

테스트 툴 & 자동화 교육



3.2 Katalon Studio 실습



3.2.1 Katalon-Studio란

❖ Katalon Studio



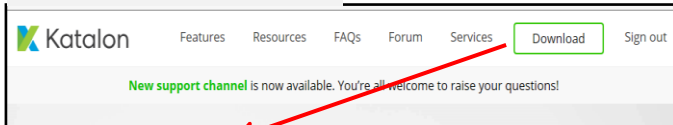
- <https://www.katalon.com/>
- KMS Technology(테스트가 발달한 인도계 회사에서 만든 듯?)
- 무료(맞춤형 고객 지원 요청 가능)
- 웹, 모바일, (REST) API 테스트 가능
- 테스트 자동 실행 뿐만 아니라 테스트 케이스, 리포트 기능 등 지원 (build-in)
- 커맨드 라인 실행을 통한 CI 연계 가능
- 스크립트 작성 기능(Groovy, Java), 커스텀 키워드 지원
- Data-driven, Keyword-driven 테스트 자동화 패턴이 녹아져 있음
- 설치가 쉬움
- (웹) Record&Play 기능 제공
- (웹) 멀티 브라우저 실행 제공

3.2.2 Katalon Studio 설치

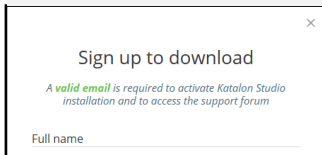
❖ Katalon Studio 설치

- 원클릭 설치(압축파일 압축해제, 이클립스에 덧씌운 형태의 IDE 및 관련 라이브러리 설치)
- 다운로드 URL : <https://www.katalon.com/#download>

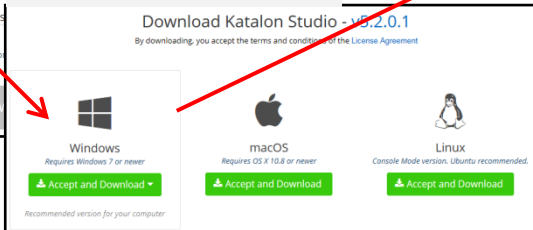
1) Download 메뉴 선택



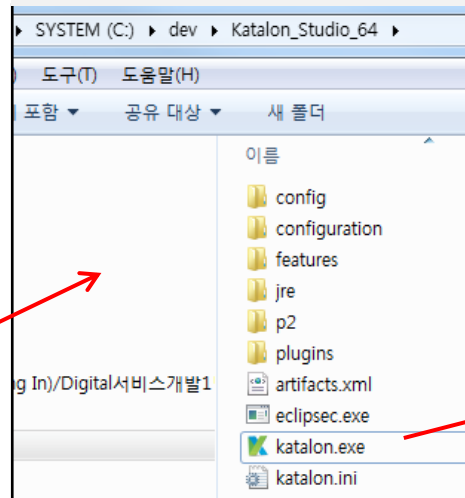
2) 회원가입, 로그인 필요



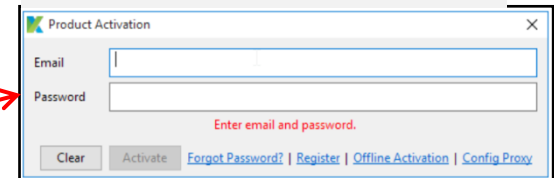
3) OS별 다운로드



4) 임의의 위치에 압축해제, katalon.exe 실행



5) 최초 실행 시 로그인 정보 입력 필요





3.2.3 Katalon Studio 작성

※ 사용 Tutorials

- 유튜브 가이드 및 문서 튜토리얼 제공 : <https://docs.katalon.com/display/KD/Tutorials>

- 목차(캡처)

• Get Started

- Installing and Setting up Katalon Studio.
- GUI Overview.
- Configuring Katalon Studio for the Web automation test project
- Setup mobile automation project in Windows.
- Setup mobile automation project in macOS.
- Setup API testing project.

• Test Design

- Create test case using Web Recorder utility.
- Create test case using Manual mode.
- Create test case using Script mode.
- Handle drag and drop testing for web applications.
- Testing mobile apps using Katalon Studio and Kobiton's cloud-based device farm
- Define and use custom keywords in your test case.
- Data-driven testing approach with Katalon Studio.
- Detect elements with XPath.
- Handling iframe
- Common validation statements with Katalon Studio
- Common Conditions and Control statements with Katalon Studio

• Test Execution

- Executing test case
- Debugging test case
- Execute/debug certain steps of a test case.
- Viewing test suite rerports
- Viewing execution logs
- Consolidate automation test into your QA process.
- Executing in console mode.

