# **Cucumber for Beginners:**

Behavior Driven Development (BDD) is a rising methodology to test and check your code. Cucumber framework is a flagship BDD tool.

**What is Cucumber?**

**Cucumber** is a testing tool that supports Behavior Driven Development (BDD). It offers a way to write tests that anybody can understand, regardless of their technical knowledge.

**Advantages of Cucumber**

1. It is helpful to involve business stakeholders who can't easily read code
2. Cucumber Testing tool focuses on end-user experience
3. Style of writing tests allow for easier reuse of code in the tests
4. Quick and easy set up and execution

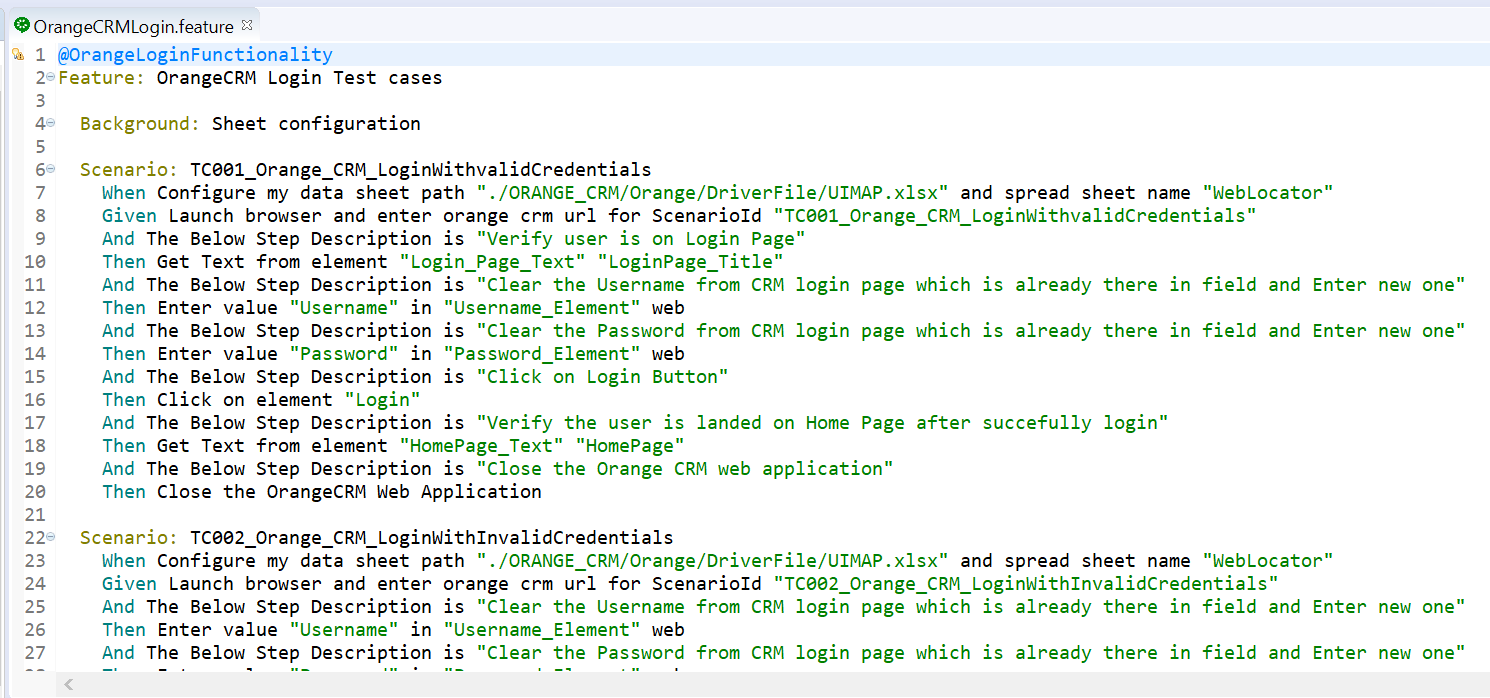
## **What is a "Feature File"?**

Features file contains high level description of the Test Scenario in simple language. It is known as Gherkin. Gherkin is a plain English text language

Feature File consist of following components -

* **Feature**: A feature would describe the current test script which has to be executed.
* **Scenario**: Scenario describes the steps and expected outcome for a particular test case.
* **Scenario Outline**: Same scenario can be executed for multiple sets of data using scenario outline. The data is provided by a tabular structure separated by (I I).
* **Given**: It specifies the context of the text to be executed. By using datatables "Given", steps can also be parameterized.
* **When**: "When" specifies the test action that has to performed
* **Then**: The expected outcome of the test can be represented by "Then"

