Basic Cucumber BDD Automation

- **Gherkin:** It is business readable, domain specific language that lets you describe software’s behaviour

Keywords used in cucumber: Feature, Feature File, Background, Scenario, Scenario Outline, Given, When, Then, And, But, Example, DataTable

- **Scenarios:** In Cucumber, test cases are represented as Scenarios.

Scenarios contains Steps which are equivalent to test Steps and use the following keywords(Gherkin syntax) to denote them: Given, When, Then, But, and And (case sensitive)

- **Given**: Preconditions are mentioned in Given keyword
- **When**: The purpose of When steps is used to describe the user actions
- **Then**: The purpose of Then steps is to observe the expected output. The observations should be related to the business value/ benefit of your Feature description
- **And**: This is used for statements that are an addition to the previous Steps and represent positive statements
- **But**: This is used for statements that are an addition to the previous Steps and represent negative statements

- When we specify a business requirement, sometimes there are multiple pre-conditions, user actions, and expected outcomes.
- We are going to add one or more Scenario and will use the **And** and **But** keywords.

Scenario: Make minimum due payment
Given User is on pay credit card page
When User fills all details and select minimum amount option
And User clicks on pay button
Then Credit card confirmation is displayed

And If reference number is displayed
But error message is not displayed

- **Feature and Feature File:**
- Feature represents Business Requirement
- Feature File acts as a Test Suite which consists of all Scenarios
- In Cucumber, Feature file contains Scenario or Scenario Outline. We can simply create feature file with .feature extension.
- Scenarios belonging to specific area of Application will be grouped into one Feature file

Feature: Credit card payment
In order to test Credit Card payment functionality
As a CC User
I want to complete the payment through online

Scenario: Make minimum due payment
Given User is on pay credit card page
When User fills all details and select Minimum Amount option
And User clicks on pay button
Then Credit card confirmation is displayed

Scenario: Pay Statement Balance
Given User is on Pay credit card page
When User fills all details and select Statement Balance option
And User clicks on Pay button
Then Credit Card confirmation page is displayed

Scenario: Enter another Amount is 0
Given User is on Pay credit card page
When User fills all details and select other Amount and enter 0
And User clicks on Pay button
Then Credit Card confirmation page is not displayed
But Error message is displayed

```
Feature: Determine if String is Palindrome or not. A string
is a palindrome if it reads
  the same backwards as forwards.

Scenario: Valid Palindrome
    Given I entered string "Refer"
    When I test it for Palindrome
    Then the result should be "true"

Scenario: Invalid Palindrome
    Given I entered string "Coin"
    When I test it for Palindrome
    Then the result should be "false"

Scenario Outline: Check if String is Palindrome
  Given I entered word <wordToTest>
  When I test it for Palindrome
  Then the output should be <output>
  Examples:
    | wordToTest | output  |
    | "Refer"     | "true"  |
    | "Coin"      | "false" |
    | "Space"     | "false" |
    | "racecar"   | "true"  |

@Given("I entered word {word}")
public void iEnteredStringWord(String word) {
    testPalindrome = word;
}

@Then("the output should be {string}")
public void theOutputShouldBeResult(String output) {
    theResultShouldBe(output);
```

https://cucumber.io/docs/cucumber/step-definitions/?lang=java
https://javapointers.com/automation/cucumber/cucumber-scenario-outline-example/
https://javapointers.com/automation/cucumber/cucumber-data-tables-example-in-java/

```
#Author: your.email@your.domain.com
#Keywords Summary :
#Feature: List of scenarios.
#Scenario: Business rule through list of steps with arguments.
#Given: Some precondition step
#When: Some key actions
#Then: To observe outcomes or validation
#And,But: To enumerate more Given,When,Then steps
#Scenario Outline: List of steps for data-driven as an Examples and <placeholder>
#Examples: Container for s table
#Background: List of steps run before each of the scenarios
#""" (Doc Strings)
#| (Data Tables)
#@ (Tags/Labels):To group Scenarios
#<> (placeholder)
#""
## (Comments)
#Sample Feature Definition Template
@tag
Feature: Title of your feature
  I want to use this template for my feature file

  @tag1
  Scenario: Title of your scenario
    Given I want to write a step with precondition
    And some other precondition
    When I complete action
    And some other action
    And yet another action
    Then I validate the outcomes
    And check more outcomes

  @tag2
  Scenario Outline: Title of your scenario outline
    Given I want to write a step with <name>
    When I check for the <value> in step
    Then I verify the <status> in step

    Examples:
      | name  | value | status  |
      | name1 |     5 | success |
      | name2 |     7 | Fail    |
```

