

KATALON STUDIO TRAINING

Presented By : Testbits Sdn Bhd



Mini Assessment 1

Duration (30mins)

- Create and Automate Test Cases on CMS Katalon Shop based on the test cases given.

LEARNING OUTCOME

- Participants able to implement test cases creation based on situation given.
- Participants able to create TS / TSC and understand their features.
- Participants able to generate various type of reports.
- Participants able to setup email notifications for TS / TSC reporting.
- Participants able to create their own custom keywords to be used in their respective TCs.

2.1 TEST SUITE & TEST SUITE COLLECTION

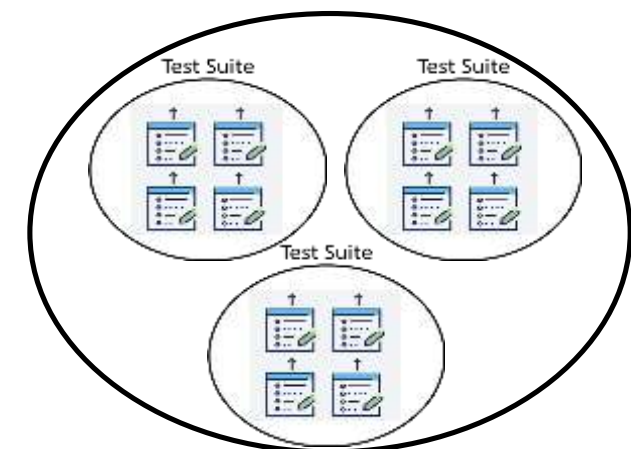
Test Suite - A collection of test cases or test scenarios that are designed to verify the functionality and quality of a software application. It includes a set of inputs, expected outputs, and the conditions under which the software should behave.

Test Suite Collection - A test suite collection is a group of related test suites that are designed to test an entire software application or a specific module or component of the application.

