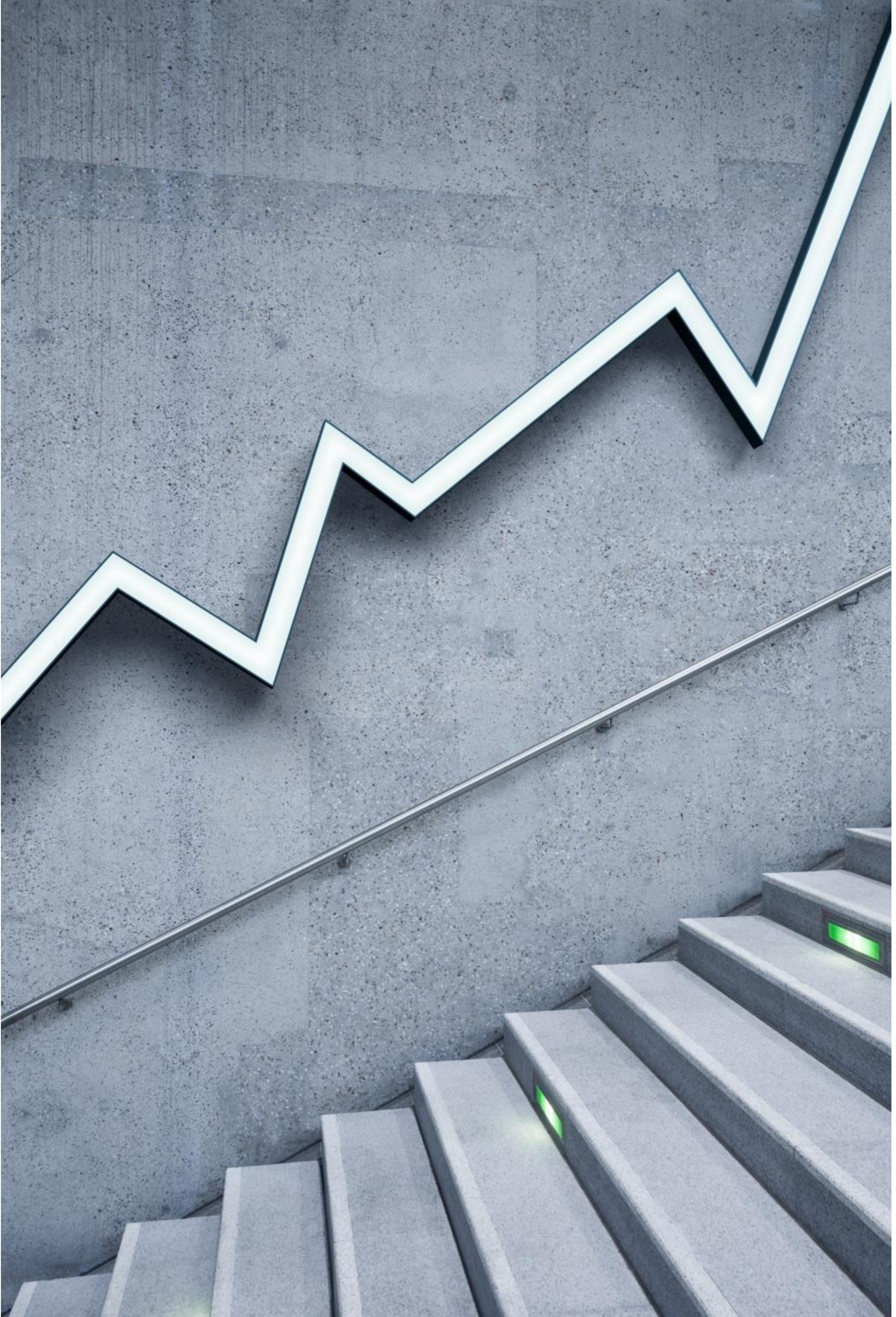
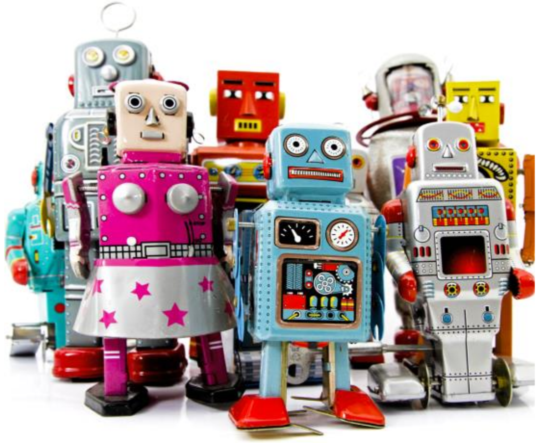


Robot Framework Training



AUN
RATTHANUN PONGSAIYA
SENIOR QUALITY ASSURANCE

Agenda

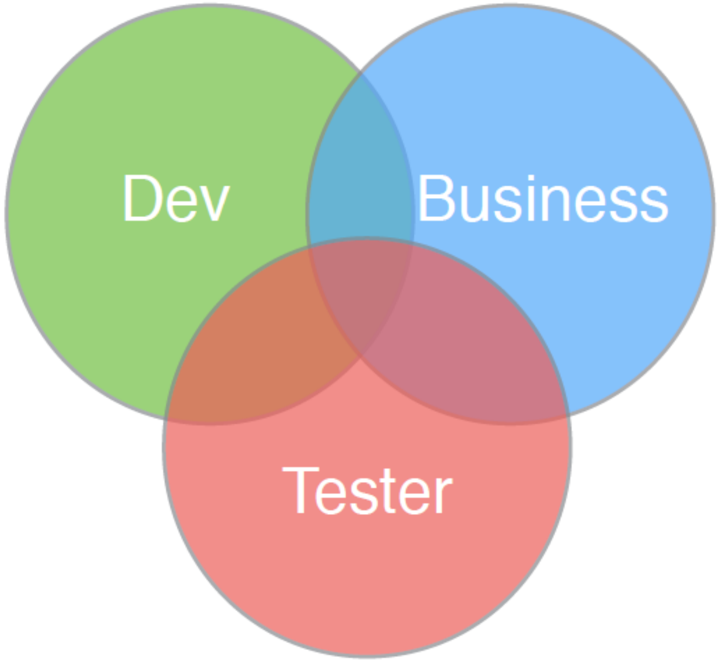
- How to create and analyze the test cases
- Acceptance Test-Driven Development
- Concept of Robot Framework
- Robot framework Installation
- Architecture of Robot Framework
- Web UI Testing
- API Testing

Acceptance Test-Driven Development

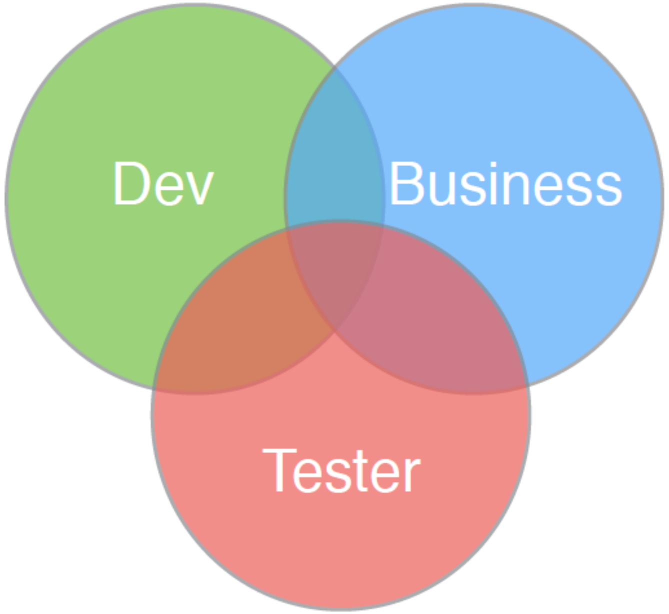
ATDD is development methodology which promote good collaboration between business and technology group. There are many similar methodologies which works more or less same way like Behavior Driven Development (BDD), Example Driven Development (EDD) and Specification by Examples etc.

- Common understanding
- Common language
- Executable requirements or examples
- Living documentation

