# Automation Testing

©Simplilearn. All rights reserved.

## Scenarios

©Simplilearn. All rights reserved.

## A Day in the Life of an Automation Test Engineer

Sam now understands how to set up a Cucumber development environment.

He has now decided to build a cucumber project. He needs to understand the scenarios to build them.

To achieve the above, he will learn a few concepts in this lesson that can help him find a solution for the scenario.

©Simplilearn. All rights reserved.

## Learning Objectives

By the end of this lesson, you will be able to:

- Describe undefined scenarios
- Illustrate pending scenarios
- Comprehend failed scenarios
- Describe variables and refactoring

©Simplilearn. All rights reserved.

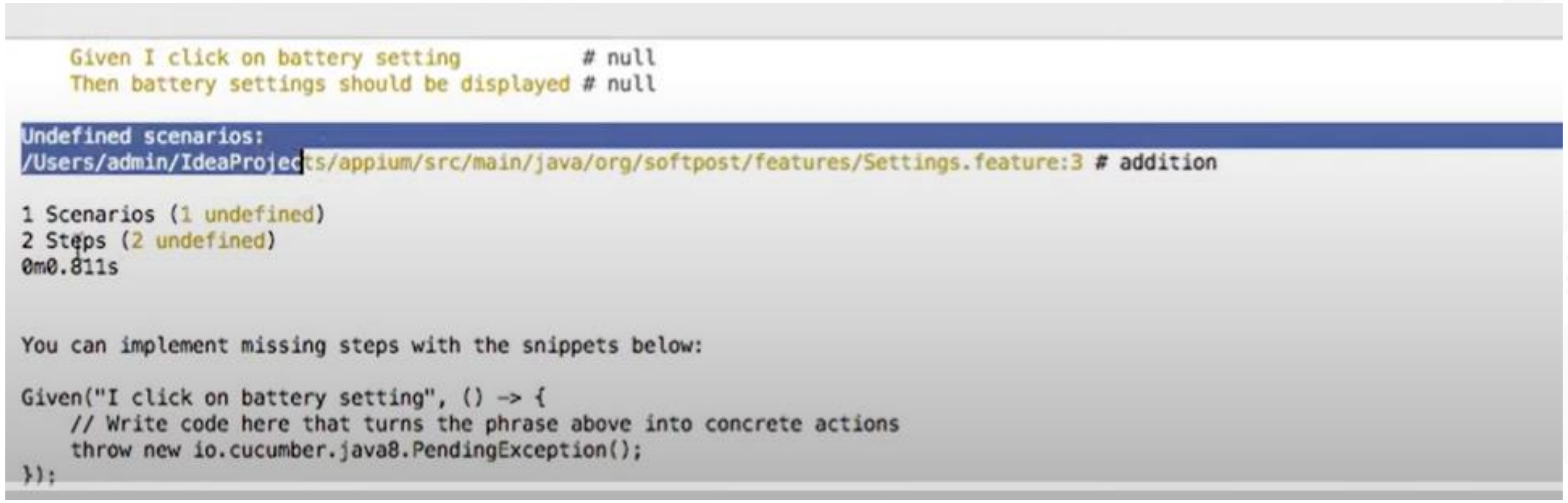
TECHNOLOGY

## Undefined Scenario

©Simplilearn. All rights reserved.

## Undefined Scenario

If Cucumber informs a user that your steps are undefinable despite the existence of defined step definitions, this indicates that Cucumber is unable to locate your step definitions.

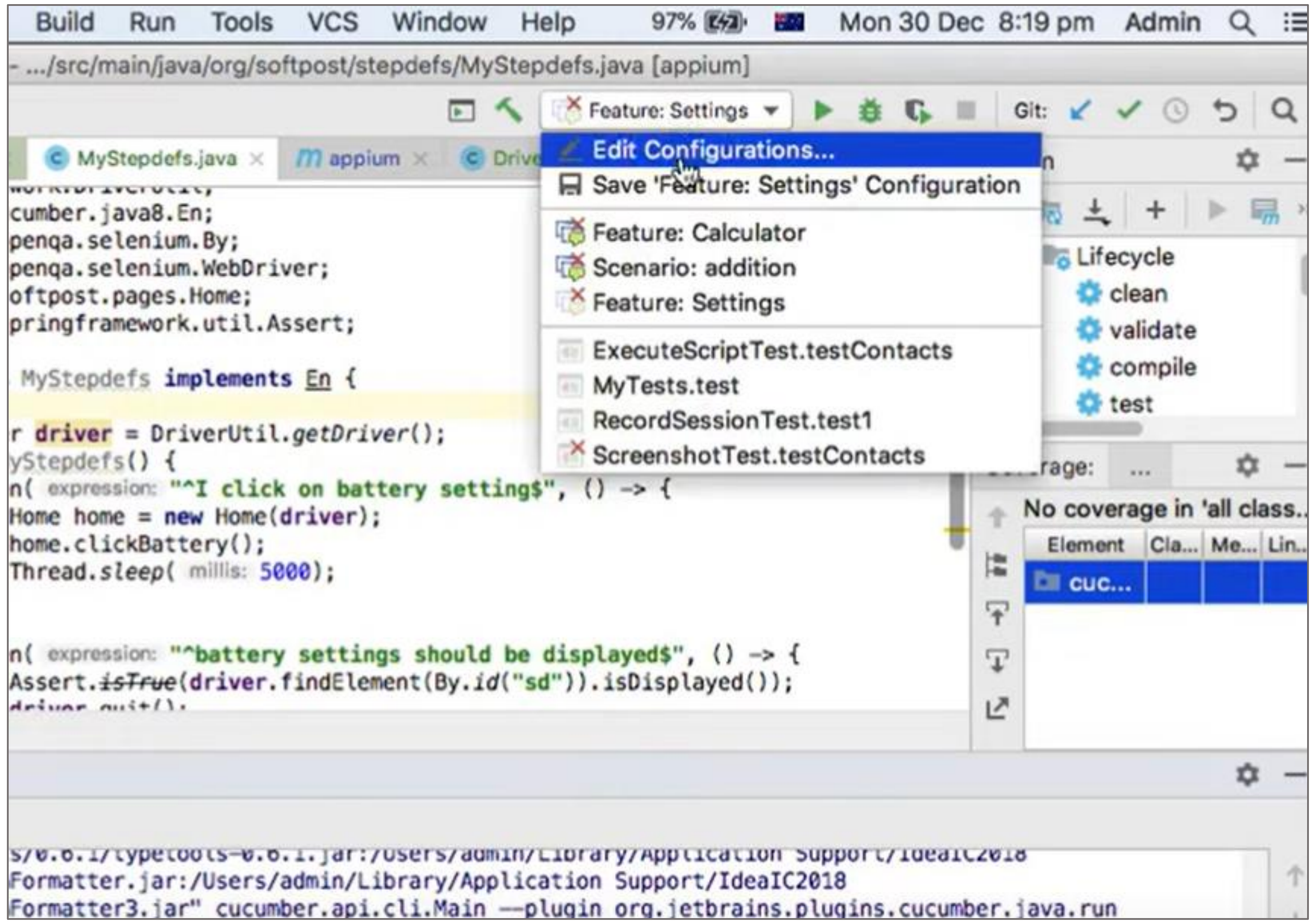


```
    Given I click on battery setting          # null
    Then battery settings should be displayed # null

Undefined scenarios:
/Users/admin/IdeaProjects/appium/src/main/java/org/softpost/features/Settings.feature:3 # addition

1 Scenarios (1 undefined)
2 Steps (2 undefined)
0m0.811s


You can implement missing steps with the snippets below:

Given("I click on battery setting", () -> {
    // Write code here that turns the phrase above into concrete actions
    throw new io.cucumber.java8.PendingException();
});
```

©Simplilearn. All rights reserved.

# Handling Undefined Scenario

These are the steps to handle an undefined scenario:


Build Run Tools VCS Window Help 97% Mon 30 Dec 8:19 pm Admin
- .../src/main/java/org/softpost/stepdefs/MyStepdefs.java [appium]
Feature: Settings
Git:
MyStepdefs.java
appium
Edit Configurations...
Save 'Feature: Settings' Configuration
Feature: Calculator
Scenario: addition
Feature: Settings
ExecuteScriptTest.testContacts
MyTests.test
RecordSessionTest.test1
ScreenshotTest.testContacts
Lifecycle
clean
validate
compile
test
No coverage in 'all class..
Element
Cla...
Me...
Lin..
cuc...

```
cumber.java8.En;
penqa.selenium.By;
penqa.selenium.WebDriver;
oftpost.pages.Home;
pringframework.util.Assert;

 MyStepdefs implements En {

r driver = DriverUtil.getDriver();
yStepdefs() {
n( expression: "^I click on battery setting$", () -> {
Home home = new Home(driver);
home.clickBattery();
Thread.sleep( millis: 5000);


n( expression: "^battery settings should be displayed$", () -> {
Assert.isTrue(driver.findElement(By.id("sd")).isDisplayed());
```

```
Formatter.jar:/Users/admin/Library/Application Support/IdeaIC2018
Formatter3.jar" cucumber.api.cli.Main --plugin org.jetbrains.plugins.cucumber.java.run
```

Step 1:

