



無料電子ブック

学習

cucumber

Free unaffiliated eBook created from
Stack Overflow contributors.

#cucumber

.....	1
1:	2
.....	2
Examples.....	2
.....	2
Ruby.....	3
Ruby.....	4
2: IntelliJCucumber	6
.....	6
.....	6
Examples.....	6
.....	6
IntelliJ Cucumber for Java PluginMac.....	7
3: Maven_cucumberpom.xml	13
.....	13
Examples.....	13
pom.xml.....	13
4:	15
.....	15
.....	15
Examples.....	15
.....	15
.....	16
.....	17
.....	18
.....	18
.....	20
5:	21
.....	21
Examples.....	21
Ruby.....	21

6:	22
.....	22
.....	22
Examples.....	22
.....	22
.....	22
.....	22
.....	24

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [cucumber](#)

It is an unofficial and free cucumber ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official cucumber.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

1: キュウリをめる

キュウリについて

キュウリはソフトウェアのなをするツールです。「フィーチャー」とばれるは、されたでかれています。Cucumberは、ステップをCucumberのでサポートされているプログラミングでかれた「ステップ」にマッピングすることで、をします。キュウリは、[Rubyオリジナル](#)、[Java](#)、[Javascriptなどのくのプログラミングでされています](#)。また、くののにされています。

Cucumberは、Behavior-Driven DevelopmentBDDとばれるアジャイルをサポートするためにされました。BDDでは、テストなどのプログラマのからではなく、ユーザーのからソフトウェアのをするけれテストをすることによって、をします。キュウリのは、これらのけれとしてつ。

に、キュウリのはがなドキュメントであり、なテストスイートでもあり、ドキュメントとテストはにしています。キュウリは、プログラマーのとやテストについてするのにです。また、プログラマーは、プログラミングのなレベルでテストをくことができます。

Cucumberは、SeleniumやPhantomJSなどのブラウザドライバをして、Webアプリケーションをおよびテストするためにもよくされます。ただし、なソフトウェアや、キュウリのでサポートされているプログラミングからやをすることができます。

そののドキュメント

は<https://cucumber.io/docs>にあります。Cucumberのをするキュウリのからされたドキュメントは

- JavaScript <https://relishapp.com/cucumber/cucumber-js/docs>
- Ruby <https://relishapp.com/cucumber/cucumber/docs>

<https://relishapp.com/explore>にはキュウリのツールやサンプルがまれていますますが、ながら、キュウリJVMはありません。

このトピック

このトピックでは、にキュウリのをするいくつかのをします。そののセクションでは、インストール、コマンドラインとIDEのい、、ステップなどのなをします。

Examples

キュウリの

Cucumberは[Gherkin](#)をして、されたでのソフトウェアのをします。

そのようなキュウリは、ユースケースのシナリオのようなものとなるテストフレームワークなが、システムのマニュアルのみ、ではないではないので。なは、のシステムのをするために、

キュウリのをすることができるといによるものです。

キュウリのドキュメントスイートはでされている`Features`ガーキンでかれており、のファイルでホストされて、それぞれがのソフトウェアのを。これらのファイルをディレクトリにすると、フィーチャをグループしてできます。

- /
 - ◦
 - `atm.feature`
 - ローン。フィーチャ
- トレーディング/
 - `portfolio.feature`
 - `intraday.feature`
- /
 - ◦
 - `accounting.feature`

`Feature`は、オプションのされていないのにのセクションと、それぞれがまたはユースケースをす1つまたはの`Scenarios`でされるプレーンテキストファイルです。

```
Feature: Documentation
As a StackOverflow user or visitor
I want to access the documentation section

  Scenario: search documentation on Stack Overflow
    Given I am on StackOverflow
    And I go to the Documentation section
    When I search for "cucumber"
    And I follow the link to "cucumber"
    Then I should see documentation for "cucumber"
```

Given、*When*、*And*、*But*、*Then*のは`Step`とされます。どのステップもにかかわらずそれらののいずれかでめることができますが、なりもなでそれらをするのがです。

フィーチャは、`Tags`、それをさらにするための`Feature`または`Scenario`にくことができるによってすることもできます。

フィーチャのは、さまざまなJava、Ruby、Scala、C / C ++でできるグルーコードをしてされます。`Step`はグルーコードとされ、`Step Definitions`に`StepDef`とされます。

