Automated Unit test tools

SE234 Advance Software Development

College of Arts, Media and Technology
Chiang Mai University

What is system test?

System

- The integration of whole units
- With the User Interface
- Similar to what the user uses

Reading the specification

- Currency exchange methods
- The method receives the amount of money, and the currency to be exchange and the target currency
- The exchange rate is given as followed

Currency	Exchange rate per USD
USD	1
THB	30
EUR	0.9

- If the target currency is USD, the method will return the input amount divide by the exchange rate in the table,
- If the source currency is USD, the method will return the input amount multiply by the rate
- Otherwise, the source amount will be exchanged to USD first, and then exchange the USD to the target currency

Work

- What are your test data? Question 2.1

Try to test your application

- The application at https://currency-exchange-969bb.web.app/currency
- How was the test?

Test script

- The script that helps other user knows what to do
- Explain steps by steps
- Normally it will similar to the execution flow you have done in the requirement specification

Activity 2

- Write down your test script, Answer in question 2.2

Test Script and test case

- The script and test case must be used together
- The script is how to run the system
- Test case
  - The input data which will add in the test script
  - The expected result we want to see from the program
- 1 test script can be used with several test cases

Check your work

- Try to use your data to run the simple app again

Test Report

- The report that show whether the expected output and the actual output are the same or not
- If it is the same
    - Pass the test
- Otherwise
    - Fail the test

Activity

- Open the currency with error application
- https://currency-exchange-2-26614.web.app/currency
  - Use your old test cases and write down the report
  - Update your test report document
  - // there are around 3 errors here
- Report the error on Question 2.3

System test

- Instead of check the expected result by yourself
- Check automatically

Katalon

- Simple deployment, easy kick-off with project templates and hundreds of keywords built-in
- Cross platforms support Windows, macOS, Linux and popular mobile and browser platforms that support CI/CD world
- Full featured to support small to enterprise automation teams
- User friendly GUI, quick learning curve
- Detailed guides, tutorials and videos by the team and community.

Katalon Benefits

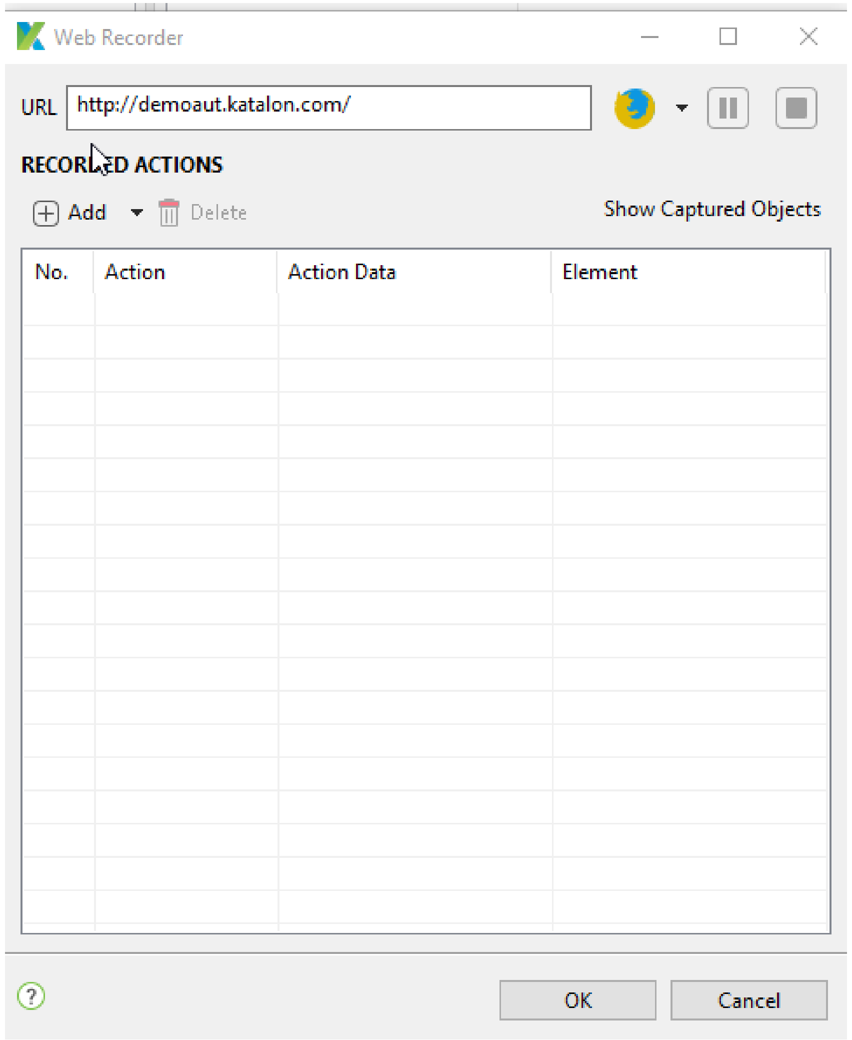
- Dual scripting modes enable effective collaboration between manual tester to automation engineers
- Supports KeyWord Driven and Data Driven Testing
- Based on Selenium and Appium, the dominant automation stacks with official support for all browsers
- Native integrations with DevOps and CI/CD ecosystems: GIT, Jenkins, qTest, JIRA, Slack and major cloud services.

Creating a test case

- Test cases consist of
  - Test script
  - Test data
- Test script
  - The steps to run the execution

Define the test script

- Using the Web recorder



URL: http://demoaut.katalon.com/

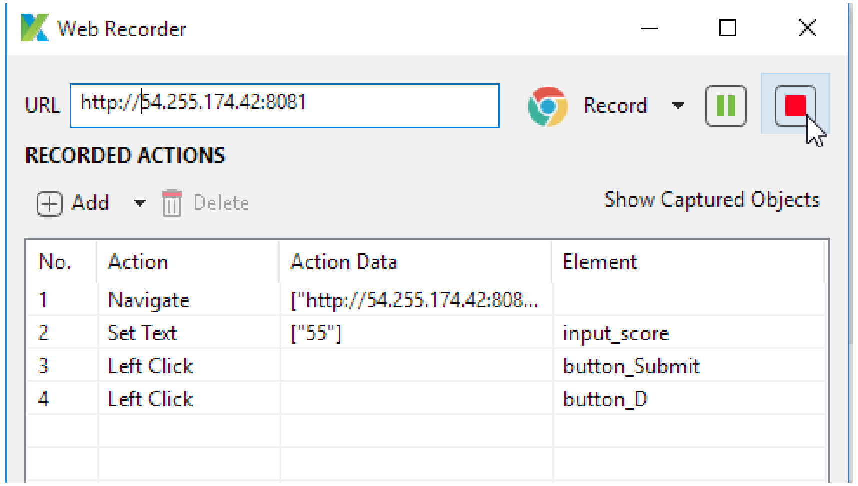
No.	Action	Action Data	Element

How it works

- Record all events occurred in the created web browser
- Generate the web object
  - For reuse purpose

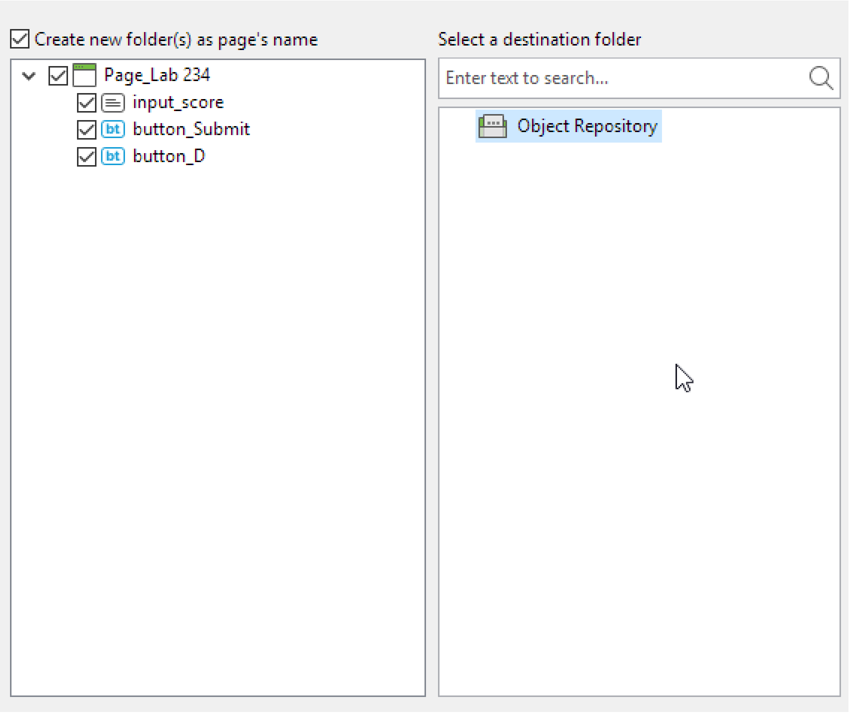
What we get?

- The script when you using the web
- The object



URL: http://54.255.174.42:8081

No.	Action	Action Data	Element
1	Navigate	["http://54.255.174.42:808...	
2	Set Text	["55"]	input_score
3	Left Click		button_Submit
4	Left Click		button_D



☑ Create new folder(s) as page's name

- ☑ Page_Lab 234
  - ☑ input_score
  - ☑ button_Submit
  - ☑ button_D

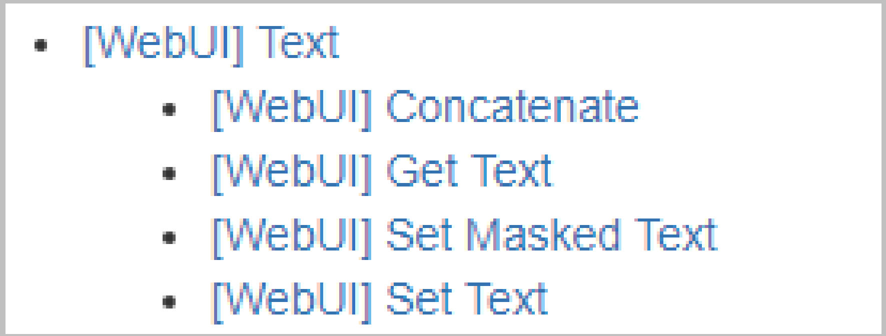
Select a destination folder

Test Script

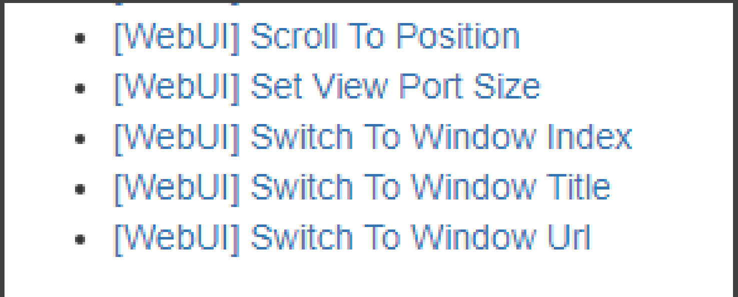
- WebUI events
  - Provide by the Katalon framework
  - Imitate how the user use the system
  - What is the normal activity we do with the web browser?