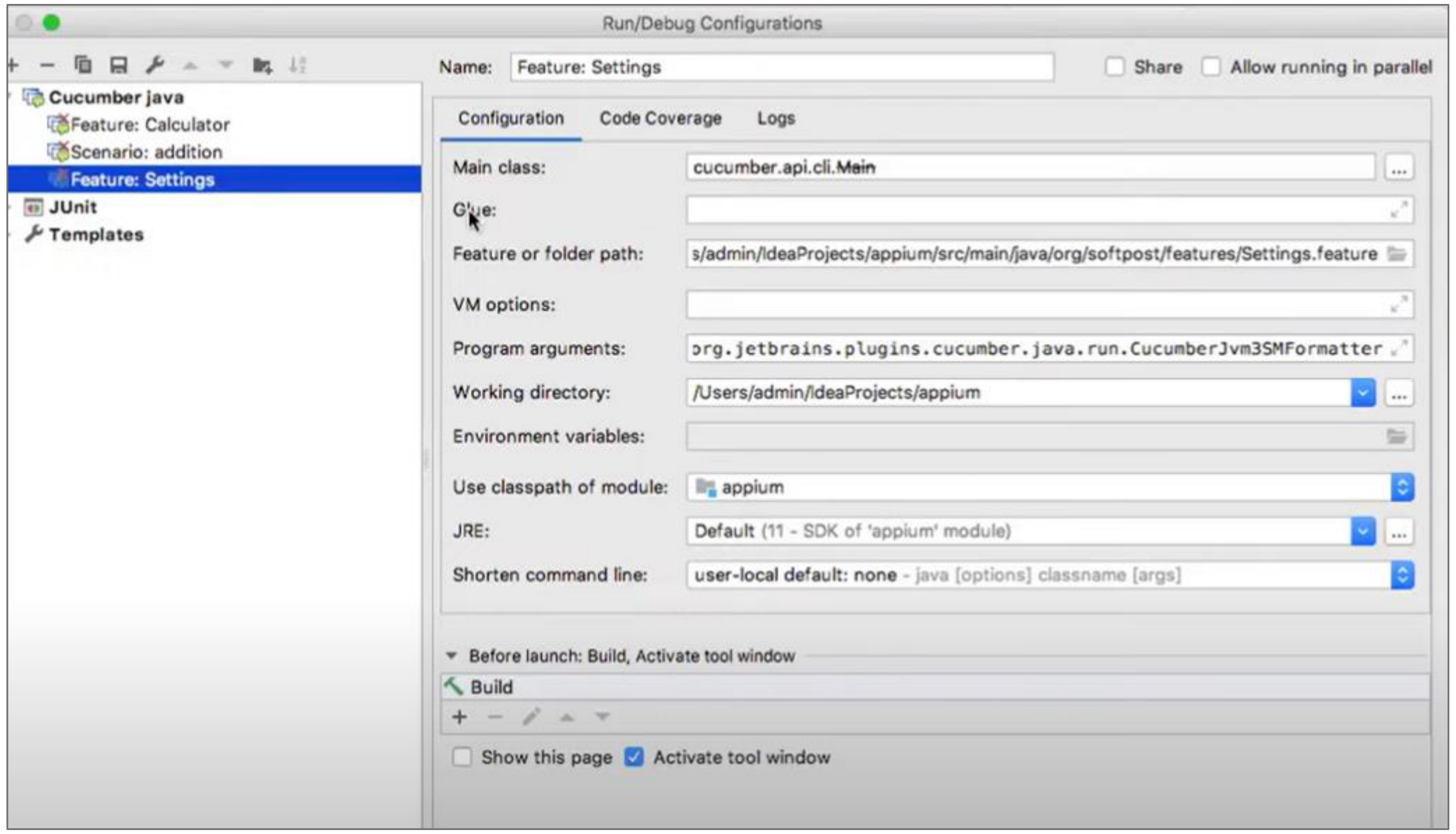
- Click on **Run** or **Feature Settings**

Step 2:

- Click on **Edit Configurations**

©Simplilearn. All rights reserved.

# Handling Undefined Scenario



Name: Feature: Settings
Share [ ] Allow running in parallel [ ]

Main class: cucumber.api.cli.Main
Glue:
Feature or folder path: s/admin/IdeaProjects/appium/src/main/java/org/softpost/features/Settings.feature
VM options:
Program arguments: org.jetbrains.plugins.cucumber.java.run.CucumberJvm3SMFormatter
Working directory: /Users/admin/IdeaProjects/appium
Environment variables:
Use classpath of module: appium
JRE: Default (11 - SDK of 'appium' module)
Shorten command line: user-local default: none - java [options] classname [args]

Before launch: Build, Activate tool window
Build

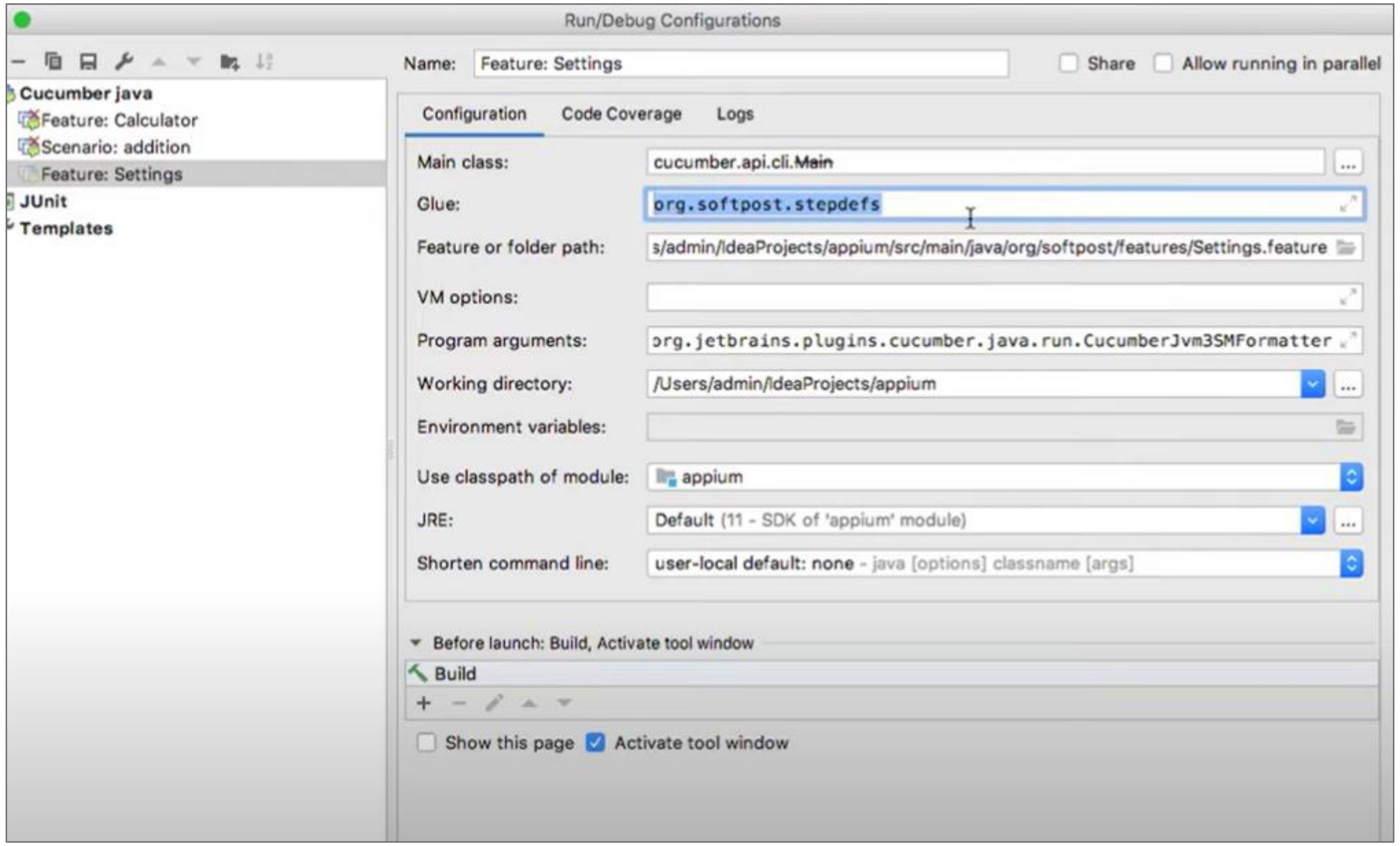
Show this page [ ] Activate tool window [✓]

**Step 3:**
Select your Cucumber java class from the "**Cucumber java**" list.

©Simplilearn. All rights reserved.

## Handling Undefined Scenario

**Step 4:**
Fill the "**Glue**" with the path where your Cucumber implementation is in.


Run/Debug Configurations

Name: Feature: Settings
Share
Allow running in parallel

Cucumber java
Feature: Calculator
Scenario: addition
Feature: Settings
JUnit
Templates

Configuration | Code Coverage | Logs

Main class: cucumber.api.cli.Main
Glue: org.softpost.stepdefs
Feature or folder path: s/admin/IdeaProjects/appium/src/main/java/org/softpost/features/Settings.feature
VM options:
Program arguments: org.jetbrains.plugins.cucumber.java.run.CucumberJvm3SMFormatter
Working directory: /Users/admin/IdeaProjects/appium
Environment variables:
Use classpath of module: appium
JRE: Default (11 - SDK of 'appium' module)
Shorten command line: user-local default: none - java [options] classname [args]

Before launch: Build, Activate tool window
Build

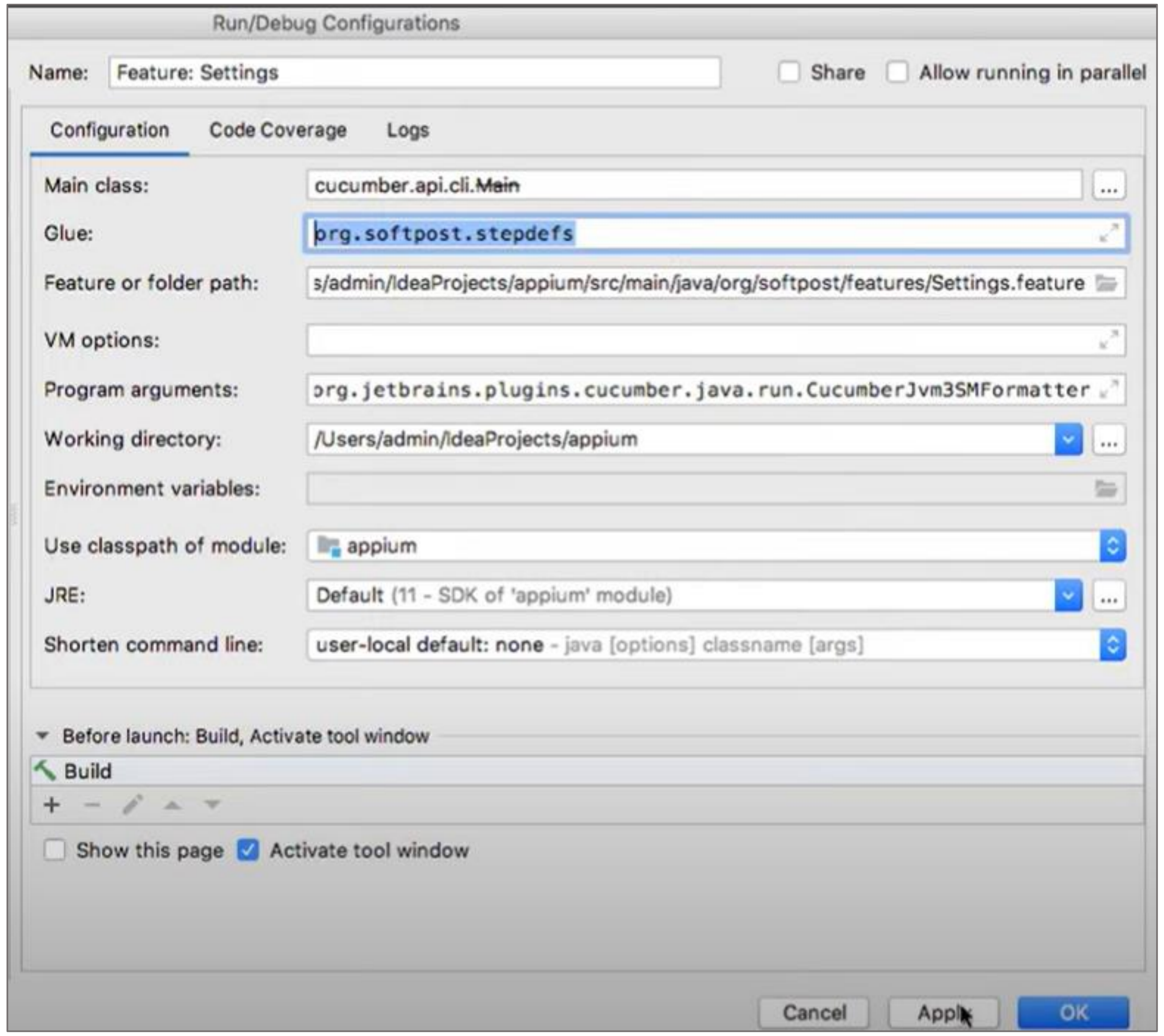
Show this page
[✓ Activate tool window]

For example, "com.dice.test.steps"

©Simplilearn. All rights reserved.