Acceptance Testing



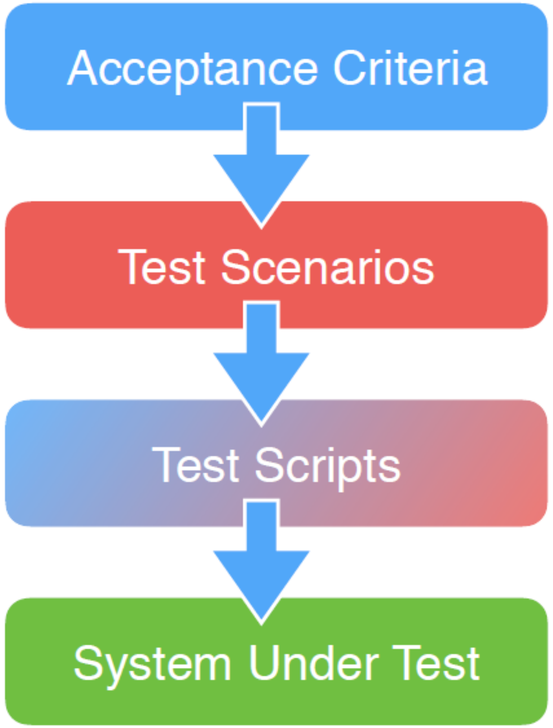
Acceptance Testing



Dev

Business

Tester



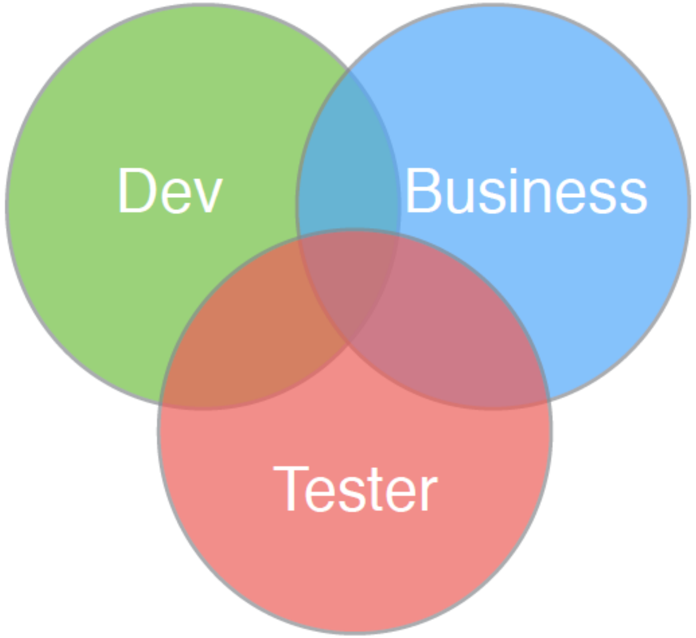
Acceptance Criteria

Test Scenarios

Test Scripts

System Under Test

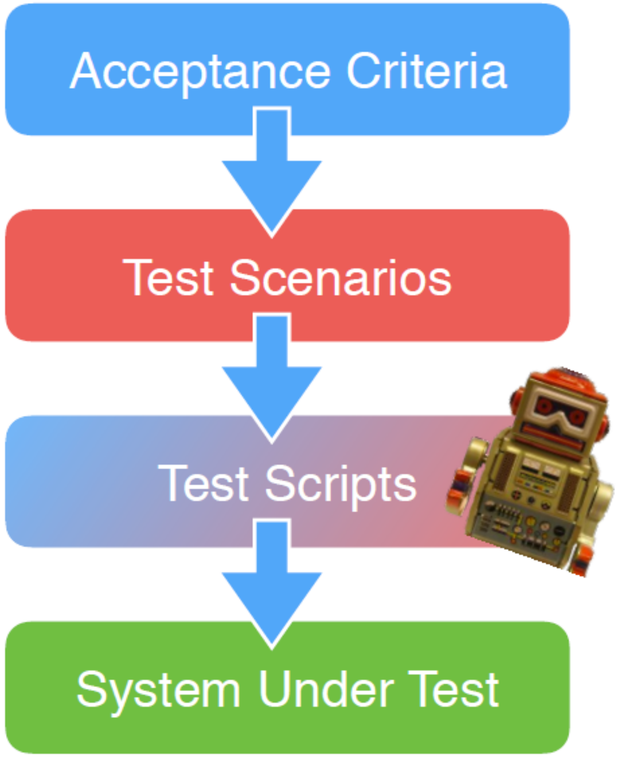
Acceptance Testing



Dev

Business

Tester



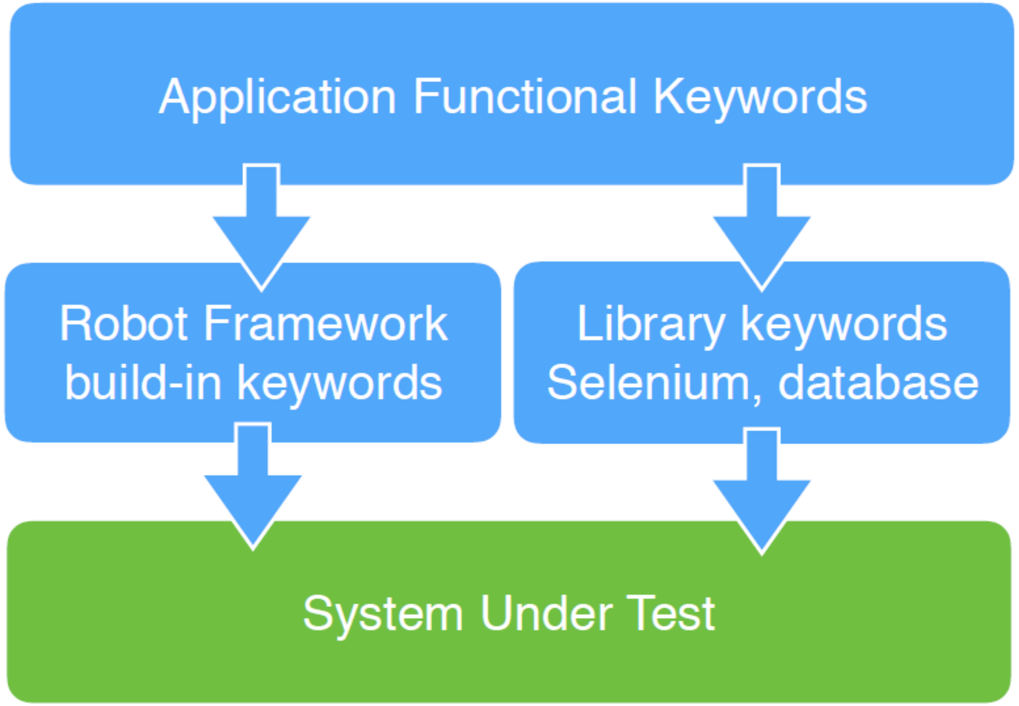
Acceptance Criteria

Test Scenarios

Test Scripts

System Under Test

Keyword Driven Development



Application Functional Keywords

Robot Framework build-in keywords

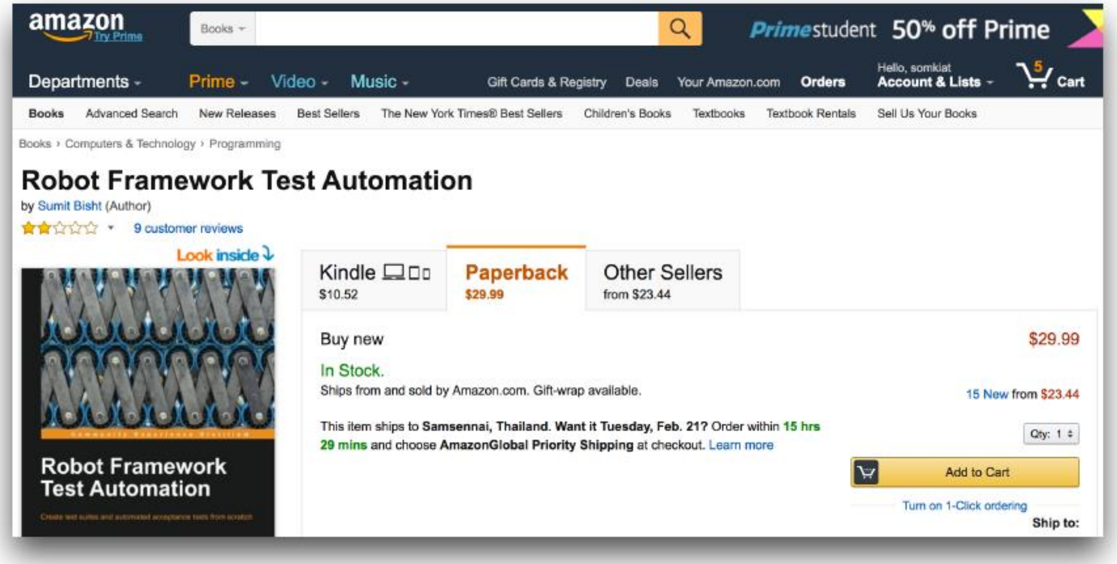
Library keywords Selenium, database

System Under Test

Keyword Driven Development

Example scenario

“Add product to cart”



Robot Framework Test Automation

by Sumit Bisht (Author)

9 customer reviews

Buy new $29.99

In Stock.

Ships from and sold by Amazon.com. Gift-wrap available.

15 New from $23.44

This item ships to **Samsennai, Thailand. Want it Tuesday, Feb. 21?** Order within 15 hrs 29 mins and choose **AmazonGlobal Priority Shipping** at checkout. Learn more