Test Suite

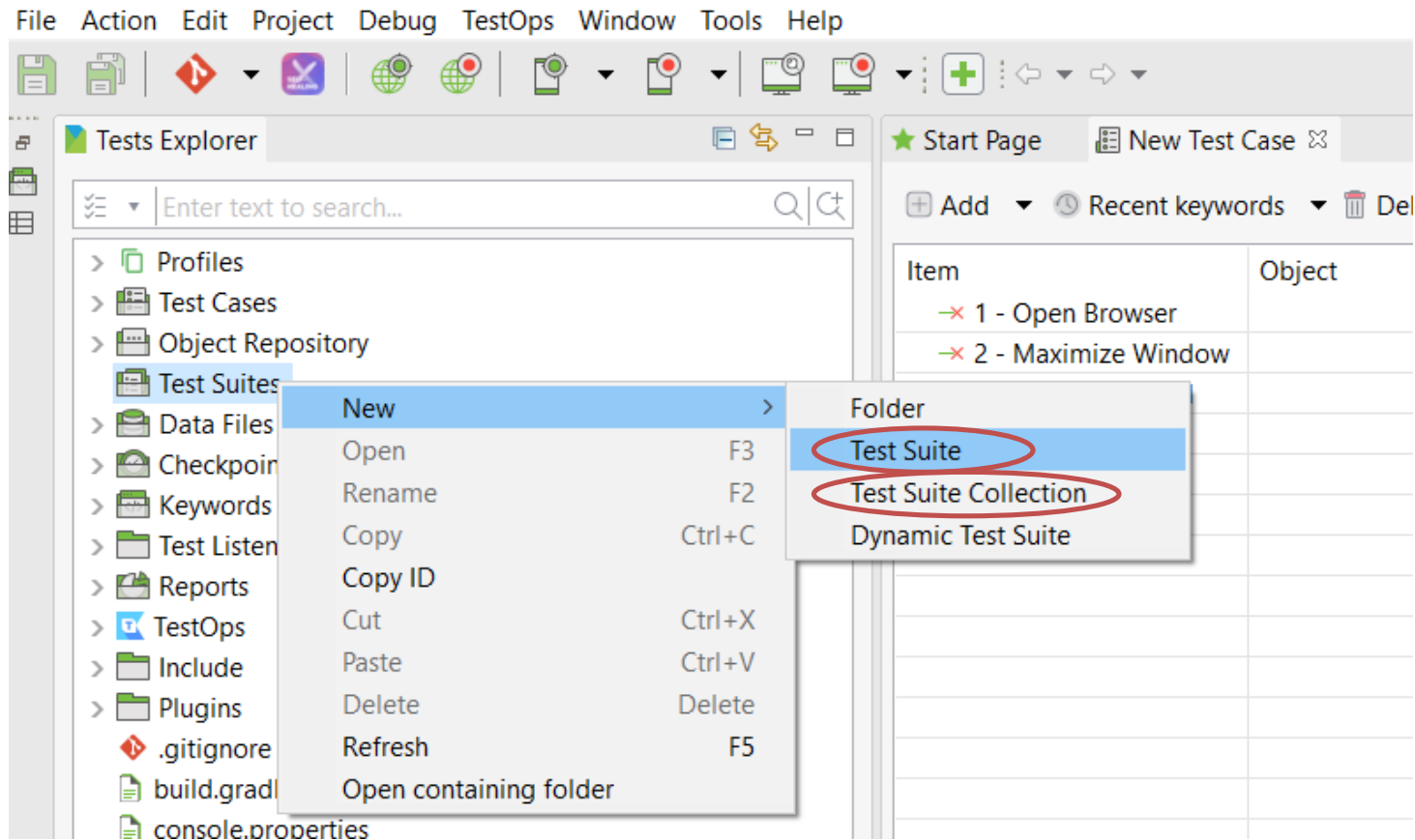


Test Suite Collection



2.1 TEST SUITE & TEST SUITE COLLECTION

Step 1: Create New Test Suite/ Test Suite Collection



1. Right Click Test Suite Folder
2. Select New
3. Click Test Suite or Test Suite Collection

2.1 TEST SUITE & TEST SUITE COLLECTION

Step 2: Add Test Case into Test Suite

Step 3: Add Test Suite into Test Suite Collection

Step 4: Execute Test Suite and Test Suite Collection

2.1.1 Multiple Browser Support (For TSC)

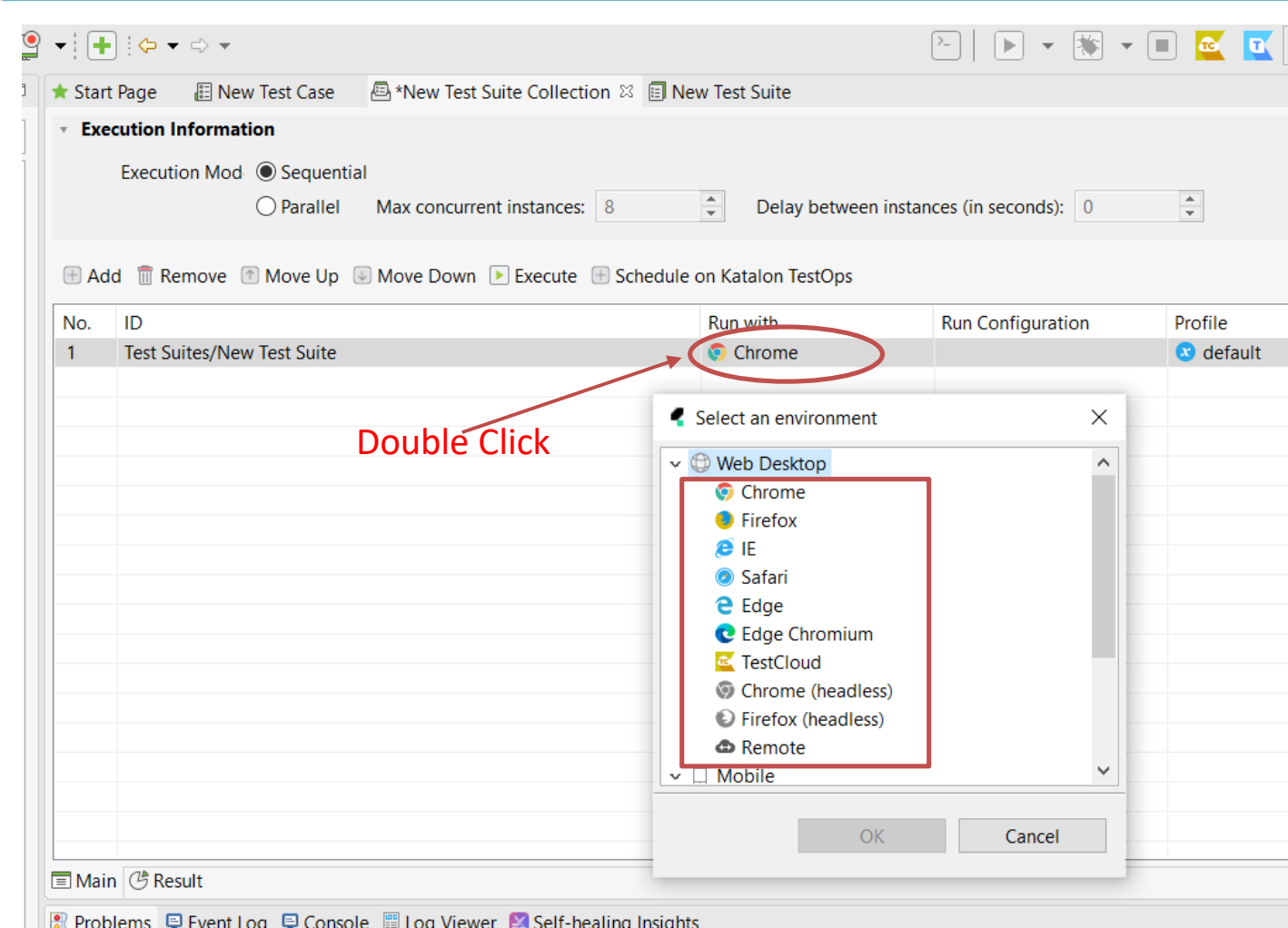
Multi-Browser Support only available in Test Suite Collection