# Handling Undefined Scenario

**Step 5:**
Click on the "**Apply**" button.


Run/Debug Configurations

Name: Feature: Settings ☐ Share ☐ Allow running in parallel

Configuration | Code Coverage | Logs

| | |
|---|---|
| Main class: | cucumber.api.cli.Main |
| Glue: | org.softpost.stepdefs |
| Feature or folder path: | s/admin/IdeaProjects/appium/src/main/java/org/softpost/features/Settings.feature |
| VM options: | |
| Program arguments: | org.jetbrains.plugins.cucumber.java.run.CucumberJvm3SMFormatter |
| Working directory: | /Users/admin/IdeaProjects/appium |
| Environment variables: | |
| Use classpath of module: | appium |
| JRE: | Default (11 - SDK of 'appium' module) |
| Shorten command line: | user-local default: none - java [options] classname [args] |

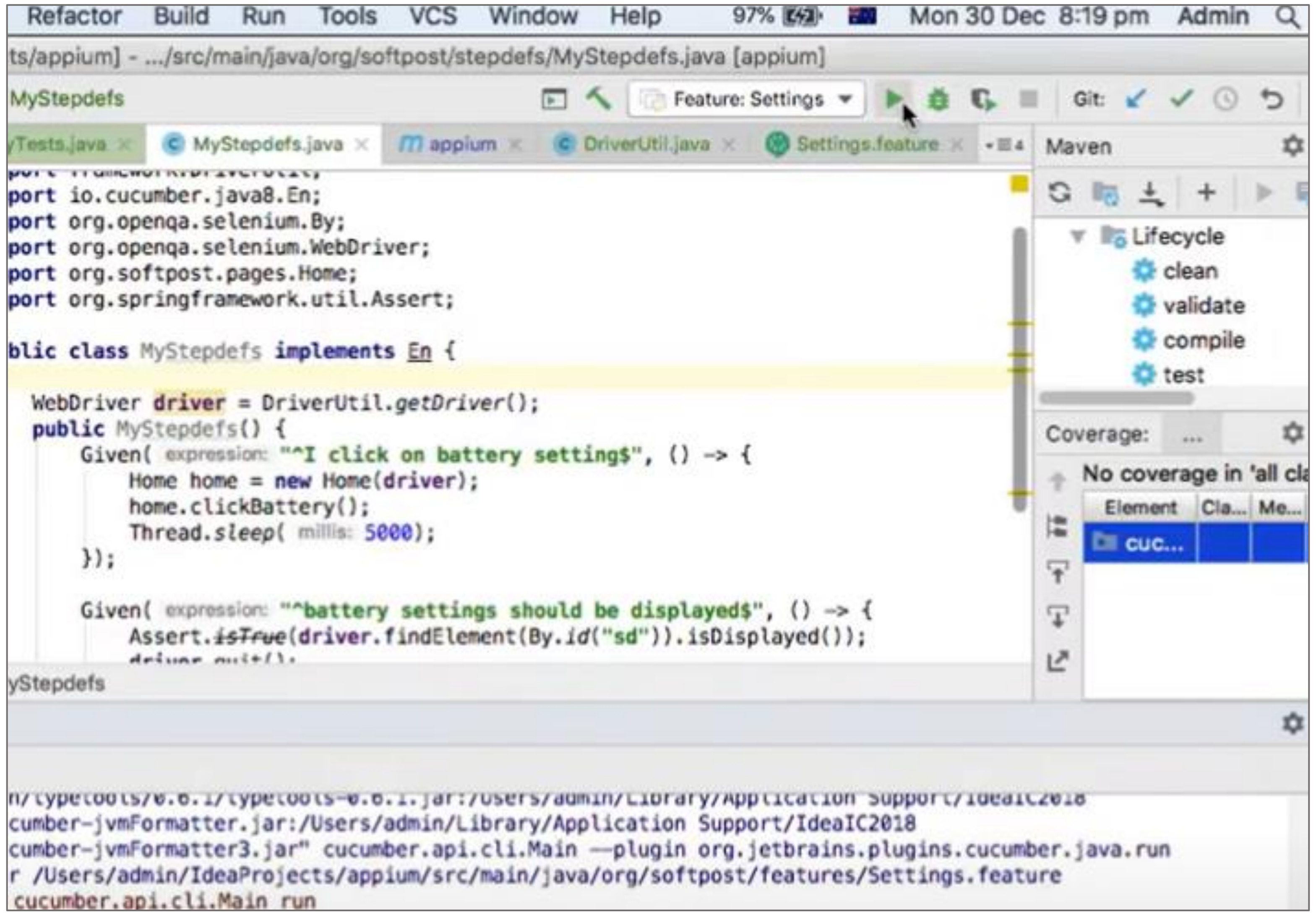
Before launch: Build, Activate tool window

Build

☐ Show this page ☑ Activate tool window

©Simplilearn. All rights reserved.

## Handling Undefined Scenario



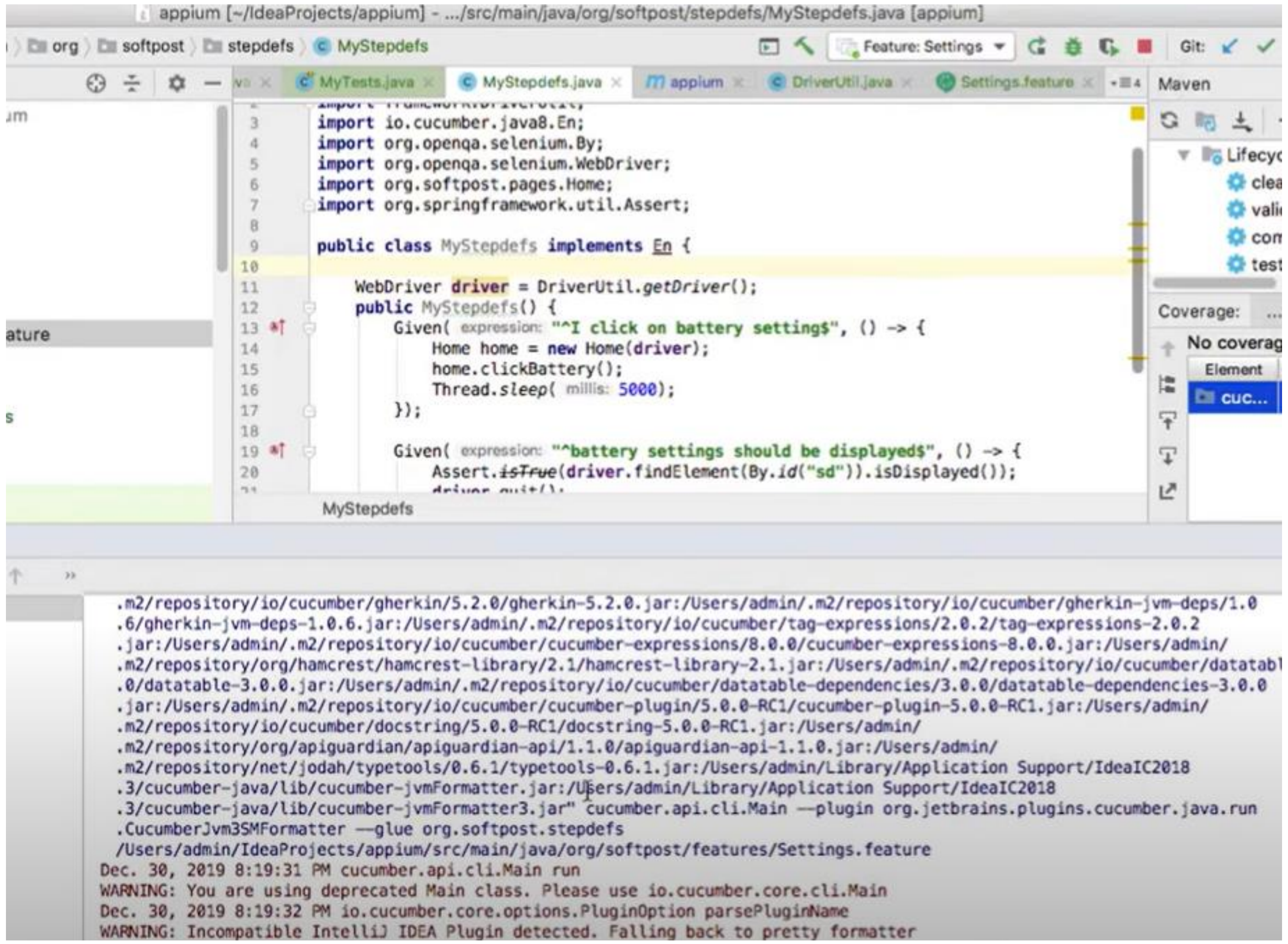
```
port io.cucumber.java8.En;
port org.openqa.selenium.By;
port org.openqa.selenium.WebDriver;
port org.softpost.pages.Home;
port org.springframework.util.Assert;

blic class MyStepdefs implements En {

  WebDriver driver = DriverUtil.getDriver();
  public MyStepdefs() {
      Given( expression: "^I click on battery setting$", () -> {
          Home home = new Home(driver);
          home.clickBattery();
          Thread.sleep( millis: 5000);
      });

      Given( expression: "^battery settings should be displayed$", () -> {
          Assert.isTrue(driver.findElement(By.id("sd")).isDisplayed());
```

```
n/typetools/0.6.1/typetools-0.6.1.jar:/Users/admin/Library/Application Support/IdeaIC2018
cumber-jvmFormatter.jar:/Users/admin/Library/Application Support/IdeaIC2018
cumber-jvmFormatter3.jar" cucumber.api.cli.Main —plugin org.jetbrains.plugins.cucumber.java.run
r /Users/admin/IdeaProjects/appium/src/main/java/org/softpost/features/Settings.feature
 cucumber.api.cli.Main run
```

**Step 6:**
 Try to **run/debug** your Cucumber feature file.

©Simplilearn. All rights reserved.