`Step`は、する`Step Definition` 1つしかありません。

`Feature`がされると、`Scenario`するそれぞれがされます。つまり、`Scenario Step`する`StepDef`がされます。

なRubyのインストール

RubyでするためにCucumberをインストールするには、のコマンドをします。

```
gem install cucumber
```

わりに、もしあなたがbundlerをっているなら、Gemfileにのをすることができます

```
gem 'cucumber'
```

そしてバンドルラーをらせる

```
bundle install
```

[はこれがのトピックであるとう、インストール。はそのトピックをし、このをそこにコピーしました。そのトピックがされたら、はそこにしてコピーをします。]

Rubyにおけるキュウリのステップ

features / step_definitions / documentation.rbで

```
When /^I go to the "([^"]+)" documentation$/ do |section|
  path_part =
    case section
    when "Documentation"
      "documentation"
    else
      raise "Unknown documentation section: #{section}"
    end
  visit "/documentation/#{path_part}/topics"
end

Then /^I should see the "([^"]+) documentation"$/ do |section|
  expect(page).to have_css('h2.doctag_title a', text: section)
end
```

これらのは、Webアプリケーションをします。らはまだであるり、できるだけです。

ステップはGherkinキーワードであります。Gherkinキーワードは、ステップファイルでCucumberにステップをするメソッドです。ステップメソッドは、シナリオのとする、シナリオがするにするときにされるブロックをります。のキャプチャ・グループは、ブロック・パラメータとしてブロックにされます。

Whenステップには、がめるからページ「ドキュメント」へのなインラインのがあります。リアルキュウリスイートは、このロジックをのでします。visitはCapybaraによってされています。カピバラはににされていますが、キュウリをするはありません。visitは、されたURLにアクセスするためにCapybaraがするブラウザにします。

Thenステップは、ページのをどのようにテストできるかをします。expect / to RSpecをび、キュウリではありませんが、ににそれをによりされます。have_cssはCapybaraによってされています。されたCSSセクタが、されたテキストをむページのとすることがされます。ブラウザのがした、このはすることにしてください。

そののについては、「ステップ」のトピックをしてください。

オンラインでキュウリをめるをむ <https://riptutorial.com/ja/cucumber/topic/4875/キュウリをめる>

2: IntelliJにcucumberプラグインをインストールする

き

IntelliJ IDEAのCucumberプラグインは、CucumberフレームワークをしてIntelliJプロジェクトのGherkinファイルをするためのなIDEをします。プラグインは、Java、Scala、Groovyのできます。

Cucumber for Java IntelliJプラグインは、CucumberをってにするためのIDEをします。

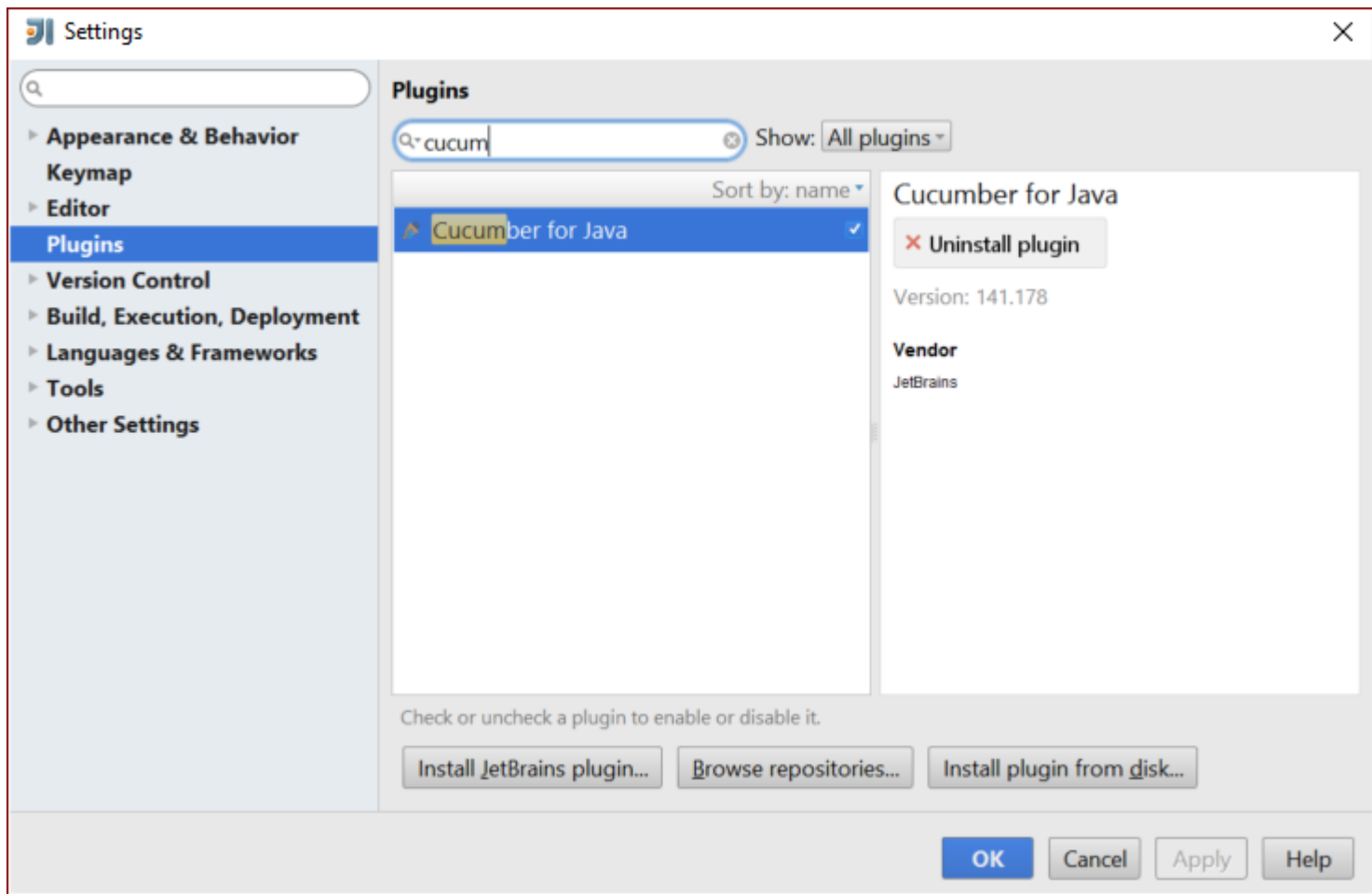
- されていないステップのGherkinステップグループ
- Gherkinのステップコードの
- ステップ・ツー・グループ・メソッド・コード・ジャンプ
- Gherkinののハイライト