WebUI command

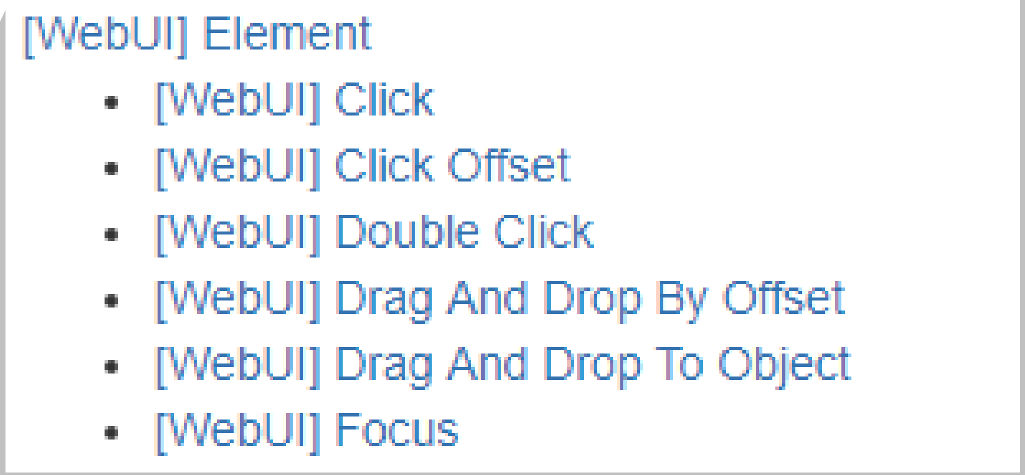
- Click
- Double click
- Focus
- Set text
- Submit



- [WebUI] Text
    - [WebUI] Concatenate
    - [WebUI] Get Text
    - [WebUI] Set Masked Text
    - [WebUI] Set Text

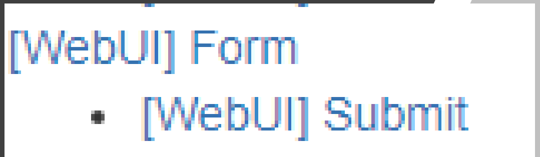


- [WebUI] Scroll To Position
- [WebUI] Set View Port Size
- [WebUI] Switch To Window Index
- [WebUI] Switch To Window Title
- [WebUI] Switch To Window Url



[WebUI] Element

- [WebUI] Click
- [WebUI] Click Offset
- [WebUI] Double Click
- [WebUI] Drag And Drop By Offset
- [WebUI] Drag And Drop To Object
- [WebUI] Focus



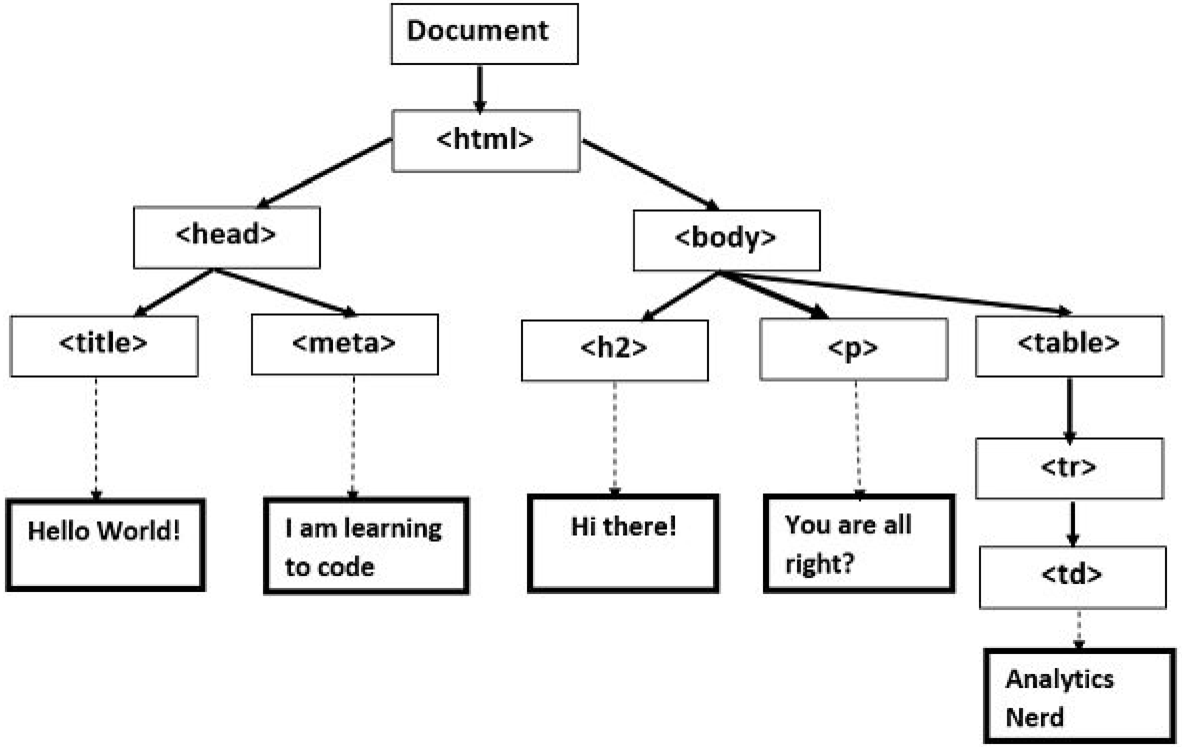
[WebUI] Form

- [WebUI] Submit

HTML code as a tree

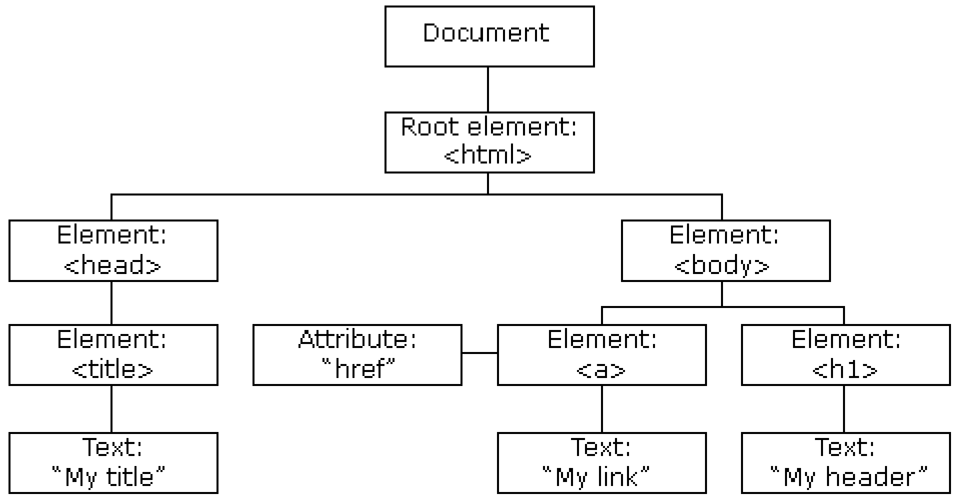
- HTML/XML can parse as a tree

```
<html>

<head>

    <title>Hello World!</title>

    <meta name="description" content="I am learning to code." />

</head>

<body>

    <h2>Hi There!</h2>

    <p>You are all right?</p>

    <table>

        <tr>

            <td>Analytics Nerd</td>

        </tr>

    </table>

</body>

</html>
```



DOM

- Document object model
  - A programming interface for HTML and XML documents
  - Represent the document as nodes and objects
- For HTML
  - JavaScript can be used to manipulate the DOM
- Each tag is the Object
  - Contains the attributes



Document

Root element: <html>

Element: <head>

Element: <body>

Element: <title>

Attribute: “href”

Element: <a>

Element: <h1>

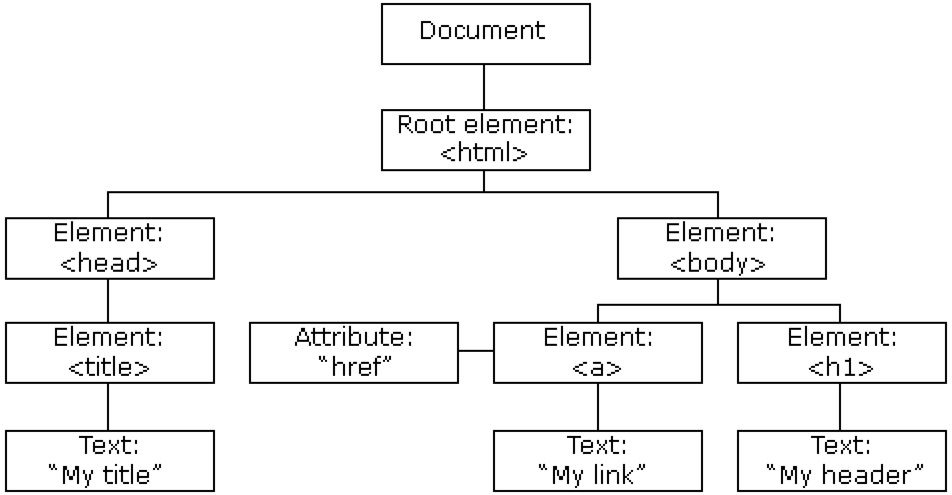
Text: “My title”

Text: “My link”

Text: “My header”