# Handling Undefined Scenario



```java
import io.cucumber.java8.En;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.softpost.pages.Home;
import org.springframework.util.Assert;

public class MyStepdefs implements En {

    WebDriver driver = DriverUtil.getDriver();
    public MyStepdefs() {
        Given( expression: "^I click on battery setting$", () -> {
            Home home = new Home(driver);
            home.clickBattery();
            Thread.sleep( millis: 5000);
        });

        Given( expression: "^battery settings should be displayed$", () -> {
            Assert.isTrue(driver.findElement(By.id("sd")).isDisplayed());
```

```
 .m2/repository/io/cucumber/gherkin/5.2.0/gherkin-5.2.0.jar:/Users/admin/.m2/repository/io/cucumber/gherkin-jvm-deps/1.0
 .6/gherkin-jvm-deps-1.0.6.jar:/Users/admin/.m2/repository/io/cucumber/tag-expressions/2.0.2/tag-expressions-2.0.2
 .jar:/Users/admin/.m2/repository/io/cucumber/cucumber-expressions/8.0.0/cucumber-expressions-8.0.0.jar:/Users/admin/
 .m2/repository/org/hamcrest/hamcrest-library/2.1/hamcrest-library-2.1.jar:/Users/admin/.m2/repository/io/cucumber/datatabl
 .0/datatable-3.0.0.jar:/Users/admin/.m2/repository/io/cucumber/datatable-dependencies/3.0.0/datatable-dependencies-3.0.0
 .jar:/Users/admin/.m2/repository/io/cucumber/cucumber-plugin/5.0.0-RC1/cucumber-plugin-5.0.0-RC1.jar:/Users/admin/
 .m2/repository/io/cucumber/docstring/5.0.0-RC1/docstring-5.0.0-RC1.jar:/Users/admin/
 .m2/repository/org/apiguardian/apiguardian-api/1.1.0/apiguardian-api-1.1.0.jar:/Users/admin/
 .m2/repository/net/jodah/typetools/0.6.1/typetools-0.6.1.jar:/Users/admin/Library/Application Support/IdeaIC2018
 .3/cucumber-java/lib/cucumber-jvmFormatter.jar:/Users/admin/Library/Application Support/IdeaIC2018
 .3/cucumber-java/lib/cucumber-jvmFormatter3.jar" cucumber.api.cli.Main --plugin org.jetbrains.plugins.cucumber.java.run
 .CucumberJvm3SMFormatter --glue org.softpost.stepdefs
 /Users/admin/IdeaProjects/appium/src/main/java/org/softpost/features/Settings.feature
Dec. 30, 2019 8:19:31 PM cucumber.api.cli.Main run
WARNING: You are using deprecated Main class. Please use io.cucumber.core.cli.Main
Dec. 30, 2019 8:19:32 PM io.cucumber.core.options.PluginOption parsePluginName
WARNING: Incompatible IntelliJ IDEA Plugin detected. Falling back to pretty formatter
```

That’s how it looks after handling an undefined scenario.

©Simplilearn. All rights reserved.

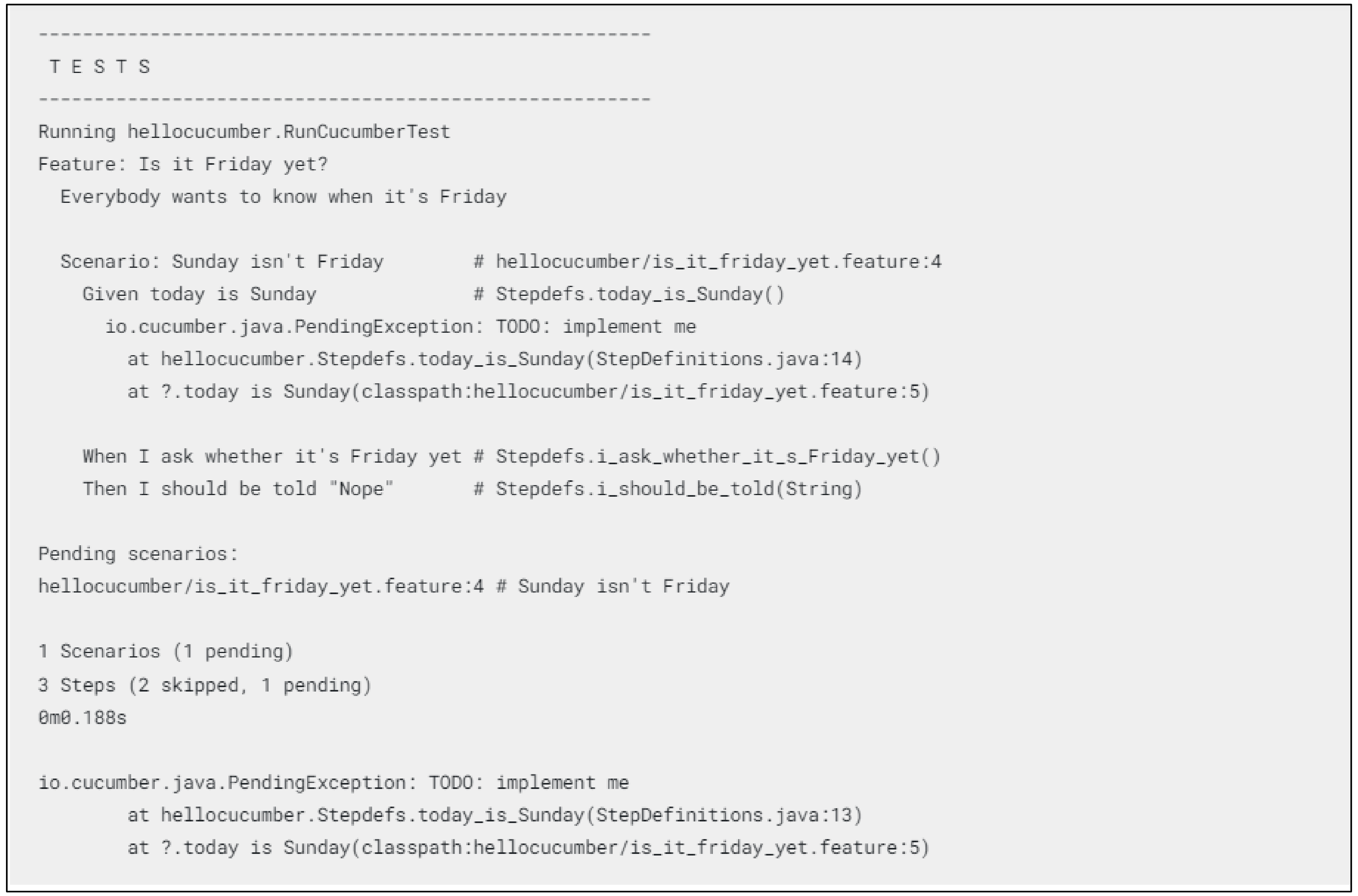
TECHNOLOGY

## Pending Scenario

©Simplilearn. All rights reserved.

## Pending Scenario

The step is indicated as pending when a step definition's method or function calls the pending method, letting you know that you still have work to perform.



```
-------------------------------------------------------
 T E S T S
-------------------------------------------------------
Running hellocucumber.RunCucumberTest
Feature: Is it Friday yet?
  Everybody wants to know when it's Friday

  Scenario: Sunday isn't Friday        # hellocucumber/is_it_friday_yet.feature:4
    Given today is Sunday              # Stepdefs.today_is_Sunday()
      io.cucumber.java.PendingException: TODO: implement me
        at hellocucumber.Stepdefs.today_is_Sunday(StepDefinitions.java:14)
        at ?.today is Sunday(classpath:hellocucumber/is_it_friday_yet.feature:5)

    When I ask whether it's Friday yet # Stepdefs.i_ask_whether_it_s_Friday_yet()
    Then I should be told "Nope"       # Stepdefs.i_should_be_told(String)

Pending scenarios:
hellocucumber/is_it_friday_yet.feature:4 # Sunday isn't Friday

1 Scenarios (1 pending)
3 Steps (2 skipped, 1 pending)
0m0.188s

io.cucumber.java.PendingException: TODO: implement me
        at hellocucumber.Stepdefs.today_is_Sunday(StepDefinitions.java:13)
        at ?.today is Sunday(classpath:hellocucumber/is_it_friday_yet.feature:5)
```

©Simplilearn. All rights reserved.

# How To Mark a Scenario As Pending?

These are the few steps through which we can mark scenarios as pending:

Write a non-defined step

Create a pending step

By default, a non-defined step will stop the execution of the scenario.

```
Scenario: login with valid credentials
            Given this is pending
            ...
```

This will show the scenario along with a note about defining the step.

©Simplilearn. All rights reserved.

## How To Mark a Scenario As Pending?

These are the few steps through which we can mark scenarios as pending:

Write a non defined step

Create a pending step

In one of the step definition files, add a step:

```
Given /^PENDING/ do
            pending
           end
```

©Simplilearn. All rights reserved.

## How To Mark a Scenario As Pending?

These are the few steps through which we can mark scenarios as pending:

Write a non defined step

Create a pending step

Then users can use it like:

```
Scenario: login with valid credentials
        Given PENDING I have valid credentials
```

©Simplilearn. All rights reserved.

TECHNOLOGY

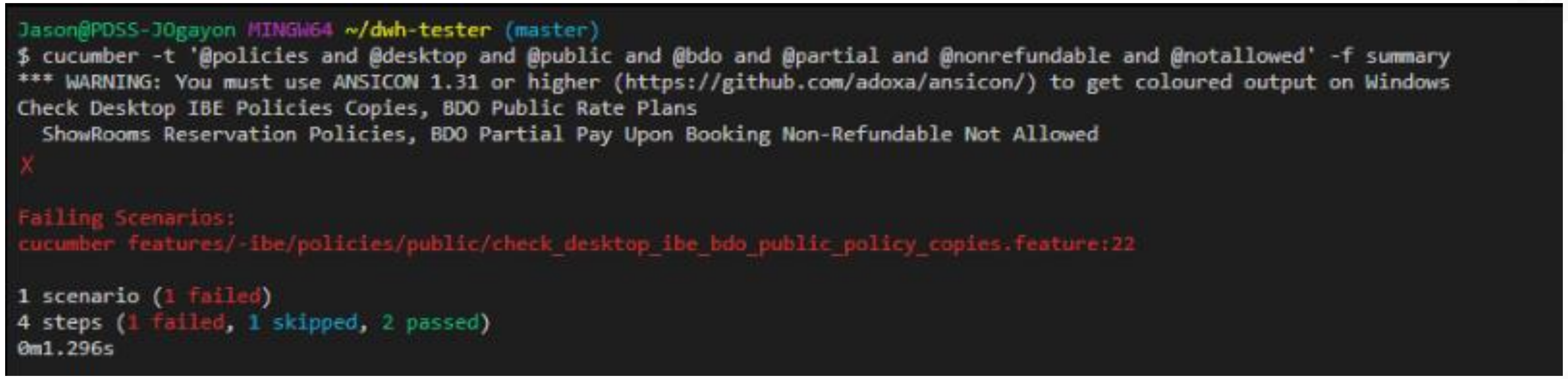
## Failed Scenario

©Simplilearn. All rights reserved.

simplilearn

## Failed Scenario

When a step definition's method or function is executed and raises an error, the step is marked as failed.



```
Jason@PDSS-JOgayon MINGW64 ~/dwh-tester (master)
$ cucumber -t '@policies and @desktop and @public and @bdo and @partial and @nonrefundable and @notallowed' -f summary
*** WARNING: You must use ANSICON 1.31 or higher (https://github.com/adoxa/ansicon/) to get coloured output on Windows
Check Desktop IBE Policies Copies, BDO Public Rate Plans
  ShowRooms Reservation Policies, BDO Partial Pay Upon Booking Non-Refundable Not Allowed
X

Failing Scenarios:
cucumber features/-ibe/policies/public/check_desktop_ibe_bdo_public_policy_copies.feature:22

1 scenario (1 failed)
4 steps (1 failed, 1 skipped, 2 passed)
0m1.296s
```

©Simplilearn. All rights reserved.

