1.Problem statement:

Process	Steps To Execute	Expected Results
Add a category	1. Login to admin portal 2. Add new category using excel and dataprovider.	1. Validate the presence of the added category. 2. Verify the presence of added category under on the database.

2. code with comments:

ExcelReader.java – to read Excel and to execute Database check

```
package com.ibm.utilities;

import java.io.FileInputStream;
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 org.apache.poi.ss.usermodel.DataFormatter;
import org.apache.poi.xssf.usermodel.XSSFSheet;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;

public class ExcelReader {
    public static void db_connect(String table, String data) throws SQLException {
        Connection
c=DriverManager.getConnection("jdbc:mysql://foodsonfinger.com:3306/foodsonfinger_atoz
groceries","foodsonfinger_atoz","welcome@123");
        Statement s=c.createStatement();
        ResultSet rs=s.executeQuery("SELECT * from " +table+ " where " +data);
        while(rs.next()) {

            System.out.println("database check for added
element:"+rs.getString("name"));
        }

        //Assert.assertEquals("Grains", rs.getString("name"));
    }
    public static Object[][] DataTable(String WBLoc, String sheetName) throws
IOException
    {
    XSSFWorkbook TestDataWB=new XSSFWorkbook(new FileInputStream(WBLoc));
    XSSFSheet loginSheet=TestDataWB.getSheet(sheetName);
    Object[][] loginInfo=new Object[loginSheet.getPhysicalNumberOfRows()-
1][loginSheet.getRow(0).getPhysicalNumberOfCells()];
    int rowcount=loginSheet.getPhysicalNumberOfRows();
    int cellcount=loginSheet.getRow(0).getPhysicalNumberOfCells();
    for(int i=1;i<rowcount;i++)
    {
            for(int j=0;j<cellcount;j++)
            {
                  DataFormatter format=new DataFormatter();
                  String
cellvalue=format.formatCellValue(loginSheet.getRow(i).getCell(j));
                  loginInfo[i-1][j]=cellvalue;
            }
      }
      return loginInfo;
      }

}
```

BaseTest.java:

```
package com.ibm.test;


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


import org.openqa.selenium.Alert;
import org.openqa.selenium.By;
import org.openqa.selenium.JavascriptExecutor;
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.support.ui.Select;

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

import org.testng.Assert;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.DataProvider;

import org.testng.annotations.Test;


import com.ibm.pages.UserPage;

import com.ibm.utilities.ExcelReader;

import com.ibm.utilities.ExcelUtil;

import com.ibm.utilities.PropertiesFileHandler;


public class BaseTest extends ExcelReader{

        WebDriverWait wait;

        WebDriver driver;

   @Test(dataProvider="CategoryData")


   public void testcase9(String CategoryName, String MegaTagTitle, String SortOrder) throws
InterruptedException, IOException, SQLException{


        FileInputStream file = new FileInputStream("./TestData/data.properties");

        Properties prop = new Properties();

        prop.load(file);

        String url = prop.getProperty("url");

        String username = prop.getProperty("user");

        String password = prop.getProperty("password");

                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);
Login login = new Login(driver, wait);
driver.get(url);

login.enterEmailAddress(username);
login.enterPassword(password);
login.clickOnLogin();

WebElement systemEle=driver.findElement(By.linkText("Catalog"));
systemEle.click();

WebElement usersEle=driver.findElement(By.linkText("Categories"));
usersEle.click();

Thread.sleep(3000);
WebElement newUserEle=driver.findElement(By.cssSelector("a.btn.btn-primary"));
newUserEle.click();

WebElement categoryName=driver.findElement(By.name("name"));
categoryName.sendKeys(CategoryName);

WebElement metaTagName=driver.findElement(By.name("tag_title"));
metaTagName.sendKeys(MegaTagTitle);
```

```
WebElement sortOrder=driver.findElement(By.name("sort"));
sortOrder.sendKeys(SortOrder);

WebElement statusEle=driver.findElement(By.name("status"));
Select select=new Select(statusEle);
select.selectByVisibleText("Enabled");

WebElement toTopEle=driver.findElement(By.id("toTop"));
toTopEle.click();
Thread.sleep(3000);
WebElement saveEle=driver.findElement(By.cssSelector("button.btn.btn-primary"));
saveEle.click();

//Assert.assertTrue(driver.findElement(By.linkText("oats")).isDisplayed());

WebElement logOutEle=driver.findElement(By.cssSelector("i.fa.fa-sign-out"));
logOutEle.click();

Thread.sleep(3000);

db_connect("as_category","name=\"oats\"");

driver.get("https://atozgroceries.com");
driver.manage().window().maximize();
driver.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);
UserPage user = new UserPage(driver, wait);

WebElement loginEle=driver.findElement(By.linkText("Login"));
loginEle.click();
```

```
                user.userName("7675058941");

                user.enterPassword("456789");

                user.clickOnLogin();

                Thread.sleep(3000);

                WebElement categoryEle=driver.findElement(By.cssSelector("#categories-menu > ul.menu > li.menu-item > span.click-categories.flaticon-bars"));

                categoryEle.click();

//the below will fail if oats are not added

                WebElement fruitsEle=driver.findElement(By.linkText("oats"));

                fruitsEle.click();

                //("as_category","name=\"oats\"");

                //Assert.assertTrue(driver.findElement(By.linkText("oats")).isDisplayed());


    }


    @DataProvider(name="CategoryData")

    public Object[][] categoryData() throws IOException {

        return ExcelReader.DataTable("./TestData/TestData.xlsx","Category");

    }


}
```