Step 1: Create New Test Suite Collection

Step 2: Add Test Suite to Test Suite Collection

Step3: Select Browser

Step4:Execute Test Suite Collection

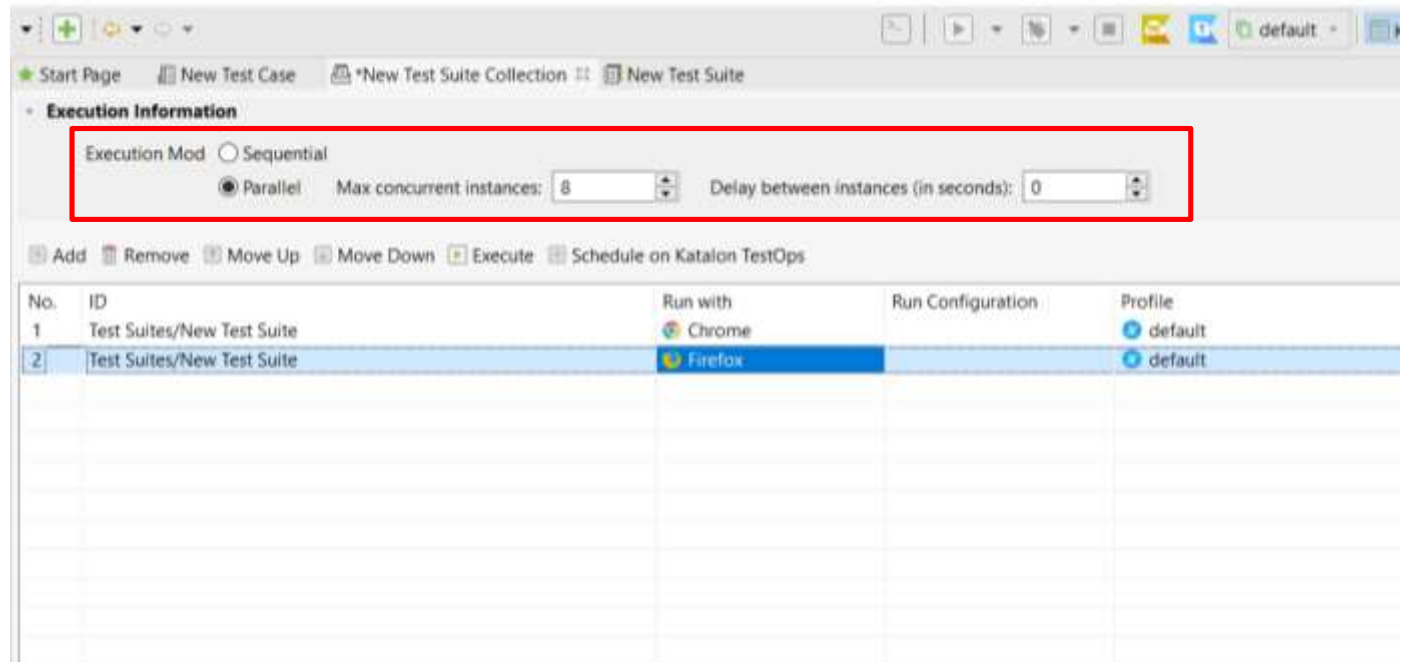


2.1.2 Sequential and Parallel (For TSC)

Sequential Execution Mod – The Test Suites will execute one-by-one according to its order.

Parallel Execution Mod – All the Test Suites will execute concurrent or simultaneously.

Ex. Both Test Suite will run for both Chrome and Firefox Browser at the same time in Parallel Mod



2.2 Logs & Reports

Upon executing Test Cases, Test Suite and Test Suite Collection regardless the results, Katalon Studio will generate Logs that can be viewed for test cases that Passed or Failed(Log Viewer).

The screenshot displays the Katalon Studio interface with the Log Viewer open. The top bar shows the menu (Action, Edit, Project, Debug, Test Suite, Window, Tools, Help) and the toolbar. The Log Viewer tab is active, showing the following summary:

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

The Log Viewer displays the following test results:

- Test Suites/New Test Suite (11.595s)
 - hostname = izzat - LAPTOP-GFVROA5E.realtek
 - os = Windows 10 64bit
 - hostAddress = 192.168.0.9
 - katalonVersion = 8.6.0.208
- Test Cases/New Test Case (10.841s)
 - 1 - openBrowser("") (2.050s)
 - 2 - maximizeWindow() (0.170s)
 - 3 - navigateToUrl("https://opensource-demo.orangehrmlive.com/web/index.php/auth/login") (3.921s)
 - 4 - setText(findTestObject("Login HRM/Input-Username"), "Admin") (1.473s)
 - 5 - setText(findTestObject("Login HRM/Input-Password"), "admin123") (0.679s)
 - 6 - click(findTestObject("Login HRM/Button-Login")) (1.596s)
 - 7 - closeBrowser() (0.652s)
 - exportKatalonReports (0.501s)

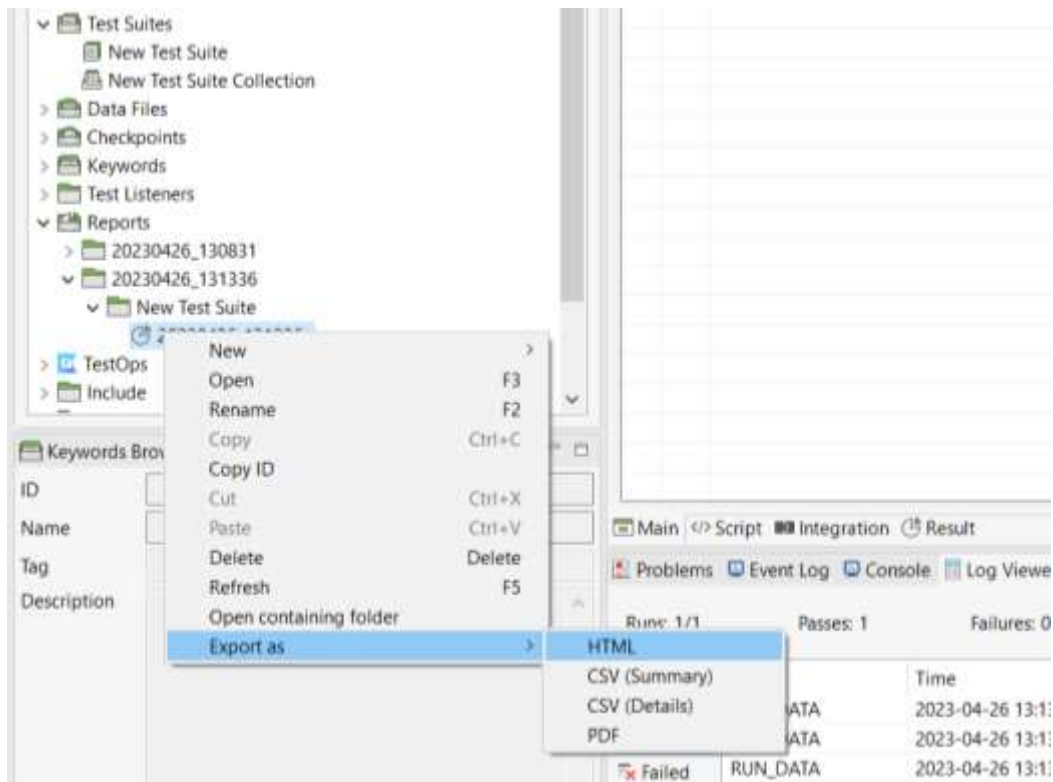
The right pane shows the test results for the selected test case, including the timestamp (04-26-2023 01:13:45 PM) and the elapsed time (11.063s).