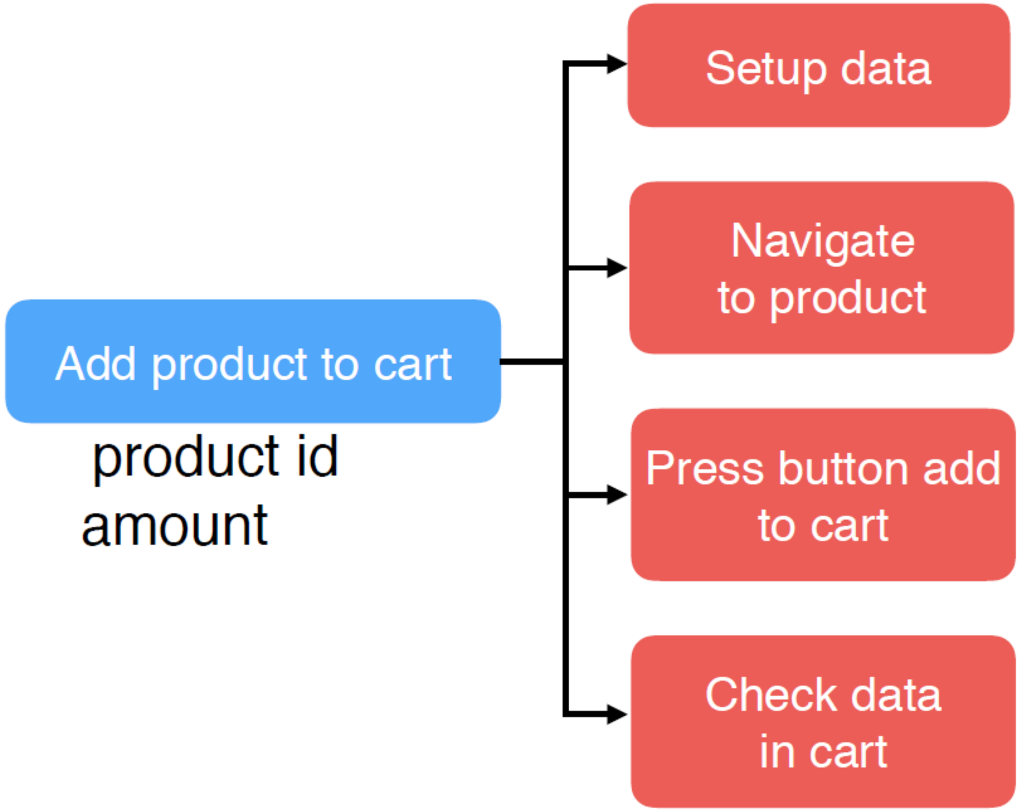
Add product to cart

Add product to cart

product id

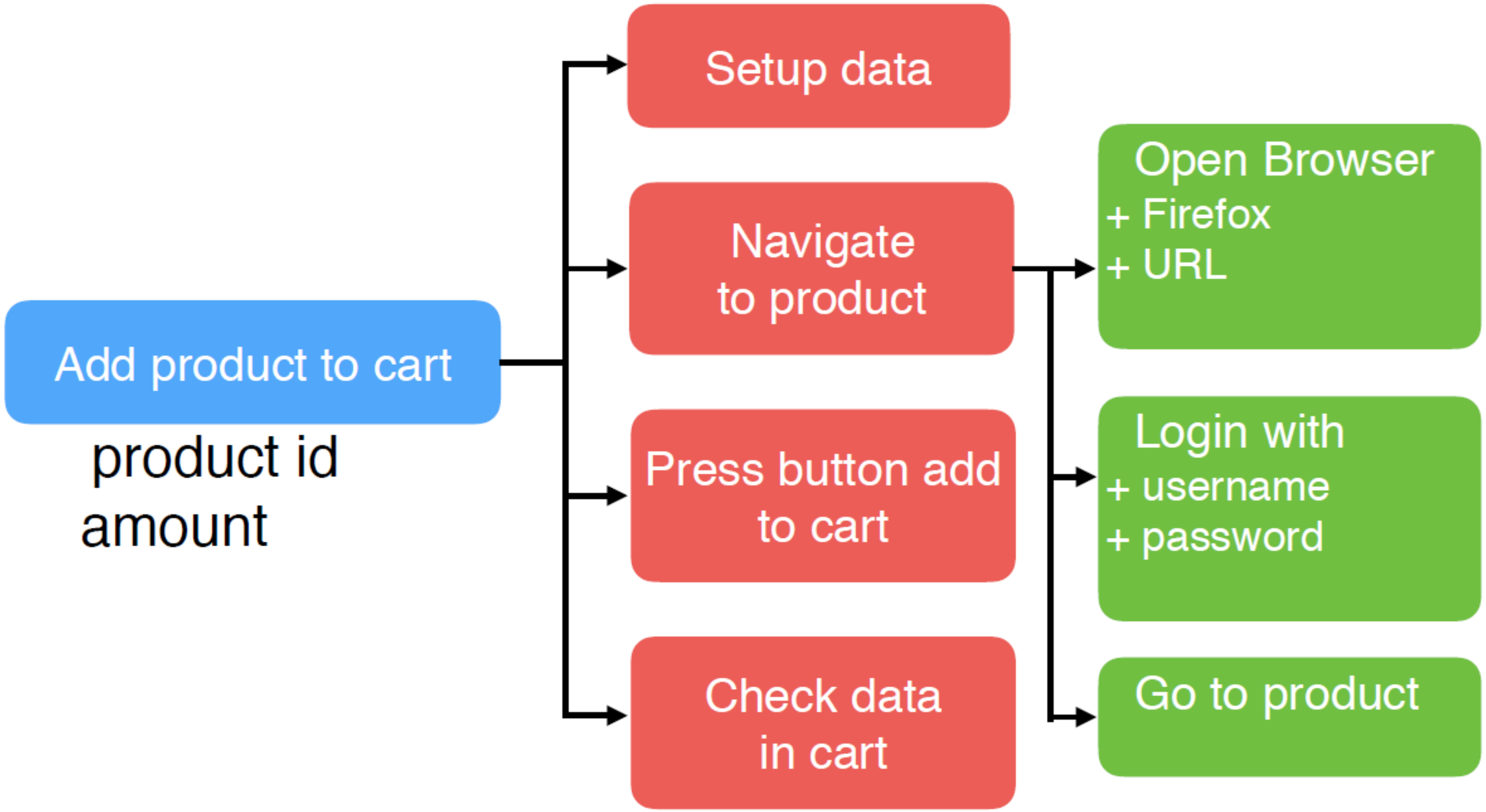
amount

Add product to cart



product id
amount

Add product to cart



Add product to cart

product id
amount

Setup data

Navigate to product

Press button add to cart

Check data in cart

Open Browser
+ Firefox
+ URL

Login with
+ username
+ password

Go to product

Test Case is ?

Test case = Business rule + Test data

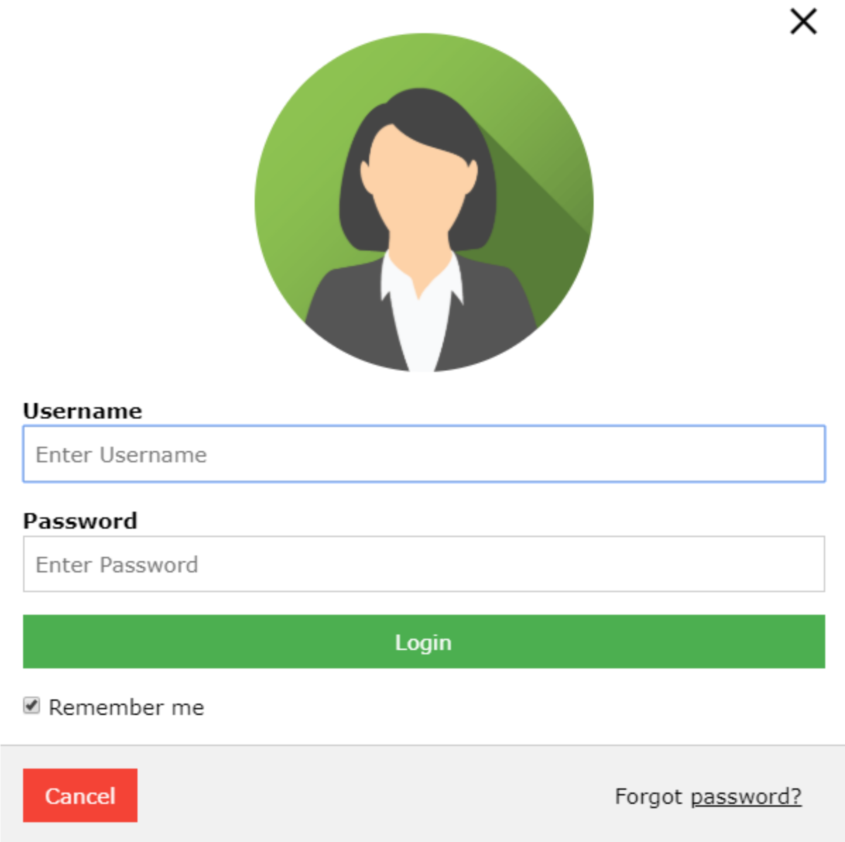
Example: Feature logging

Login with 3 roles and make a permission

Admin = Creator and Editor

Manager = Approver

Staff = View



Test case

- Login Success with valid username and password
- Login Fail with invalid username and password
- Login with inactive account
- Login with deleted account
- Display the data depend on the permission

Write the test case with Gherkin language

It is a Business Readable, Domain Specific Language that lets you describe software's behavior without detailing how that behavior is implemented.

There are 10 main keywords that helps in writing features and acceptance tests/scenarios/examples.

1. Feature – To describe feature name and functionality
2. Scenario – To describe scenarios of feature
3. Given – Pre-condition for a test/examples
4. When – Action to be performed for test/examples
5. Then – Expected outcome for a test/examples
6. And – Use to write multiples given/when/then
7. But – Similar to "And" to describe examples/test
8. Background – To write repeated steps for scenario
9. Scenario Outline – Used with "Examples" to write multiple scenarios together by having placeholders
10. Examples – Multiples scenarios for scenario outline

Login Success with valid username and password

Feature: Login

Scenario: I want to login into the system with valid username and password by role admin

Given: I open chrome browser

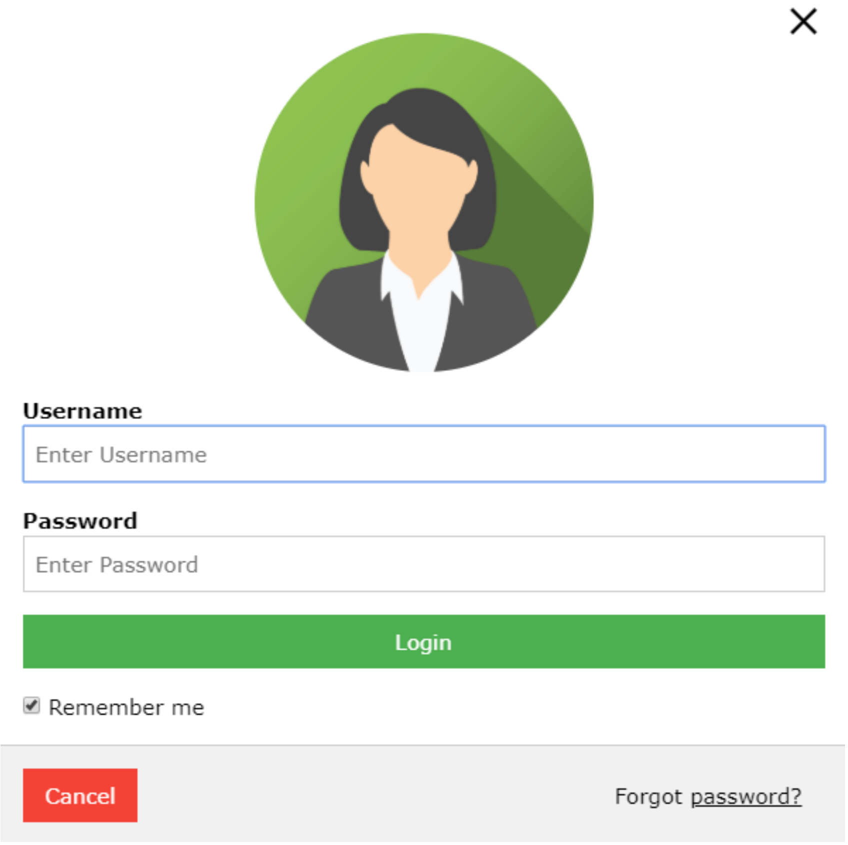
And: I fill username with admin

And: I fill password with passwordAdmin

When: I click login button

Then: I should see the dashboard page

And: I should see my name



Test case

- Login Success with valid username and password
- Login Fail with invalid username and password
- Login with inactive account
- Login with deleted account
- Display the data depend on the permission