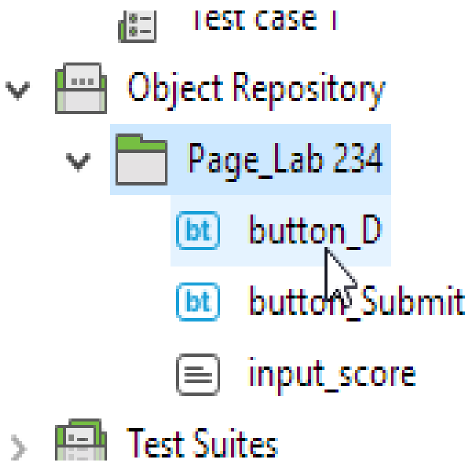
Object Repository

- The object to be selected
- Use to refer as the target of the event
  - The node in the DOM tree

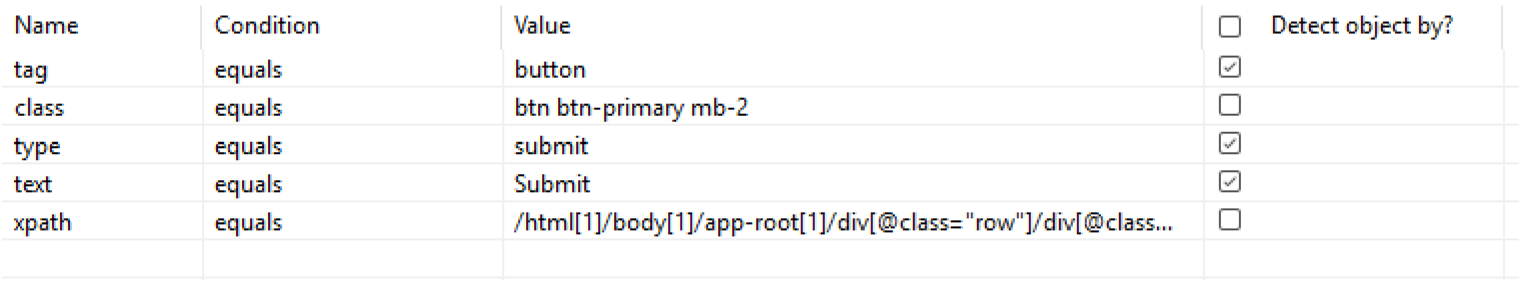


Object Selector

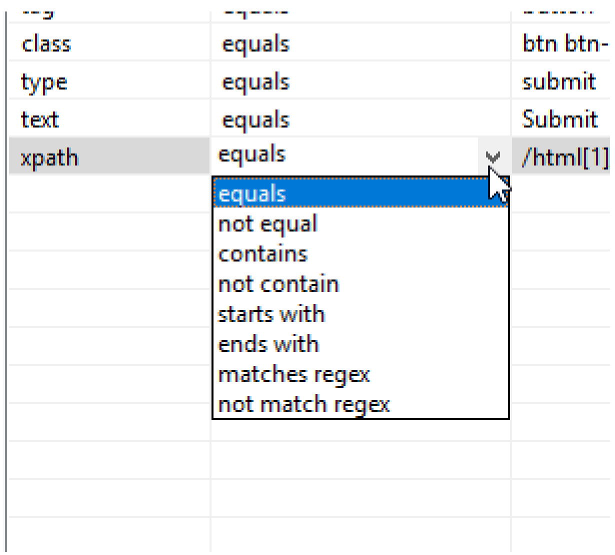
- Programmatically select the object
- The default name normally come from the text in the elements
  - Need to be changed for easier understanding



Object Selector



Name	Condition	Value	□ Detect object by?
tag	equals	button	☑
class	equals	btn btn-primary mb-2	□
type	equals	submit	☑
text	equals	Submit	☑
xpath	equals	/html[1]/body[1]/app-root[1]/div[@class="row"]/div[@class...	□



class	equals	btn btn-
type	equals	submit
text	equals	Submit
xpath	equals	/html[1]

- equals
- not equal
- contains
- not contain
- starts with
- ends with
- matches regex
- not match regex

XPath

- XPath is a syntax used for selecting parts of an XML document
- The way XPath describes paths to elements is similar to the way an operating system describes paths to files
- XPath is almost a small programming language; it has functions, tests, and expressions
- XPath is a W3C standard
- XPath is not itself written as XML, but is used heavily in XSLT

Overview of XPath

- Finding the elements
- Use the / to trace the object from the root

- Use // to start from anywhere in the documents

- Xpath without the / mean start from the current element

Overview of XPath

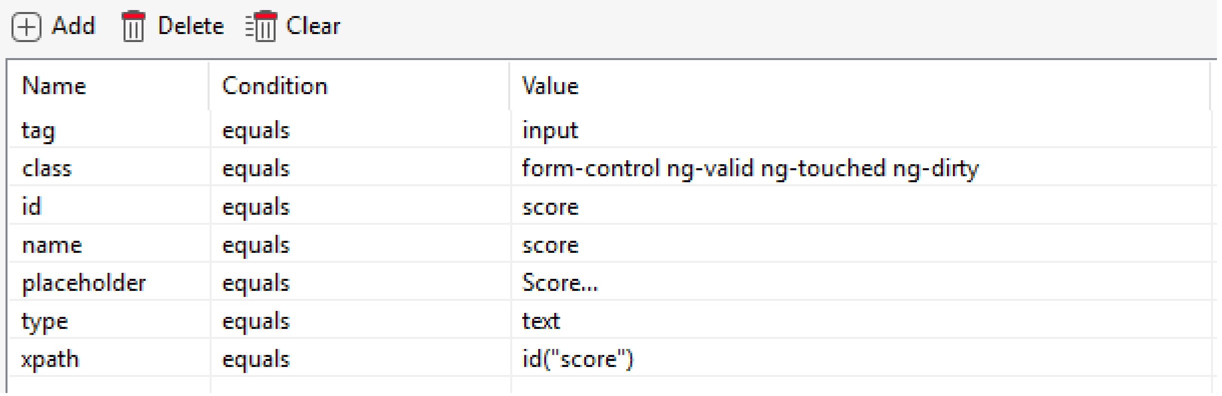
- [] if there are more than one element type as children
  - Access to the array of the elements
  - //tbody/tr[0] = the first tr elements under the tbody
- last() -> the the last elements in the list
  - //tbody/tr[0] = the the last tr elements under the tbody
- @ select the element which has the attribute value as given
- The selector can return only one element or a list of all elements

Problem with xpath

- Long xpath
  - May not sure where is it?
  - /html[1]/body[1]/app-root[1]/div[@class="row"]/div[@class="col-md-6"]/form[@class="form-inline ng-valid ng-touched ng-dirty"]/button[@class="btn btn-primary mb-2"]

Using id attribute

- Easy to track
  - Inspect the id of the element
  - Need to talk with the programmer



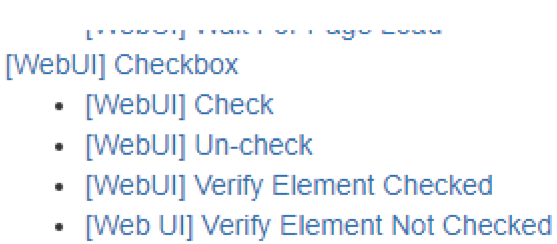
Name	Condition	Value
tag	equals	input
class	equals	form-control ng-valid ng-touched ng-dirty
id	equals	score
name	equals	score
placeholder	equals	Score...
type	equals	text
xpath	equals	id("score")

Running the test script

- Everything fine
    - As we imitate how the software have been launch

Verify

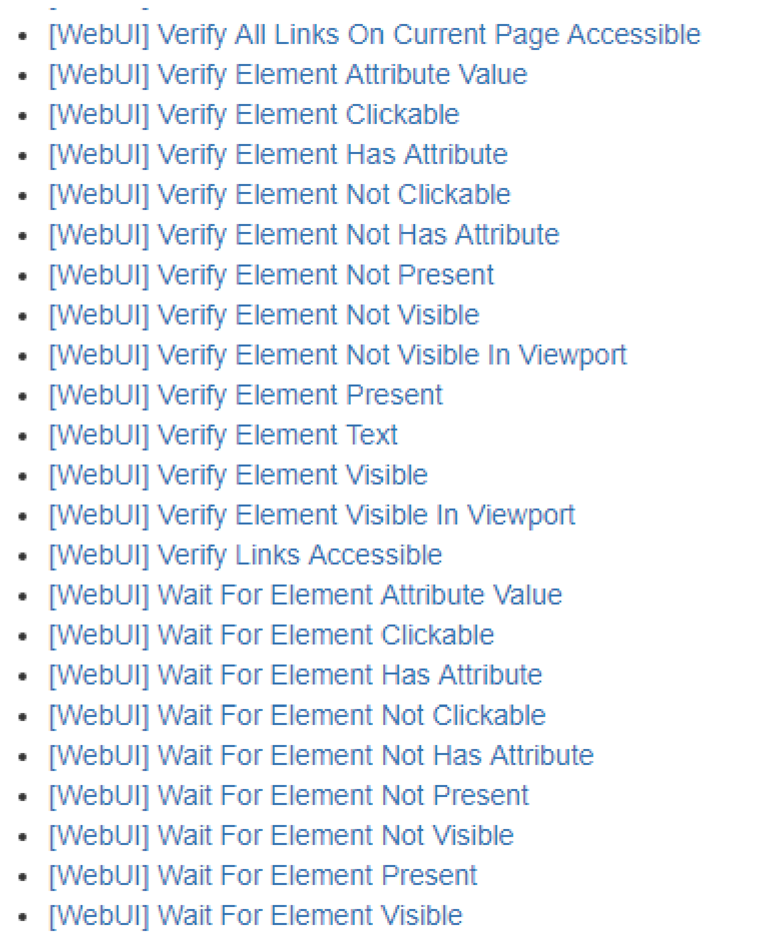
- Check that the value is as it should be or not



