5.12 Cucumber Data Tables

****

This section will guide you on:

* What is Cucumber Data Table

**Development Environment:**

* JRE: OpenJDK Runtime Environment 11.0.2
* Eclipse IDE for Enterprise Java Developers v2019-03 (4.11.0)
* TestNG
* Selenium jars
* Cucumber jars

This guide has two subsections, namely:

## 5.12.1 Data table in Cucumber

5.12.2 Pushing code to your GitHub repositories

**Step 5.12.1:** Data Tables

* Data table is a set of inputs to be provided for a single tag. This tag can be GIVEN, WHEN, or THEN.
* Create a package named **dataTable** under **src/test/java**
* Create a Feature file.
* Create a feature file, named as **dataTable**.feature inside the package dataTable (see section scenario outline for more detailed steps).
* Write the following data table

|  |
| --- |
| Feature �' Data table  Verify that the new user registration is unsuccessful after passing incorrect inputs  Scenario:  Given I am on the new user registration page  When I enter invalid data on the page  | Fields | Values |  | First Name | Tom |  | Last Name | Kenny |  | Email Address | someone@someone.com |  | Re-enter Email Address | someone@someone.com |  | Password | Password1 |  | Birthdate | 01 | |

* + Save the file.
* Create a step definition file.
* Create the step definition file named as ‘dataTable.java’ inside the package dataTable (see section scenario outline for more detailed steps).
* Write the following code:

|  |
| --- |
| package dataTable;  **import** java.util.List;  **import** org.openqa.selenium.By;  **import** org.openqa.selenium.WebDriver;  **import** org.openqa.selenium.WebElement;  **import** org.openqa.selenium.firefox.FirefoxDriver;  **import** org.openqa.selenium.support.ui.Select;  **import** cucumber.annotation.en.Given;  **import** cucumber.annotation.en.Then;  **import** cucumber.annotation.en.When;  **import** cucumber.table.DataTable;  public class stepdefinition {  WebDriver driver = null;    @Given("^I am on new user registration page$")  public **void** goToFacebook() {  //Intiate web browser instance. driver = new FirefoxDriver();  driver.navigate().to("https://www.facebook.com/");  }    @When("^I enter invalid data on the page$")  public **void** enterData(DataTable table){  //Initialize data table  List<list> data = table.raw();  System.out.println(data.get(1).get(1));    //Enter data  driver.findElement(By.name("firstname")).sendKeys(data.get(1).get(1));  driver.findElement(By.name("lastname")).sendKeys(data.get(2).get(1));  driver.findElement(By.name("reg\_email\_\_")).sendKeys(data.get(3).get(1));  driver.findElement(By.name("reg\_email\_confirmation\_\_")).  sendKeys(data.get(4).get(1));  driver.findElement(By.name("reg\_passwd\_\_")).sendKeys(data.get(5).get(1));    Select dropdownB = new Select(driver.findElement(By.name("birthday\_day")));  dropdownB.selectByValue("15");    Select dropdownM = new Select(driver.findElement(By.name("birthday\_month")));  dropdownM.selectByValue("6");    Select dropdownY = new Select(driver.findElement(By.name("birthday\_year")));  dropdownY.selectByValue("1990");    driver.findElement(By.className("\_58mt")).click();  // Click submit button driver.findElement(By.name("websubmit")).click();  }    @Then("^User registration should be unsuccessful$")  public **void** User\_registration\_should\_be\_unsuccessful() {  **if**(driver.getCurrentUrl().equalsIgnoreCase("https://www.facebook.com/")){  System.out.println("Test Pass");  } **else** {  System.out.println("Test Failed");  }  driver.close();  }  } |

* Save the file
* Create runner class
* Create runner class named as runTest.java inside the package.
* Write the following code.

|  |
| --- |
| package dataTable;  **import** org.junit.runner.RunWith;  **import** cucumber.junit.Cucumber;  @RunWith(Cucumber.class)  @Cucumber.Options(format = {"pretty", "html:target/cucumber"})    public class runTest { } |

* Save the file.
* Run the test using the option
  + Select runTest.java file from the package explorer.
  + Right-click and select the option, Run as.
  + Select JUnit test.

**Step 5.12.2:** Pushing code to your GitHub repositories

Open your command prompt and navigate to the folder where you have created your files

cd <folder path>

Initialize your repository using the following command:

git init

Add all the files to your git repository using the following command:

git add . 

Commit the changes using the following command:

git commit . -m “Changes have been committed.”

Push the files to the folder you initially created using the following command:

git push -u origin master