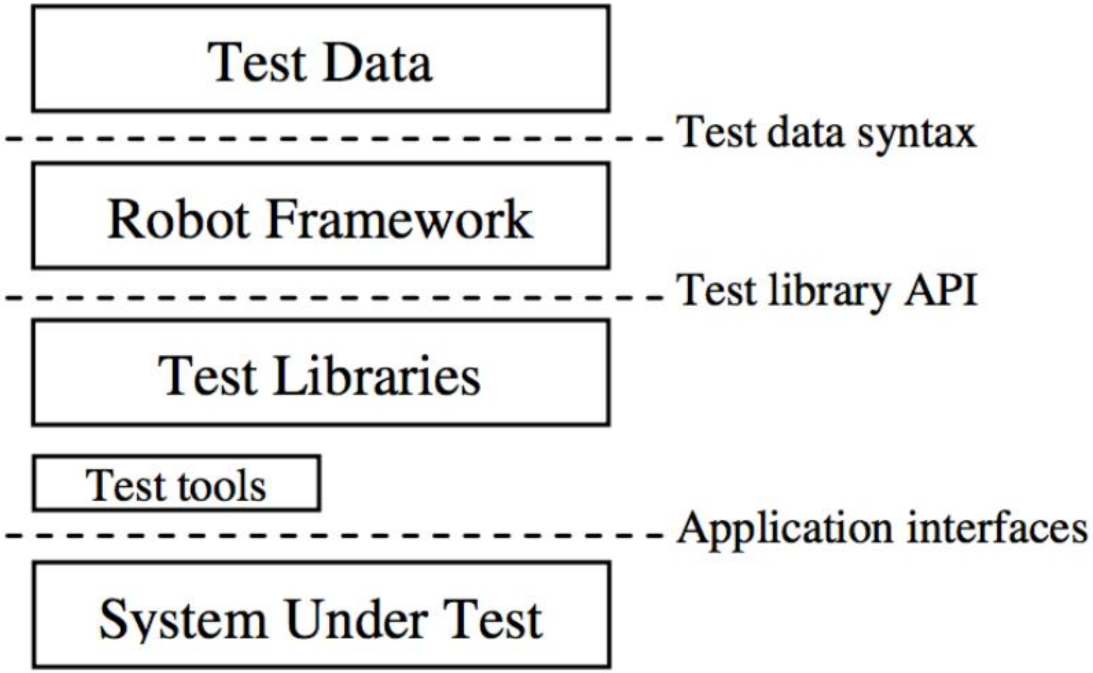
API Methods

1. GET   Use GET requests **to retrieve resource representation/information only**
2. POST Use POST APIs **to create new subordinate resources**, e.g., a file is subordinate to a directory containing it or a row is subordinate to a database table.
3. PUT Use PUT APIs primarily **to update existing resource** (if the resource does not exist, then API may decide to create a new resource or not).
4. DELETE As the name applies, DELETE APIs are used **to delete resources**
5. HEAD method is almost identical to GET, except without the response body.
6. PATCH requests are **to make partial update on a resource**.
7. OPTION request should **return data describing what *other* methods and operations the server supports** at the given URL.

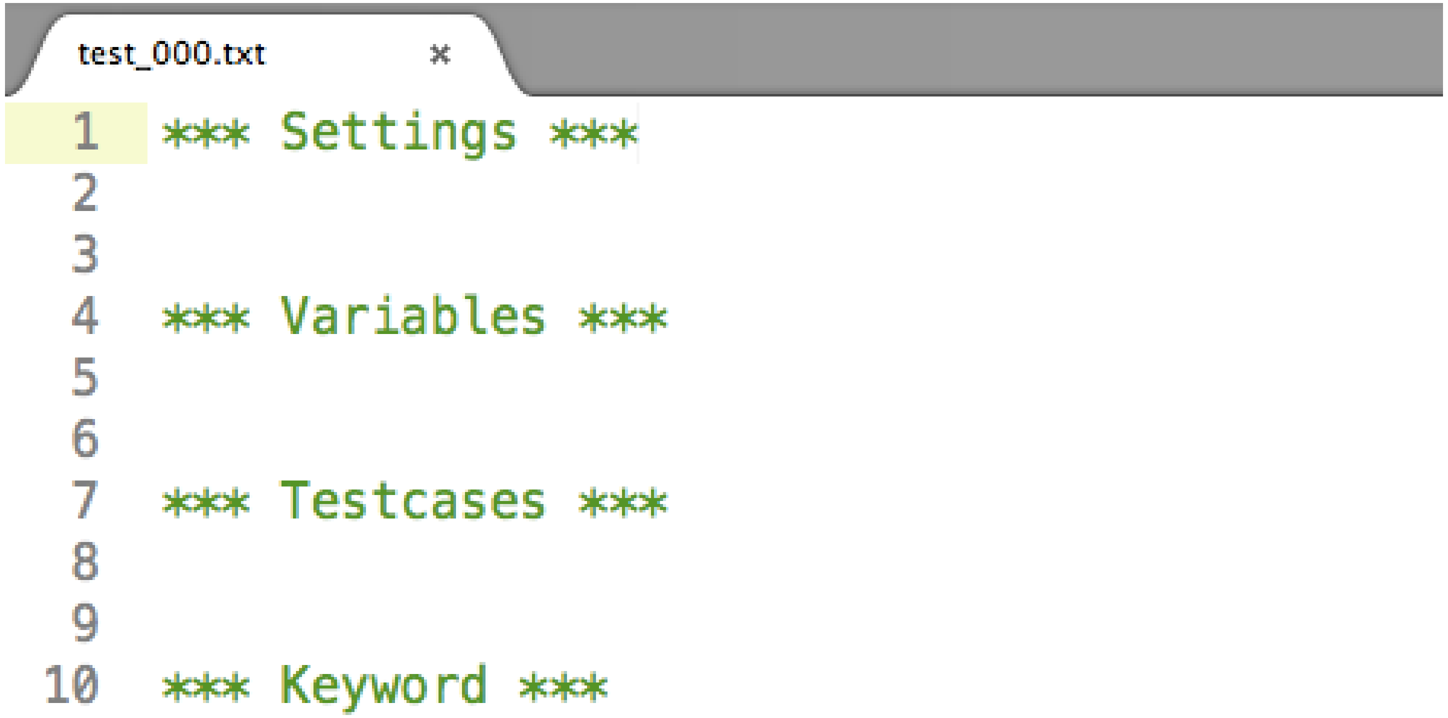
Try it with the postman

http://dummy.restapiexample.com/

High level architecture



Test structure



```
*** Settings ***


*** Variables ***


*** Testcases ***


*** Keyword ***
```

Settings

Alias Name => Setting, Settings,

Metadata

Settings

Define metadata for test suites and test cases

Import test libraries

Resource files

Variables

Alias Name => Variable, Variables

Test cases

Alias Name => Testcases, Test Case, Test Cases

Create test cases from keywords

Keywords

Alias Name => Keyword, Keywords, User Keyword, User Keywords

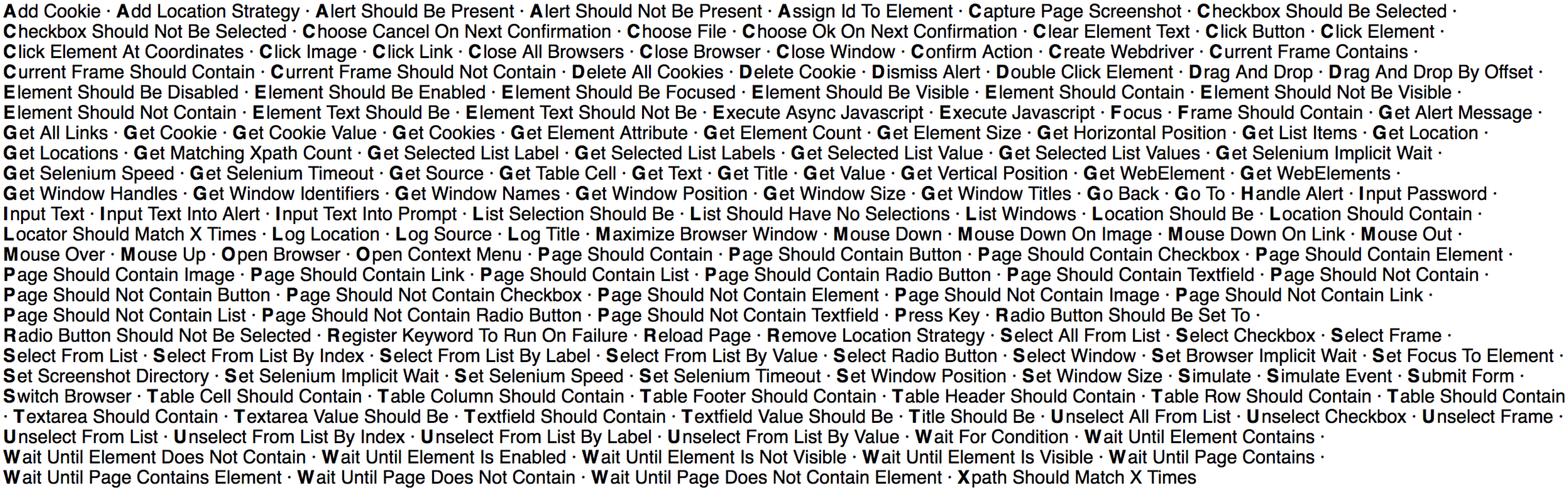
Create user keywords from existing low-level keyword

