FileUtils;
import org.openqa.selenium.Alert;
import org.openqa.selenium.By;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Keys;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.remote.LocalFileDetector;
import org.openqa.selenium.support.ui.Select;
import org.openqa.selenium.By;
import org.testng.annotations.AfterMethod;
import org.testng.annotations.AfterTest;
import org.testng.annotations.DataProvider;
import org.testng.annotations.Test;

import com.ibm.pages.Login;

import com.ibm.pages.*;
import com.ibm.utilities.ExcelUtil;
import com.ibm.utilities.PropertiesFileHandler;
import com.ibm.utilities.JDBC_Connection;
import org.testng.Assert;


//jars would be in the link
// https://drive.google.com/open?id=1Oa7AzFN8nQapvatjRToU_bwDgpf18QKI
public class BaseTest extends WebDriverLaunch
{
	@Test
	public void login() throws FileNotFoundException, IOException,
InterruptedException
	 {

		String url = data.get("url");
		String userName = data.get("user");
		String password = data.get("password");
		driver.get(url);

		Login loginele = new Login(driver, wait);
		loginele.enterEmailAddress(userName);
		loginele.enterPassword(password);
		loginele.clickOnLogin();
		Properties p = new Properties();
		p.load(new FileInputStream("TestData/data.properties"));

		AdminPage admin = new AdminPage(driver, wait);

		admin.clickonSystemtab();
		admin.clickonUsertab();

	driver.findElement(By.xpath("//*[@id='dataTableExample2']/tbody/tr[8]/td[5]/di
v/button")).click();

	driver.findElement(By.xpath("//*[@id='dataTableExample2']/tbody/tr[8]/td[5]/di
v/ul/li[1]/a")).click();

	driver.findElement(By.xpath("//input[@name='pword']")).sendKeys("12345678
9");
```

```
        driver.findElement(By.xpath("//input[@name='cpword']")).sendKeys("1234567
89");

                WebElement saveEle=
driver.findElement(By.xpath("//button[@type='Submit']"));
                Thread.sleep(2000);
                JavascriptExecutor js=(JavascriptExecutor) driver;
                js.executeScript("arguments[0].click();",saveEle );

                WebElement message= driver.findElement(By.xpath("//*[@id='page-
wrapper']/div/div[2]"));
                String msg =message.getText();
                System.out.println(msg);
                String pagesource = driver.getPageSource();

                assertTrue(pagesource.contains("Success: You have successfully
updated user!"), msg);

                Thread.sleep(3000);
                loginele.clickOnLogout();

         }

        @Test

        public void Databaseconnection() throws SQLException

        {
                String exp = JDBC_Connection.singleDataQuery("SELECT * from
as_user WHERE username = 'Sweetynew'");

                System.out.println(exp);
        }
        }
```

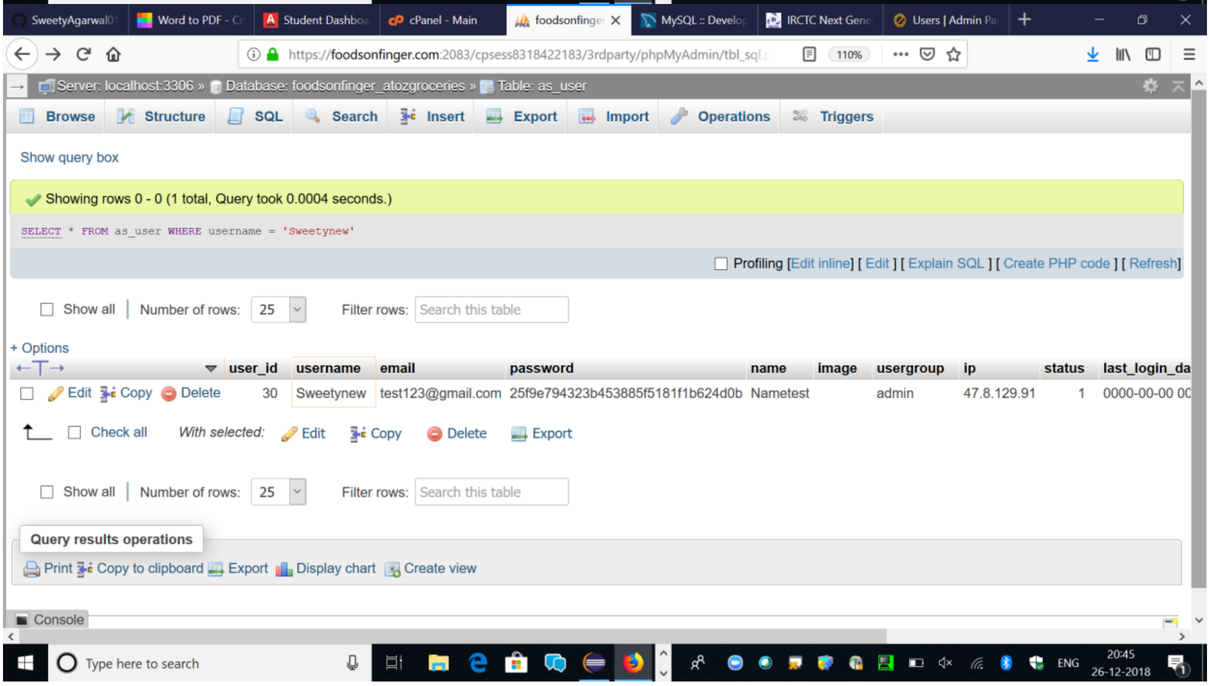
JDBC_Connection Class

```
package com.ibm.utilities;
//We will understand the construction of this page
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.ResultSetMetaData;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;


public class JDBC_Connection
{


public static String singleDataQuery(String query) throws SQLException
             {
                          String text=null;

             Connection con =
DriverManager.getConnection("jdbc:mysql://foodsonfinger.com:3306/foodsonfinger_ato
zgroceries","foodsonfinger_atoz",
             "welcome@123");

             Statement s = con.createStatement();

             ResultSet rs = s.executeQuery(query);

             if(rs.next())
              {

             text = rs.getString(1);

              }
             return text;

             }

}
```

Screen shot



#	Username	Status	Date Added	Action
1	demouser	Enabled	2018-12-10 18:10:31	Action ▾
2	test01@gmail.com	Enabled	2018-12-17 21:11:27	Action ▾
3	Krishna007	Enabled	2018-12-18 21:57:09	Action ▾
4	shobha12	Enabled	2018-12-20 10:38:19	Action ▾
5	Jesurathinam	Enabled	2018-12-21 16:42:25	Action ▾
6	aaa	Enabled	2018-12-23 20:29:18	Action ▾
7	Test9Demo	Enabled	2018-12-24 19:42:16	Action ▾
8	Sweetynew	Enabled	2018-12-25 22:52:20	Action ▾



✓ Showing rows 0 - 0 (1 total, Query took 0.0004 seconds.)

```
SELECT * FROM as_user WHERE username = 'Sweetynew'
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table

	user_id	username	email	password	name	image	usergroup	ip	status	last_login_da
□ Edit Copy Delete	30	Sweetynew	test123@gmail.com	25f9e794323b453885f5181f1b624d0b	Nametest		admin	47.8.129.91	1	0000-00-00 00

□ Check all *With selected:* Edit Copy Delete Export

□ Show all | Number of rows: 25 Filter rows: Search this table

Query results operations

Print Copy to clipboard Export Display chart Create view