Properties file:

```
url=https://atozgroceries.com/admin
user=demo@atozgroceries.com
password=456789
url1=https://atozgroceries.com
user1=7675058941
password1=456789
```

UserPage.java

```
package com.ibm.pages;
```

```
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;

import com.ibm.test.BaseTest;

public class UserPage extends BaseTest{

        @FindBy(xpath="//*[@id=\"pnum2\"]")
        WebElement userEle;

        @FindBy(id="pword2")
        WebElement passEle;

        @FindBy(id="mem_login")
        WebElement loginEle;
        WebDriverWait wait;
        WebDriver driver;

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

        public void userName(String user1)
        {
            userEle.sendKeys(user1);
        }
        public void enterPassword(String password1)
        {
            passEle.sendKeys(password1);
        }
        public void clickOnLogin()
        {
            loginEle.click();
        }

    /*  @FindBy(xpath="//*cvdcg")
        WebElement tabNotPresentEle;

        public boolean getTabNotPresent()
        {
        boolean deletedTab=tabNotPresentEle.isDisplayed();
        return deletedTab;
        }*/

}
```

Login.java:

```
package com.ibm.test;

import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.util.HashMap;
import java.util.Properties;
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.FindBy;

import org.openqa.selenium.support.PageFactory;

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

import org.testng.Assert;

import org.testng.annotations.BeforeSuite;

import org.testng.annotations.Test;


import com.ibm.pages.AdminPage;

import com.ibm.utilities.PropertiesFileHandler;


public class Login extends BaseTest{


        @FindBy(name="email")

        WebElement emailEle;


        @FindBy(name="pword")

        WebElement passEle;


        @FindBy(xpath="/html/body/div/div/div/div[2]/form/button")

        WebElement loginEle;

        WebDriverWait wait;

        WebDriver driver;
```