Web UI Testing

SeleniumLibrary

https://github.com/robotframework/SeleniumLibrary/

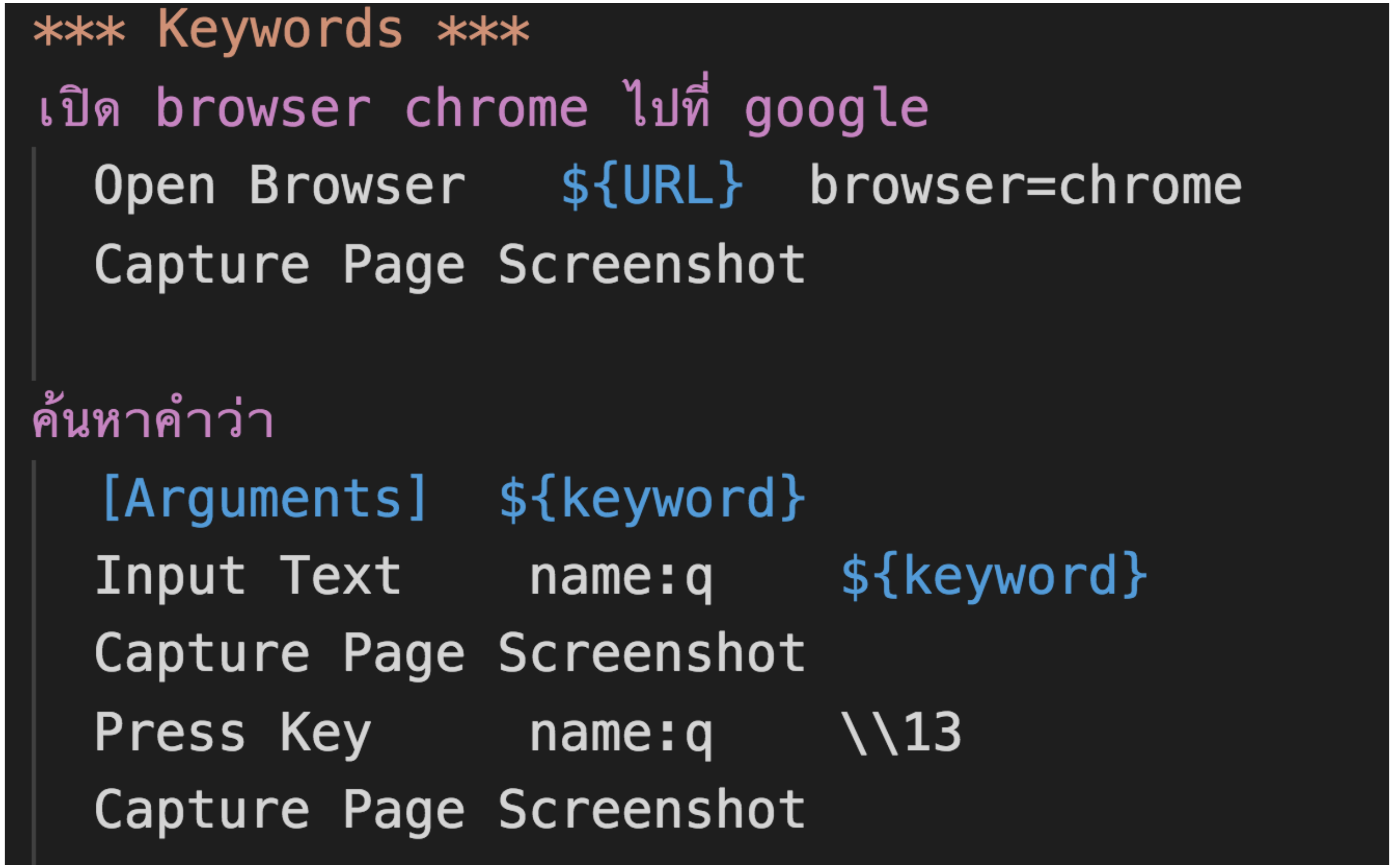
SeleniumLibrary Keywords



Add Cookie · Add Location Strategy · Alert Should Be Present · Alert Should Not Be Present · Assign Id To Element · Capture Page Screenshot · Checkbox Should Be Selected · Checkbox Should Not Be Selected · Choose Cancel On Next Confirmation · Choose File · Choose Ok On Next Confirmation · Clear Element Text · Click Button · Click Element · Click Element At Coordinates · Click Image · Click Link · Close All Browsers · Close Browser · Close Window · Confirm Action · Create Webdriver · Current Frame Contains · Current Frame Should Contain · Current Frame Should Not Contain · Delete All Cookies · Delete Cookie · Dismiss Alert · Double Click Element · Drag And Drop · Drag And Drop By Offset · Element Should Be Disabled · Element Should Be Enabled · Element Should Be Focused · Element Should Be Visible · Element Should Contain · Element Should Not Be Visible · Element Should Not Contain · Element Text Should Be · Element Text Should Not Be · Execute Async Javascript · Execute Javascript · Focus · Frame Should Contain · Get Alert Message · Get All Links · Get Cookie · Get Cookie Value · Get Cookies · Get Element Attribute · Get Element Count · Get Element Size · Get Horizontal Position · Get List Items · Get Location · Get Locations · Get Matching Xpath Count · Get Selected List Label · Get Selected List Labels · Get Selected List Value · Get Selected List Values · Get Selenium Implicit Wait · Get Selenium Speed · Get Selenium Timeout · Get Source · Get Table Cell · Get Text · Get Title · Get Value · Get Vertical Position · Get WebElement · Get WebElements · Get Window Handles · Get Window Identifiers · Get Window Names · Get Window Position · Get Window Size · Get Window Titles · Go Back · Go To · Handle Alert · Input Password · Input Text · Input Text Into Alert · Input Text Into Prompt · List Selection Should Be · List Should Have No Selections · List Windows · Location Should Be · Location Should Contain · Locator Should Match X Times · Log Location · Log Source · Log Title · Maximize Browser Window · Mouse Down · Mouse Down On Image · Mouse Down On Link · Mouse Out · Mouse Over · Mouse Up · Open Browser · Open Context Menu · Page Should Contain · Page Should Contain Button · Page Should Contain Checkbox · Page Should Contain Element · Page Should Contain Image · Page Should Contain Link · Page Should Contain List · Page Should Contain Radio Button · Page Should Contain Textfield · Page Should Not Contain · Page Should Not Contain Button · Page Should Not Contain Checkbox · Page Should Not Contain Element · Page Should Not Contain Image · Page Should Not Contain Link · Page Should Not Contain List · Page Should Not Contain Radio Button · Page Should Not Contain Textfield · Press Key · Radio Button Should Be Set To · Radio Button Should Not Be Selected · Register Keyword To Run On Failure · Reload Page · Remove Location Strategy · Select All From List · Select Checkbox · Select Frame · Select From List · Select From List By Index · Select From List By Label · Select From List By Value · Select Radio Button · Select Window · Set Browser Implicit Wait · Set Focus To Element · Set Screenshot Directory · Set Selenium Implicit Wait · Set Selenium Speed · Set Selenium Timeout · Set Window Position · Set Window Size · Simulate · Simulate Event · Submit Form · Switch Browser · Table Cell Should Contain · Table Column Should Contain · Table Footer Should Contain · Table Header Should Contain · Table Row Should Contain · Table Should Contain · Textarea Should Contain · Textarea Value Should Be · Textfield Should Contain · Textfield Value Should Be · Title Should Be · Unselect All From List · Unselect Checkbox · Unselect Frame · Unselect From List · Unselect From List By Index · Unselect From List By Label · Unselect From List By Value · Wait For Condition · Wait Until Element Contains · Wait Until Element Does Not Contain · Wait Until Element Is Enabled · Wait Until Element Is Not Visible · Wait Until Element Is Visible · Wait Until Page Contains · Wait Until Page Contains Element · Wait Until Page Does Not Contain · Wait Until Page Does Not Contain Element · Xpath Should Match X Times