そののなをえています。

Examples

キュウリプラグインをインストールする

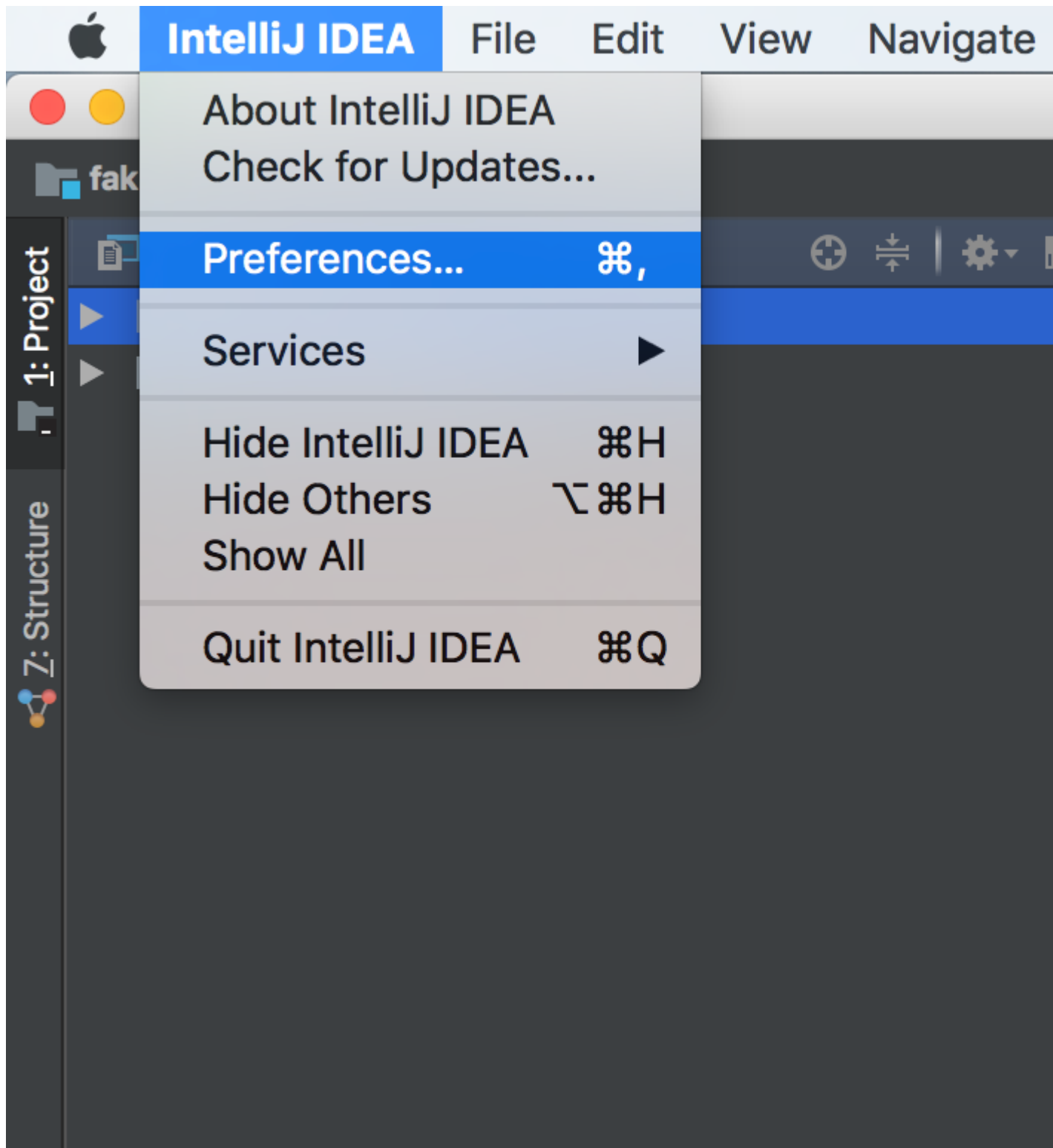
[ファイル] -> [] -> ペインのプラグインをクリック -> キュウリを -> プラグインをインストール