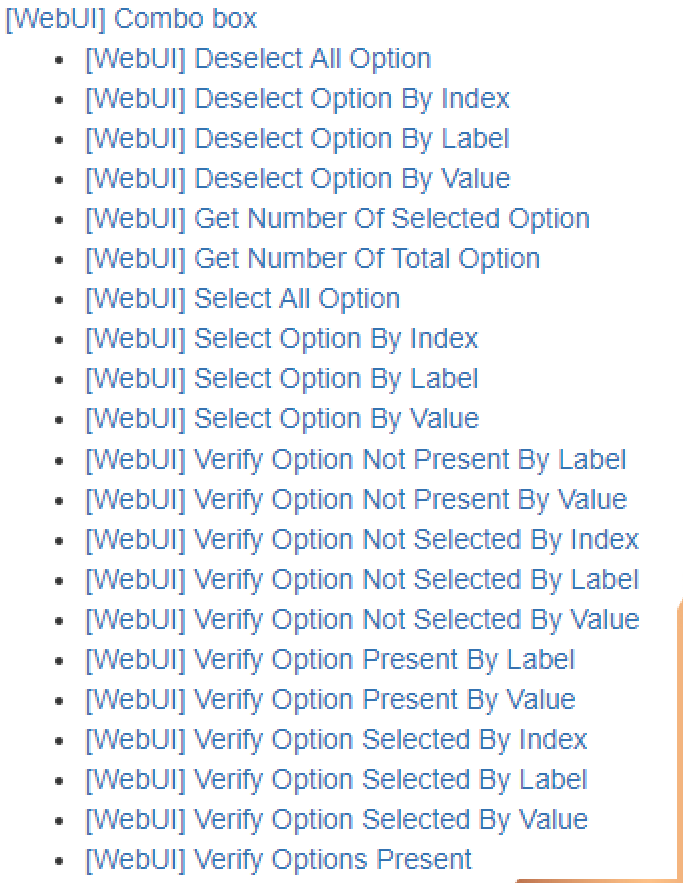
- [WebUI] Wait For Page Load

[WebUI] Checkbox

- [WebUI] Check
- [WebUI] Un-check
- [WebUI] Verify Element Checked
- [Web UI] Verify Element Not Checked



- [WebUI] Verify All Links On Current Page Accessible
- [WebUI] Verify Element Attribute Value
- [WebUI] Verify Element Clickable
- [WebUI] Verify Element Has Attribute
- [WebUI] Verify Element Not Clickable
- [WebUI] Verify Element Not Has Attribute
- [WebUI] Verify Element Not Present
- [WebUI] Verify Element Not Visible
- [WebUI] Verify Element Not Visible In Viewport
- [WebUI] Verify Element Present
- [WebUI] Verify Element Text
- [WebUI] Verify Element Visible
- [WebUI] Verify Element Visible In Viewport
- [WebUI] Verify Links Accessible
- [WebUI] Wait For Element Attribute Value
- [WebUI] Wait For Element Clickable
- [WebUI] Wait For Element Has Attribute
- [WebUI] Wait For Element Not Clickable
- [WebUI] Wait For Element Not Has Attribute
- [WebUI] Wait For Element Not Present
- [WebUI] Wait For Element Not Visible
- [WebUI] Wait For Element Present
- [WebUI] Wait For Element Visible

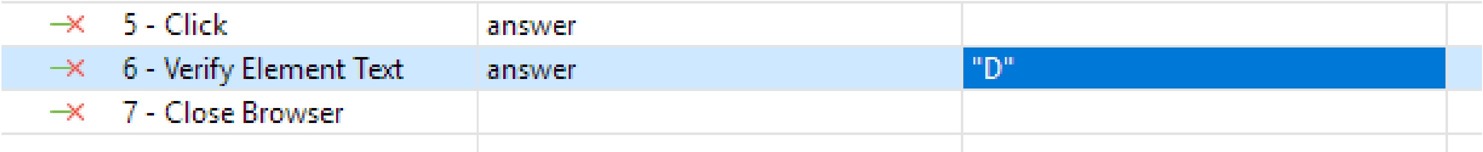


[WebUI] Combo box

- [WebUI] Deselect All Option
- [WebUI] Deselect Option By Index
- [WebUI] Deselect Option By Label
- [WebUI] Deselect Option By Value
- [WebUI] Get Number Of Selected Option
- [WebUI] Get Number Of Total Option
- [WebUI] Select All Option
- [WebUI] Select Option By Index
- [WebUI] Select Option By Label
- [WebUI] Select Option By Value
- [WebUI] Verify Option Not Present By Label
- [WebUI] Verify Option Not Present By Value
- [WebUI] Verify Option Not Selected By Index
- [WebUI] Verify Option Not Selected By Label
- [WebUI] Verify Option Not Selected By Value
- [WebUI] Verify Option Present By Label
- [WebUI] Verify Option Present By Value
- [WebUI] Verify Option Selected By Index
- [WebUI] Verify Option Selected By Label
- [WebUI] Verify Option Selected By Value
- [WebUI] Verify Options Present

Verify for the object

- Check that it is as verify or not



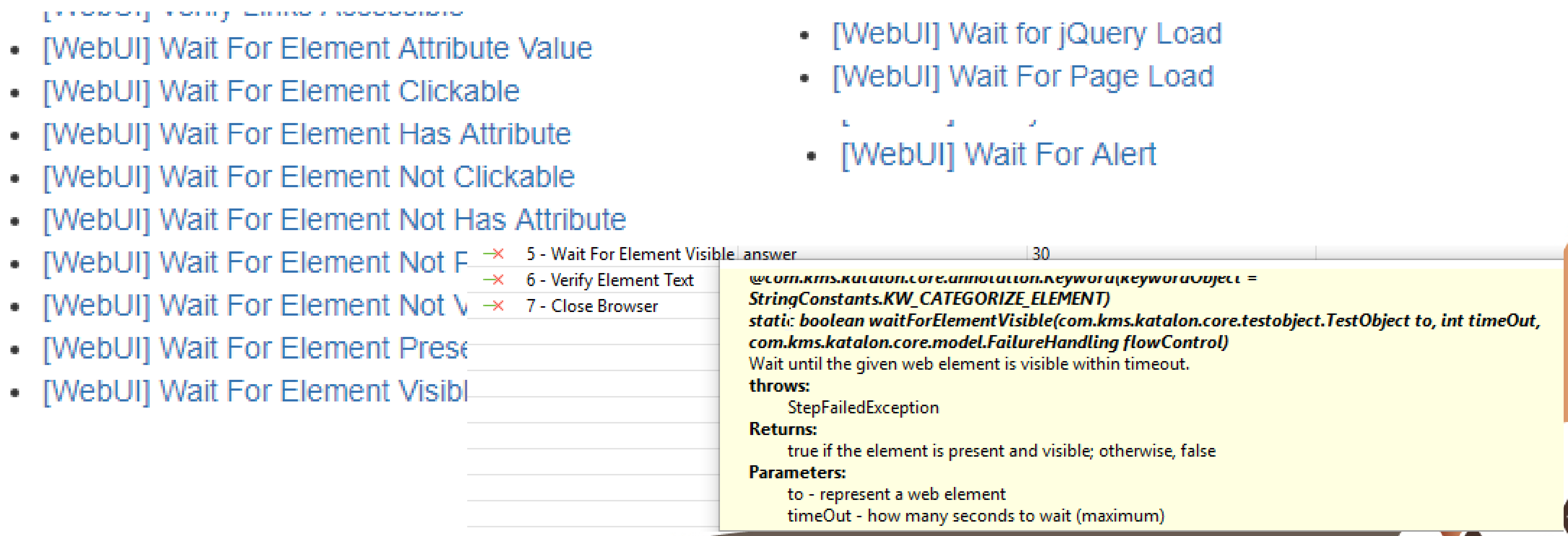
5 - Click	answer	
6 - Verify Element Text	answer	"D"
7 - Close Browser		

Error that can occurred

- Expected value is not as actual output
- The object could not be found
  - Wrong selector
  - Object is not presented yet
    - Due to the lack of network etc.

Wait for

- Wait until some thing happen
  - Need the parameter to add the seconds which you will wait for



- [WebUI] Verify Links Accessible
- [WebUI] Wait For Element Attribute Value
- [WebUI] Wait For Element Clickable
- [WebUI] Wait For Element Has Attribute
- [WebUI] Wait For Element Not Clickable
- [WebUI] Wait For Element Not Has Attribute
- [WebUI] Wait For Element Not F
- [WebUI] Wait For Element Not V
- [WebUI] Wait For Element Prese
- [WebUI] Wait For Element Visibl

- [WebUI] Wait for jQuery Load
- [WebUI] Wait For Page Load
- [WebUI] Wait For Alert

```
5 - Wait For Element Visible  answer  30
6 - Verify Element Text
7 - Close Browser
```

@com.kms.katalon.core.annotation.Keyword(keywordObject =
StringConstants.KW_CATEGORIZE_ELEMENT)
static boolean waitForElementVisible(com.kms.katalon.core.testobject.TestObject to, int timeOut, com.kms.katalon.core.model.FailureHandling flowControl)

Wait until the given web element is visible within timeout.

throws:
StepFailedException

Returns:
true if the element is present and visible; otherwise, false

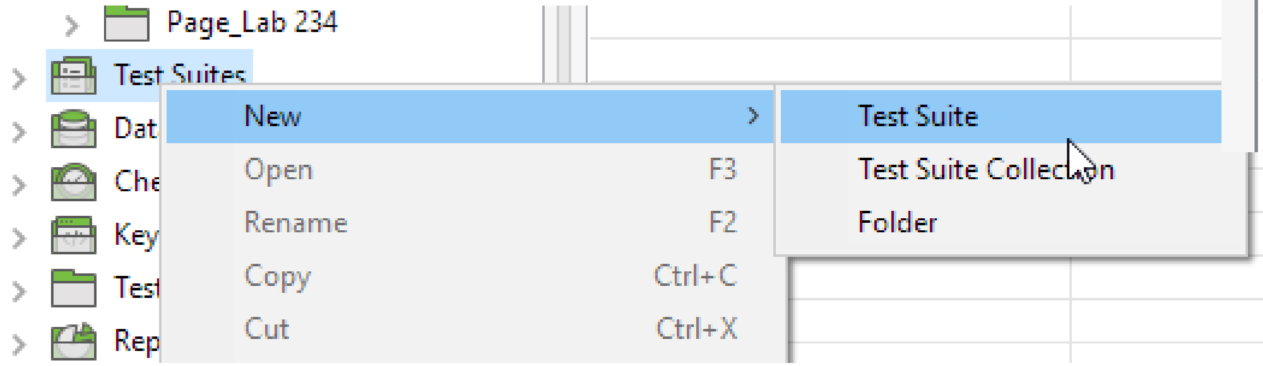
Parameters:
to - represent a web element
timeOut - how many seconds to wait (maximum)