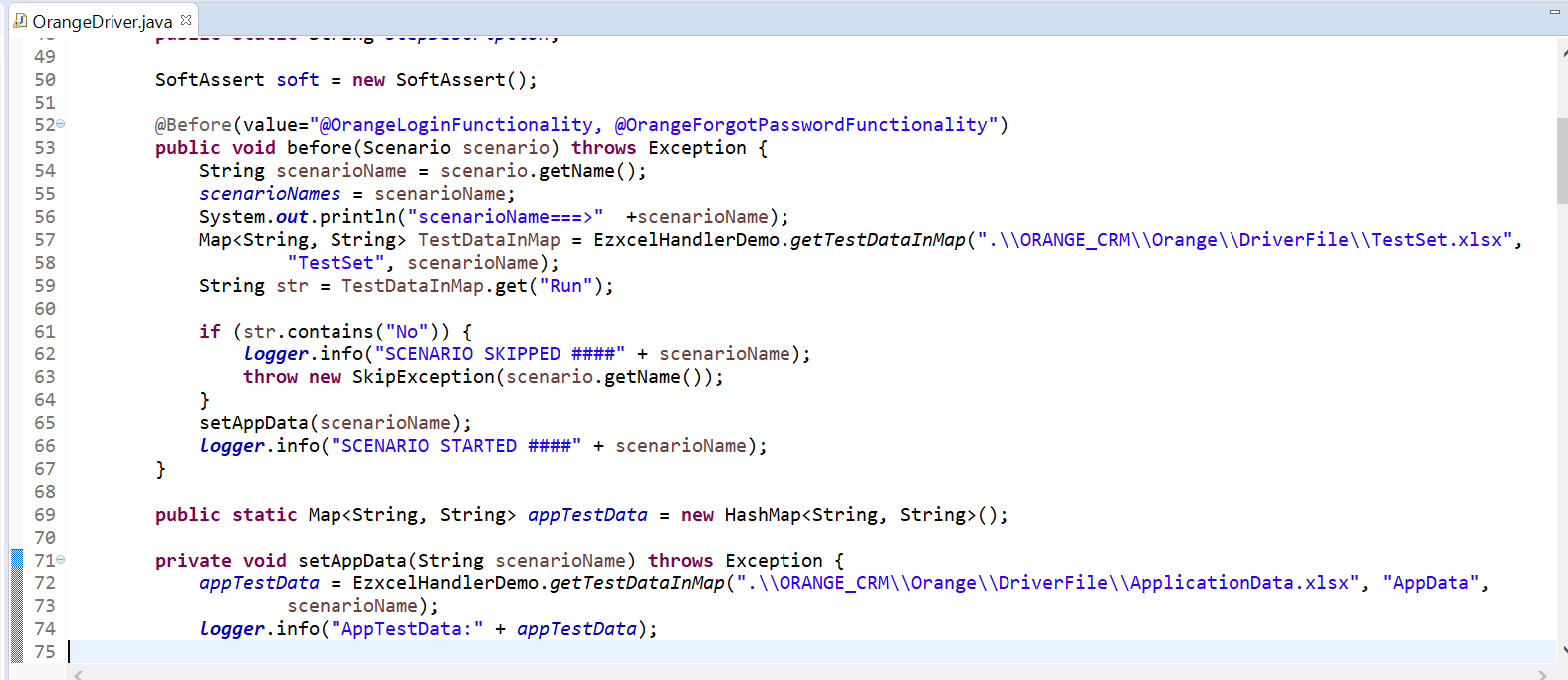
**Sample Feature File Example:**

****

## What is "Step Definition"?

Step definition maps the Test Case Steps in the feature files(introduced by Given/When/Then) to code. It executes the steps on Application Under Test and checks the outcomes against expected results. For a step definition to be executed, it must match the given component in a feature.

**Example for Step Definition**:



**Summary:**

* You need 2 Files – Features and Step Definition to execute a Cucumber test scenario
* Features file contain high level description of the Test Scenario in simple language
* Steps Definition file contains the actual code to execute the Test Scenario in the Features file.

## **What is Gherkin Language?**

Gherkin is a business readable language which helps you to describe business behavior without going into details of implementation. It is a domain specific language for defining tests in Cucumber format for specifications. It uses plain language to describe use cases and allows users to remove logic details from behavior tests.

## **Important Terms used in Gherkin**

* Feature
* Background
* Scenario
* Given
* When
* Then
* And
* But
* Scenario Outline Examples

### **Feature:**

The file should have extension .feature and each feature file should have only one feature.

**Scenario**:

Each feature file may have multiple scenarios, and each scenario starts with Scenario: followed by scenario name.

### **Background**:

Background keywords help you to add some context to the scenario. It can contain some steps of the scenario, but the only difference is that it should be run before each scenario.

### **Given:**

The use of the Given keyword is to put the system in a familiar state before the user starts interacting with the system. However, you can omit writing user interactions in Given steps if Given in the "Precondition" step.