Annotations in the image point to the Log Viewer tab, the summary statistics, the test results tree, and the right pane, with the following text:

- Click on Log Viewer to see all steps executed for Passes, Failures, Errors and Skips
- Click on Tree View to see all or filter steps executed for Passes, Failures, Errors and Skips
- Tree View

2.2 Logs & Reports

Overall Reports only generated everytime Test Suite or Test Suite Collection is executed and can be exported as html, PDF and CSV Format.



Export/ View Report

Step 1: Open Reports Folder in the Test Explorer

Step 2: Right-Click on the Test Suite/Test Suite Collection Folder

Step3: Select Export as HTML,CSV or PDF

2.2 Logs & Reports

Host name: izzat - LAPTOP-GFVROA5E.reelek
Local OS: Windows 10 64bit
Katalon version: 8.6.0.208
Browser: Chrome 112.0.0.0

Test Execution Log

TEST SUITE: New Test Suite

Full Name: New Test Suite
Start / End / Elapsed: 2023-04-26 13:13:45.413 / 2023-04-26 13:13:56.476 / 00:00:11.063
Status: 1 test total, 1 passed, 0 failed, 0 error, 0 incomplete, 0 skipped

TEST CASE: Test Cases/New Test Case

Full Name: New Test Suite/Test Cases/New Test Case
Start / End / Elapsed: 2023-04-26 13:13:45.635 / 2023-04-26 13:13:56.476 / 00:00:10.841
Status: PASSED

TEST STEP: openBrowser()

Start / End / Elapsed: 2023-04-26 13:13:45.932 / 2023-04-26 13:13:47.982 / 00:00:02.050
13:13:46.164 INFO Starting 'chrome' driver
13:13:46.203 INFO Action delay is set to 0 milliseconds
13:13:47.082 PASSED Browser is opened with url: ''

TEST STEP: maximizeWindow()

Start / End / Elapsed: 2023-04-26 13:13:47.963 / 2023-04-26 13:13:48.153 / 00:00:00.170
13:13:48.153 PASSED Current window maximized

TEST STEP: navigateToUrl("https://opensource-demo.orangehrmlive.com/web/index.php/auth/login")

Start / End / Elapsed: 2023-04-26 13:13:48.154 / 2023-04-26 13:13:52.075 / 00:00:03.921
13:13:52.074 PASSED Navigate to 'https://opensource-demo.orangehrmlive.com/web/index.php/auth/login' successfully

TEST STEP: setText(findTestObject("Login_HRM/Input-Username"), "Admin")

Start / End / Elapsed: 2023-04-26 13:13:52.075 / 2023-04-26 13:13:53.548 / 00:00:01.473
13:13:53.548 PASSED Text 'Admin' is set on object 'Object Repository/Login_HRM/Input-Username'.

TEST STEP: setText(findTestObject("Login_HRM/Input-Password"), "admin123")

Start / End / Elapsed: 2023-04-26 13:13:53.548 / 2023-04-26 13:13:54.227 / 00:00:00.679
13:13:54.226 PASSED Text 'admin123' is set on object 'Object Repository/Login_HRM/Input-Password'.