• Integration

- Kobiton Integration.
- JIRA Integration.
- CI/CD integration.
- Git integration.

• How-to

- How to create a custom keyword
- How to import external library
- How to perform Database testing using Katalon Studio

• Troubleshooting

- Solving Pop-up dialog issue
- Solving common issue - Wait-time

• Sample Project

- Sample Web Automation Test Project

3.2.3 Katalon Studio 작성

❖ Katalon Studio 구성

- 1)상단 Top 메뉴 영역 : 프로젝트 열기/저장, 글씨체 등 옵션, Groovy/Java 설정 등
- ※ Spy Web/Mobile : 떠있는 웹/모바일 오브젝트에 대한 속성정보 확인,
- ※ Record Web/Mobile : 떠있는 웹/모바일 상에서 테스트 기록해서 테스트 생성하기
- ※ Run/Debug/Stop : 테스트 (한단계씩) 실행하기

The screenshot shows the Katalon Studio interface with several key areas highlighted and labeled:

- 1)상단 Top 메뉴 영역**: The top menu bar and toolbar containing options like File, Edit, Project, Debug, Window, Help, and various icons for saving, running, and debugging.
- 2)Test Explorer**: The left sidebar showing a tree view of test cases, object repositories, and test suites.
- 3)상세 테스트 정보 영역**: The central pane displaying a table of test steps with columns for Item, Object, Input, Output, and Description.
- 4)실행 정보 영역**: The bottom pane showing execution details, including a list of steps with their execution times and a summary of the run results (Runs: 1/1, Passes: 1, Failures: 0, Errors: 0).
- 5)기타**: The right sidebar containing panels for Global Variables, Job Progress, and other utility tools.

2)Test Explorer :
테스트 케이스 / 테스트류음 관리,
데이터, 리포트 등 관리

4)실행 정보 영역



3.2.3 Katalon Studio 작성

❖ 전체 실습 순서

(1) 테스트 케이스 "NaverMovieSearch" 레코딩 및 실행

- (a) www.naver.com 열기
- (b) 영화 카테고리 검색 클릭
- (c) 검색어 "명량" 입력 후 검색
- (d) 영화 상세 정보로 이동, 검색 결과에 키워드 있는지 검증
- (e) 다시 첫 페이지로 돌아오기

