3 Gherkin: Given, When, Then, and Background Steps****

This section will guide you to:

* Understand the Gherkin keyword formats

**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:

5.3.1 Demonstrating Gherkin keyword/scenarios explanation

5.3.2 Pushing the code to your GitHub repositories

**Step 5.3.1:** DemonstratingGherkin keyword/scenarios explanation

Each scenario should follow **given**, **when**, and **then** format.

* Given: n Specifies the preconditions. It is basically a known state.
* When: This is used when some action is to be performed. Then: The expected outcome or result should be placed here. For Instance, verify if the login and page navigation is successful.
* Background: Whenever any step is required to be performed in each scenario, then those steps are to be placed in the Background. For Instance, If user needs to clear the database before each scenario, then those steps can be put in the background.
* And: Used to combine two or more actions of the same type.
* But: Signifies logical OR condition between any two statements. OR can be used in conjunction with GIVEN, WHEN ,and THEN statement.

|  |
| --- |
| *Feature: Add to Cart*  *This feature will test functionality of adding different products to the User basket from different flow*  *Background: User is Logged In*  *Scenario: Search a product and add the first result/product to the User basket*  *Given User searched for Lenovo Laptop*  *When Add the first laptop that appears in the search result to the basket*  *Then User basket should display with 1 item* |

**Step 5.3.2:** Pushing the 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