# How To Rerun a Failed Scenario?

These are the few steps through which we can rerun the failed scenarios:

Modify runner class

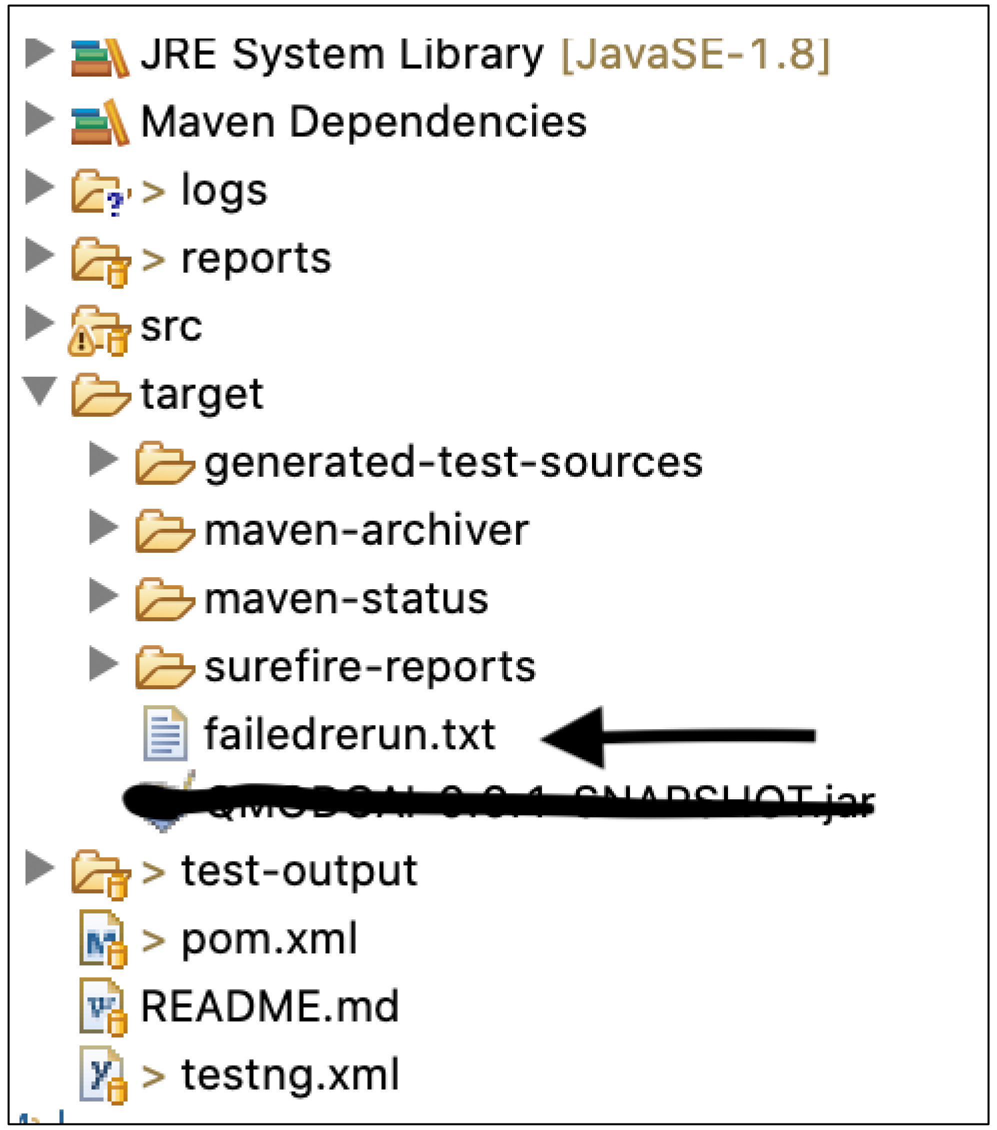
```
@CucumberOptions(
               features = { "src/test/resources/features" },
               glue = { "com.app.stepdefinition" }, // path of
step definition
               tags = "@Testcase1",

               plugin = { "pretty",
               "html:./reports/cucumber-reports/cucumber-
html/index.html",
                "rerun:target/failedrerun.txt"}
               )

public class TestRunner extends AbstractTestNGCucumberTests {}
```

Add **rerun:target/failedrerun.txt** in the plugin.

©Simplilearn. All rights reserved.

## How To Rerun a Failed Scenario?

In case of execution failure, it will generate a text file in the target folder. This text file will contain the information on the scenarios that get failed.



```
▶ JRE System Library [JavaSE-1.8]
▶ Maven Dependencies
▶ > logs
▶ > reports
▶ src
▼ target
    ▶ generated-test-sources
    ▶ maven-archiver
    ▶ maven-status
    ▶ surefire-reports
      failedrerun.txt  ◀
      [redacted]
▶ > test-output
  > pom.xml
  README.md
  > testng.xml
```

©Simplilearn. All rights reserved.

## How To Rerun a Failed Scenario?

Create new runner class

```
@CucumberOptions(

        glue = { "com.app.stepdefinition" }, // path of step
definition

        plugin = { "pretty",
                "html:./reports/cucumber-reports/cucumber-
html/index.html",
                "rerun:target/failedrerun.txt"},
                monochrome =true,
                features = { "@target/failedrerun.txt" }

                )

public class FailedRun extends AbstractTestNGCucumberTests{}
```

Now users can simply run the FailedRun class after automation suite execution in case of test failures. It will execute only the failed scenarios and update the text file again.

©Simplilearn. All rights reserved.

TECHNOLOGY

## Variables

©Simplilearn. All rights reserved.

## Variables

Cucumber uses environment variables to enable certain features, such as publishing cucumber reports.

- It guides on how to define the **CUCUMBER_PUBLISH_TOKEN** environment variable with value **some-secret-token**.

©Simplilearn. All rights reserved.

## Variables

For security reasons users should not define environment variables containing secrets globally.

For MacOS and Linux users this means you should not define them in **~/.bashrc**, **~/.bash_profile**, **~/.zshrc**, **/etc.profile** or similar.

For Windows users this means you should not define them via **System/Control Panel** or **setx.exe**.

©Simplilearn. All rights reserved.

## Variables

Defining variables on different platforms:

**Terminal**

- If users are using a terminal to run Cucumber, you should define environment variables in the same terminal.
- This also applies to terminals embedded in an editor such as Visual Studio Code or IntelliJ IDEA.

**Windows**

```
setx /M CUCUMBER_PUBLISH_TOKEN "some-secret-token"
```

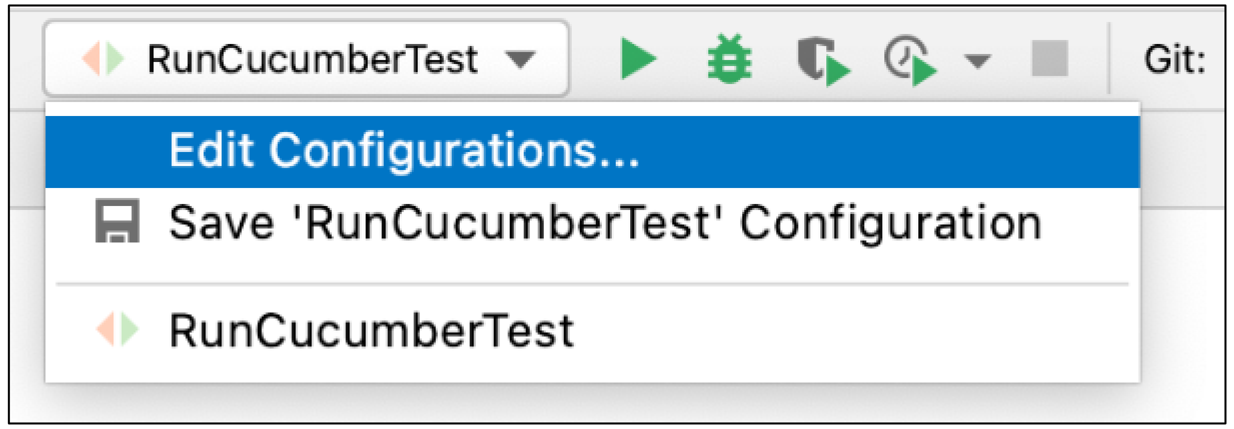
©Simplilearn. All rights reserved.

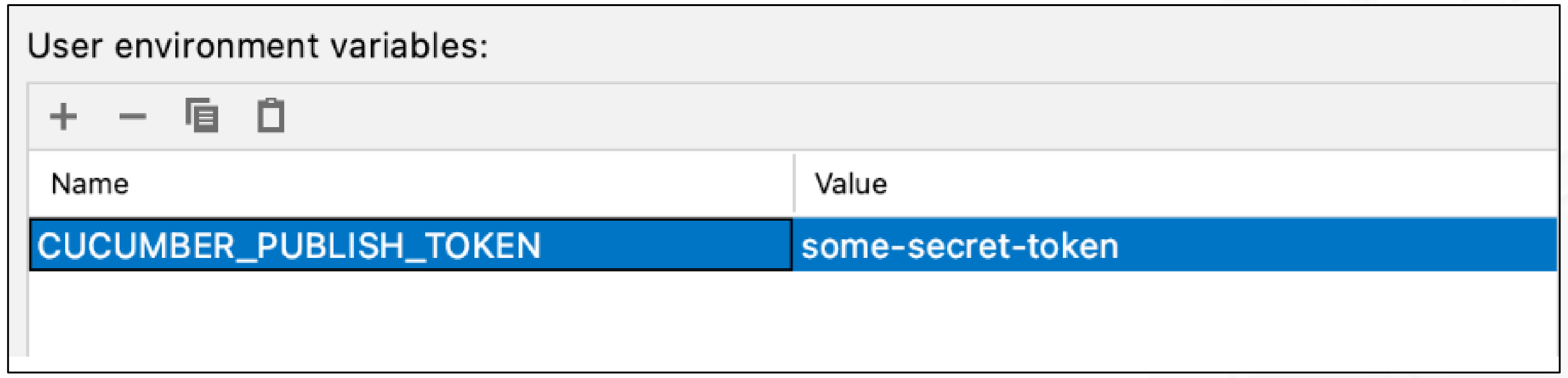
## Variables

Defining variables on different platforms:

**IntelliJ IDEA / WebStorm / RubyMine**

Click the **Run/Debug Configuration** dropdown in the toolbar:




User environment variables:

| Name | Value |
|---|---|
| CUCUMBER_PUBLISH_TOKEN | some-secret-token |

©Simplilearn. All rights reserved.