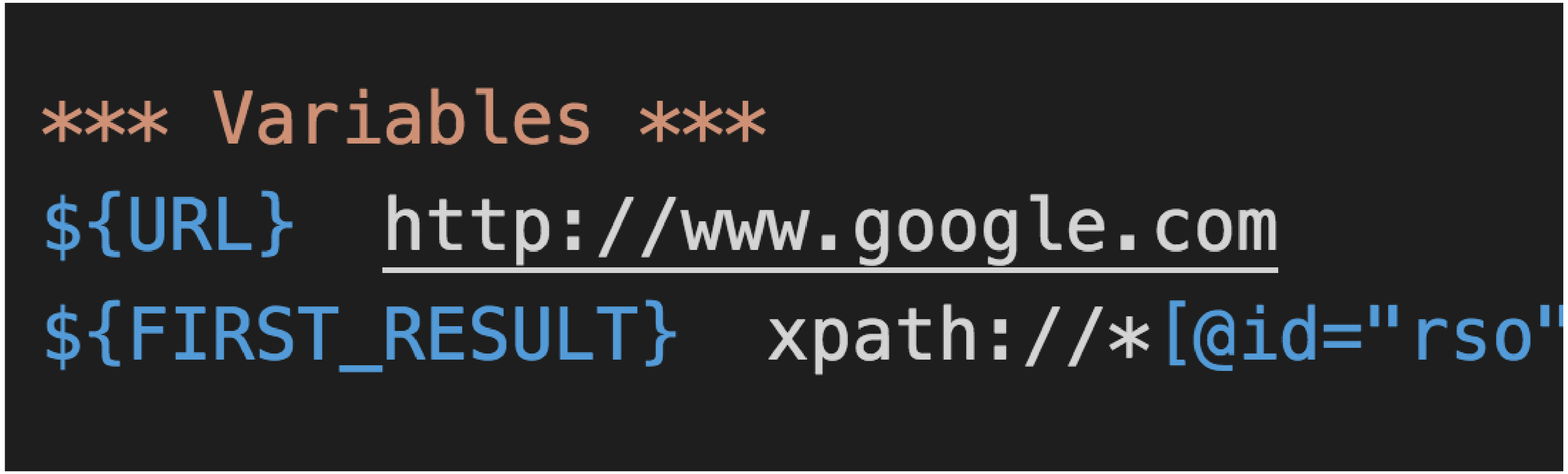
Write your keyword

Keywords



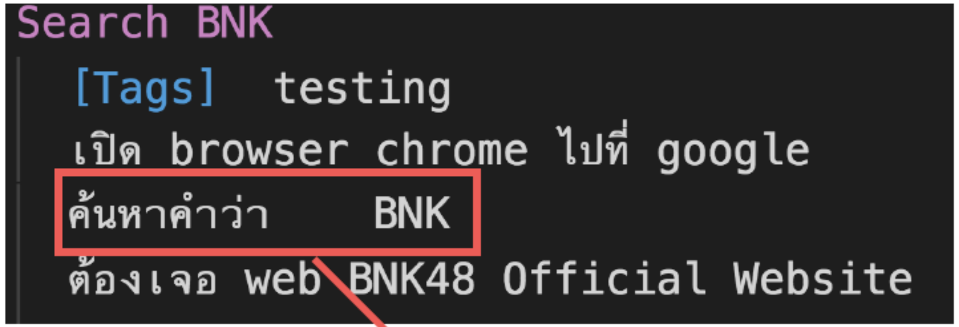
```
*** Keywords ***
เปิด browser chrome ไปที่ google
  Open Browser   ${URL}  browser=chrome
  Capture Page Screenshot

ค้นหาคำว่า
  [Arguments]  ${keyword}
  Input Text    name:q    ${keyword}
  Capture Page Screenshot
  Press Key     name:q    \\13
  Capture Page Screenshot
```

Variables

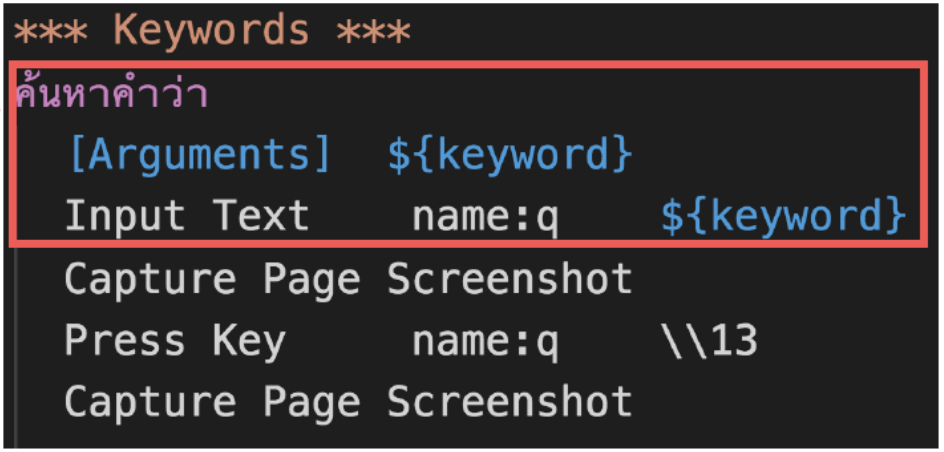


```
*** Variables ***
${URL}  http://www.google.com
${FIRST_RESULT}  xpath://*[@id="rso"
```

Keyword Arguments



```
Search BNK
    [Tags]   testing
    เปิด browser chrome ไปที่ google
    ค้นหาคำว่า     BNK
    ต้องเจอ web BNK48 Official Website
```

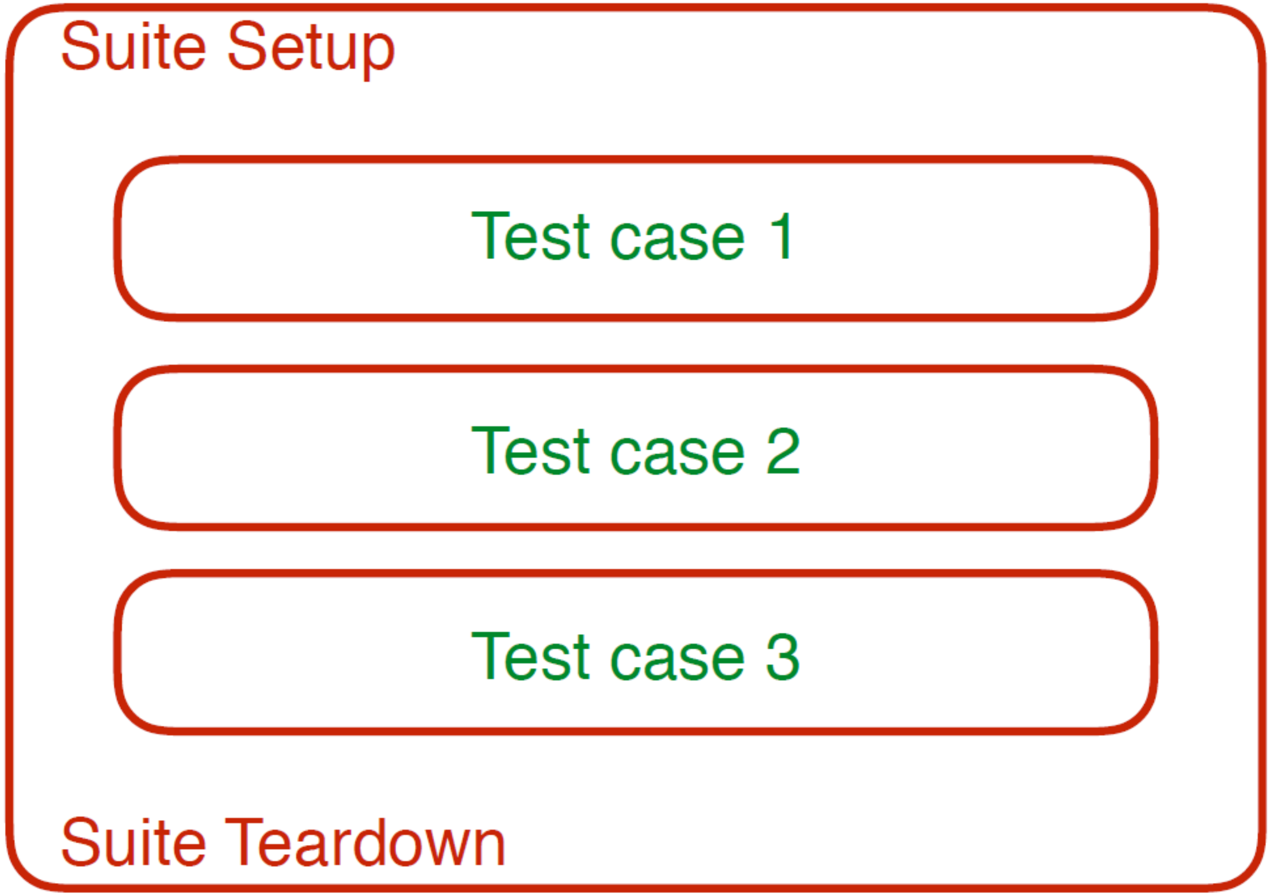


```
*** Keywords ***
ค้นหาคำว่า
    [Arguments]   ${keyword}
    Input Text      name:q     ${keyword}
    Capture Page Screenshot
    Press Key       name:q     \\13
    Capture Page Screenshot
```

