# How to make new cucumber steps

Contents

[How to make new cucumber steps 1](#_Toc513646787)

[Introduction 1](#_Toc513646788)

[Group several cucumber steps under one new step. 1](#_Toc513646789)

[Example of Grouping together cucumber scripts 1](#_Toc513646790)

[New cucumber step that calls Ruby commands. 2](#_Toc513646791)

[Example of cucumber scripts using Ruby 2](#_Toc513646792)

[Potential problems with creating customized cucumber steps 2](#_Toc513646793)

[Duplicate steps- two steps that perform the same step 2](#_Toc513646794)

[Ambitious step names 2](#_Toc513646795)

[Steps with confusing names, confusing case, and extra spacing 3](#_Toc513646796)

## Introduction

Adding new steps to cucumber is like adding new functions or routines in other languages, it gives more power and versatility to that language. New steps in cucumber can be created either by grouping together other cucumber steps together under one new step, or the new cucumber step can be a collection of ruby commands, this give the ability to perform task outside of cucumber’s ability, such as opening file, accessing database, etc.

## Group several cucumber steps under one new step.

Creating a new step by combining several steps together under one single step has two benefits, It decreases the size of the script one step replace hundreds of lines in the script, such as taking a large script of over 1500 lines, and reduce it down to 100 lines, or performing routine task such as login. Besides reducing script size, combing scripts into one step decreases maintenance time.

### Example of Grouping together cucumber scripts

#### No parameters used

Then /^I use my new cucumber script $/ do

steps %{

<A single line or Block of Cucumber steps>

Then I should…..

And I should ….

}

End

Example: Then I use my new cucumber script

#### Using parameters, the parameters names have to match case name and spaces.

Then /^I should log into the system as ^(.\*) with my ^(.\*)$/ do |name, credentials|

steps %{

<A single line or Block of Cucumber steps>

Given the user #{name} exists

Given I log in as #{credentials}

}

end

Example: *Then I should log into the system as “me” with my “login”*

## New cucumber step that calls Ruby commands.

Adding cucumber steps that uses another language, such as a ruby, adds robust and gives cucumber the ability to perform task that it could not do in its native language, such as reading and writing to database files, spreadsheets, and flat files.

### Example of cucumber scripts using Ruby

#### No Parameters used

Then /^I should use my new cucumber step$/ do

< any ruby code>

End

Example: *Then I should use my new cucumber*

Parameters used

Then /^I should log into the system as ^(.\*) with my ^(.\*)$/ do |1st Parameter, 2nd Parameter|

< any ruby code>

<ruby code using vars name and credentials, from input parameters>

Example: *Then I should log into the system as “me” with my “login”*

## Potential problems with creating customized cucumber steps

There is a complete set of problems that can occur when creating customized script for cucumber, ranging from creating a new script that does the same task as a create that has already been created, create a script with a duplicate name, mismatching the name of the script, i.e. misuse of case and or spaces.

### Duplicate steps, two steps that perform the same step

Examples: Same function, but same task

*And I login my server*

*And I log into “my server”*

### Ambitious step names

Steps that have the identical name, even though they may be contain in different files will generate an ambitious error and will abort the script

Example: Two different files containing the same step name

*File 1 contains a step called “And I check the value on the page”*

*File 2 contains a step called “And I check the value on the page”*

### Steps with confusing names, confusing case, and extra spacing

Keep the step names as short as possible but still convey the function of the step, avoid abbreviations, unnecessary spacing,

Example:

*“And I fill in all the fields on the Mailing Address page”-> wordy*

*“And I fill in all the fields on the Mailing Addr page”-> abbreviations*

*“And I fill in the Mailing Address page”->case*

*“And I fill in the Mailing Address page ”->spaces on the end*