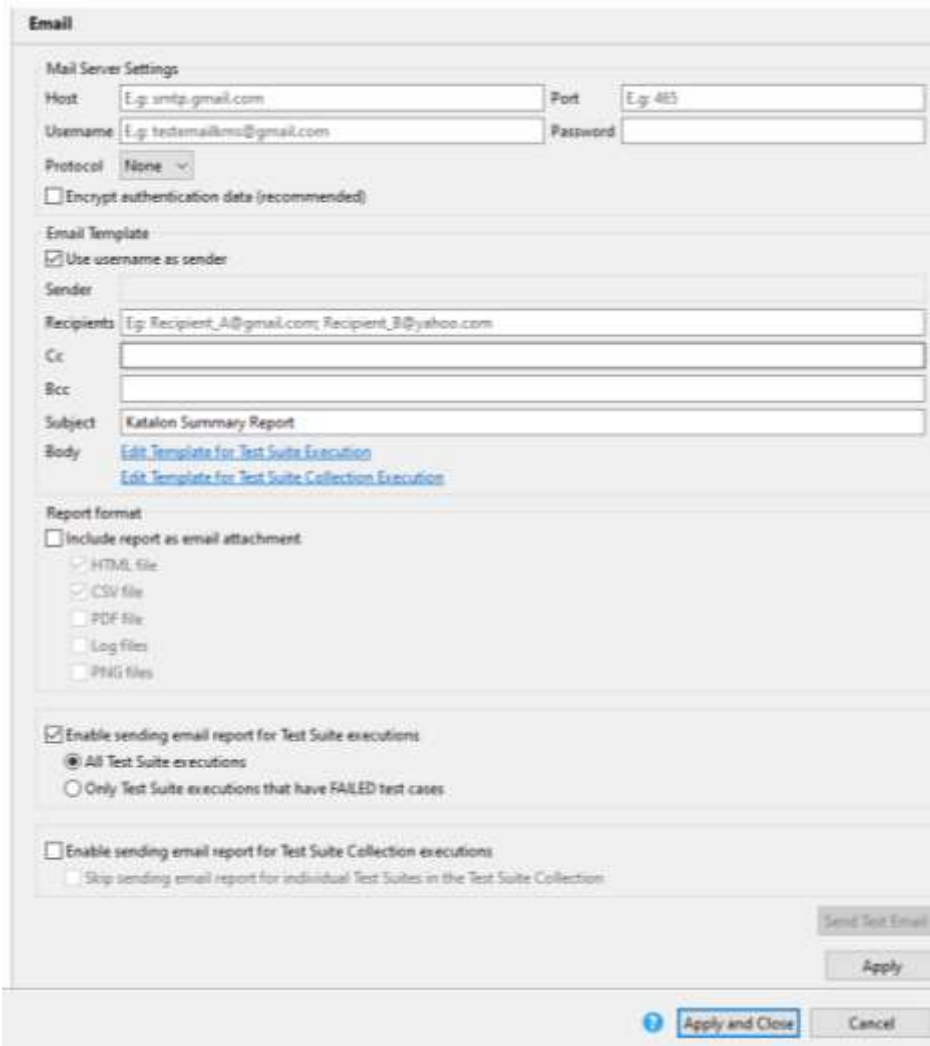
TEST STEP: click(findTestObject("Login_HRM/Button-Login"))

Start / End / Elapsed: 2023-04-26 13:13:54.227 / 2023-04-26 13:13:55.823 / 00:00:01.596
13:13:55.822 PASSED Object: 'Object Repository/Login_HRM/Button-Login' is clicked on

TEST STEP: closeBrowser()

Start / End / Elapsed: 2023-04-26 13:13:55.824 / 2023-04-26 13:13:56.476 / 00:00:00.652
13:13:56.475 PASSED Browser is closed

2.3 Email Notifications



The screenshot shows the 'Email' configuration window with the following sections:

- Mail Server Settings:** Host (E.g. smtp.gmail.com), Port (E.g. 465), Username (E.g. testemailms@gmail.com), Password, Protocol (None), and an unchecked checkbox for 'Encrypt authentication data (recommended)'.
- Email Template:** A checked checkbox for 'Use username as sender', a Sender field, a Recipients field (E.g. Recipient_A@gmail.com; Recipient_B@yahoo.com), and empty Cc and Bcc fields.
- Subject:** Katalon Summary Report
- Body:** Two links: 'Edit Template for Test Suite Execution' and 'Edit Template for Test Suite Collection Execution'.
- Report format:** An unchecked checkbox for 'Include report as email attachment' and five radio button options: HTML file, CSV file, PDF file, Log files, and PNG files.
- Execution Settings:** A checked checkbox for 'Enable sending email report for Test Suite executions' with two radio button options: 'All Test Suite executions' (selected) and 'Only Test Suite executions that have FAILED test cases'. Below it is an unchecked checkbox for 'Enable sending email report for Test Suite Collection executions' with a sub-option 'Skip sending email report for individual Test Suites in the Test Suite Collection'.

At the bottom right are buttons for 'Send Test Email', 'Apply', and 'Apply and Close' (highlighted with a red box). A 'Cancel' button is at the bottom right.

After a test suite or test suite collection execution, you might want to automatically send summary reports to your own email or other stakeholders to notify them about the test result. This document shows you how to set up your mail server and customize email reports to automatically send out a summary report email whenever a test execution finishes.

configure global email settings in
Project > Settings > Email

2.3 Email Notifications

Email

Mail Server Settings

Host	E.g: smtp.gmail.com	Port	E.g: 465
Username	E.g: testemailkms@gmail.com	Password	
Protocol	None ▾		
<input type="checkbox"/> Encrypt authentication data (recommended)			

Email Template

☒ Use username as sender

Sender

Recipients E.g: Recipient_A@gmail.com; Recipient_B@yahoo.com

Cc

Bcc

Subject Katalon Summary Report

Body [Edit Template for Test Suite Execution](#)
[Edit Template for Test Suite Collection Execution](#)

•**Username:** Your full email account to authenticate with the server (e.g., [yourusername@gmail.com](#))

•**Password:** Your email password to authenticate with the server. This could be a password generated from App Passwords.

Email sever	Host	Port	Reference
Gmail	smtp.gmail.com	465 or 587	Check Gmail through other email platforms
Outlook	smtp.office365.com	587 or 25	How to set up a multifunction device or application to send email using Microsoft 365 or Office 365
Yahoo! Mail	smtp.mail.yahoo.com	465	POP access settings and instructions for Yahoo Mail

2.4 Custom Keywords

A custom keyword is a User define keyword that extends the capabilities of Katalon Studio in addition to built-in Keywords.