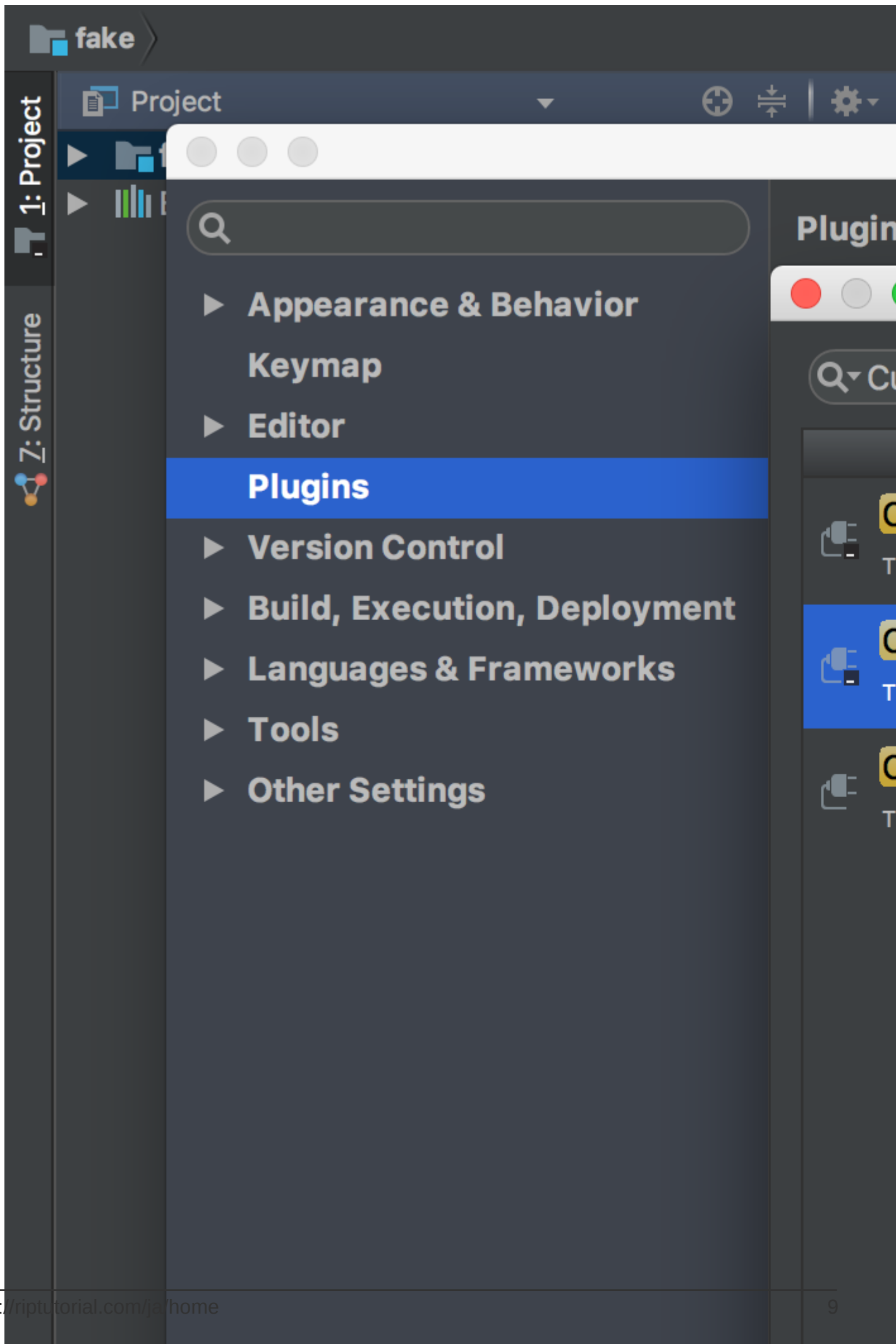
IntelliJ Cucumber for Java PluginをインストールするMac