(2) 테스트 폴더 구성, Suite "TS_NaverMovieSearch" 생성 및 TC 할당

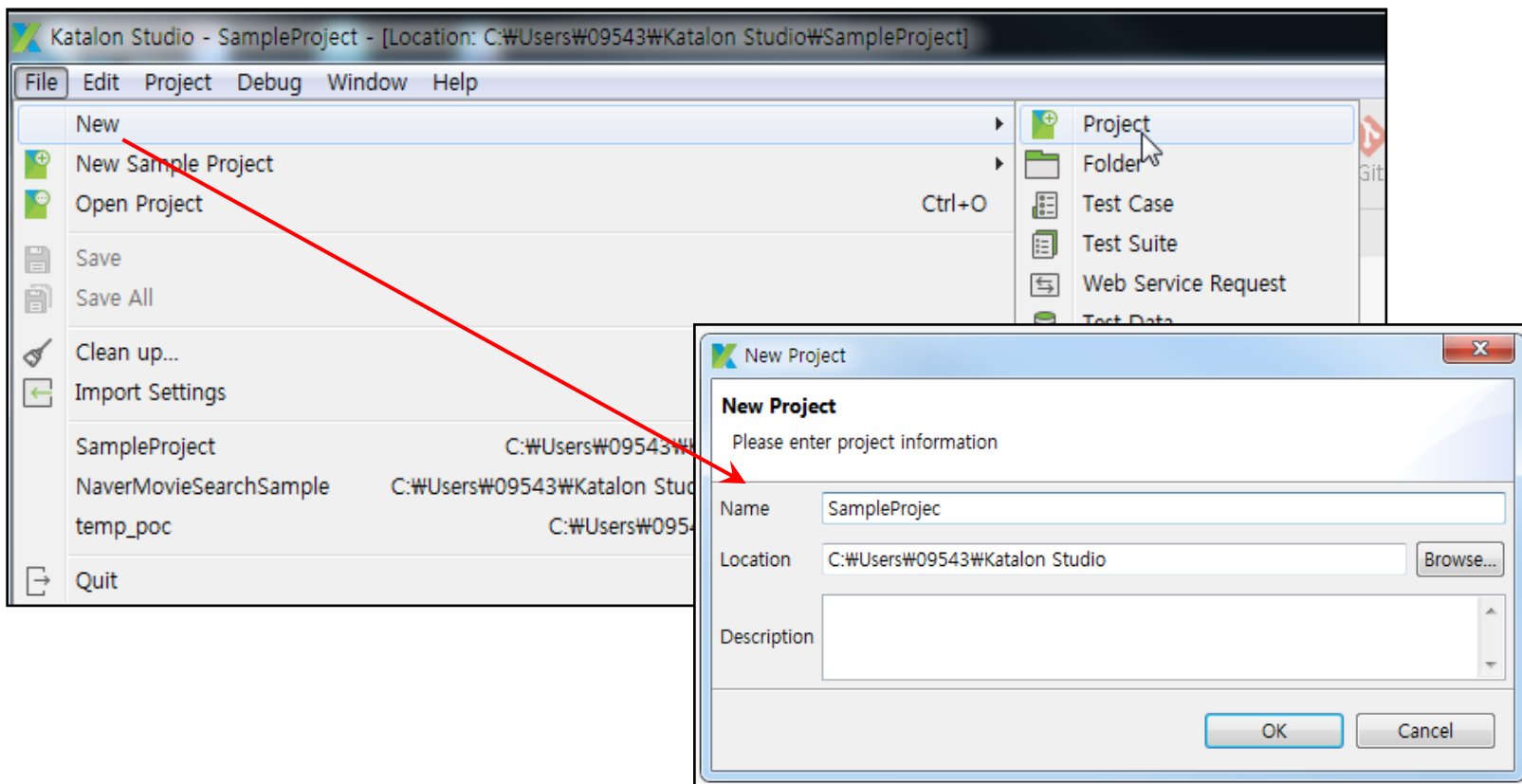
(3) 테스트 케이스 이름 변경, 검증 포인트 설정

(4) 빌드 명령어 생성

3.2.3 Katalon Studio 작성