AbstractTestNGCucumberTests

AbstractTestNGCucumberTests class is used in TestNG to make TestNg file compatible with Cucumber.

For example:

```
package cucumberOptions;

import io.cucumber.testng.AbstractTestNGCucumberTests;
import io.cucumber.testng.CucumberOptions;

@CucumberOptions(features = "src/test/java/features",
glue = "stepDefinitions", monochrome = true )
public class TestNGRunner extends
AbstractTestNGCucumberTests {

}
```

monochrome = chrome -> to display results in plain English text instead of encoded 64 bit

dryRun = true -> is used to compile the cucumber code such feature file scenarios mapped to corresponding step definition or not like java compile or maven compile. It won't execute the code

Cucumber Exception:

```
io.cucumber.testng.UndefinedStepException: The step 'User is on NetBanking landing page' and 3 other step(s) are undefined.
You can implement these steps using the snippet(s) below:

@Given("User is on NetBanking landing page")
public void user_is_on_net_banking_landing_page() {
   // Write code here that turns the phrase above into concrete actions
   throw new io.cucumber.java.PendingException();
}
@When("User login into application")
public void user_login_into_application() {
   // Write code here that turns the phrase above into concrete actions
   throw new io.cucumber.java.PendingException();
}
@Then("Home Page is displayed")
public void home_page_is_displayed() {
   // Write code here that turns the phrase above into concrete actions
   throw new io.cucumber.java.PendingException();
}
@Then("Cards are displayed")
public void cards_are_displayed() {
   // Write code here that turns the phrase above into concrete actions
   throw new io.cucumber.java.PendingException();
}
```

POM.XML using testNG for cucumber project

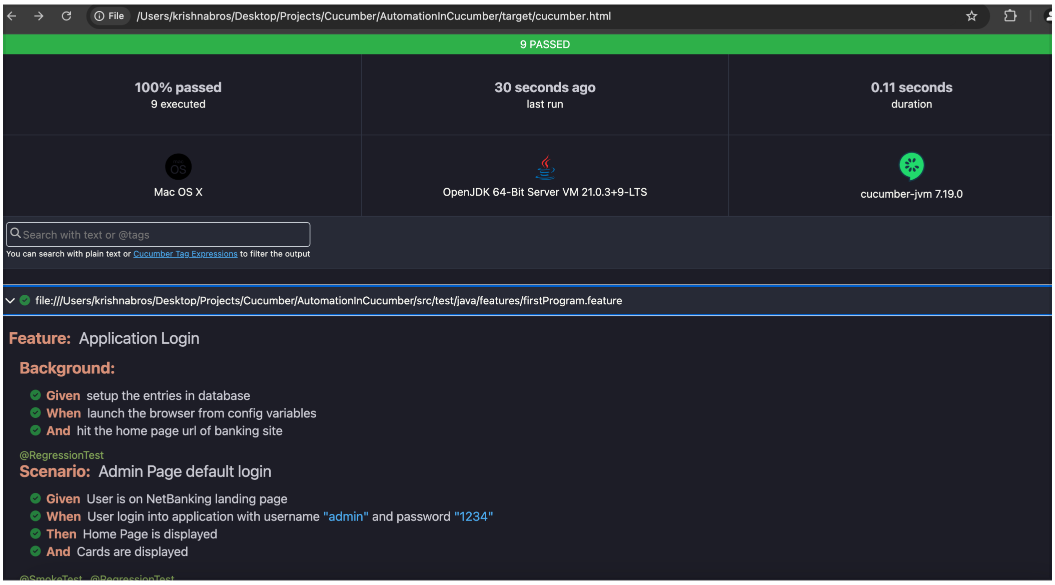
```
<dependencies>
    <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->
    <dependency>
        <groupId>io.cucumber</groupId>
        <artifactId>cucumber-java</artifactId>
        <version>7.19.0</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-testng -->
    <dependency>
        <groupId>io.cucumber</groupId>
        <artifactId>cucumber-testng</artifactId>
        <version>7.19.0</version>
    </dependency>
</dependencies>
```