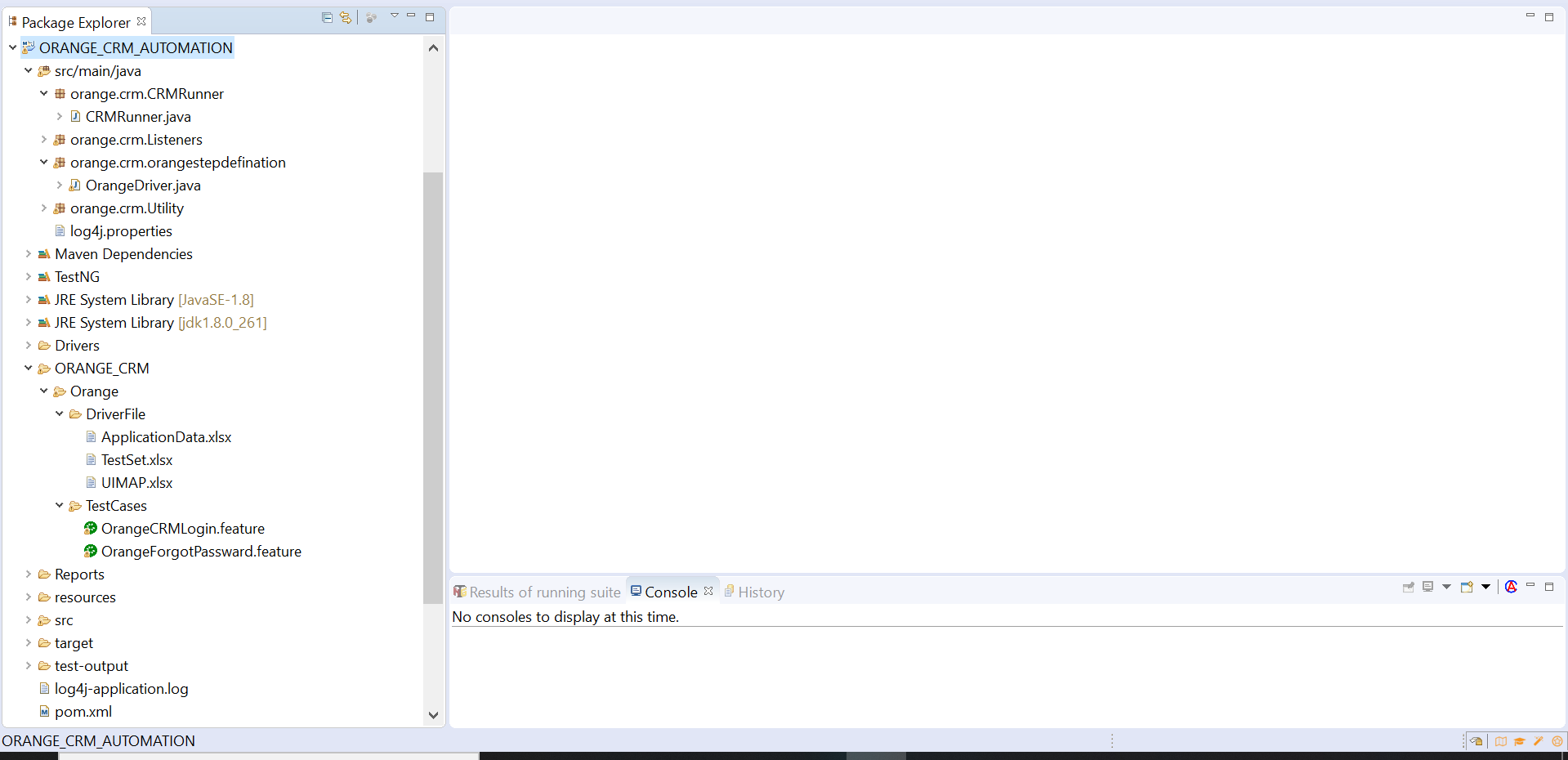
### **When**:

When the step is to define action performed by the user.

### **Then:**

The use of 'then' keyword is to see the outcome after the action in when step. However, you can only verify noticeable changes.

## **Framework structure:**

****

### **DriverFile (Folder) :** It has 3 different sheets

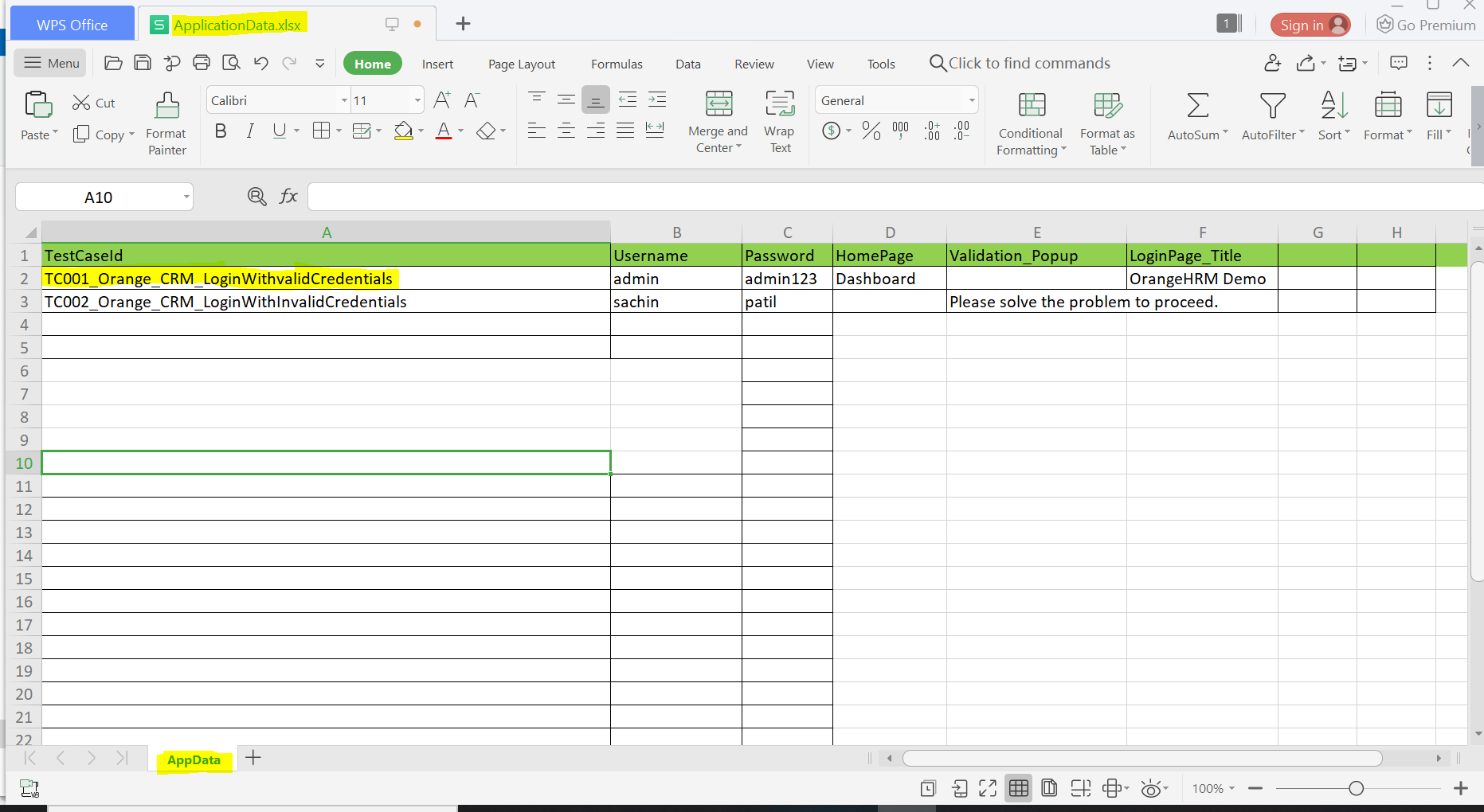
#### **ApplicationData :** This file contains the our test data which we use for our test case execution