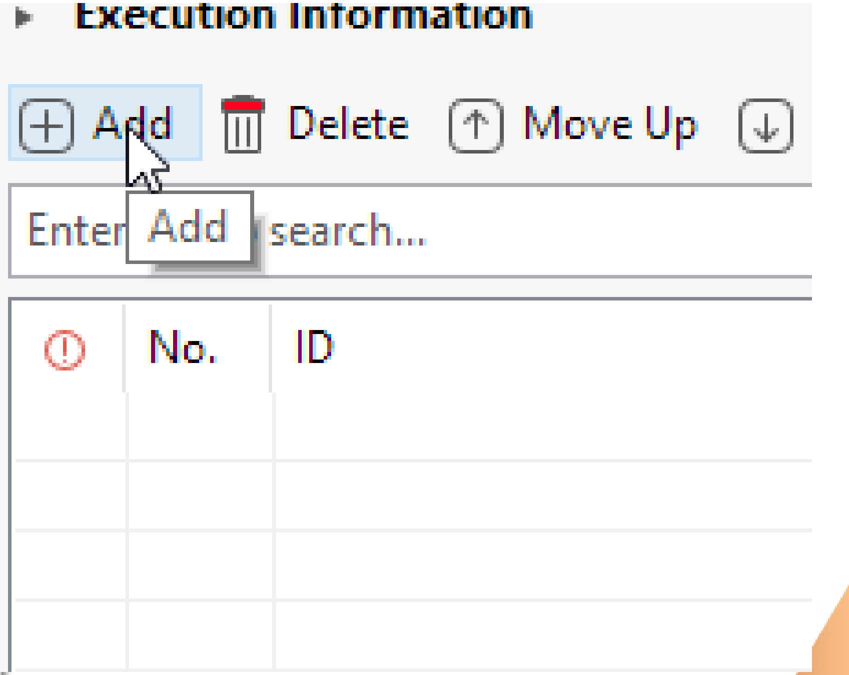
Test cases

- Test case can be copy
  - Update only some input, and expected output to run the test
  - But can run test case by test case

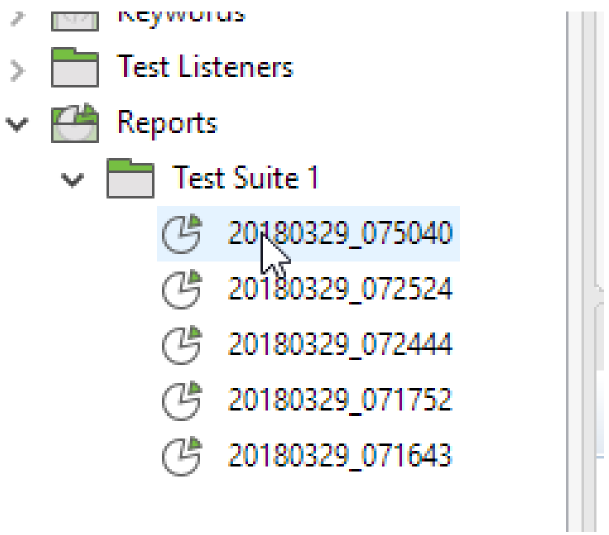
Test Suite

- A set of test case
- Can be run at once
- Reduce the duplicate calling test case

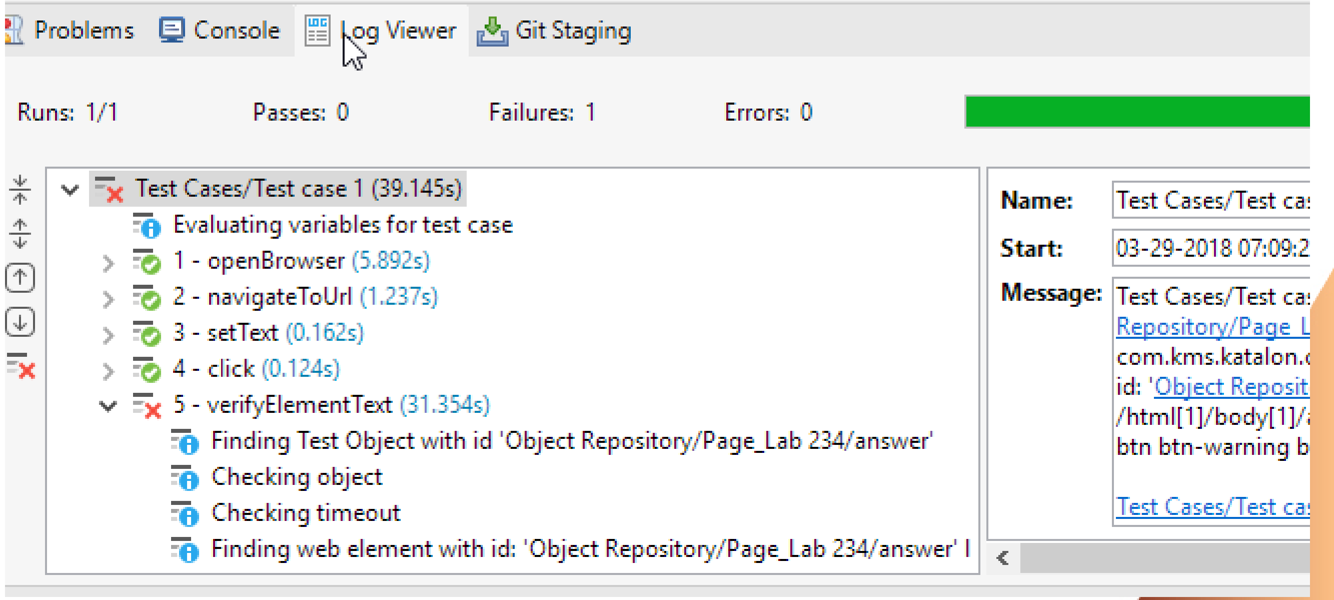




The report is also provided



- Keywords
- Test Listeners
- Reports
  - Test Suite 1
    - 20180329_075040
    - 20180329_072524
    - 20180329_072444
    - 20180329_071752
    - 20180329_071643



Runs: 1/1 Passes: 0 Failures: 1 Errors: 0

```
Test Cases/Test case 1 (39.145s)
    Evaluating variables for test case
  > 1 - openBrowser (5.892s)
  > 2 - navigateToUrl (1.237s)
  > 3 - setText (0.162s)
  > 4 - click (0.124s)
  v 5 - verifyElementText (31.354s)
        Finding Test Object with id 'Object Repository/Page_Lab 234/answer'
        Checking object
        Checking timeout
        Finding web element with id: 'Object Repository/Page_Lab 234/answer' l
```

Name: Test Cases/Test ca

Start: 03-29-2018 07:09:2

Message: Test Cases/Test ca
Repository/Page_L
com.kms.katalon.
id: 'Object Reposit
/html[1]/body[1]/
btn btn-warning b

Test Cases/Test ca

Data driven testing

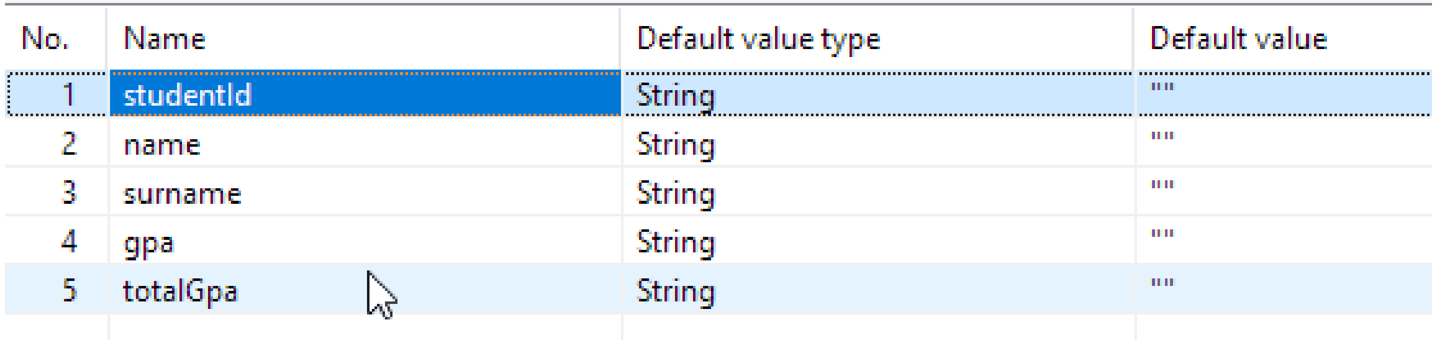
- Same script
- Set of data
- Test different cases

Data driven

- Requires
  - Set of data
  - Where to insert the data

Variable

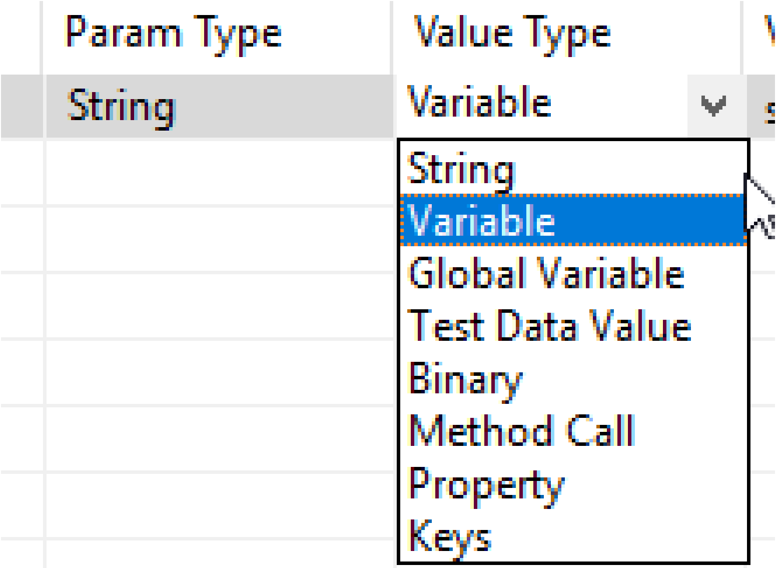
- Set the location to put the data to the test script
- Declare the variable first



No.	Name	Default value type	Default value
1	studentId	String	""
2	name	String	""
3	surname	String	""
4	gpa	String	""
5	totalGpa	String	""