MacでIntelliJのCucumber for Javaプラグインをインストールするには、

1. IntelliJ IDEAをします。
2. バーの "IntelliJ IDEA"タブをクリックします。



3. 「」をクリックします。
4. Preferences / Settingsで、のペインの "Plugins"をクリックします。
5. [リポジトリの]ボタンをクリックすると、しいウィンドウがされます。
6. バーで「キュウリ」をします。



7. "Cucumber for Java"プラグインをインストールします。
8. プラグインがになるようにIDEをします。 Cucumber for Javaがインストールされました。



IntelliJ IDEA

File

Edit

View

Navigate

fake

1: Project

7: Structure



▶ Appearance & Behavior

Keymap

▶ Editor

Plugins

▶ Version Control

▼ Build, Execution, Deployment

▶ Build Tools

▼ Compiler

Excludes

Java Compiler

Annotation Processors

Validation

RMI Compiler

Groovy Compiler

Android Compilers

Kotlin Compiler

▶ Debugger

Coverage

Plugin



<https://riptutorial.com/ja/cucumber/topic/8356/intellij>にcucumberプラグインをインストールする

3: Maven_cucumberプロジェクトのpom.xml。

き

のプロジェクトオブジェクトモデルはテンプレートpom.xmlです。キュウリプロジェクトをつ
mavenをするは、のをテンプレートとしてできます

Examples

pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
```

4.0.0

```
<groupId>Project name</groupId>
<artifactId>MulitClients</artifactId>
<version>1.0-SNAPSHOT</version>

<dependencies>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>4.11</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>info.cukes</groupId>
    <artifactId>cucumber-core</artifactId>
    <version>1.2.0</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>info.cukes</groupId>
    <artifactId>cucumber-testng</artifactId>
    <version>1.2.0</version>
  </dependency>
  <dependency>
    <groupId>info.cukes</groupId>
    <artifactId>cucumber-junit</artifactId>
    <version>1.2.0</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>info.cukes</groupId>
    <artifactId>cucumber-java</artifactId>
    <version>1.2.0</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>info.cukes</groupId>
    <artifactId>gherkin</artifactId>
```