#### **TestSet :** This file contains the list of test cases and also we can mark the test cases as “No” if we don’t want to run any particular test case.

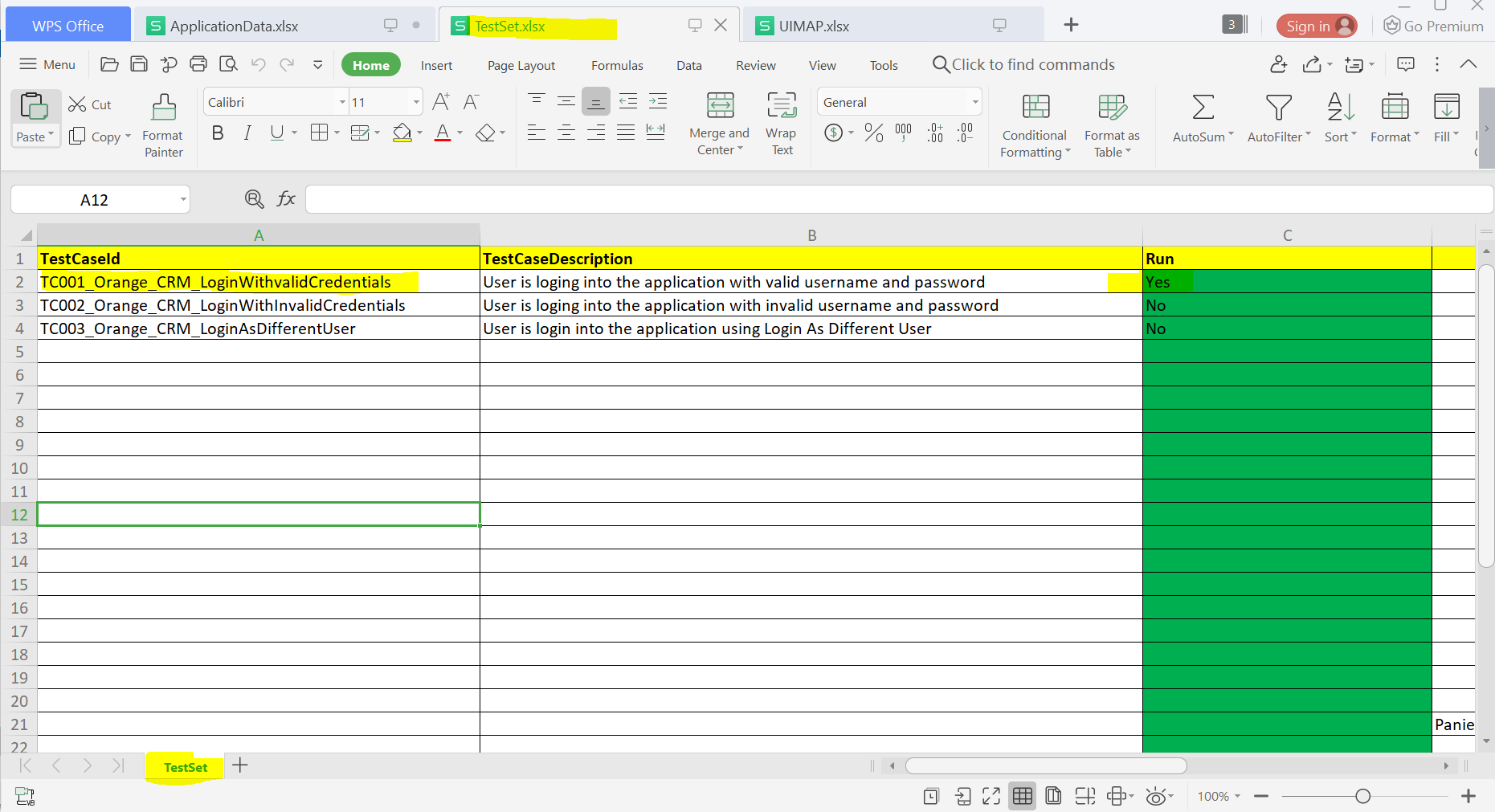
#### **UIMAP :** This file contains the webelement