Use variable

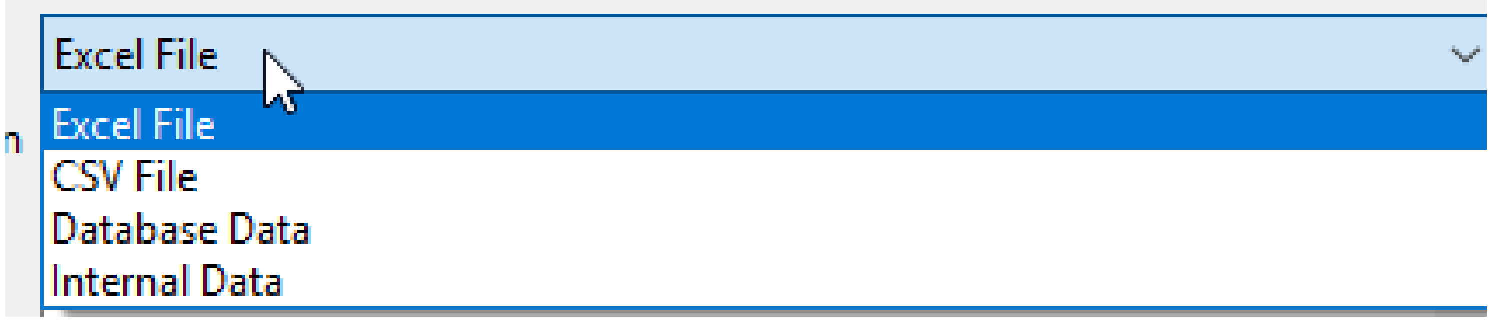
- Add it to the parameter of Verify



Param Type	Value Type
String	Variable

Data files

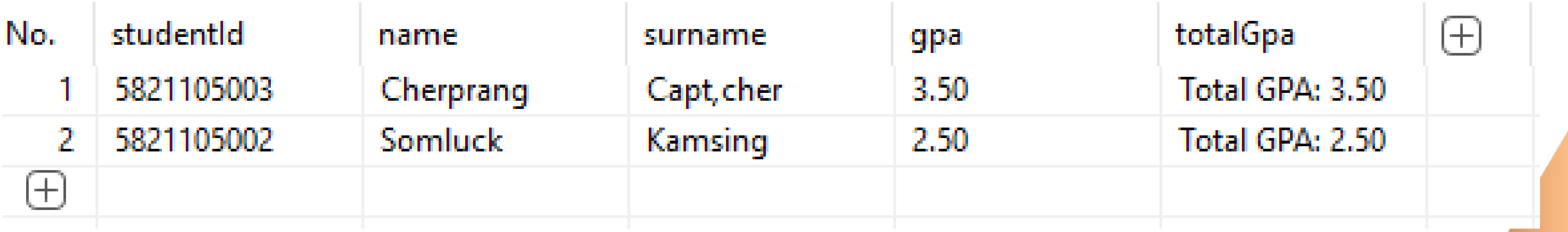
- Data to be used



- Many format
- Need the column name to refer

Internal data

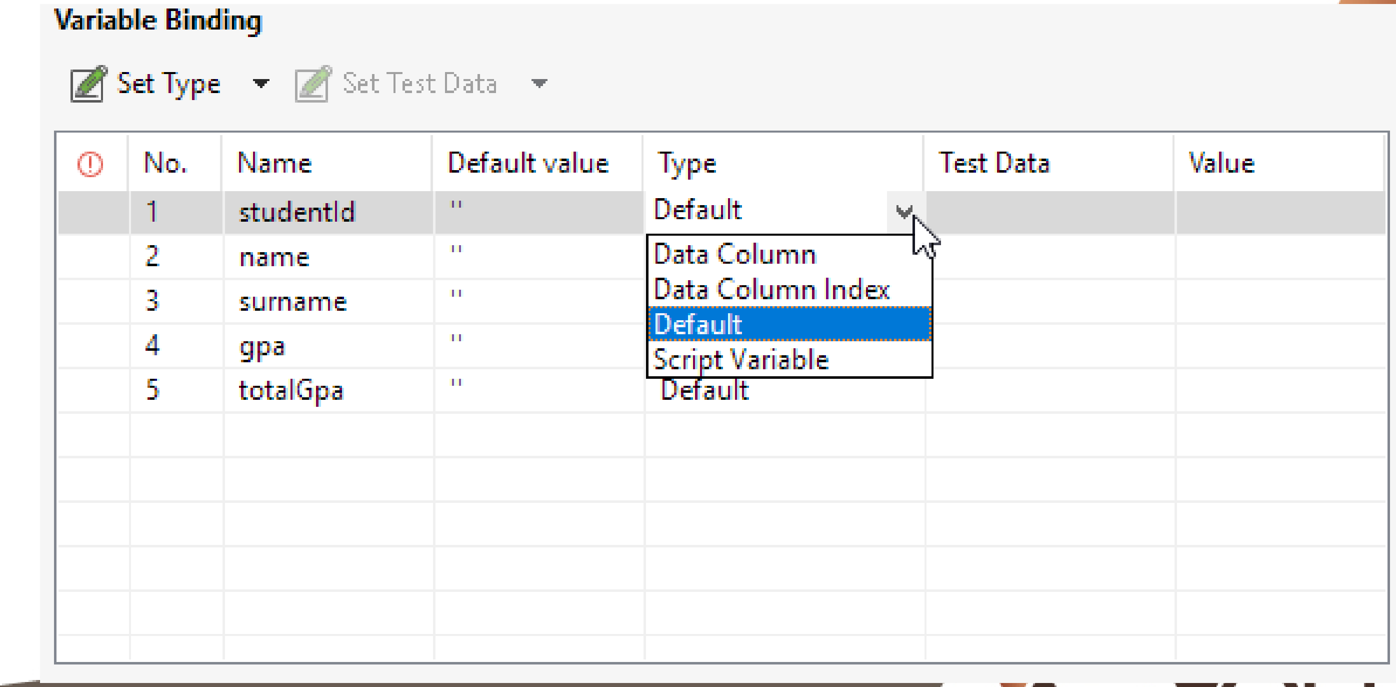
- Create your own table



No.	studentId	name	surname	gpa	totalGpa	⊞
1	5821105003	Cherprang	Capt,cher	3.50	Total GPA: 3.50	
2	5821105002	Somluck	Kamsing	2.50	Total GPA: 2.50	
⊞						

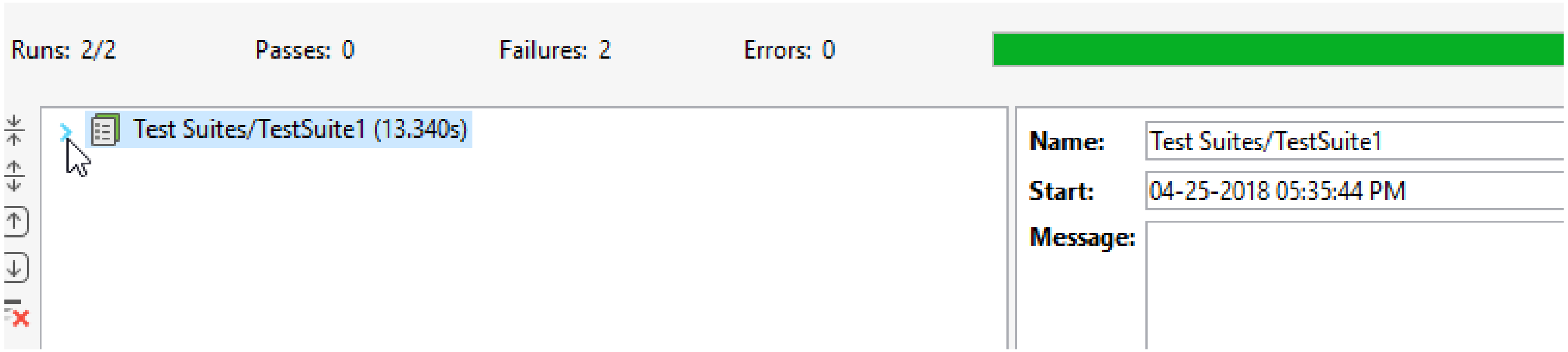
Mapping the Data source with variable

- Crete the test suite
  - Add the test case
  - Map the variable to any data sources
  - One data source per test case



	No.	Name	Default value	Type	Test Data	Value
	1	studentId	''	Default		
	2	name	''			
	3	surname	''			
	4	gpa	''			
	5	totalGpa	''	Default		

Running the test



Runs: 2/2 Passes: 0 Failures: 2 Errors: 0

Test Suites/TestSuite1 (13.340s)

Name: Test Suites/TestSuite1

Start: 04-25-2018 05:35:44 PM

Message:

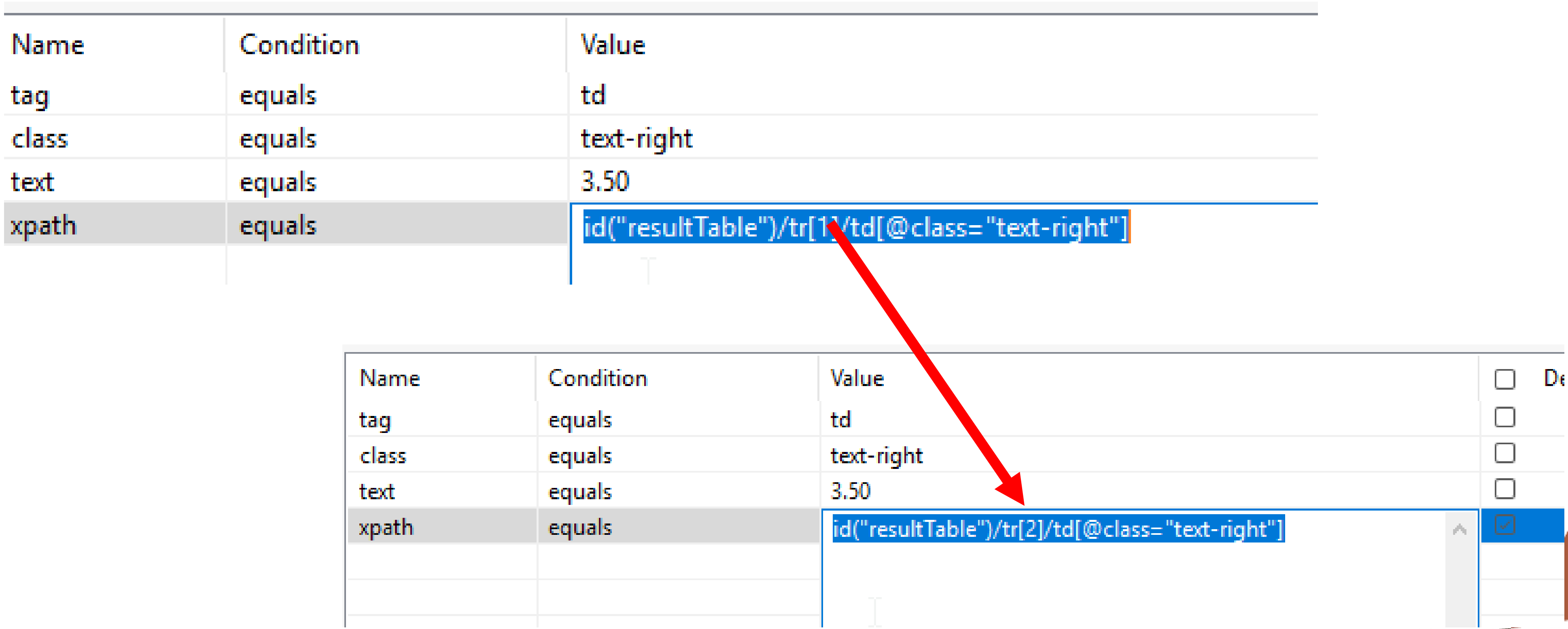
Customization

- The object can be customized
  - Enough background
  - Create an object to inspect without recording it
- Save time
- Can handle the complicate situation

Object Selection

- Edit the selector command
- Using Xpath is recommend
    - Easy to handle the row elements
    - In case that there are many rows

Update the selector

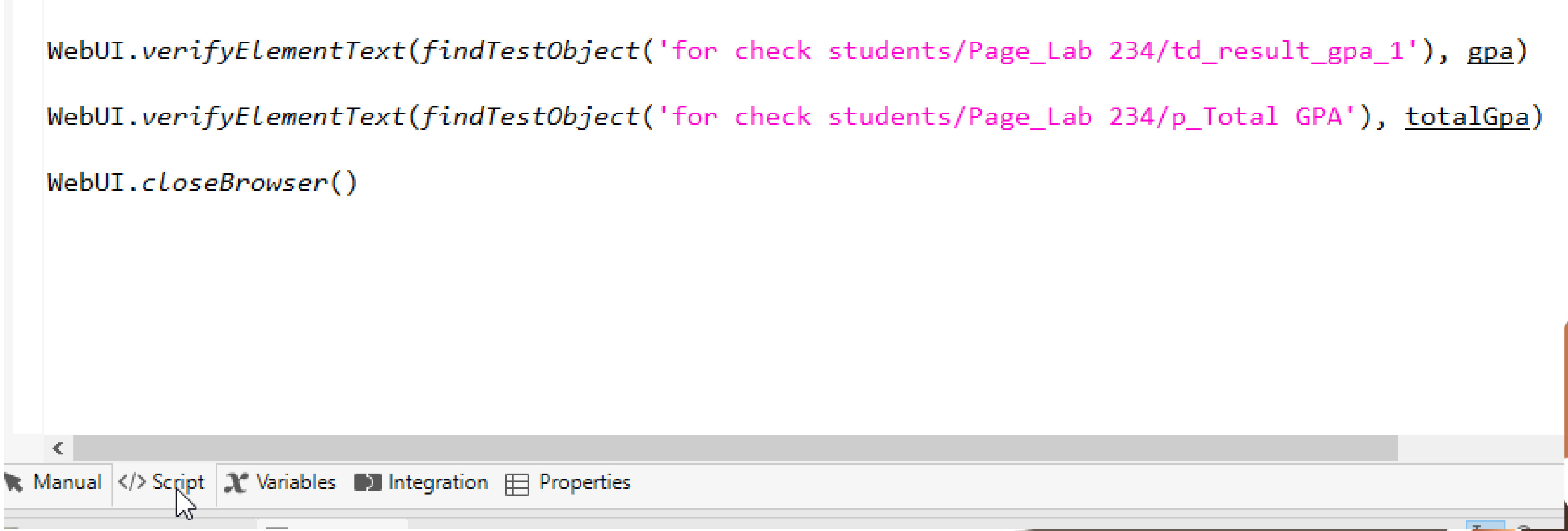


Name	Condition	Value
tag	equals	td
class	equals	text-right
text	equals	3.50
xpath	equals	id("resultTable")/tr[1]/td[@class="text-right"]

Name	Condition	Value	☐	De
tag	equals	td	☐	
class	equals	text-right	☐	
text	equals	3.50	☐	
xpath	equals	id("resultTable")/tr[2]/td[@class="text-right"]	☑	

Customize the Script

- Use external library
- To select and get some data



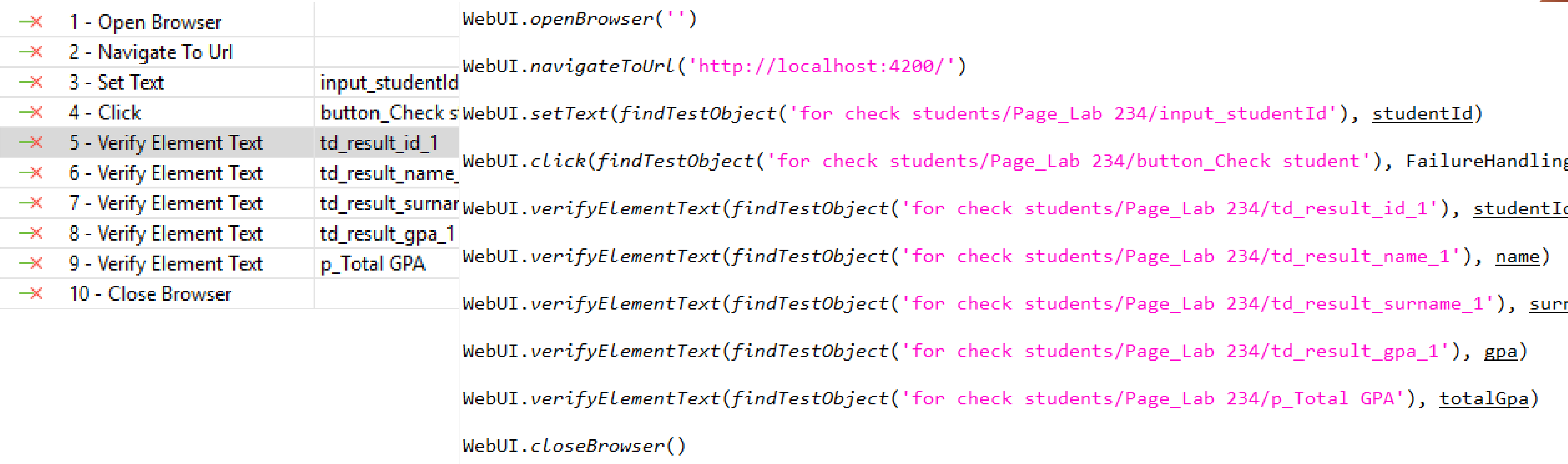
```
WebUI.verifyElementText(findTestObject('for check students/Page_Lab 234/td_result_gpa_1'), gpa)

WebUI.verifyElementText(findTestObject('for check students/Page_Lab 234/p_Total GPA'), totalGpa)

WebUI.closeBrowser()
```

Katalon Script

- Groovy
  - powerful, optionally typed and dynamic language
  - static-typing and static compilation capabilities
  - familiar and easy to learn syntax
  - Domain-Specific Language
  - meta-programming and functional programming
  - Seamlessly Support Java platform

Script

- Each line of code is matched with the script in the script pane



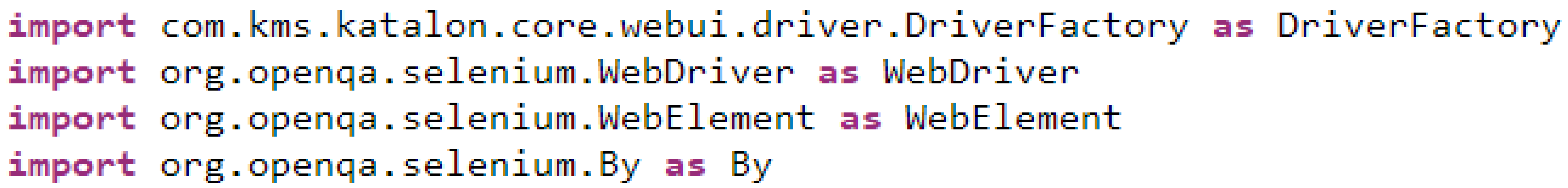
1 - Open Browser	
2 - Navigate To Url	
3 - Set Text	input_studentId
4 - Click	button_Check s
5 - Verify Element Text	td_result_id_1
6 - Verify Element Text	td_result_name_
7 - Verify Element Text	td_result_surnar
8 - Verify Element Text	td_result_gpa_1
9 - Verify Element Text	p_Total GPA
10 - Close Browser	

```
WebUI.openBrowser('')

WebUI.navigateToUrl('http://localhost:4200/')

WebUI.setText(findTestObject('for check students/Page_Lab 234/input_studentId'), studentId)

WebUI.click(findTestObject('for check students/Page_Lab 234/button_Check student'), FailureHandling

WebUI.verifyElementText(findTestObject('for check students/Page_Lab 234/td_result_id_1'), studentI

WebUI.verifyElementText(findTestObject('for check students/Page_Lab 234/td_result_name_1'), name)

WebUI.verifyElementText(findTestObject('for check students/Page_Lab 234/td_result_surname_1'), surn

WebUI.verifyElementText(findTestObject('for check students/Page_Lab 234/td_result_gpa_1'), gpa)

WebUI.verifyElementText(findTestObject('for check students/Page_Lab 234/p_Total GPA'), totalGpa)

WebUI.closeBrowser()
```