```
        <version>2.12.2</version>
        <scope>test</scope>
    </dependency>
    <dependency>
        <groupId>org.seleniumhq.selenium</groupId>
        <artifactId>selenium-java</artifactId>
        <version>2.53.0</version>
        <scope>test</scope>
    </dependency>
    <dependency>
        <groupId>org.seleniumhq.selenium</groupId>
        <artifactId>selenium-firefox-driver</artifactId>
        <version>2.53.0</version>
        <scope>test</scope>
    </dependency>
    <dependency>
        <groupId>org.seleniumhq.selenium</groupId>
        <artifactId>selenium-htmlunit-driver</artifactId>
        <version>2.53.0</version>
        <scope>test</scope>
    </dependency>
    <dependency>
        <groupId>org.yaml</groupId>
        <artifactId>snakeyaml</artifactId>
        <version>1.13</version>
    </dependency>
    <dependency>
        <groupId>com.esotericsoftware.yamlbeans</groupId>
        <artifactId>yamlbeans</artifactId>
        <version>1.06</version>
    </dependency>
</dependencies>
```

オンラインでMaven_cucumberプロジェクトのpom.xml。をむ

<https://riptutorial.com/ja/cucumber/topic/8331/maven-cucumberプロジェクトのpom-xml->

4: ガーキンの

き

Gherkinは、テストのとテストのためのビジネスでめるです。これはキュウリによってされ、にのツールとしてします。

- このキーワードは、テストまたはされているのなまたはです。
- このキーワードは、のすべてのシナリオのにされるステップをします。
- シナリオこのキーワードは、をテストするのシナリオのまたはなをします。
- シナリオこのキーワードは、でまれたでにされたにリストされているすべてののについてシナリオがNされることをします。
- このキーワードは、シナリオのにされるのリストをします。
- えられたこのキーワードは、えられたステップ、またはするにされるをします。アレンジ、アクト、アサートのパラダイムでは、えられた "アレンジ" をします。
- Whenこのキーワードは、whenステップ、またはアサートされるべきをします。Arrange、Act、Assertのパラダイムでは、えられた "Act" をします。
- にこのキーワードは、ステップ、つまりのがされるステップをします。Arrange、Act、Assertパラダイムでは、えられた "Assert" をします。
- Andこのキーワードは、のいずれかのキーワードとみわせてされます。2つのされたステートメントがある、Givenをに2びすのではなく、"AとBをする"とすることができます。

Examples

このでは、Gherkinのキュウリのファイルのについてします。ファイルは、なでいくつかのキーワードをします。

なキーワードを試みましょう

- このキーワードは、テストまたはされているのなまたはです。
- シナリオこのキーワードは、をテストするのシナリオのまたはなをします。
- このキーワードをえると、えられたステップ、またはするにしているをしています。アレンジ、アクト、アサートのパラダイムでは、えられた "アレンジ" をします。
- このキーワードはときステップ、またはするをす。Arrange、Act、Assertのパラダイムでは、えられた "Act" をします。
- に、このキーワードは、ステップ、すなわちのがされるステップをします。Arrange、Act、Assertパラダイムでは、えられた "Assert" をします。
- そして、このキーワードは、のいずれかのキーワードとみわせてされます。2つのされたステートメントがある、Givenをに2びすのではなく、"AとBをする"とすることができます。
- しかし、このキーワードは、**Given**、**When**と**Then**とにされ、かがこらないことをします。その、AではなくB。

すべてのキーワードは、しいになければならず、ガーキンパーザによってされるためには、しいののでなければなりません。フィーチャーとシナリオのキーワードは、のにすように、にコロンをけるがあります。「When、Then、And」はコロンをとしません。

キーワードにえて、やコメントをくことができます。はキーワードのにあります、じにあり、キーワードののにコメントがされます。フィーチャーコメントをくときは、のなるにつながるエッジとのをにするのがです。

```
Feature: Product Login
  As a user, I would like to be able to use my credentials to successfully login.

  Rules:
  - The user must have a valid username
  - The user must have a valid password
  - The user must have an active subscription
  - User is locked out after 3 invalid attempts
  - User will get a generic error message following login attempt with invalid credentials

  Scenario: The user successfully logs in with valid credentials
    This scenario tests that a user is able to successfully login provided they enter a valid username, valid password, and currently have an active subscription on their account.

    Given the user is on the login page
    When the user signs in with valid credentials
    Then the user should be logged in
```

パラメータされたステップ

Gherkinをくときには、テストプランをしているエンジニアがのためのステップをパラメータすることがあります。ステップは、キャプチャグループをしてパラメータをけります。エンジニアリングノートのキャプチャグループごとにするパラメータがない、「CucumberExceptionArity mismatch」がスローされることがされますのでは、をでむことにしましたasはとしてをけれます。

```
Feature: Product Login
  As a user, I would like to be able to use my credentials to successfully login.

  Rules:
  - The user must have a valid username
  - The user must have a valid password
  - The user must have an active subscription
  - User is locked out after 3 invalid attempts
  - User will get a generic error message following login attempt with invalid credentials

  Scenario: The user successfully logs in with valid credentials
    This scenario tests that a user is able to successfully login provided they enter a valid username, valid password, and currently have an active subscription on their account.
```

```
Given the user is on the login page
When the user signs in with "valid" credentials
Then the user should be logged in
```

Scenario: The user attempts to log in with invalid credentials
This scenario tests that a user is not able to log in when they enter invalid credentials

```
Given the user is on the login page
When the user signs in with "invalid" credentials
Then the user should be logged in
```

Scenario: The user is locked out after too many failed attempts
This scenario validates that the user is locked out of their account after failing three consecutive attempts to log in

```
Given the user is on the login page
When the user fails to log in 3 times
Then the user should be locked out of their account
```

の

のでついたように、じをもきしています

```
Given the user is on the login page
```

になことがあります。に、されるステップがあるはにそうです。Gherkinはこれをするために、となるのキーワードをします。

キーワードは、フィーチャーのすべてのシナリオのに、そのにされたステップをするためにされます。すべてのシナリオでであることがでないり、バックグラウンドステップをしないでください。のキーワードとに、にはやがき、そのにコメントがされます。フィーチャーとシナリオのように、バックグラウンドはコロンでするがあります。

Feature: Product Login

```
As a user, I would like to be able to use my credentials to successfully login.
```

Rules:

- The user must have a valid username
- The user must have a valid password
- The user must have an active subscription
- User is locked out after 3 invalid attempts
- User will get a generic error message following login attempt with invalid credentials

Background: The user starts out on the login page
Given the user is on the login page

Scenario: The user successfully logs in with valid credentials
This scenario tests that a user is able to successfully login provided they enter a valid username, valid password, and currently have an active subscription on their account.