#### **Sheets**

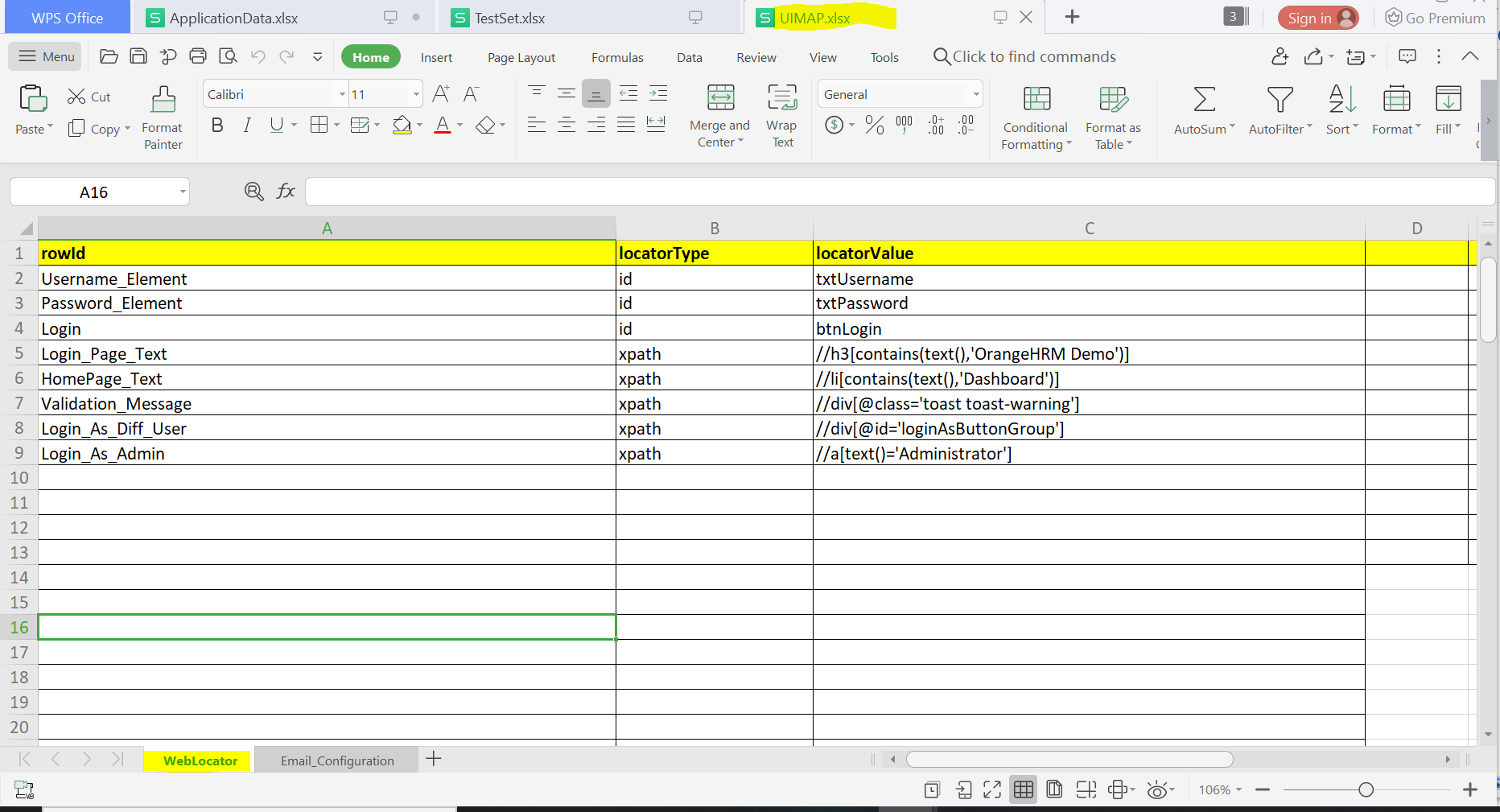
**ApplicationData :**

****

**TestSet :**

****

**UIMAP :**

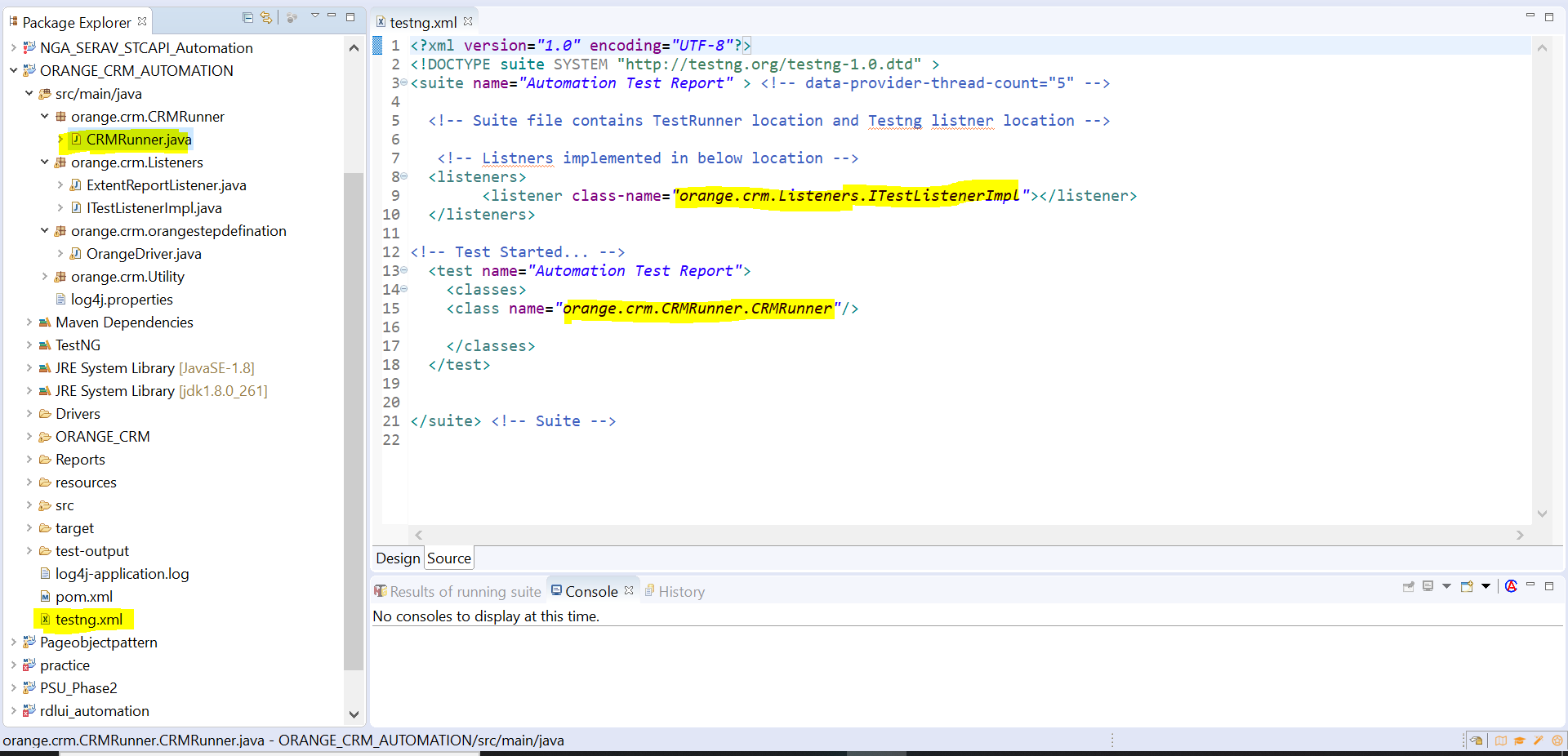
****

**Note :** Test case Id which is in ApplicationData