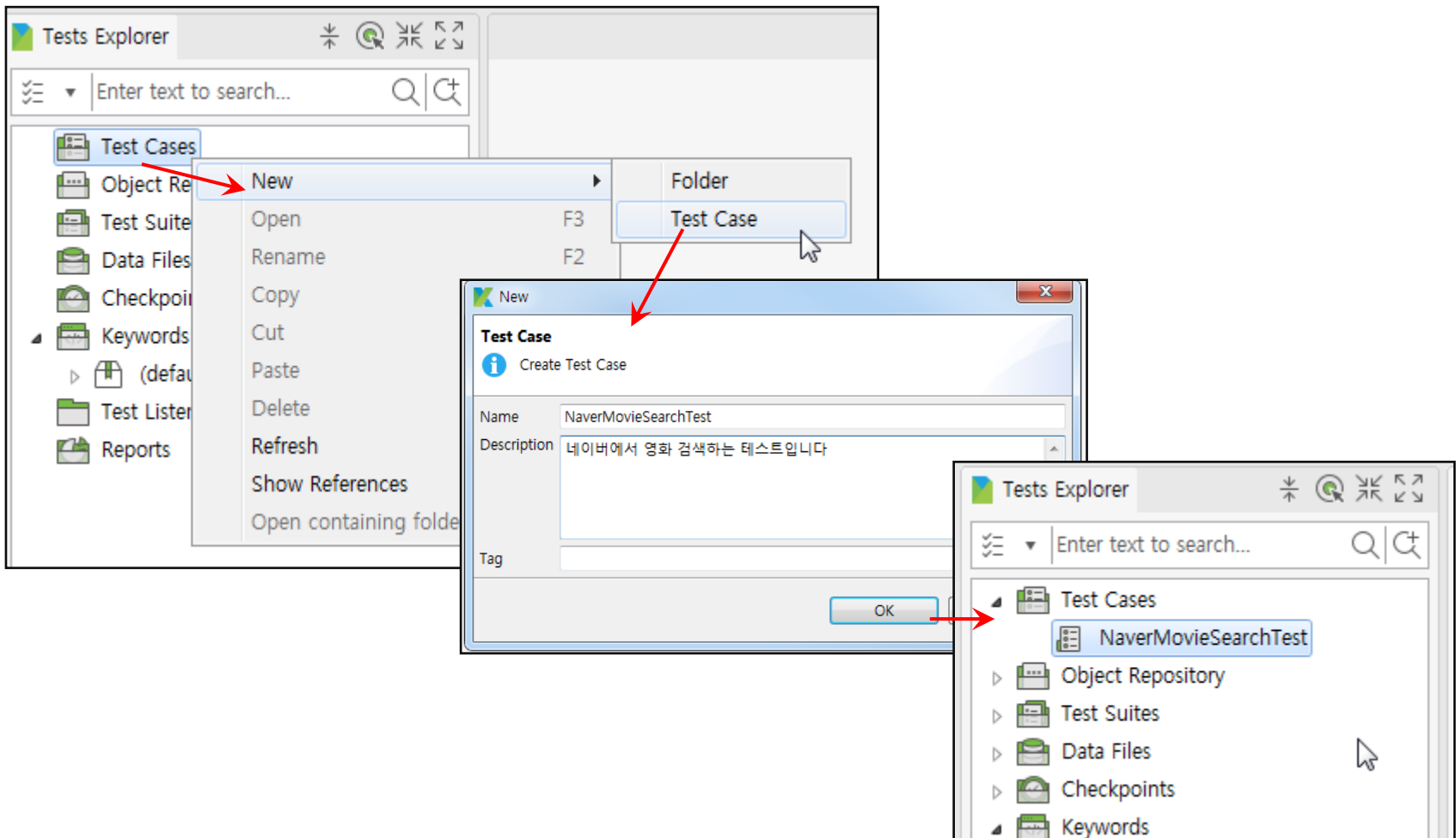
(1) 테스트 케이스 “NaverMovieSearch” 레코딩 및 실행

1) 상단 File > New 메뉴에서 새 프로젝트를 생성한다



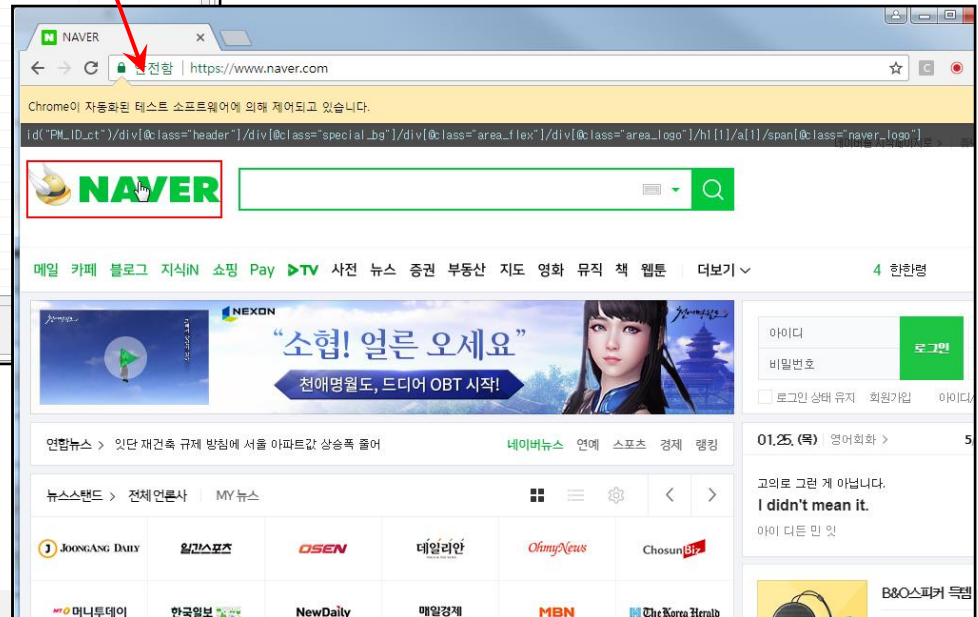
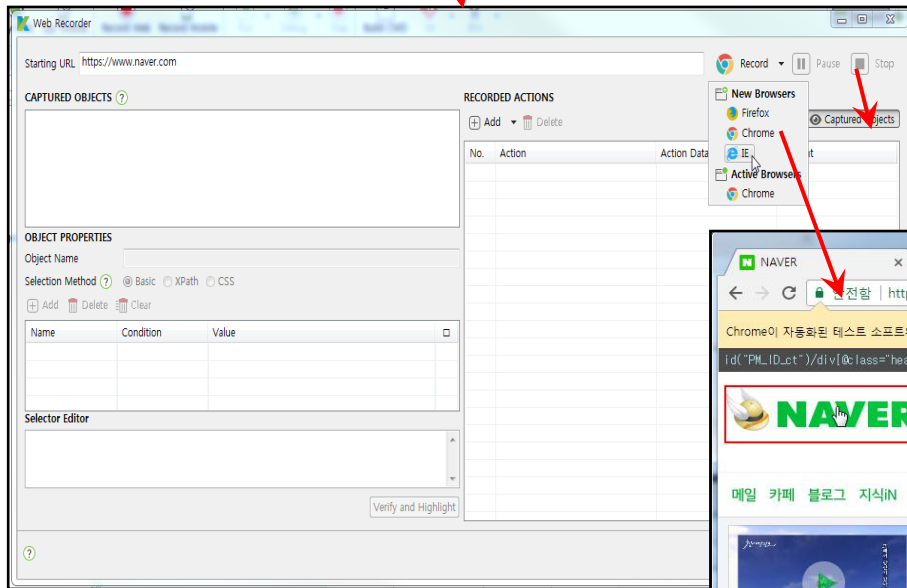
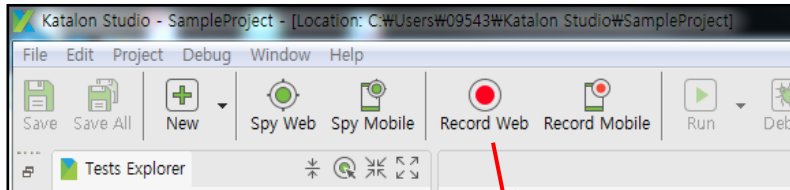
3.2.3 Katalon Studio 작성

- 2) 왼쪽 Tests Explorer에서 TestCases를 선택하고 New > Test Case를 선택해 테스트 케이스 아이템을 생성한다
생성된 테스트 케이스를 더블클릭해서 오픈한다