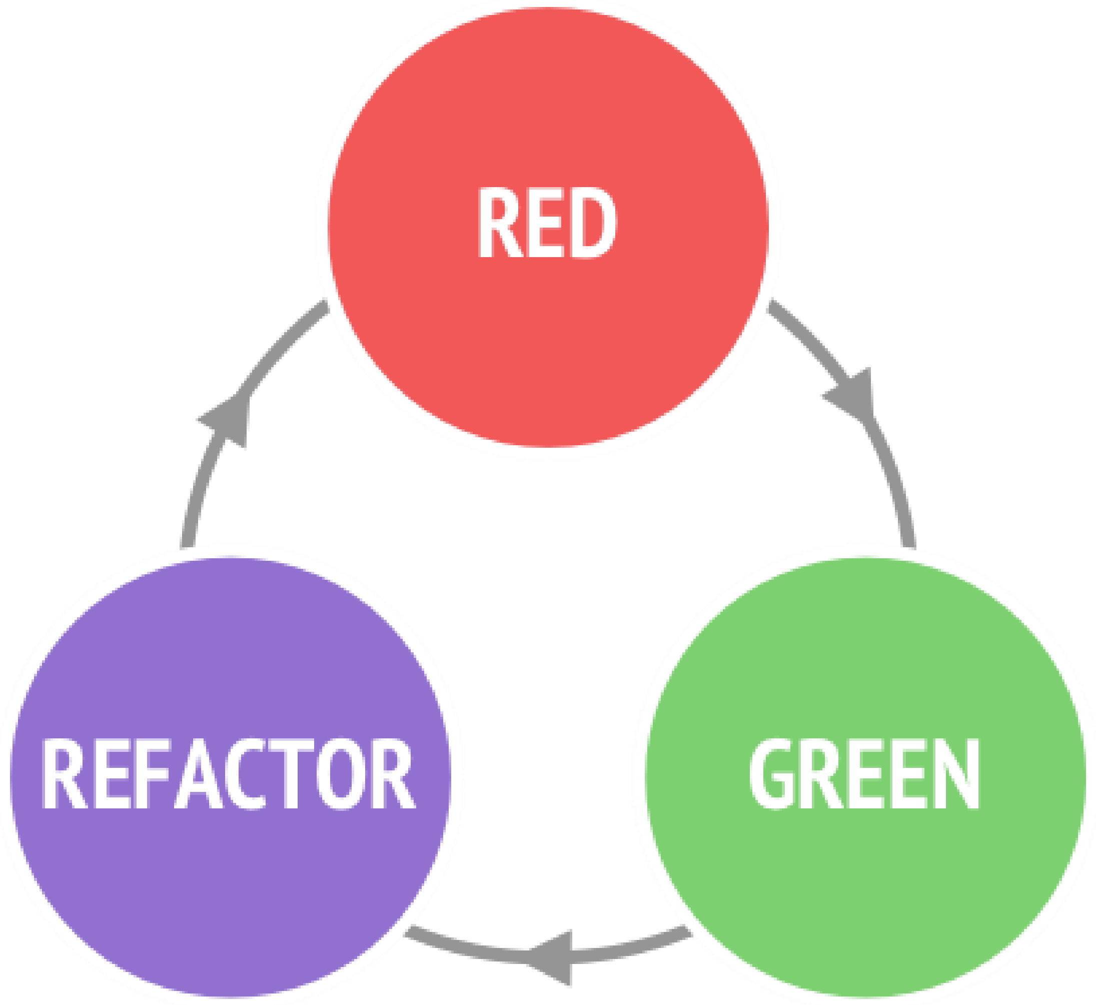
## Refactoring

©Simplilearn. All rights reserved.

simplilearn

## Refactoring

The process of restructuring the code, while not changing its original functionality. Tests should be reviewed and refactored continuously, just like code.



©Simplilearn. All rights reserved.

## Refactoring

This section describes the main refactoring principles:

Promote

Inline

Rename

- When user have a group of steps which they want to reuse across multiple scenarios, they can extract them into an action word.
- This refactoring technique makes your steps reusable and easy to maintain.

©Simplilearn. All rights reserved.

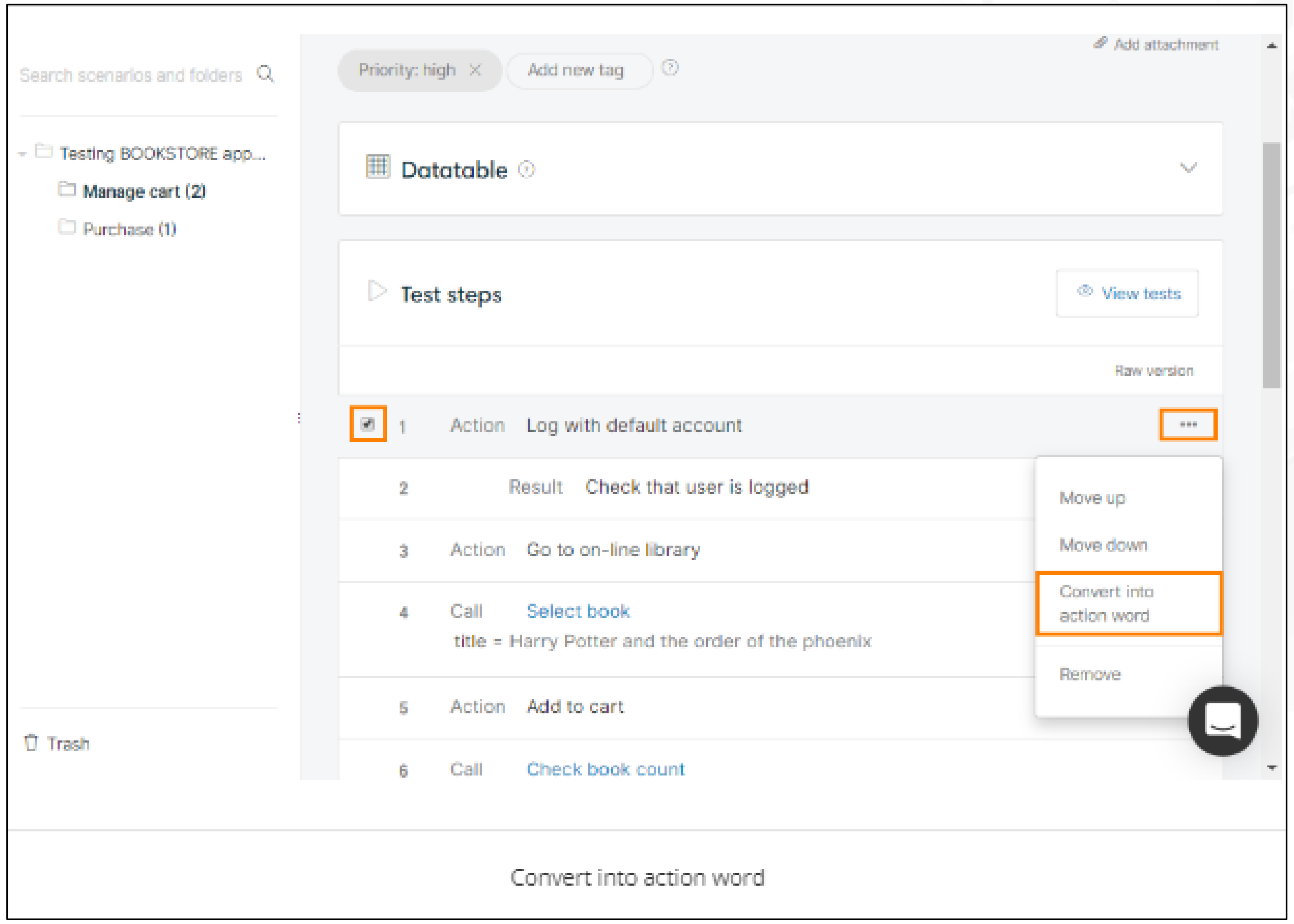
# Refactoring

Promote

Inline

Rename

First, select the **group of steps** and select **Convert** into action word:



1 Action Log with default account
2 Result Check that user is logged
3 Action Go to on-line library
4 Call Select book
title = Harry Potter and the order of the phoenix
5 Action Add to cart
6 Call Check book count

Convert into action word

©Simplilearn. All rights reserved.

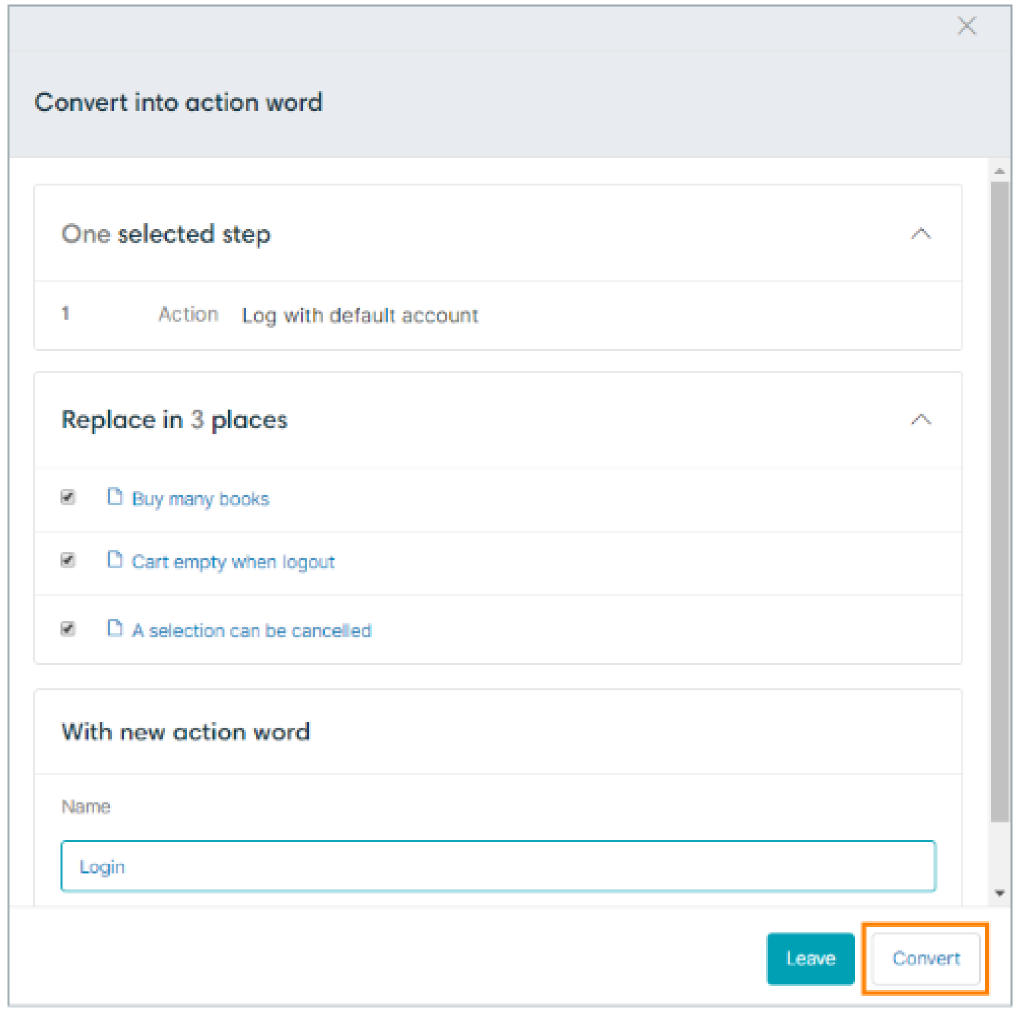
# Refactoring

Promote

Inline

Rename

If several scenarios already use this group of steps, then select the ones users want to apply the refactoring to:



Convert into action word

One selected step

1 Action Log with default account

Replace in 3 places

- [x] Buy many books
- [x] Cart empty when logout
- [x] A selection can be cancelled

With new action word

Name: Login

Convert into action word modal

©Simplilearn. All rights reserved.