sheet and TestSet sheet should be the same.

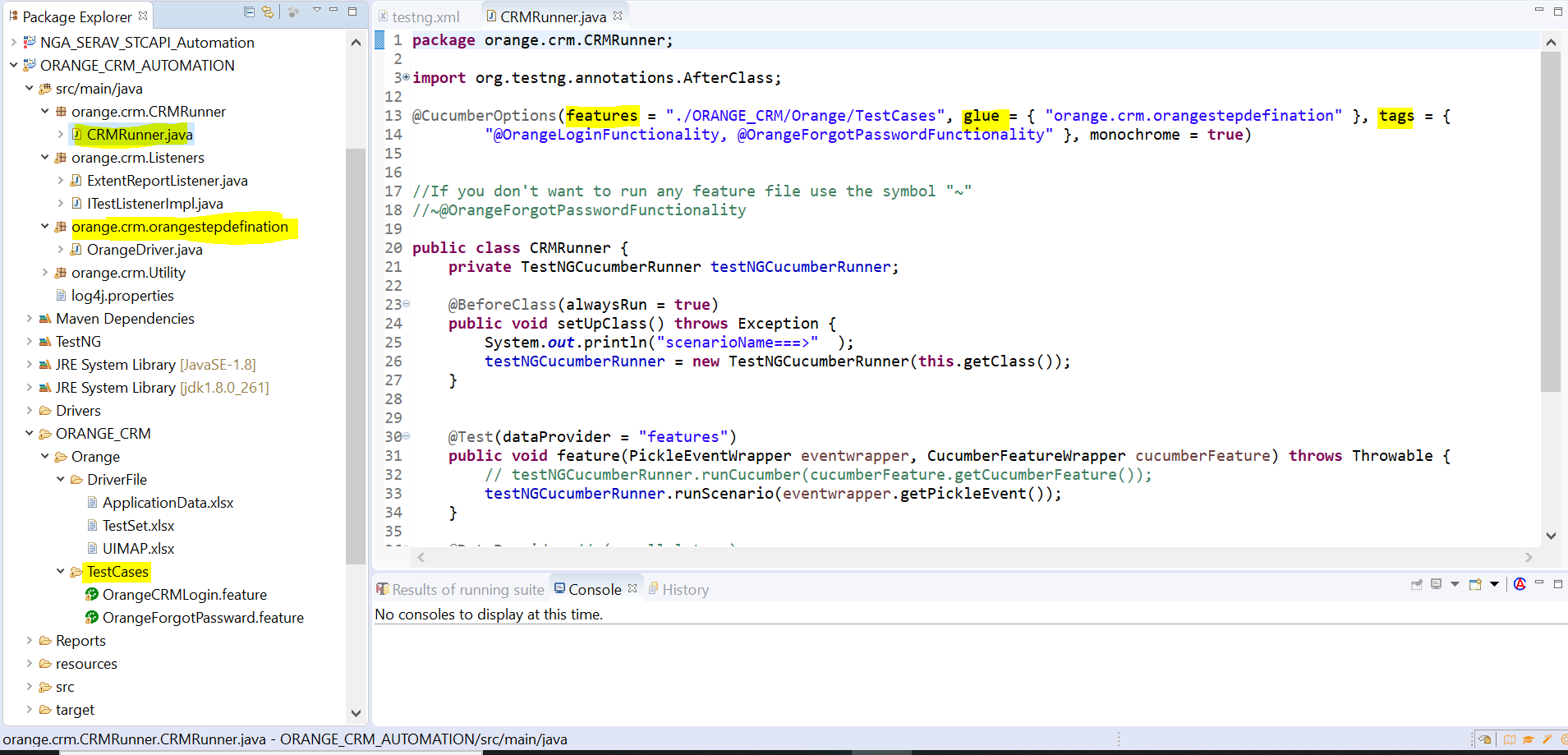
Basically, fetching the test case id, Application test data and Webelement from the 3 different sheets as mentioned above.