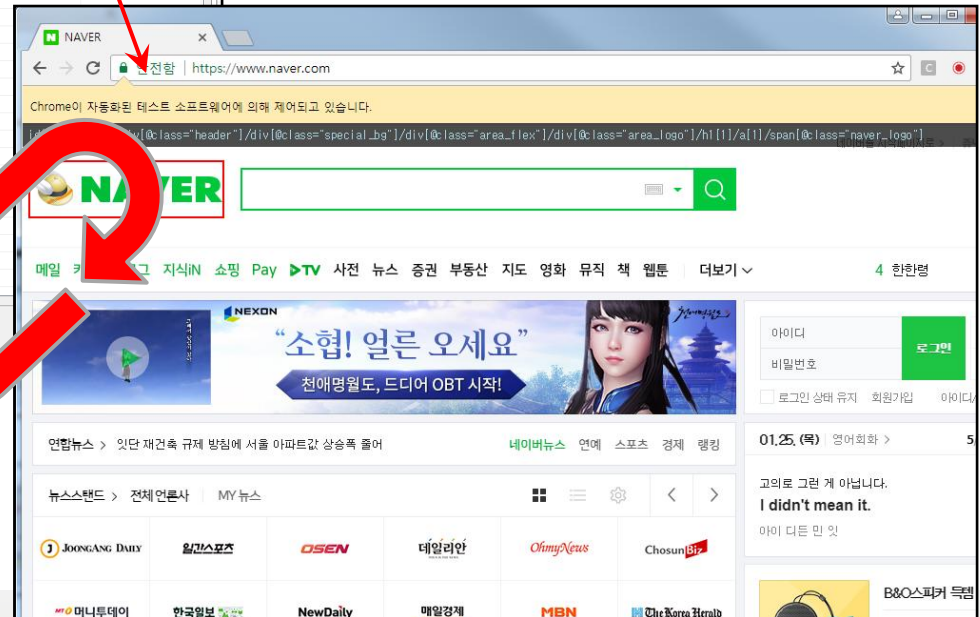
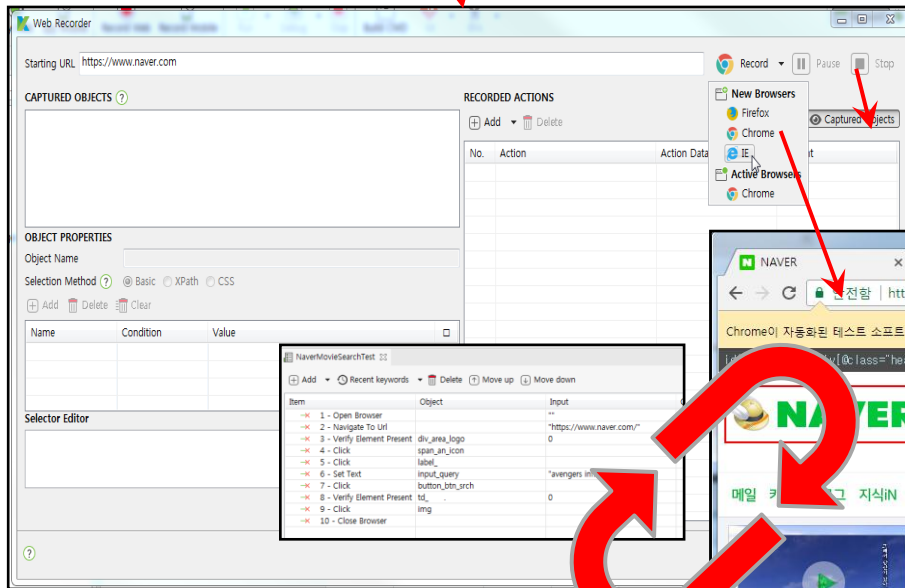
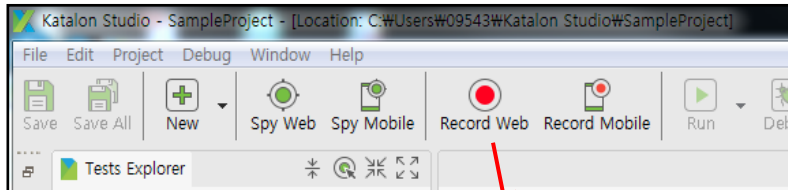
3.2.3 Katalon Studio 작성

3) 상단 Record Web 클릭 > 팝업 창의 우상단 Record 버튼과 녹화를 수행할 브라우저를 선택한다



3.2.3 Katalon Studio 작성

4) 녹화를 진행할 동작을 수행하면 각 단계가 RECORDED ACTIONS에 기록된다



3.2.3 Katalon Studio 작성

※ Tip. 엘리먼트(오브젝트)의 인식 방법

웹 엘리먼트에 대해 tag, class, xpath 등 다양한 방식으로 인식 설정할 수 있습니다

로그인 영화검색 검색

<- 컨트롤하고 싶은 웹 엘리먼트

↓ 실제 html 소스

```
<button type="submit" title="검색" class="btn_srch" onClick="clickcr(this,'GNB.search','','',event); delayed_submit(this);">검색</button>
```

button_btn_srch

Settings

Have parent object?

☒ No

☐ Parent iframe

☐ Shadow Root Parent

Use relative path for image

Image

Object's Properties

Selection Method ☒ Basic ☐ CSS ☐ XPath

+ Add - Delete Clear

Name	Condition	Value	Detect object by?
tag	equals	button	<input checked="" type="checkbox"/>
type	equals	submit	<input checked="" type="checkbox"/>
title	equals	검색	<input checked="" type="checkbox"/>
class	equals	btn_srch	<input type="checkbox"/>
onclick	equals	clickcr(this,'GNB.search','','',event); delayed_submit(this);	<input type="checkbox"/>
xpath	equals	id("SearchArea")/div[@class="srch field on_view"]/button[...	<input type="checkbox"/>

Selector Editor

```
//button[count(. //*[@type = 'submit']) = count(. //*[@type = 'submit'])][count(. //*[@title = '검색']) = count(. //*[@title = '검색'])]
```