Parameterization of scenario:

If we want to parameterize the test scenario, we need to use Examples. Executes scenarios for all parameters

Datatable usage

If we want to pass multiple data into single test case, we need to use DataTable. Executes only once using list of data passed


File
/Users/krishnabros/Desktop/Projects/Cucumber/AutomationInCucumber/target/cucumber.html

9 PASSED

100% passed
9 executed

30 seconds ago
last run

0.11 seconds
duration

Mac OS X

OpenJDK 64-Bit Server VM 21.0.3+9-LTS

cucumber-jvm 7.19.0

Search with text or @tags

You can search with plain text or Cucumber Tag Expressions to filter the output

file:///Users/krishnabros/Desktop/Projects/Cucumber/AutomationInCucumber/src/test/java/features/firstProgram.feature

Feature: Application Login

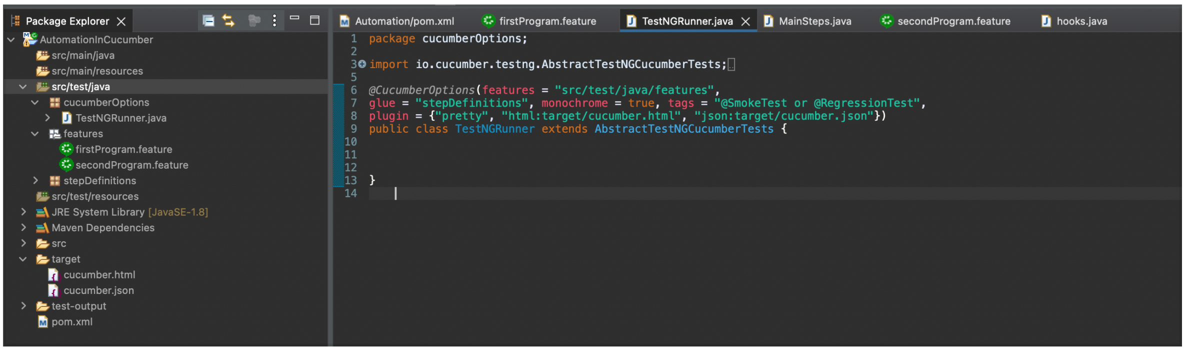
Background:

✓ **Given** setup the entries in database
✓ **When** launch the browser from config variables
✓ **And** hit the home page url of banking site

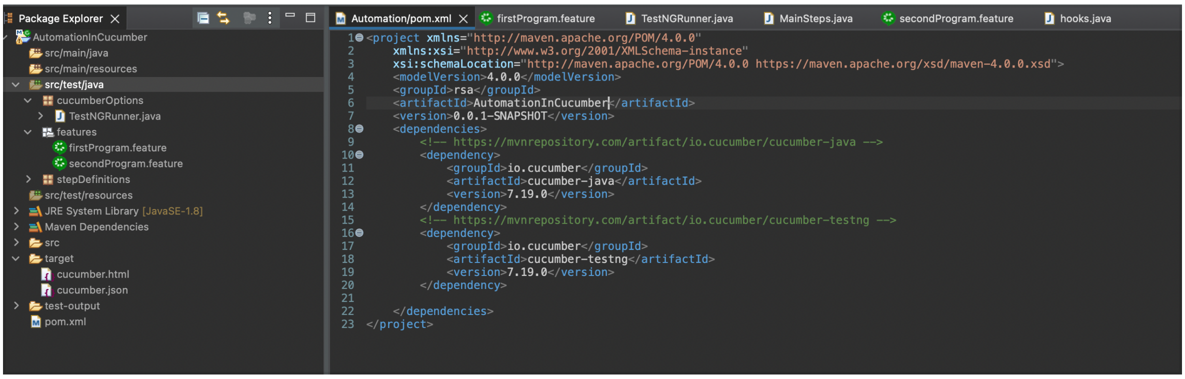
@RegressionTest
Scenario: Admin Page default login