#### **TestNG.xml:**

In this have configured the runner class and listener

****

#### CRMRunner:

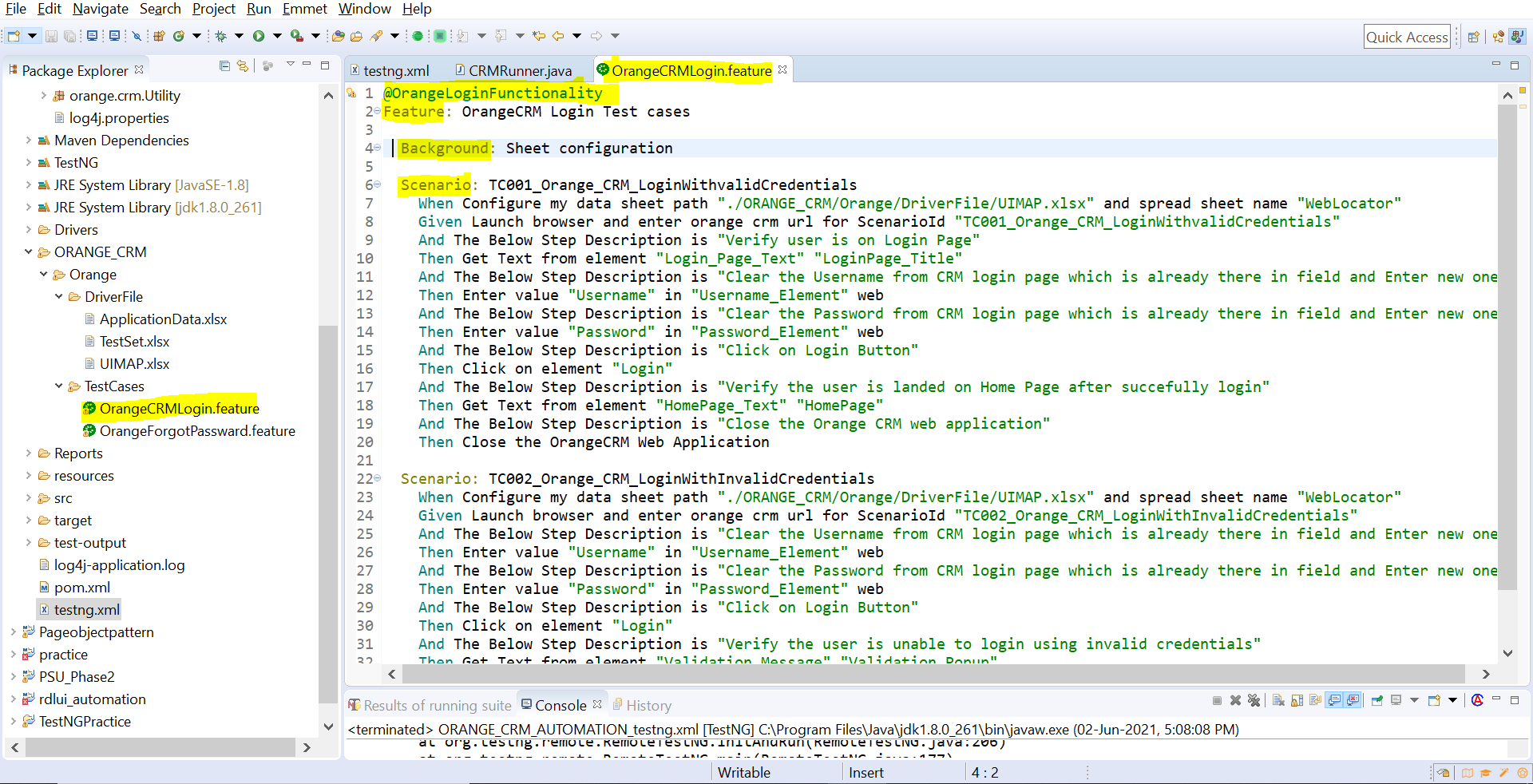
****

**features:** The features is a part of Cucumber options that describes the location and path of the features file.

**glue :** The glue is a part of Cucumber options that describes the location and path of the step definition file.

**tags:** Once you define a tag at the feature level, it ensures that all the scenarios within that feature file inherits that tag. Depending on the nature of the scenario, we can use more than one tag for the single feature. Whenever Cucumber finds an appropriate call, a