Test Life Cycle

Suite

Suite



Suite Setup

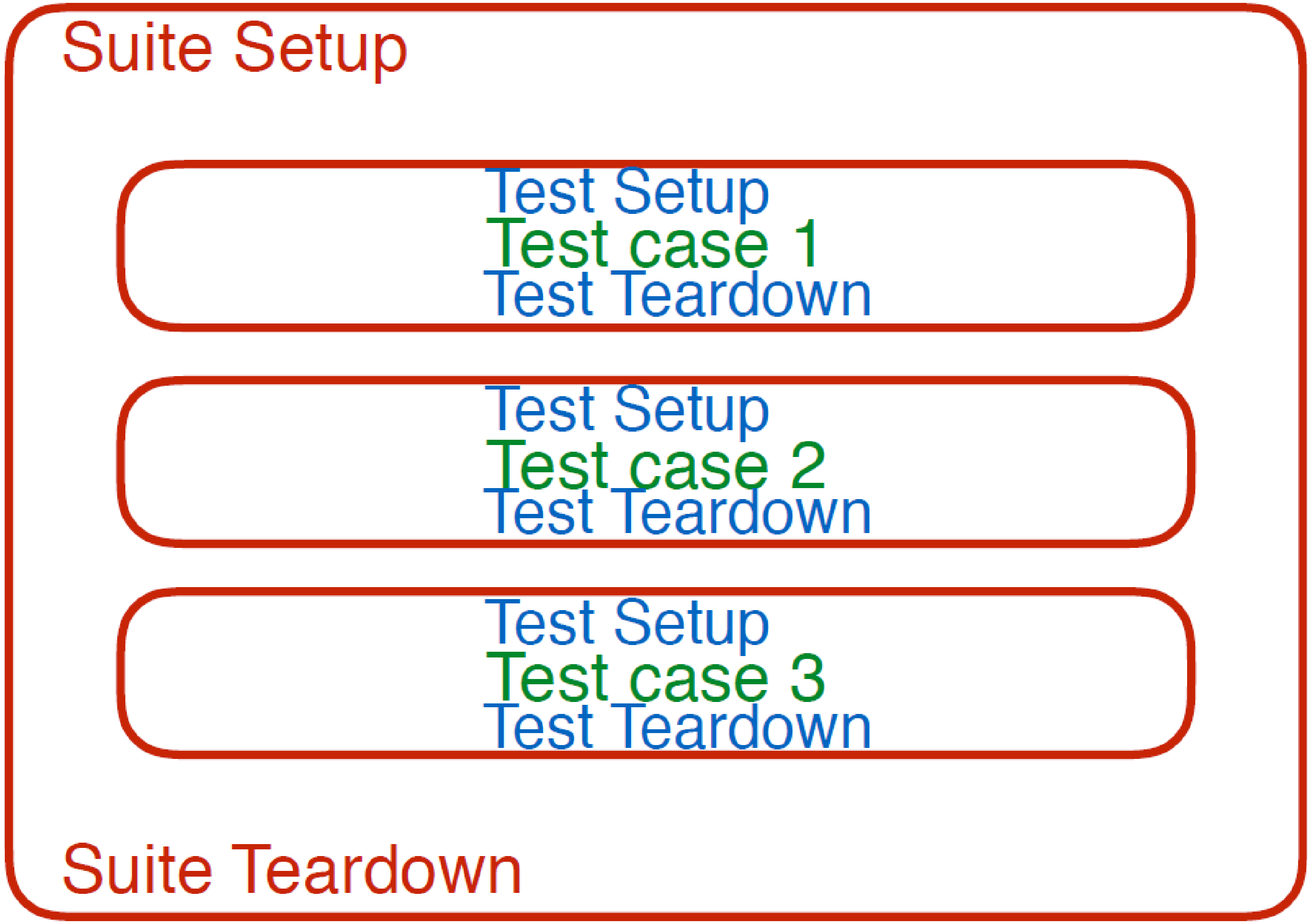
Test case 1

Test case 2

Test case 3

Suite Teardown

Suite



Suite Setup

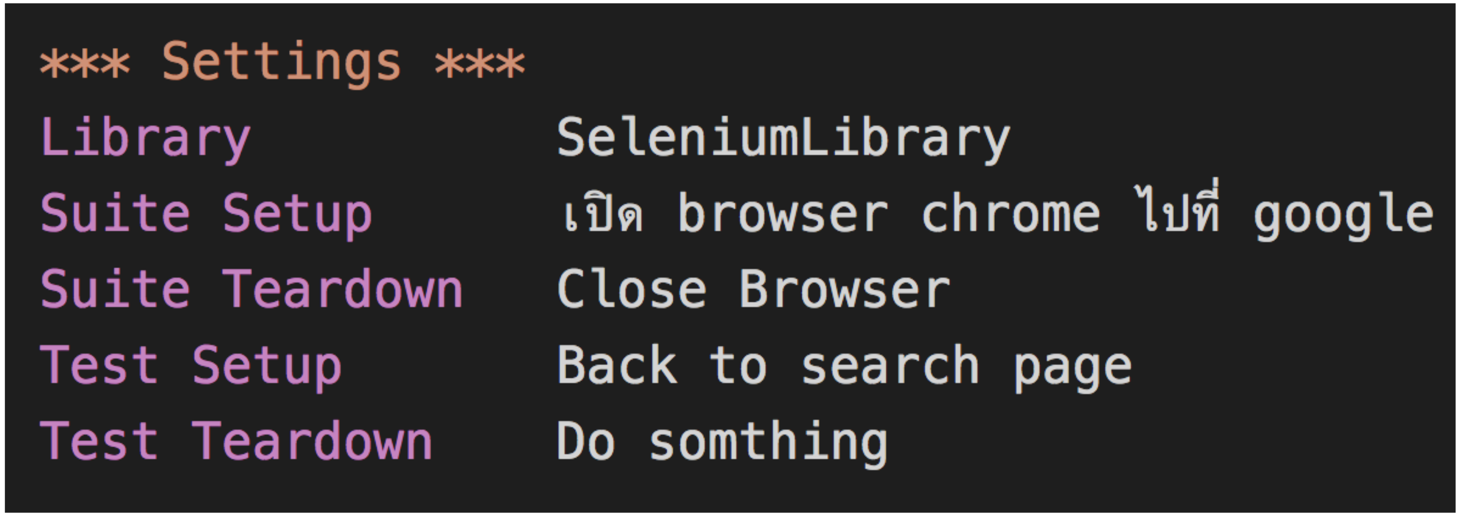
Test Setup
Test case 1
Test Teardown

Test Setup
Test case 2
Test Teardown

Test Setup
Test case 3
Test Teardown

Suite Teardown

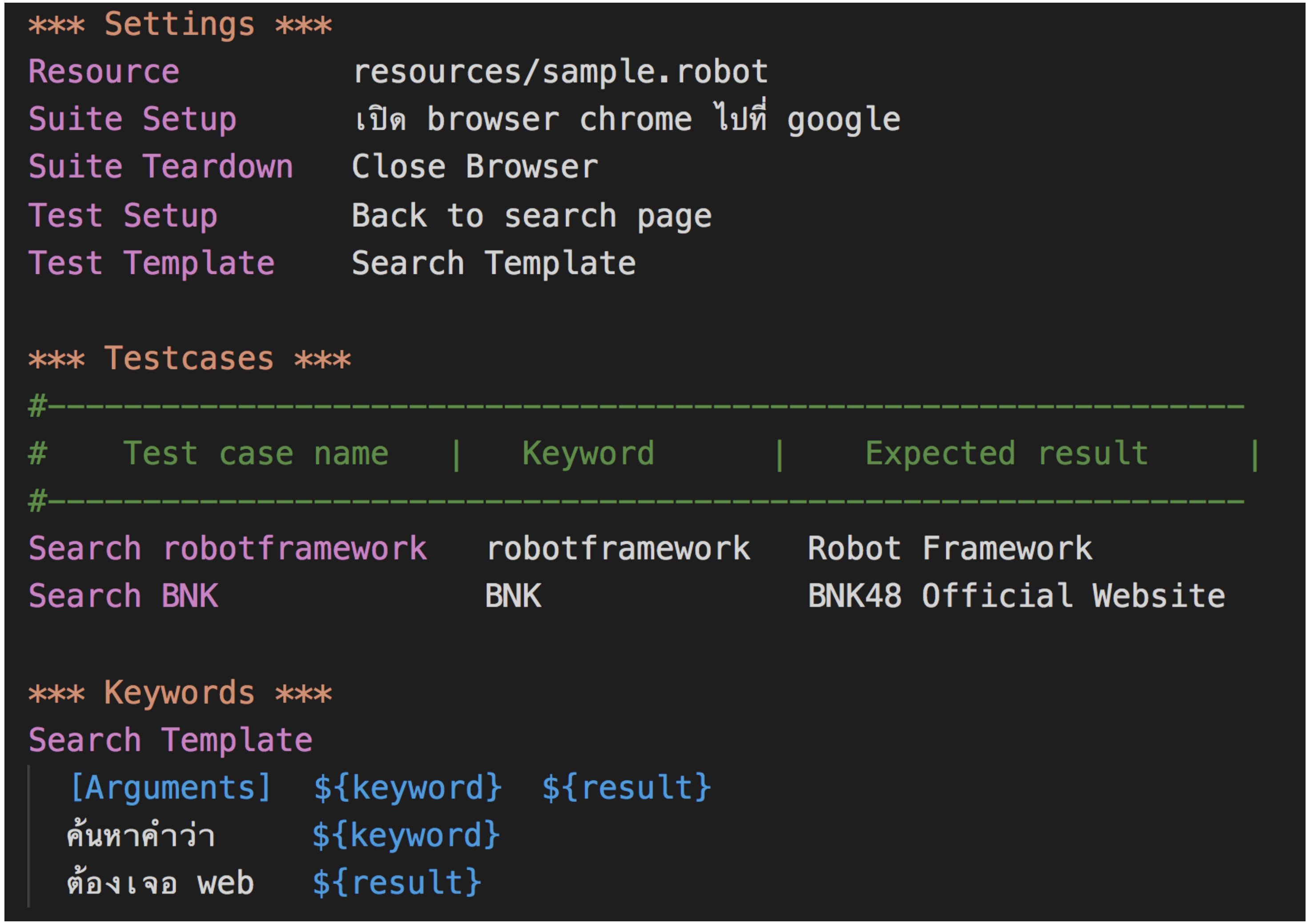
Test life cycle



```
*** Settings ***
Library          SeleniumLibrary
Suite Setup      เปิด browser chrome ไปที่ google
Suite Teardown   Close Browser
Test Setup       Back to search page
Test Teardown    Do somthing
```