Import other Java components

- Selenium
  - WebDriver is a tool for automating web application testing, and in particular to verify that they work as expected.
  - It’s not tied to any particular test framework, so it can be used equally well in a unit testing or from a plain old “main” method
  - Provided in many programming language
  - Ability
    - to select elements
    - Read some data from the elements
- Cons
  - You need to add all the script and find object manaully

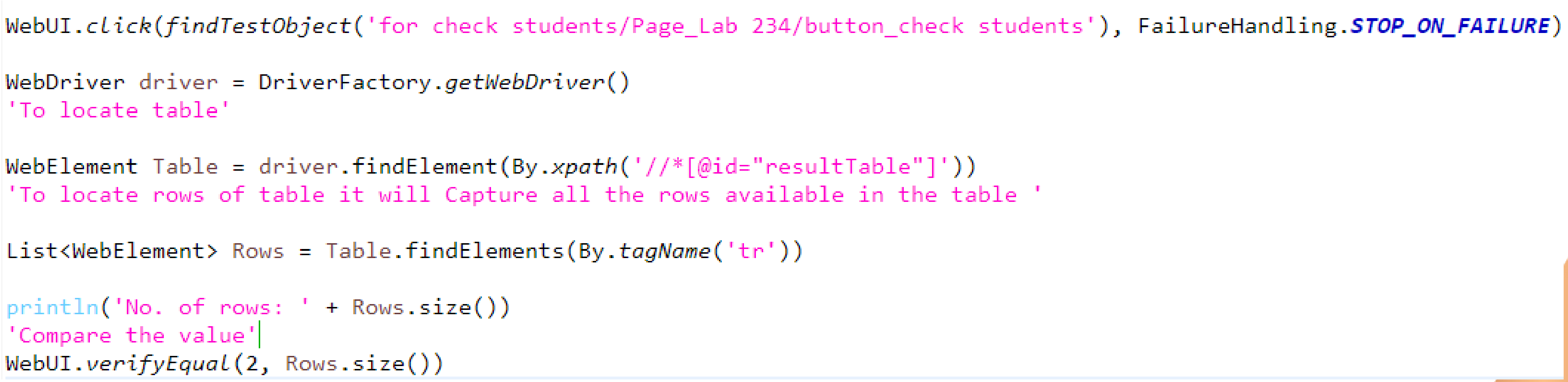
Import the component



```
import com.kms.katalon.core.webui.driver.DriverFactory as DriverFactory
import org.openqa.selenium.WebDriver as WebDriver
import org.openqa.selenium.WebElement as WebElement
import org.openqa.selenium.By as By
```

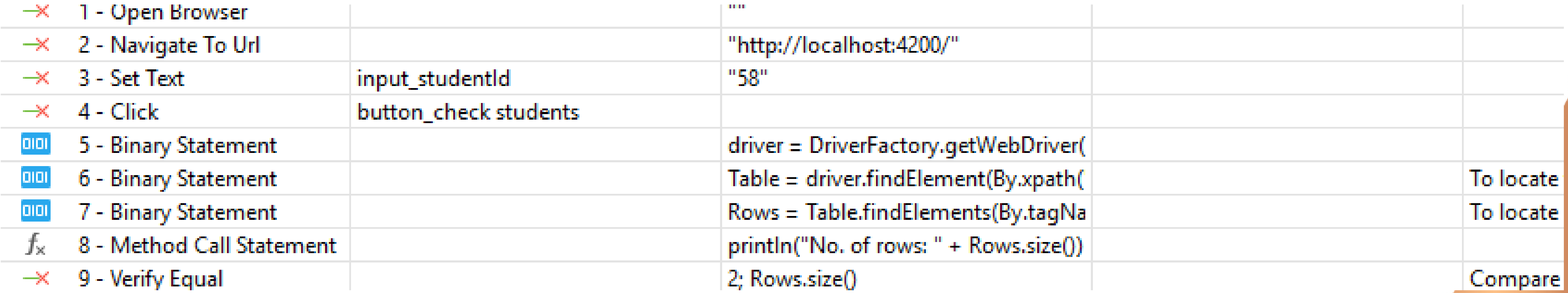
- Selenium use driver to execute the test
- Drive can get Dom element

Compare by the number of row



```
WebUI.click(findTestObject('for check students/Page_Lab 234/button_check students'), FailureHandling.STOP_ON_FAILURE)

WebDriver driver = DriverFactory.getWebDriver()
'To locate table'

WebElement Table = driver.findElement(By.xpath('//*[@id="resultTable"]'))
'To locate rows of table it will Capture all the rows available in the table '

List<WebElement> Rows = Table.findElements(By.tagName('tr'))

println('No. of rows: ' + Rows.size())
'Compare the value'
WebUI.verifyEqual(2, Rows.size())
```

In the manual pane



1 - Open Browser		""		
2 - Navigate To Url		"http://localhost:4200/"		
3 - Set Text	input_studentId	"58"		
4 - Click	button_check students			
5 - Binary Statement		driver = DriverFactory.getWebDriver(		
6 - Binary Statement		Table = driver.findElement(By.xpath(		To locate
7 - Binary Statement		Rows = Table.findElements(By.tagNa		To locate
8 - Method Call Statement		println("No. of rows: " + Rows.size())		
9 - Verify Equal		2; Rows.size()		Compare

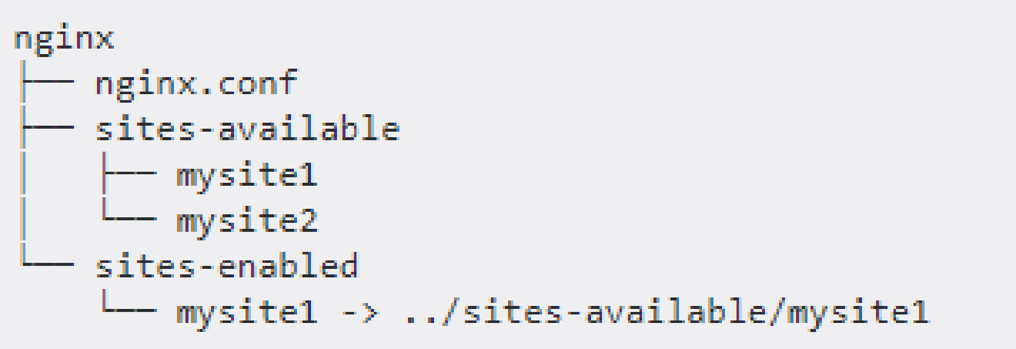
- The Binary is shown
  - Not the Katalon components

Customized

- Other Java test component can be import
- As a part of Groovy
    - All the Java components can be used

Docker mapping

- Normal server setting

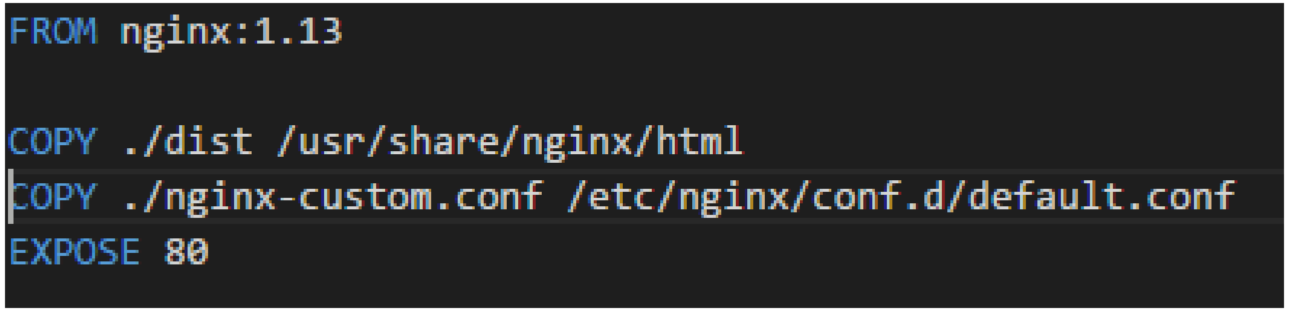


```
nginx
├── nginx.conf
├── sites-available
│   ├── mysite1
│   └── mysite2
└── sites-enabled
    └── mysite1 -> ../sites-available/mysite1
```

- Configuration file must be in the proper directory

How

- Add the file when the create image



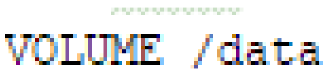
```
FROM nginx:1.13

COPY ./dist /usr/share/nginx/html
COPY ./nginx-custom.conf /etc/nginx/conf.d/default.conf
EXPOSE 80
```

- Copy the configuration file to the image
- When run the container, the setting is as we created

Problem?

- Not flexible
- Some data are written in the writeable layer
- When the container stop
  - All written data gone
- Hard to set a new configuration
  - A new build is required
  - Also the login and push image is required

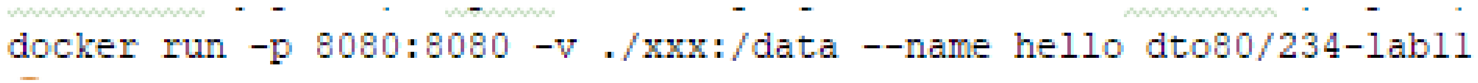
Mounting Volume



```
VOLUME /data
```

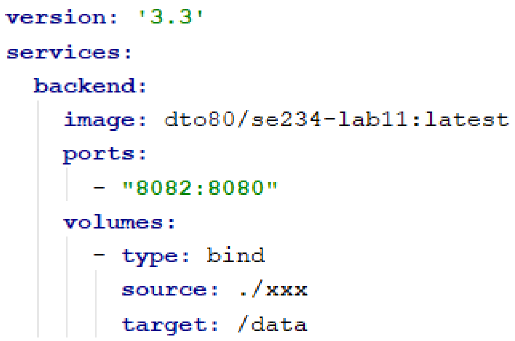
- Open the place to mount value
- Mount the value from the host, and the container

To mount volume



```
docker run -p 8080:8080 -v ./xxx:/data --name hello dto80/234-lab11
```

- -v mount the host dir : the container dir



```
version: '3.3'
services:
  backend:
    image: dto80/se234-lab11:latest
    ports:
      - "8082:8080"
    volumes:
      - type: bind
        source: ./xxx
        target: /data
```

What will happen?

- Files in your host directory will override what is in the container
- The generated file will be written in the host folder
  - Even stop the container
  - When restart
    - The file will be read and continue the work

Q & A

CAMT
College of Arts, Media and Technology
Chiang Mai University