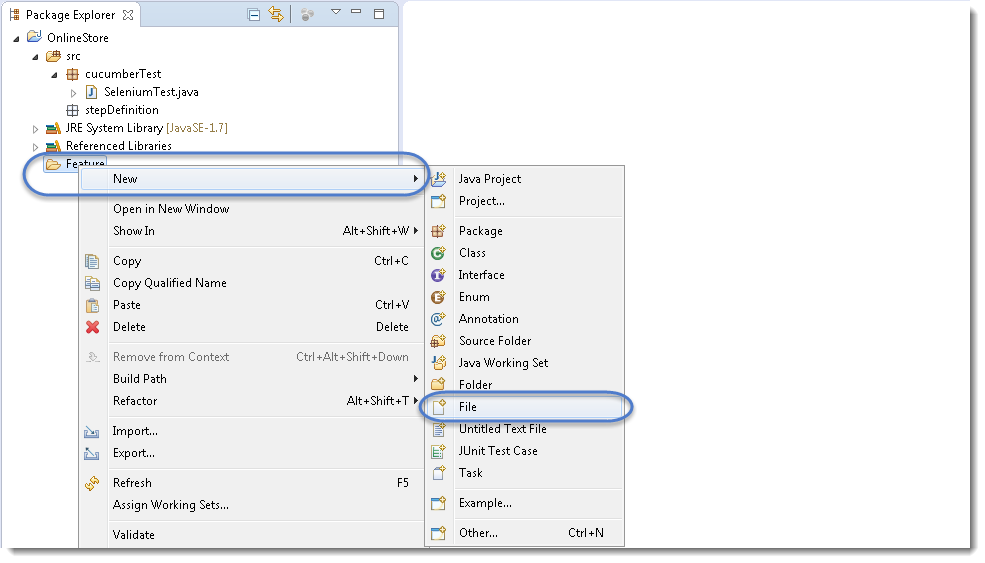
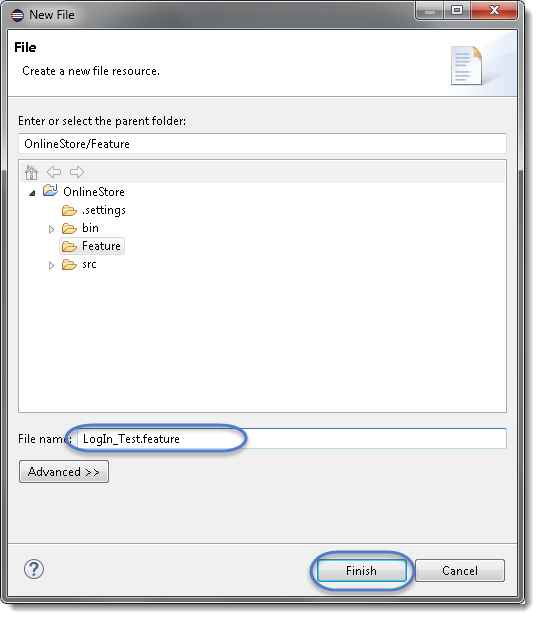
**Cucumber Feature File**

A ***Feature File*** is an entry point to the *Cucumber* tests. This is a file where you will describe your tests in Descriptive language (Like English). It is an essential part of Cucumber, as it serves as an automation test script as well as live documents. A feature file can contain a scenario or can contain many scenarios in a single feature file but it usually contains a list of scenarios. Let’s create one such file.

1) On the ***Feature*** folder *Right click* and select ***New > File***



2) In order for Cucumber to automatically detect the stories (or ***features***, as they’re known in *Cucumber*), you need to make sure that they carry the ‘***.feature***‘ file extension. For example, in this case, I’ve named my user story ‘***LogIn\_Test.feature***‘. Every ‘*.feature*‘ file conventionally consists of a single feature.



***Note****: In case you get a pop up from Eclipse which suggest you to install the better Editor for BDD files, please go ahead and install that. At the botttom of the chapter, steps to install the better editor is given.*

3) Write the first cucumber script. In BDD terms the scenario would look like the following.