✓ **Given** User is on NetBanking landing page
✓ **When** User login into application with username "admin" and password "1234"
✓ **Then** Home Page is displayed
✓ **And** Cards are displayed

@SmokeTest @RegressionTest



```java
package cucumberOptions;

import io.cucumber.testng.AbstractTestNGCucumberTests;

@CucumberOptions(features = "src/test/java/features",
glue = "stepDefinitions", monochrome = true, tags = "@SmokeTest or @RegressionTest",
plugin = {"pretty", "html:target/cucumber.html", "json:target/cucumber.json"})
public class TestNGRunner extends AbstractTestNGCucumberTests {



}

```



```xml
<project xmlns="http://maven.apache.org/POM/4.0.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <groupId>rsa</groupId>
    <artifactId>AutomationInCucumber</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <dependencies>
        <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->
        <dependency>
            <groupId>io.cucumber</groupId>
            <artifactId>cucumber-java</artifactId>
            <version>7.19.0</version>
        </dependency>
        <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-testng -->
        <dependency>
            <groupId>io.cucumber</groupId>
            <artifactId>cucumber-testng</artifactId>
            <version>7.19.0</version>
        </dependency>

    </dependencies>
</project>
```

10 Common problems which needs to be addressed in Cucumber Framework Design

1. How to write and maintain reusable code across the framework?
2. How to data drive the data into testcase?
3. How to parameterize the test cases with multiple set of data?
4. How to achieve 100% test coverage through framework?

5. How to run all the tests on single click?
6. How to control the execution of tests to select and run only few based on needs?
7. How to implement logic to separate test code from common Post and prerequisites of test?
8. How to generate HTML reports and Junit reports?
9. How to develop maven framework with existing code?

Sample Selenium Cucumber Project

In case If you are looking to implement the cucumber Framework for Selenium Project, Check out below github link.. There is one sample project available which you can download and play around..
https://github.com/selenium-cucumber/selenium-cucumber-java-maven-example

Sample Appium Cucumber Project

In case If you are looking to implement the cucumber Framework for Selenium Project, Check out below github link.. There is one sample project available which you can download and play around.. All the Best
https://github.com/mubbashir/appium-java-cucumber-gradle

Cucumber Selenium Framework from Scratch

- Build Maven Project with all the framework dependencies
- Develop end to end selenium web scenarios in the feature file
- Implement TestNG/ Junit Runner to run the framework
- Understand the single responsibility principle to distribute the implementation into multiple step definitions
- Importance of Dependency injection to share the data between the Step file
- Understand the Page object pattern to drive the locators from page files
- Implement Factory Design Pattern by writing PageObjectManager class to create objects of all PO classes
- How driver can be configured and distributed across the files in the framework
- How to run Cucumber scenarios in parallel mode using Cucumber TestNG runner
- Build Test utilities for reusable Selenium methods
- Implement Cucumber Hooks to capture screenshots on test failures
- Implement Cucumber tags to run selected tests in framework
- Parameterizing the test data to run the scenarios with multiple data sets
- Creating HTML and Extent reports to run Cucumber Selenium Tests
- Running tests in parallel mode and generate Extent reports with

screenshots attached for failed scenarios

- Run the cucumber tests using maven & command line options
- Integrate the framework to CI/ CD Jenkins and schedule the jobs on regular time intervals
- Create Parameterized Jenkins job to dynamically send the global properties at run time of job execution