specific scenario will be executed. And also we can skip the particular tag with the help symbol (~@OrangeLoginFunctionality). And also we have to pass all the tags runner which we have defined in the feature file.

#### **OrangeCRMLogin.feature**

****

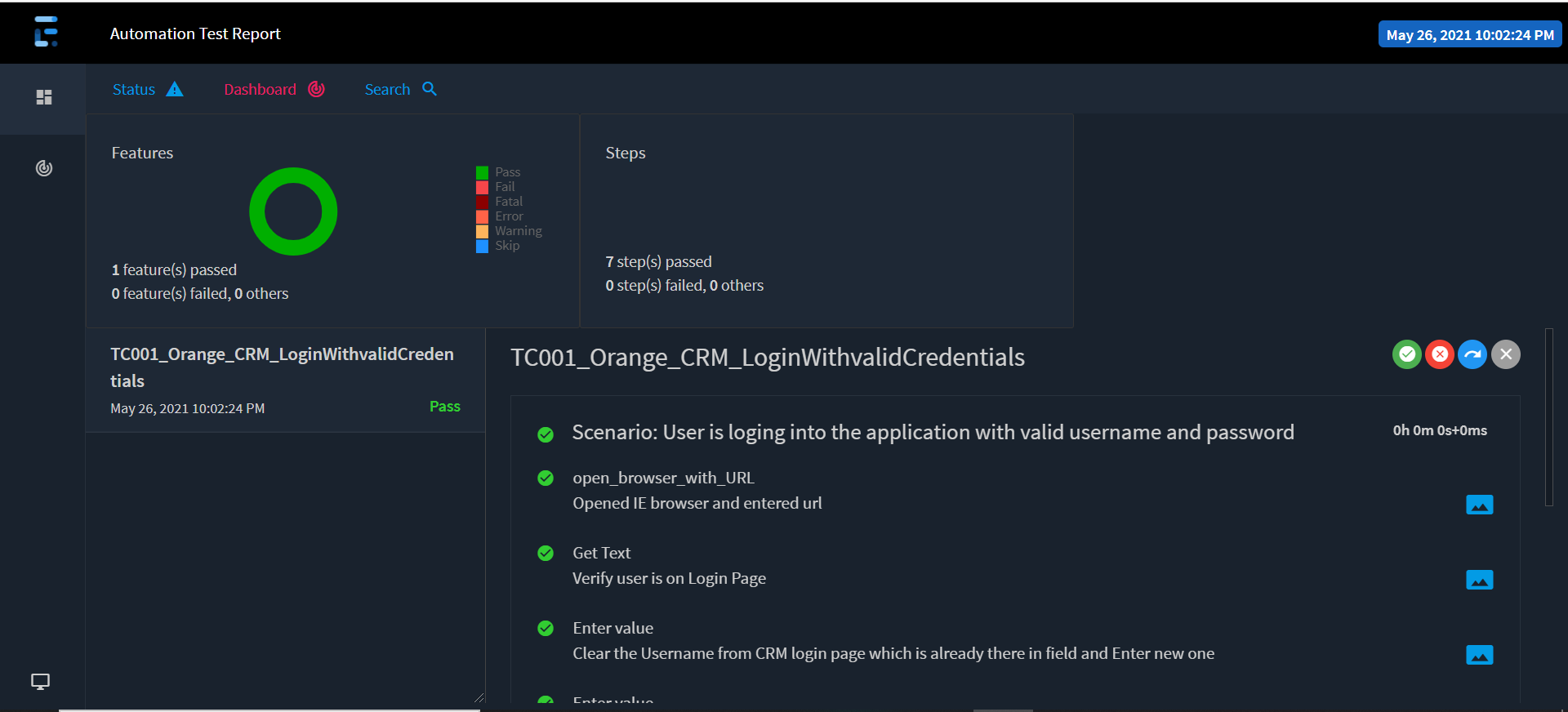
Please refer to **Important Terms used in Gherkin** to get understanding Feature, Tags, Background and Scenario.

#### **To Run Test Cases:**

**Steps:**

1. Right click on testng.xml file
2. Run As
3. TestNG Suite

#### **Extent Report**:

****

## 

## The framework can be cloned from below url

## **GitLinkURL** : <https://github.com/kantilal1/OrangeHRM>