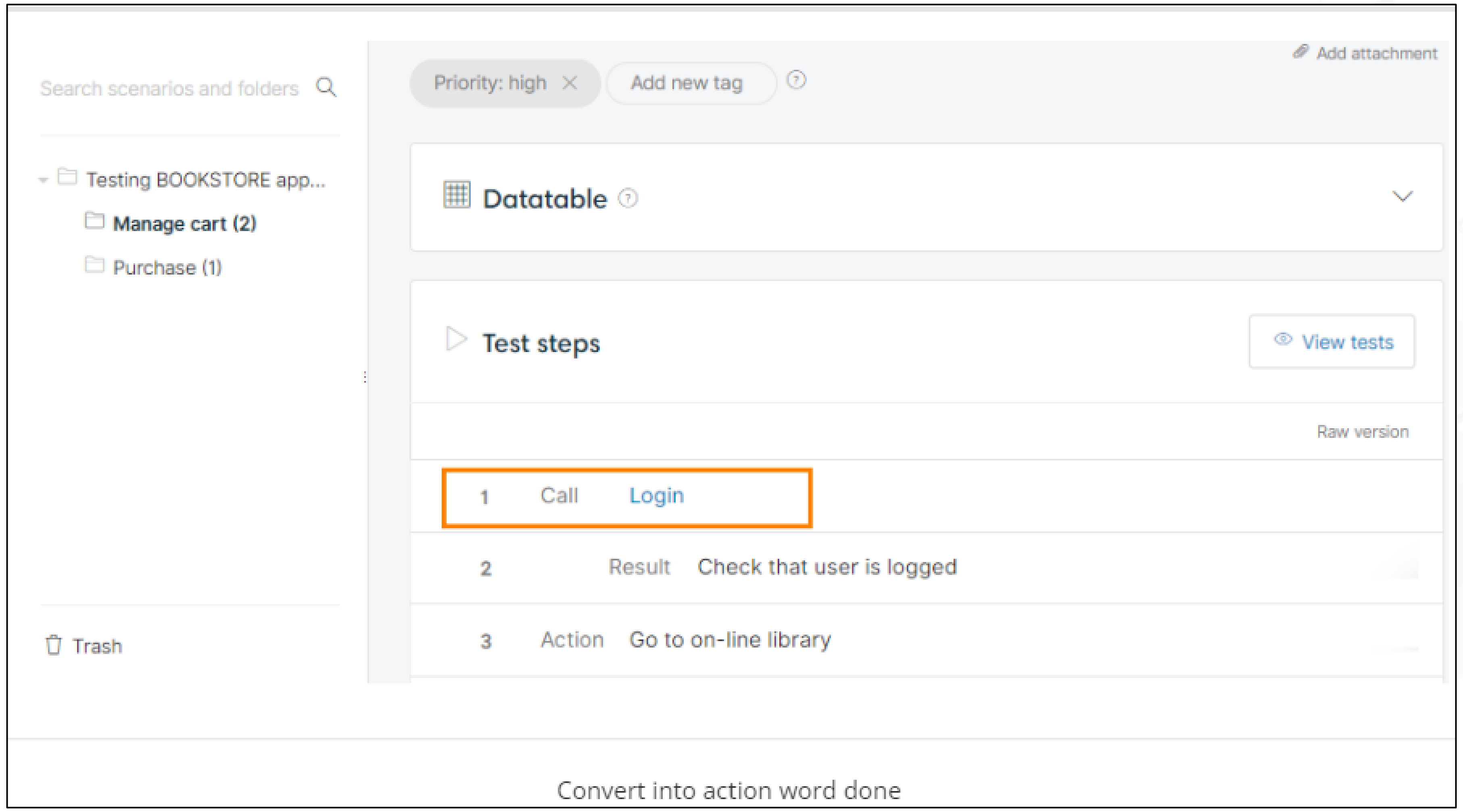
# Refactoring

Promote

Inline

Rename

Refactoring is completed.



| | | |
|---|---|---|
| 1 | Call | Login |
| 2 | Result | Check that user is logged |
| 3 | Action | Go to on-line library |

Convert into action word done

©Simplilearn. All rights reserved.

## Refactoring

Promote

Inline

Rename

- Inline is the opposite of promote.
- It replaces an action word by its steps in every object where this action word is called.

©Simplilearn. All rights reserved.

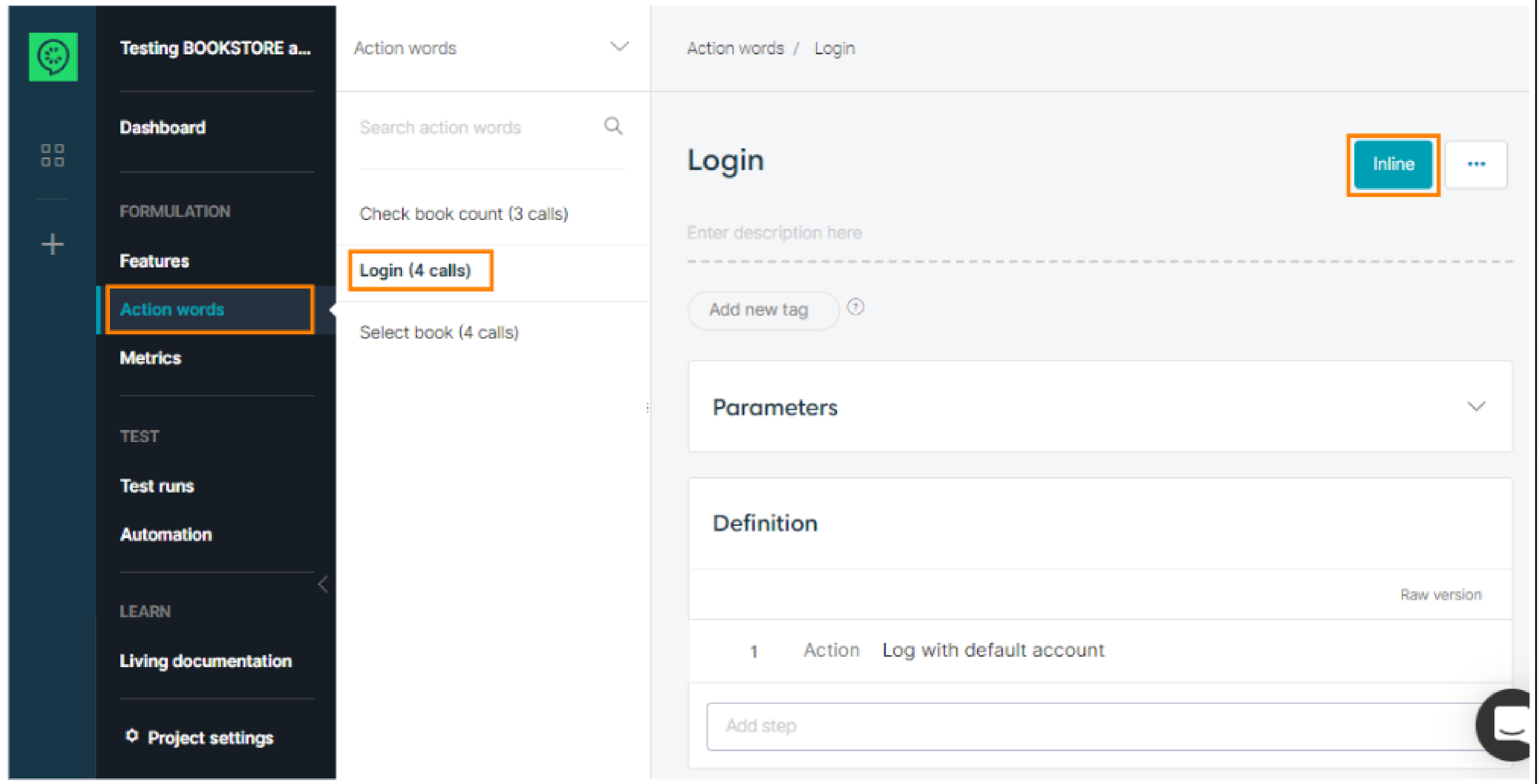
# Refactoring

Promote

Inline

Rename

Select **Action words** and click on **Inline**:



Action word inline

©Simplilearn. All rights reserved.