Test Template 1

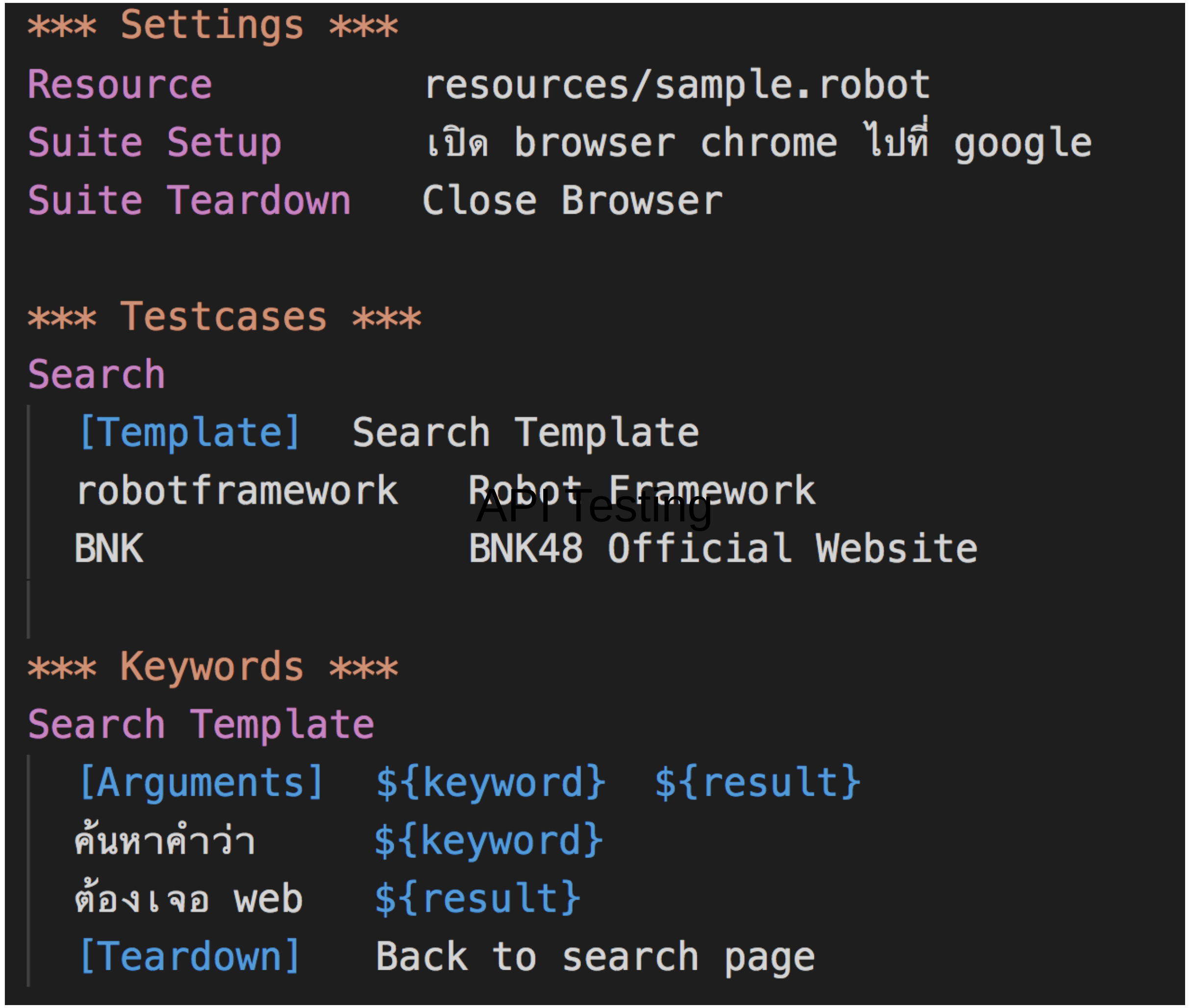
Test Template for all test



```
*** Settings ***
Resource            resources/sample.robot
Suite Setup         เปิด browser chrome ไปที่ google
Suite Teardown      Close Browser
Test Setup          Back to search page
Test Template       Search Template

*** Testcases ***
#---------------------------------------------------------------
#    Test case name    |   Keyword        |    Expected result     |
#---------------------------------------------------------------
Search robotframework    robotframework    Robot Framework
Search BNK               BNK               BNK48 Official Website

*** Keywords ***
Search Template
    [Arguments]   ${keyword}   ${result}
    ค้นหาคำว่า      ${keyword}
    ต้องเจอ web     ${result}
```

Test Template 2

Use Template in each test

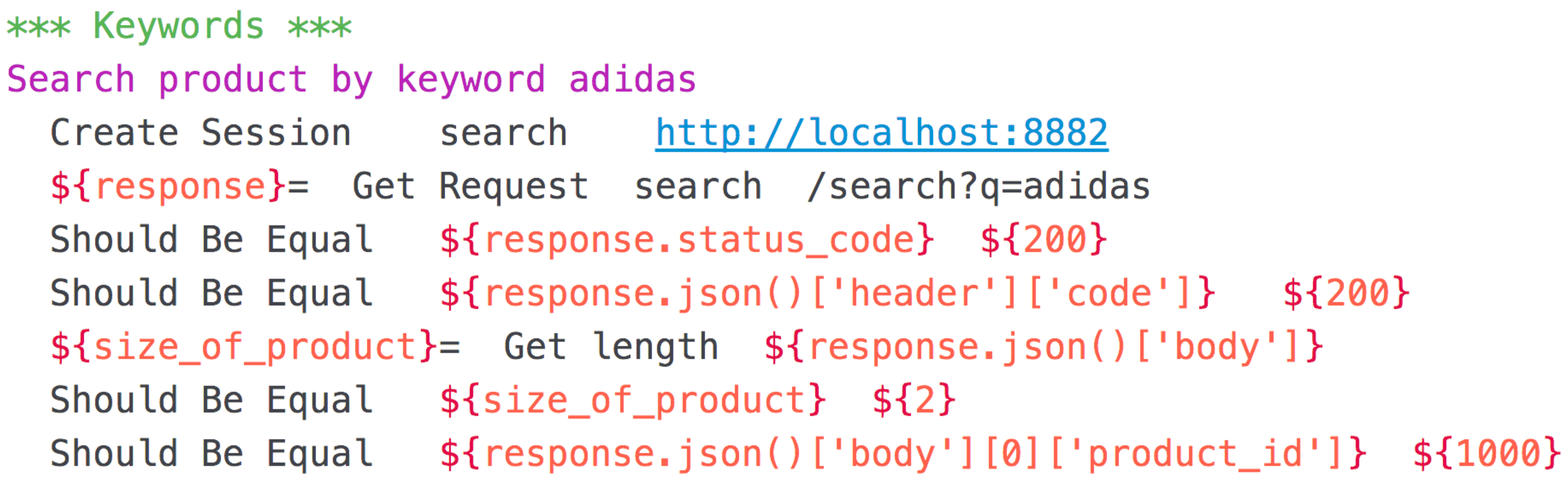


```
*** Settings ***
Resource           resources/sample.robot
Suite Setup        เปิด browser chrome ไปที่ google
Suite Teardown     Close Browser

*** Testcases ***
Search
    [Template]   Search Template
    robotframework     Robot Framework
    BNK                BNK48 Official Website

*** Keywords ***
Search Template
    [Arguments]   ${keyword}   ${result}
    ค้นหาคำว่า       ${keyword}
    ต้องเจอ web     ${result}
    [Teardown]    Back to search page
```

API Testing

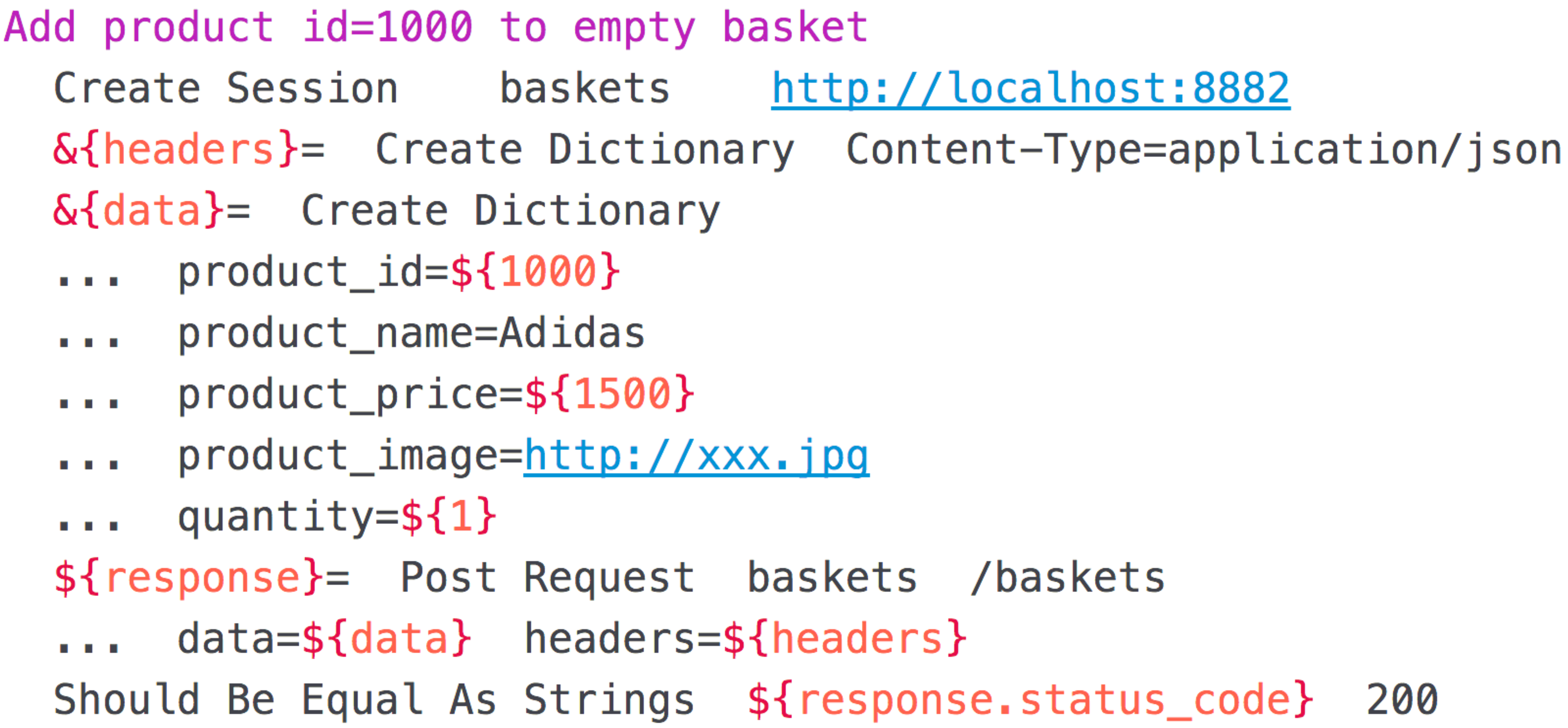
API Testing

HTTP GET



```
*** Keywords ***
Search product by keyword adidas
  Create Session    search    http://localhost:8882
  ${response}=  Get Request  search  /search?q=adidas
  Should Be Equal   ${response.status_code}  ${200}
  Should Be Equal   ${response.json()['header']['code']}   ${200}
  ${size_of_product}=  Get length  ${response.json()['body']}
  Should Be Equal   ${size_of_product}  ${2}
  Should Be Equal   ${response.json()['body'][0]['product_id']}  ${1000}
```

HTTP POST



```
Add product id=1000 to empty basket
  Create Session    baskets    http://localhost:8882
  &{headers}=  Create Dictionary  Content-Type=application/json
  &{data}=  Create Dictionary
  ...  product_id=${1000}
  ...  product_name=Adidas
  ...  product_price=${1500}
  ...  product_image=http://xxx.jpg
  ...  quantity=${1}
  ${response}=  Post Request  baskets  /baskets
  ...  data=${data}  headers=${headers}
  Should Be Equal As Strings  ${response.status_code}  200
```

May the force be will you.