How to Create Custom Keywords:

Step 1: Create New Package

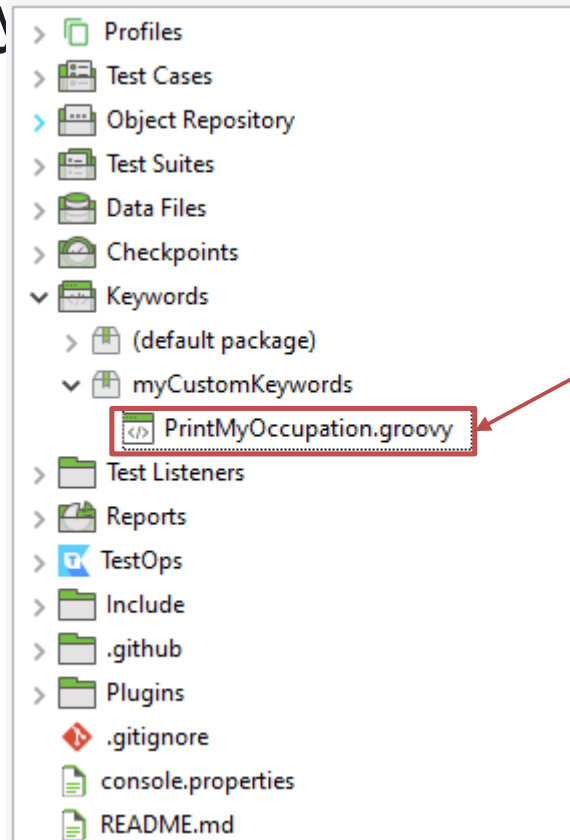
- Right Click at the “Keywords” Folder and select “New” and Choose “Package”
- Give a name to your package. (Make sure the Package name starts in lowercase Ex. “myCustomKeywords”)

Step 2: Create New Keyword

- Right Click on the newly create Package folder and select “New” and Choose “Keyword”
- Give a name to your Keyword and it will create a new .groovy script.
(Ex. PrintMyOccupation)

2.4 Custom Keywords

Step 2: Create New Key



```
★ Welcome | PrintMyOccupation.groovy ✕
1 package myCustomKeywords
2
3 import static com.kms.katalon.core.checkpoint.CheckpointFactory.findCheckpoint
4 import static com.kms.katalon.core.testcase.TestCaseFactory.findTestCase
5 import static com.kms.katalon.core.testdata.TestDataFactory.findTestData
6 import static com.kms.katalon.core.testobject.ObjectRepository.findTestObject
7 import static com.kms.katalon.core.testobject.ObjectRepository.findWindowsObject
8
9 import com.kms.katalon.core.annotation.Keyword
10 import com.kms.katalon.core.checkpoint.Checkpoint
11 import com.kms.katalon.core.cucumber.keyword.CucumberBuiltinKeywords as CucumberKW
12 import com.kms.katalon.core.mobile.keyword.MobileBuiltinKeywords as Mobile
13 import com.kms.katalon.core.model.FailureHandling
14 import com.kms.katalon.core.testcase.TestCase
15 import com.kms.katalon.core.testdata.TestData
16 import com.kms.katalon.core.testobject.TestObject
17 import com.kms.katalon.core.webservice.keyword.WSBuiltInKeywords as WS
18 import com.kms.katalon.core.webui.keyword.WebUiBuiltInKeywords as WebUI
19 import com.kms.katalon.core.windows.keyword.WindowsBuiltinKeywords as Windows
20
21 import internal.GlobalVariable
22
23 public class PrintMyOccupation {
24 }
25
```

It should look like this with the Package folder and custom keyword created

2.4 Custom Keywords

Step 3: Create New Function

```
14 import com.kms.katalon.core.testcase.TestCase
15 import com.kms.katalon.core.testdata.TestData
16 import com.kms.katalon.core.testobject.TestObject
17 import com.kms.katalon.core.webservice.keyword.WSBuiltInKeywords as WS
18 import com.kms.katalon.core.webui.keyword.WebUiBuiltInKeywords as WebUI
19 import com.kms.katalon.core.windows.keyword.WindowsBuiltinKeywords as Windows
20
21 import internal.GlobalVariable
22
23 public class PrintMyOccupation {
24
25     @Keyword (keywordObject = "<category_name>")
26     def keywordName(parameters...) {
27         // enter your code here
28         // you can use either Groovy or Java
29     }
30
31
32
33 }
```

Item	Description	Required
@Keyword	The annotation to indicate that the block of code below is the definition of a keyword.	Yes
keywordObject	The category of your custom keyword (available from version 7.5.5).	No
keywordName	The name for your custom keyword.	Yes
parameters	The list of parameters to be used in the custom keyword.	No

2.4 Custom Keywords

Step 3: Create New

```

8
9 import com.kms.katalon.core.annotation.Keyword
10 import com.kms.katalon.core.checkpoint.Checkpoint
11 import com.kms.katalon.core.cucumber.keyword.CucumberBuiltinKeywords as CucumberKW
12 import com.kms.katalon.core.mobile.keyword.MobileBuiltinKeywords as Mobile
13 import com.kms.katalon.core.model.FailureHandling
14 import com.kms.katalon.core.testcase.TestCase
15 import com.kms.katalon.core.testdata.TestData
16 import com.kms.katalon.core.testobject.TestObject
17 import com.kms.katalon.core.webservice.keyword.WSBuiltInKeywords as WS
18 import com.kms.katalon.core.webui.keyword.WebUiBuiltInKeywords as WebUI
19 import com.kms.katalon.core.windows.keyword.WindowsBuiltinKeywords as Windows
20
21 import internal.GlobalVariable
22
23 public class PrintMyOccupation {
24
25 @Keyword(keywordObject = "PrintFunction")
26 def SelectOccupation (int input) {
27
28     switch (input) {
29
30         case 1:
31             println("Automation Tester");
32             break;
33
34         case 2:
35             println("Manual Tester");
36             break;
37
38         case 3:
39             println("Automation and Manual Tester");
40             break;
41
42     }
43 }
44