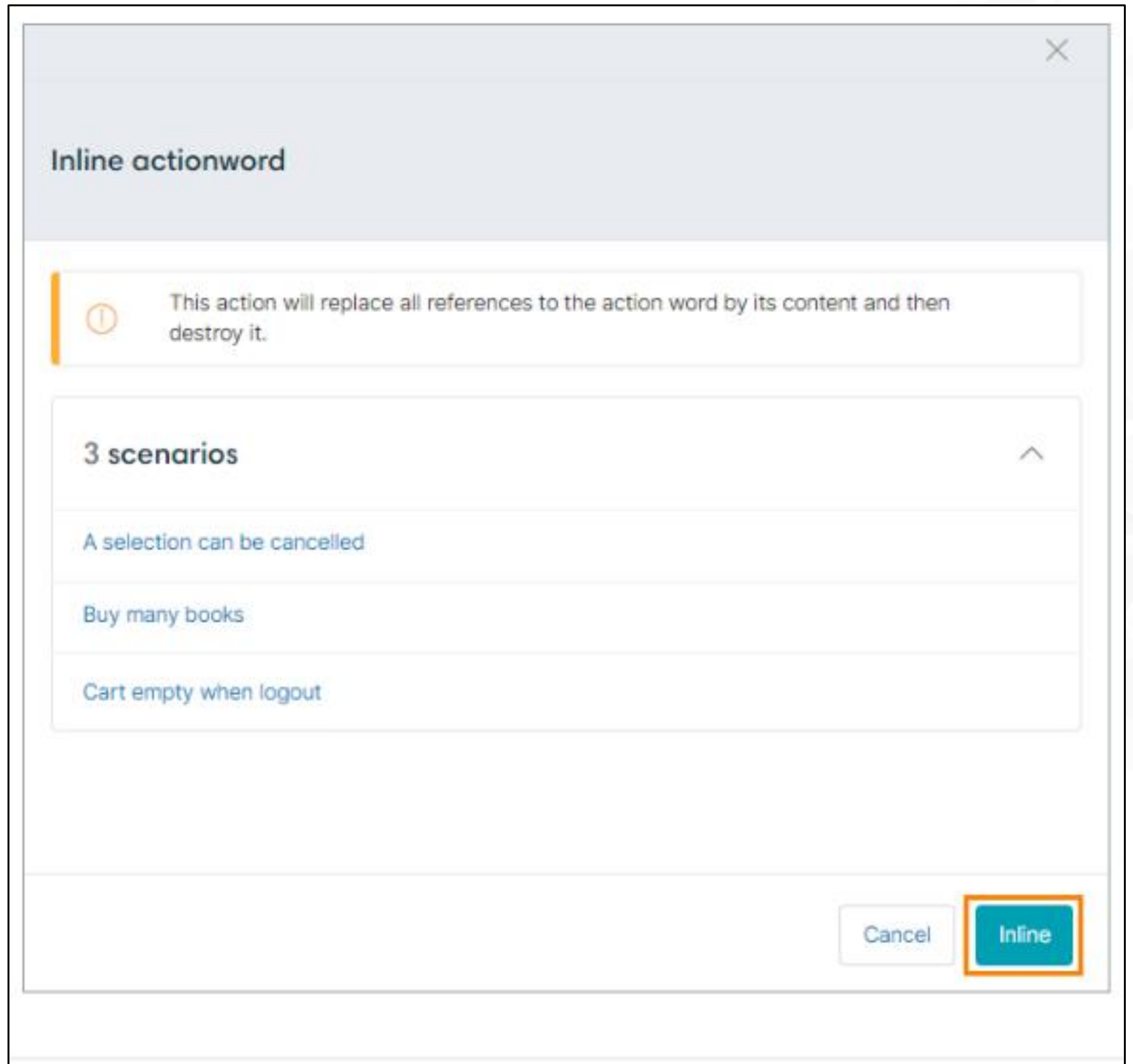
## Refactoring

Promote

Inline

Rename

Review all the objects (scenarios and action words) calling the action word to be inlined.


Inline actionword

This action will replace all references to the action word by its content and then destroy it.

3 scenarios

A selection can be cancelled

Buy many books

Cart empty when logout

Action word inline modal

©Simplilearn. All rights reserved.

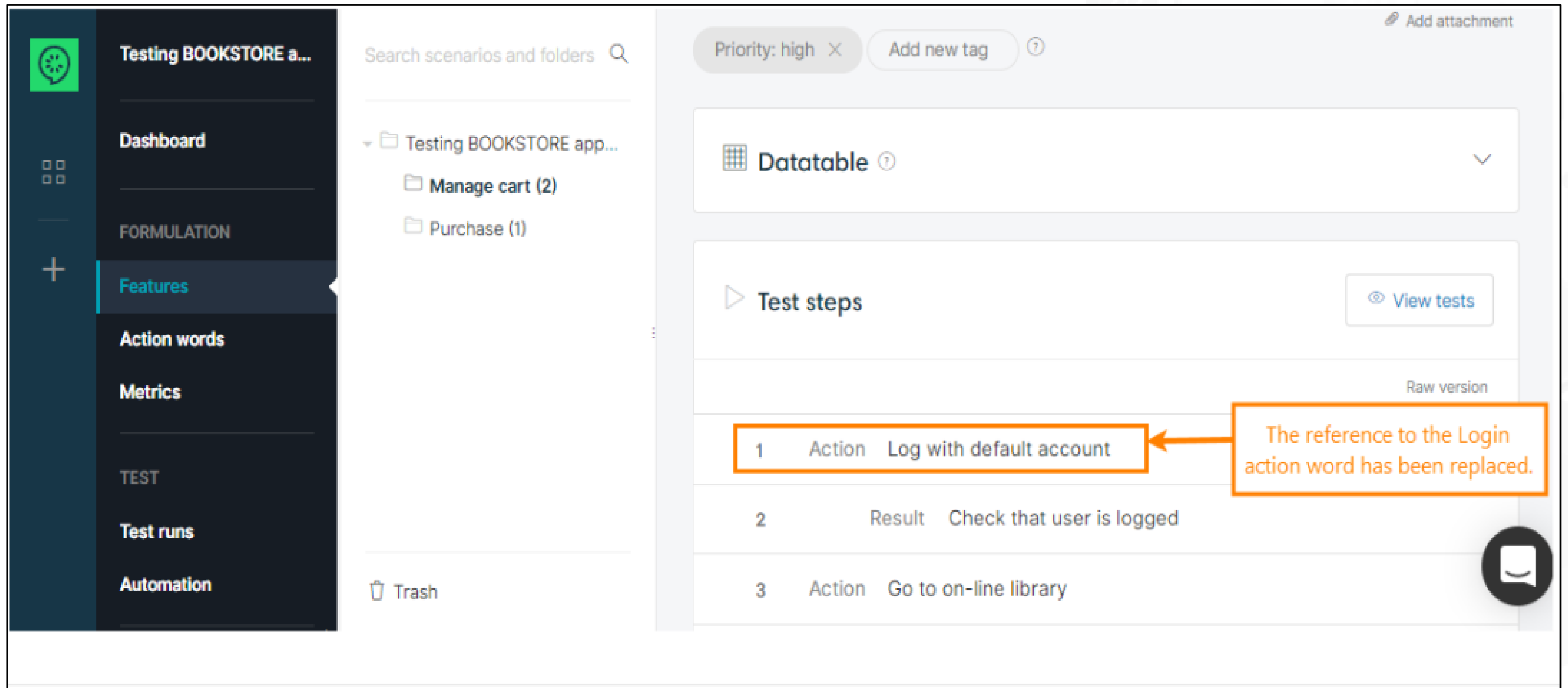
# Refactoring

Promote

Inline

Rename

All the references to this action word have been replaced by the steps of the action word.



1 Action Log with default account

2 Result Check that user is logged

3 Action Go to on-line library

The reference to the Login action word has been replaced.

Action word inline done

©Simplilearn. All rights reserved.

## Refactoring

Promote

Inline

Rename

Make sure that the action word naming is meaningful because it keeps test readability and do not hesitate to rename an action word: all the references (call) to this action word are updated automatically.

©Simplilearn. All rights reserved.

simplilearn

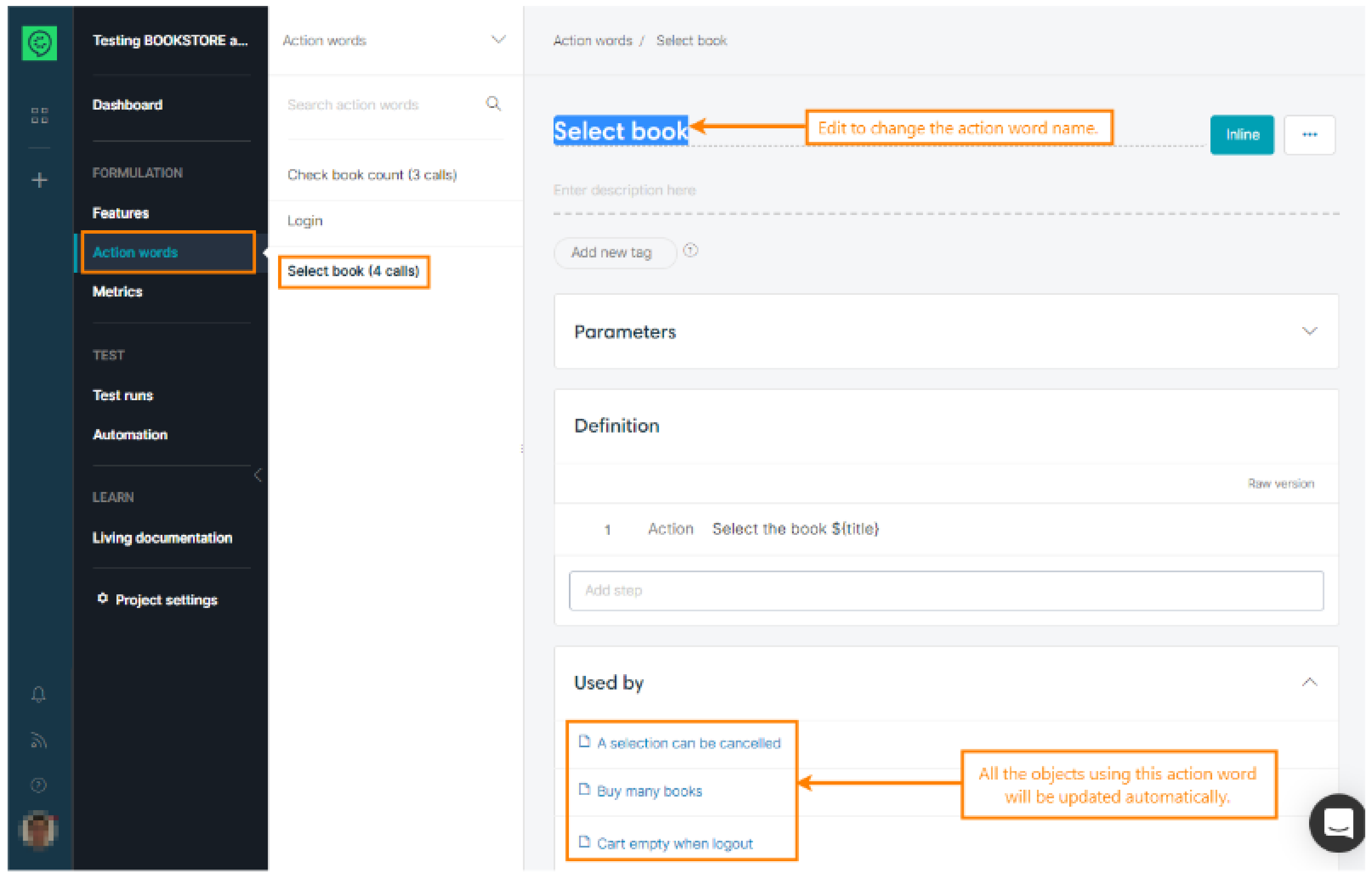
# Refactoring

Promote

Inline

Rename

Action word renaming:



Action word renaming

©Simplilearn. All rights reserved.

## Key Takeaways

- Steps that are undefinable despite the existence of defined step definitions are called undefined steps.
- When a step definition's method or function is executed and raises an error, the step is marked as failed.
- Cucumber uses environment variables to enable certain features.
- The process of restructuring code, while not changing its original functionality is known as refactoring.

©Simplilearn. All rights reserved.