```
When the user signs in with "valid" credentials
Then the user should be logged in
```

```
Scenario: The user attempts to log in with invalid credentials
This scenario tests that a user is not able to log in when
they enter invalid credentials
```

```
When the user signs in with "invalid" credentials
Then the user should be logged in
```

```
Scenario: The user is locked out after too many failed attempts
This scenario validates that the user is locked out
of their account after failing three consecutive
attempts to log in
```

```
When the user fails to log in 3 times
Then the user should be locked out of their account
```

シナリオの

によっては、をしてじシナリオをももやりすことができます。この、Gherkinはこのにするためのいくつかのしいキーワードをしています。シナリオと。シナリオアウトラインScenario Outline キーワードは、シナリオがリストからきをきえてされることをCucumberにえます。リストがにされるに、Examplesキーワードがびされます。シナリオのアウトラインのはカッコでむがあります。のでは、カッコでまれたは、にされているにしています。はバーでられ、のにがされます。

```
Feature: Product Login
```

```
As a user, I would like to be able to use my credentials to successfully
login.
```

```
Rules:
```

- The user must have a valid username
- The user must have a valid password
- The user must have an active subscription
- User is locked out after 3 invalid attempts
- User will get a generic error message following login attempt with invalid credentials

```
Background: The user starts out on the login page
```

```
Given the user is on the login page
```

```
Scenario Outline: The user successfully logs in with their account
```

```
This scenario outlines tests in which various users attempt
to sign in successfully
```

```
When the user enters their <username>
```

```
And the user enters their <password>
```

```
Then the user should be successfully logged on
```

```
Examples:
```

username	password
frank	1234
jack	4321

タグ

ので、テストまたはシナリオをカテゴリにフィルタリングすることができます。は、じカテゴリについてテストをすることができます。 Gherkinでは、タグをしてフィーチャと々のシナリオをできます。のでは、のFeatureキーワードがタグ "@Automation"であることにしてください。Gherkinはこれを "@"のユーザーがタグとしてします。このでは、エンジニアはこれらのテストがにされていることをにしたいとえています。すべてのテストができるわけではなく、のテストはQAでうがあります。

に、@Productionというタグがシナリオテストのユーザーロックアウトにされていることにもしてください。これは、このシナリオがアプリケーションのでのみなためです。はにサンドボックスアカウントをロックアウトしたくない。このタグをすると、このテストがにしてのみされるようすることができます。

に、シナリオアウトラインには@Stagingタグがいています。このでは、されているアカウントがステージングアカウントであり、のではしないためです。 @Productionタグとに、これらのテストはステージングでのみされます。

これらは、タグをする、およびのほんのいくつかのです。には、これらのタグはあなたとにとってをつことになります。

```
@Automation
Feature: Product Login
  As a user, I would like to be able to use my credentials to successfully login.

  Rules:
  - The user must have a valid username
  - The user must have a valid password
  - The user must have an active subscription
  - User is locked out after 3 invalid attempts
  - User will get a generic error message following login attempt with invalid credentials

  Background: The user starts out on the login page
    Given the user is on the login page

  Scenario: The user successfully logs in with valid credentials
    This scenario tests that a user is able to successfully login provided they enter a valid username, valid password, and currently have an active subscription on their account.

    When the user signs in with "valid" credentials
    Then the user should be logged in

  Scenario: The user attempts to log in with invalid credentials
    This scenario tests that a user is not able to log in when they enter invalid credentials

    When the user signs in with "invalid" credentials
    Then the user should be logged in

@Production
Scenario: The user is locked out after too many failed attempts
  This scenario validates that the user is locked out of their account after failing three consecutive
```

```
attempts to log in
```

```
When the fails to log in 3 times
```

```
Then the user should be locked out of their account
```

```
@Staging
```

```
Scenario Outline: The user successfully logs in with their account
```

```
    This scenario outlines tests in which various users attempt  
    to sign in successfully
```

```
    When the user enters their <username>
```

```
    And the user enters their <password>
```

```
    Then the user should be successfully logged on
```

```
Examples:
```

username	password
frank	1234
jack	4321

ガーキンのヒント

- シナリオは1つのをテストします
- シナリオはなでされます
- シナリオでのなの
- らかにしない
- なステップをける
- シナリオをくしておく
- じでくのシナリオをするはありません
- なシナリオをする
- 1つしかないステップ
- その、「ステップ」で「」をします。

オンラインでガーキンのをむ <https://riptutorial.com/ja/cucumber/topic/9296/ガーキンの>

5: ステップ

ステップは、Cucumberののによってサポートされるプログラミングである。このトピックでは、サポートされているプログラミングのステップのと、ステップでCucumber APIびしをするをします。

Examples

いくつかのなRubyのステップ

features / step_definitions / documentation.rbで