```

Example to print
Occupation

```

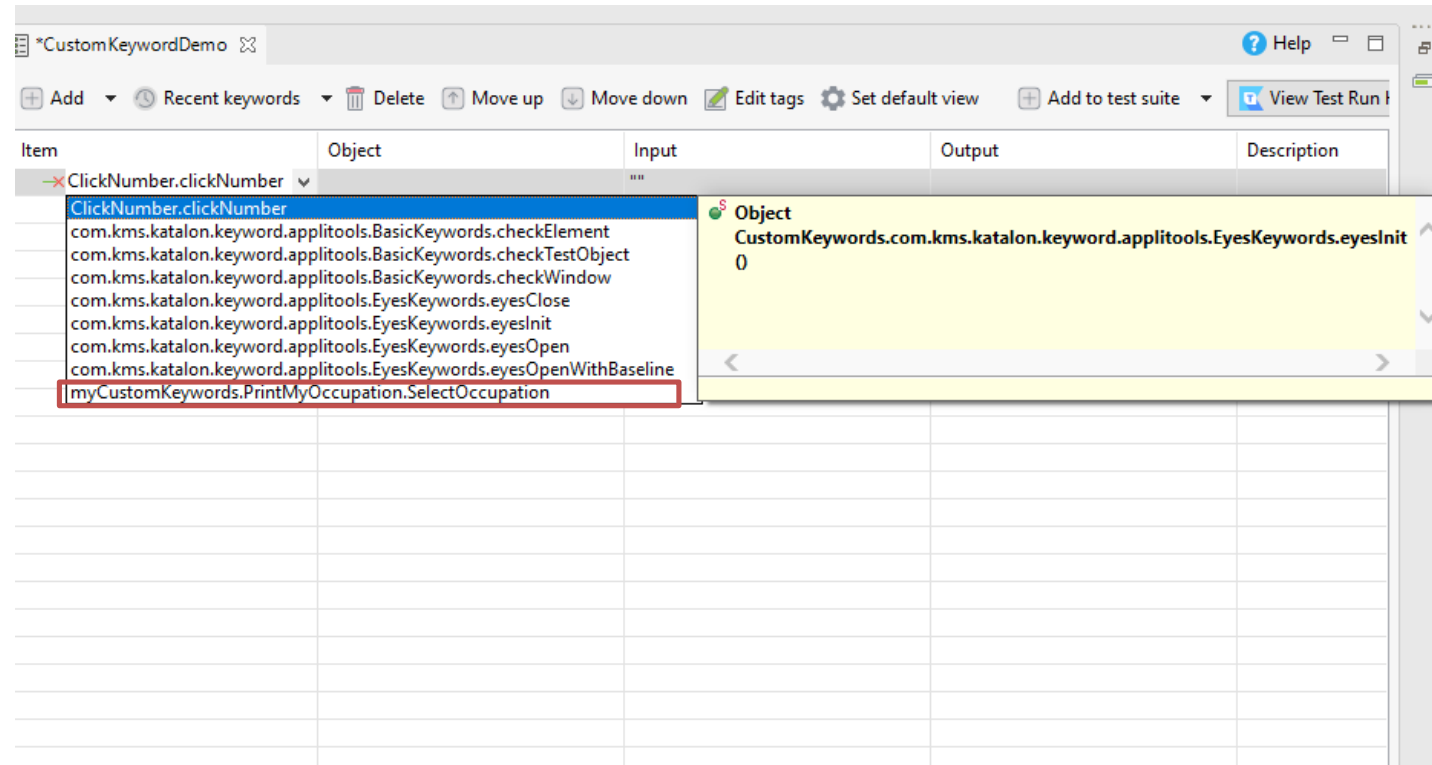
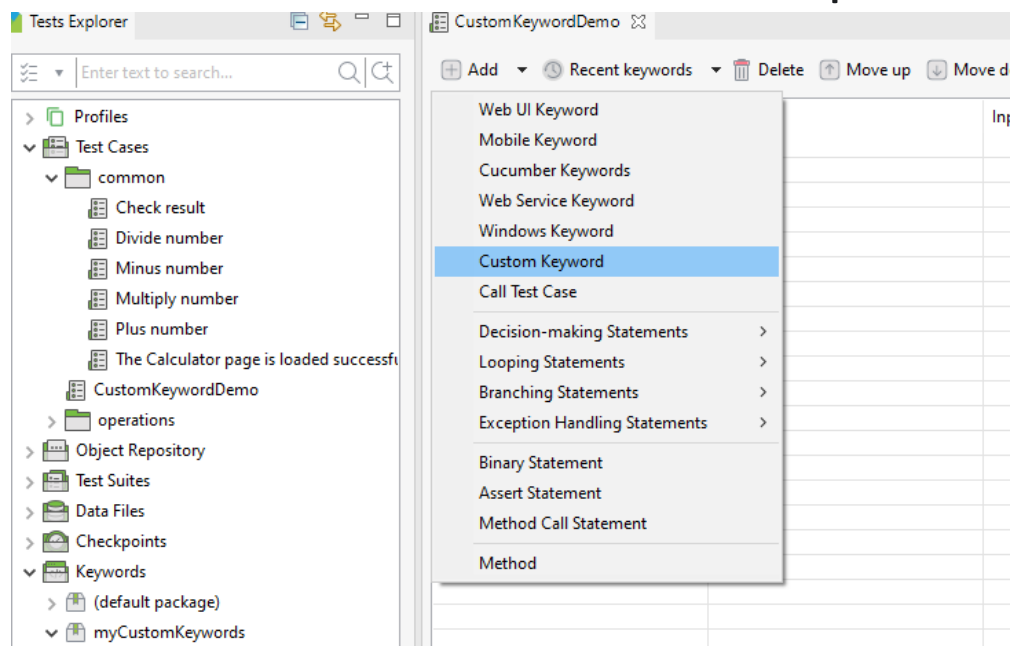
23 public class PrintMyOccupation {
24
25 @Keyword (keywordObject = "PrintFunction")
26 def SelectOccupation (int input) {
27
28     switch (input) {
29
30         case 1:
31             println("Automation Tester");
32             break;
33
34         case 2:
35             println("Manual Tester");
36             break;
37
38         case 3:
39             println("Automation and Manual Tester");
40             break;
41
42     }
43 }
44