***Cucumber Test Script***

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | Feature: Login Action    Scenario: Successful Login with Valid Credentials  Given User is on Home Page  When User Navigate to LogIn Page  And User enters UserName and Password  Then Message displayed Login Successfully    Scenario: Successful LogOut  When User LogOut from the Application  Then Message displayed LogOut Successfully |

***Note:****This is a simple test in Cucumber. Don’t worry about the syntax if you don’t understand it. Ideally you should be able to understand the intent of the test just by reading a test in feature file. We will discuss this in more details in next chapter.*

***Keywords***

Now moving forward we have just defined a test. You will notice colored part of the tests (***Feature, Scenario, Given, When, And and Then***). These are keywords defined by ***Gherkin***. *Gherkin* has more keywords and we will discuss those in following tutorials. But to start off we can quickly explain some of the keywords in one line. Note this is not complete listing of Keywords:

***Feature: Defines what feature you will be testing in the tests below***

***Given: Tells the pre-condition of the test***

***And: Defines additional conditions of the test***

***Then: States the post condition. You can say that it is expected result of the test****.*

***Gherkin***

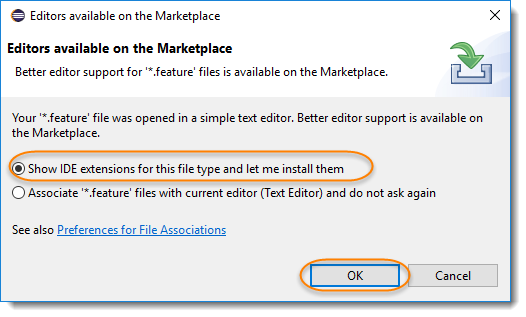
A language above is called ***Gherkin*** and it implements the principles of ***Business readable domain specific language(BRDSL)***. Domain specific language gives you the ability to describe your application behavior without getting into details of implementation. What does that mean? If we go back to our tutorial in [***TDD***](http://toolsqa.com/cucumber/test-driven-development-tdd/) we saw that we wrote test code before writing any application code. In a way we described what is the expected behavior of our application in terms of tests. On *TDD* those tests were pure Java tests, in your case those might be a C++ or C# tests. But the basic idea is that those are core technical tests.

If we now come back to [***BDD/BRDSL***](http://toolsqa.com/cucumber/behavior-driven-development/) we will see that we are able to describe tests in a more readable format. In the above test it’s quite clear and evident, just by reading, what test would do. At the same time of being a test it also documents the behavior of application. This is the true power of *BDD/BRDSL* and it will become the power of cucumber eventually because cucumber works on the same principles.

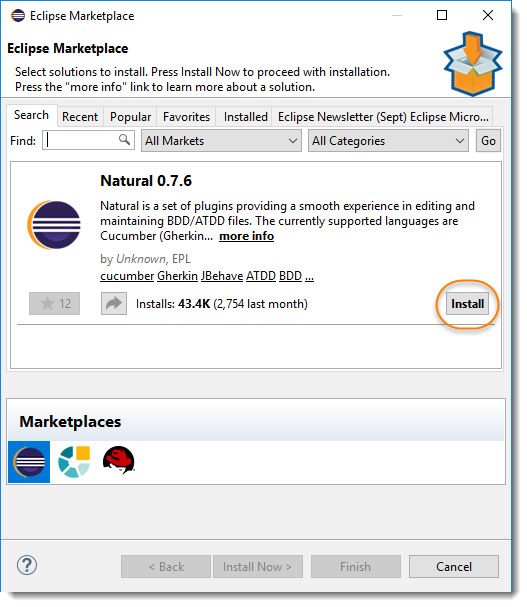
***Steps to install the Natural Eclipse Editor for Gherkin***

You get this option automatically when try to create a new file with .feature ext. But if you do not get that one, you can anytime go to Eclipse Marketplace and look for the same to install it.

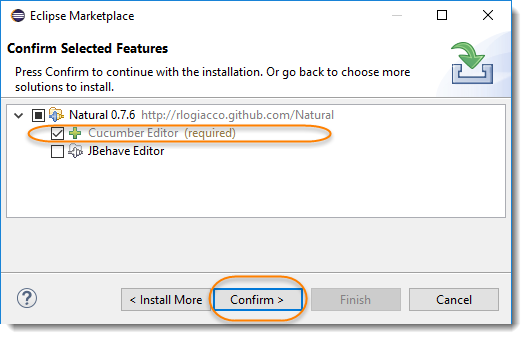
1)Just select the first option of ***Show IDE extensions*** if it is not pre-selected and click OK.



2) Natural is the name of the plugin, so this can also be found [***Eclipse Marketplace***](https://marketplace.eclipse.org/content/natural#group-details). Just click Install.



3) This will give you an option to select, whether you like to use it for Cucumber or JBehave(Another BDD Framework). Go for Cucumber.



4) Last step is to accept the Terms and Conditions.