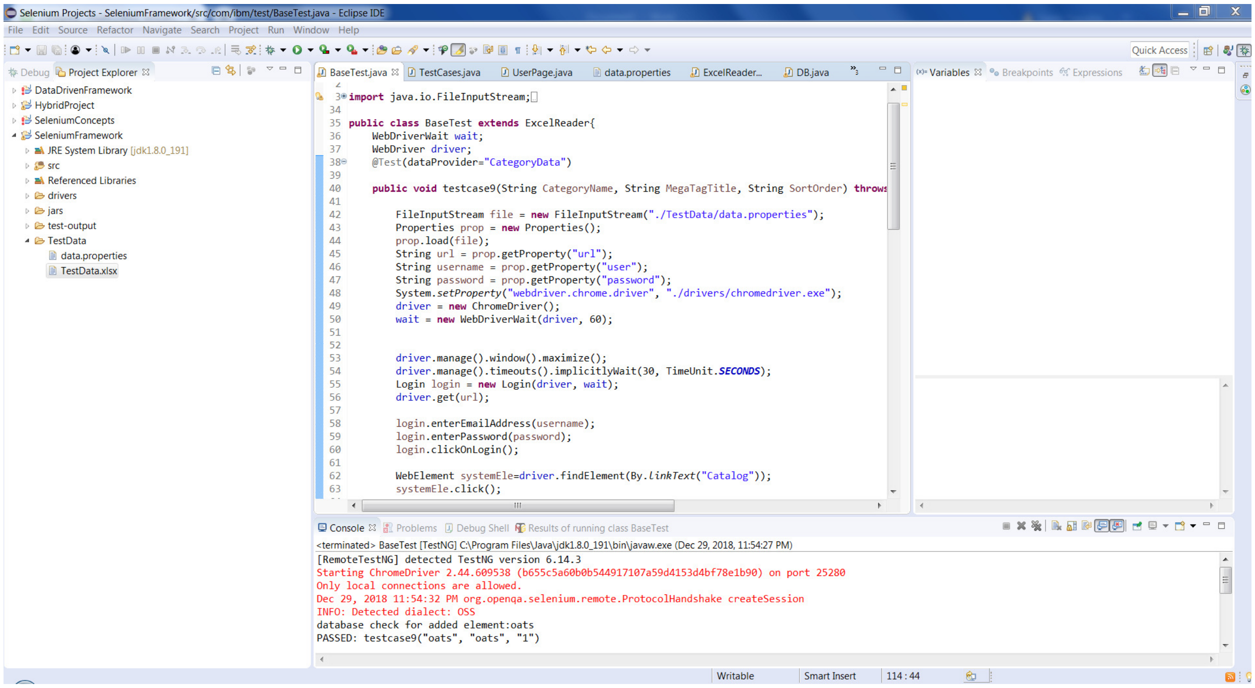
```
    public Login(WebDriver driver,WebDriverWait wait) {
        PageFactory.initElements(driver, this);
        this.driver=driver;
        this.wait=wait;
    }

    public void enterEmailAddress(String user)
    {
        emailEle.sendKeys(user);
    }
    public void enterPassword(String password)
    {
        passEle.sendKeys(password);
    }
    public void clickOnLogin()
    {
        loginEle.click();
    }

}
```

3. Explanation of the code: UserPage.java has Userpage login code and Login.java has admin Page login code these both are extending the BaseTest.java code where I have written the code for the testcase code. I have used data provider concept to use the data from excel sheet. ExcelReader.java has excel data reading as well database connection code.

4. Result flow in detail:

UserPage.java has Userpage login code and Login.java has admin Page login code these both are extending the BaseTest.java code where I have written the code for the testcase code. I have used data provider concept to use the data from excel sheet. ExcelReader.java has excel data reading as well database connection code.

5. Output screenshot:



```
import java.io.FileInputStream;

public class BaseTest extends ExcelReader{
    WebDriverWait wait;
    WebDriver driver;
    @Test(dataProvider="CategoryData")

    public void testcase9(String CategoryName, String MegaTagTitle, String SortOrder) throws

        FileInputStream file = new FileInputStream("./TestData/data.properties");
        Properties prop = new Properties();
        prop.load(file);
        String url = prop.getProperty("url");
        String username = prop.getProperty("user");
        String password = prop.getProperty("password");
        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);
        Login login = new Login(driver, wait);
        driver.get(url);

        login.enterEmailAddress(username);
        login.enterPassword(password);
        login.clickOnLogin();

        WebElement systemEle=driver.findElement(By.linkText("Catalog"));
        systemEle.click();
```

```
<terminated> BaseTest [TestNG] C:\Program Files\Java\jdk1.8.0_191\bin\javaw.exe (Dec 29, 2018, 11:54:27 PM)
[RemoteTestNG] detected TestNG version 6.14.3
Starting ChromeDriver 2.44.609538 (b655c5a60b0b544917107a59d4153d4bf78e1b90) on port 25280
Only local connections are allowed.
Dec 29, 2018 11:54:32 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: OSS
database check for added element:oats
PASSED: testcase9("oats", "oats", "1")
```