

# Lesson Objectives



To understand the following topics:

Introduction to Gherkin Language
Generating Behavior test cases
Running test cases
Creating tags
Categorizing tests based on tags



# Introduction to Gherkin Language



- Gherkin is a language, that is used to write **Features, Background, Scenarios, and Steps**.
- Gherkin helps us to write concrete requirements.
- Gherkin files are plain text Files and have the extension .feature.
- · Gherkin syntax is simple and readable
- · Each line that is not blank has to start with a Gherkin keyword, followed by any text you like.
- Gherkin is based on <u>TreeTop Grammar</u> which exists in 37+ languages thus you can write your gherkin in 37+ spoken languages.

Other Keywords: Scenario Outline Examples """ (Doc Strings) | (Data Tables) @ (Tags) # (Comments)

# Example



**Feature:** Search feature for users

This feature is very important because it will allow users to

filter products

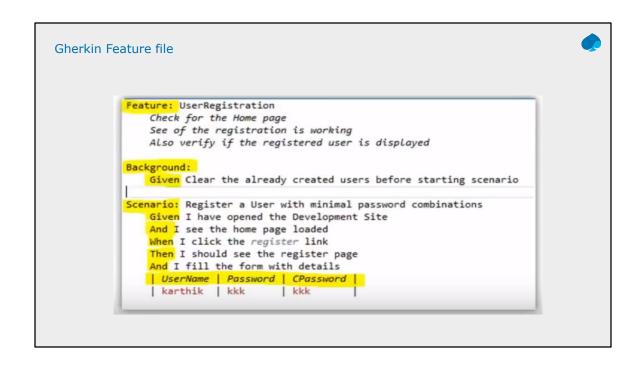
**Scenario:** When a user searches, without spelling mistake, for a product name present in inventory. All the products with similar name should be displayed

**Given** User is on the main page of www.myshopingsite.com

**When** User searches for laptops

**Then** search page should be updated with the lists of laptops

# Rules for using Gherkin Single Gherkin source file should have a description of the single feature. the source file should have the .feature extension.



### Feature



Feature acts as a heading for a Scenario

Feature has a list of steps to be performed in particular scenario as a whole

The **Feature** keyword is used to describe a software feature, and to group the related scenarios.

A Feature has three basic elements

- The keyword Feature.
- The name of the feature, provided on the same line as the Feature keyword.
- · An optional description that can span multiple lines

Feature files are the base files to hold all the scenarios

Description in feature holds all the text between the line containing the keyword Feature, and a line that starts with Scenario, Background, or Scenario Outline.

The parser will not interpret any line under feature, since they are just information to the user (Business Analyst, Quality Analyst or Developer)

# Background



Backgrounds are used with the scenarios which have common functionalities

They will be executed before every scenario and perform the operation specified

For Example:

In scenarios like checking user settings, profile settings, user information, the common operations are

Launching the application

Login into the application

# Background:

Given Clear the already created users before starting scenario

### Feature



- Based on the example given anybody can understand the working of the test and what it is intending to do.
- It gives an unexpected powerful impact by enabling people to visualize the system even before it has been built.
- Any of the business user would read and understand the test and would be able to give the feedback that whether it reflects their understanding of what the system should do, and any other scenario that needs to be considered.

# Summary



- In this lesson, you have learnt:

  Gherkhin Language and its syntax
  Creating feature file and generating test cases



Q.4.getTitle()
Q.5.True
Q.6. sendKeys(Review Question



1. What will be extension of a feature file?

.exe a. .obj .feature b. c. d. .sfj