```

2.4 Custom Keywords

Step 4: Add Custom Keyword to Test Case

-Create a new test case named “CustomKeywordDemo” and “add” Custom Keyword and choose the function “SelectOccupation”



2.4 Custom Keywords

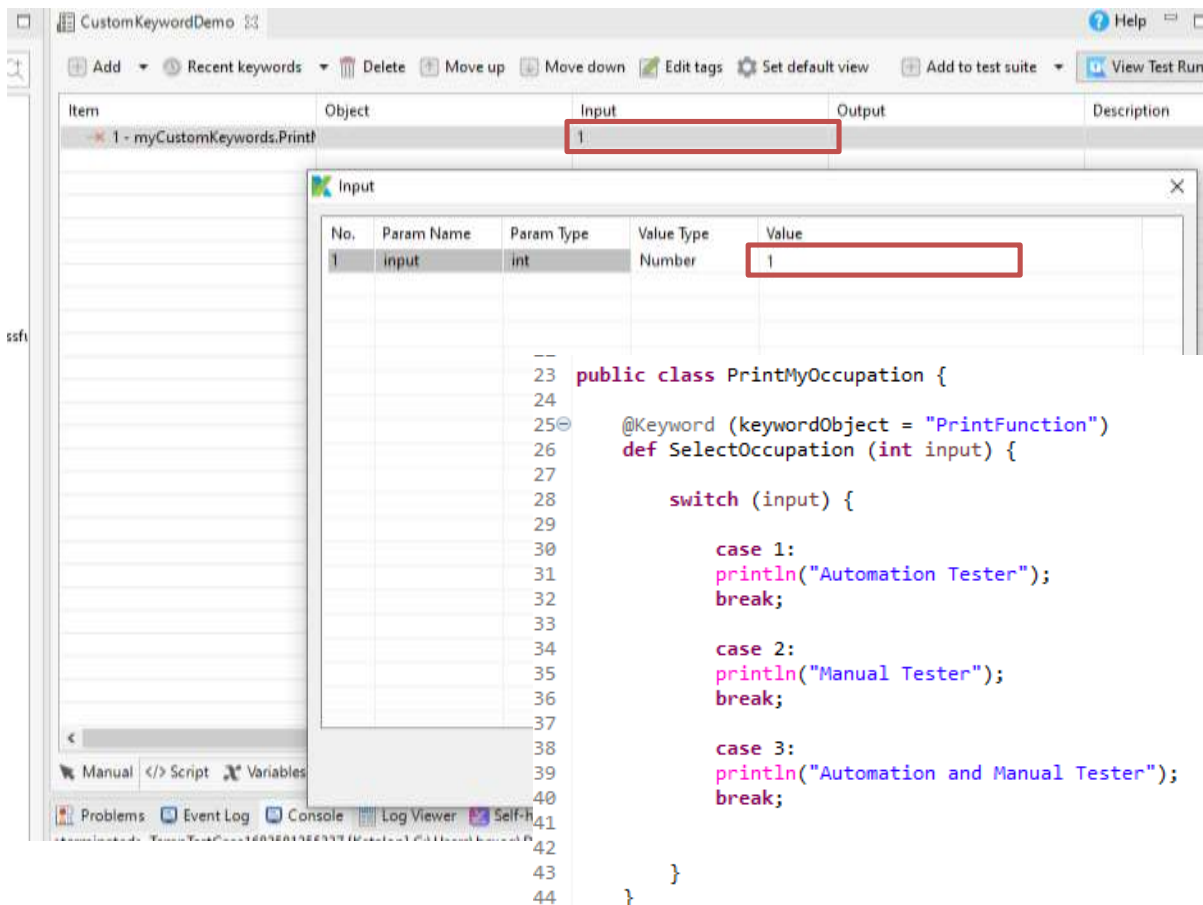
Step 5: Execute the Test Case

Change the Input to 1,2 or 3 and execute the test case while observing the result on the console.

Input 1 = Automation Tester


Input 2 = Manual Tester

Input 3 = Automation and Manual Tester




Q&A

THE END

 +60388005219

 <https://mytestbits.com/>

 contactus@mytestbits.com

 B-01-02, Tamarind Square, Persiaran Multimedia, Cyber 10,
63000 Cyberjaya, Selangor, Malaysia