```
When /^I go to the "([^"]+)" documentation$/ do |section|
  path_part =
    case section
    when "Documentation"
      "documentation"
    else
      raise "Unknown documentation section: #{section}"
    end
  visit "/documentation/#{path_part}/topics"
end

Then /^I should see the "([^"]+)" documentation"$/ do |section|
  expect(page).to have_css('h2.doctag_title a', text: section)
end
```

これらは、Webアプリケーションをします。らはまだであり、できるだけです。

ステップはGherkinキーワードでまります。Gherkinキーワードは、ステップファイルでCucumberにステップをするメソッドです。ステップメソッドは、シナリオのとする、シナリオがするにするときにはされるブロックをります。のキャプチャ・グループは、ブロック・パラメータとしてブロックにされます。

Whenステップには、がめるからページ「ドキュメント」へのなインラインのがあります。リアルキュウリスイートは、このロジックをのでします。visitはCapybaraによってされています。カピバラはににされていますが、キュウリをするはありません。visitは、されたURLにアクセスするためにCapybaraがするブラウザにします。

Thenステップは、ページのをどのようにテストできるかをします。expect / to RSpecをび、キュウリではありませんが、ににそれをによりされます。have_cssはCapybaraによってされています。されたCSSセレクトアが、されたテキストをむページのとすることがされます。ブラウザのがした、このはすることにしてください。

オンラインでステップをむ <https://riptutorial.com/ja/cucumber/topic/5681/ステップ>

6:

き

あなたはQTPとSeleniumのプラグインとしてキュウリをうことができます。キュウリシナリオのステップはグローバルです。してもびすことができます。したがって、コードのメンテナンスが、なときにじコードをすることができます。

キュウリのは、Gherkinでかれ、`.feature`つファイルにされます。このトピックでは、Gherkinののをします。

Examples

キュウリの

フィーチャー/ドキュメンテーション。フィーチャー

```
Feature: Documentation

  Scenario: User views documentation
    When I go to the "Cucumber" documentation
    Then I should see the "Cucumber" documentation
```

のには、`Feature`ラインと、`When`、`Then`またはのGherkinキーワードでまる1つのステップがある`Scenario`があります。

なシナリオにはおそらくのステップがあります。

シナリオの

のテンプレート

```
Scenario Outline: As a homemaker i want to buy and pay for the below product
  Given I purchase <a product>
    And I require a carry bag to take things to home
  When I pay bill using <payment method> to successfully checkout
  Then I should have a receipt
```

Examples:

a product	payment method
Cake	Visa
Coke	Paypal

の

```
Feature: Some terse yet descriptive text of what is desired
```

```
Textual description of the business value of this feature
Business rules that govern the scope of the feature
Any additional information that will make the feature easier to understand
```

Background:

```
Given some precondition needed for all scenarios in this file
And another precondition
```

Scenario: Some determinable business situation

```
Textual description of the business value of this scenario
Business rules that govern the scope of the scenario
Any additional information that will make the scenario easier to understand
Given some precondition
And some other precondition
When some action by the actor
And some other action
And yet another action
Then some testable outcome is achieved
And something else we can check happens too
But something else we can check does not happen
```

Scenario Outline: Some determinable business situation

```
Given I am <precondition>
And some other precondition
When some action by the actor
Then I have <outcome> rights
```

Examples:

```
| precondition | outcome |
| username1   | customer |
| username2   | admin   |
```

のキーワードは入れることができますが、によってはするがいかもしれません

- Feature: | Ability: | Business Need:
- Scenario Outline: | Scenario Template:
- Examples: | Scenarios:
- Given | | When | Then | And | But | * |

オンラインでをむ <https://riptutorial.com/ja/cucumber/topic/6023/>

クレジット

S. No		Contributors
1	キュウリをめる	Community , Dave Schweisguth , Mo H. , Roberto Lo Giacco , SirLenz0rlot , user3554664
2	IntelliJにcucumberプラグインをインストールする	George Pantazes , Priya
3	Maven_cucumberプロジェクトのpom.xml。	user
4	ガーキンの	jordiPons , tramstheman , user3554664
5	ステップ	Dave Schweisguth
6		Dave Schweisguth , Kyle Fairns , Priya