Katalon Studio

엘리먼트 인식 방식 설정 창 →

(따로 선택할 필요는 없지만 향후

인식이 잘 안 되거나, 유사한 다른

엘리먼트가 인식, 엘리먼트 변경 등의

경우 선택을 바꿀 수 있다)

3.2.3 Katalon Studio 작성

※ Tip. Spy 기능을 이용해 Recording 중이 아니더라도 특정 오브젝트의 식별 값을 가져올 수 있다

The screenshot shows the Katalon Studio interface with the Spy Web tool active. The Object Spy window is open, displaying the following information:

- Starting URL:** www.naver.com
- CAPTURED OBJECTS:**
 - Page_1516857939970
 - Page_NAVER
 - span_
- OBJECT PROPERTIES:**
 - Object Name: span_
 - Selection Method: Basic (selected), XPath, CSS
 - Table of properties:
- Selector Editor:**

```
//*[@text() = '네이버']
```

A yellow starburst with the text "오브젝트 선택" (Object Selection) points to the 'span_' object in the CAPTURED OBJECTS list.

Name	Condition	Value	
tag	equals	span	<input checked="" type="checkbox"/>
class	equals	naver_logo	<input checked="" type="checkbox"/>
text	equals	네이버	<input checked="" type="checkbox"/>
xpath	equals	id("PM_ID_ct")/div[@class="header"]...	<input type="checkbox"/>

3.2.3 Katalon Studio 작성

(검증 포인트 추가)

5) 검증을 수행할 부분의 기존 스텝을 선택하고 Insert>Insert after>Web UI Keyword를 선택

6) Verify 검증을 추가한다

(예) 특정 엘리먼트가 표시되어야 한다 -> Verify Element Present, 존재해야 하는 엘리먼트 입력 or 선택

The image consists of two screenshots from the Katalon Studio interface, illustrating the steps to add a verification keyword to a test case.

Left Screenshot: Shows a test case named 'NaverMovieSearchTest' with a table of steps. The third step, '3 - Verify Element', is selected. A context menu is open, showing options like 'Add', 'Delete', 'Copy', etc. The 'Insert' option is chosen, and a sub-menu is displayed with 'Insert before' and 'Insert after'. The 'Insert after' option is selected, leading to a list of keywords. 'Web UI Keyword' is highlighted.

Right Screenshot: Shows the same test case table. The 'Verify Element Present' keyword has been added as the third step. A dropdown menu for this keyword is open, showing a list of verification options. 'Verify Element Present' is selected.

Item	Object	Input	Output
1 - Open Browser		""	
2 - Navigate To Url		"https://www.naver.com/"	
3 - Verify Element Present		0	

Item	Object	Input	Output
1 - Open Browser		""	
2 - Navigate To Url		"https://www.naver.com/"	
3 - Verify Element Present		0	



3.2.3 Katalon Studio 작성

※ 주요 검증(Verify) 메소드

검증 함수 명	인자	설명
verifyAlertPresent/Not	N/A	Alert 창이 표시되는지 검증
verifyElementClickable/Not	TestObject to	해당 오브젝트가 클릭이 가능한지 검증
verifyElementHasAttribute/Not	TestObject to, String attributeName	해당 오브젝트에 해당 속성이 존재하는지 검증
verifyElementChecked/Not	TestObject to	해당 오브젝트가 'check'되어 있는지 검증
verifyElementPresent/Not	TestObject to	해당 오브젝트가 존재하는지 검증
verifyElementText	TestObject to, String expectedText	해당 오브젝트의 text값이 기대한 text와 동일한지 검증
verifyMatch	String actualText, String expectedText, Boolean isRegex	특정 텍스트가 regular exp 구문에 매칭되는지 검증
verifyElementVisible/Not	TestObject to	해당 오브젝트가 visible 한지 검증
verifyElementAttributeValue	TestObject to, String attributeName, String attributeValue	해당 오브젝트의 특정 attribute 값이 기대한 값과 동일한지 검증

※Tip. Wait 명령어

GUI 테스트 수행 상 테스트 스텝과 스텝과의 로딩 과정이 발생하므로 필요한 경우 verifyXXX 메소드와 비슷한 형태의 waitXXX 명령들이 존재함. 이를 적절히 사용하여야 안정적인 테스트 자동 수행 가능



3.2.3 Katalon Studio 작성

(네이버 영화검색 기본 흐름 작성1)

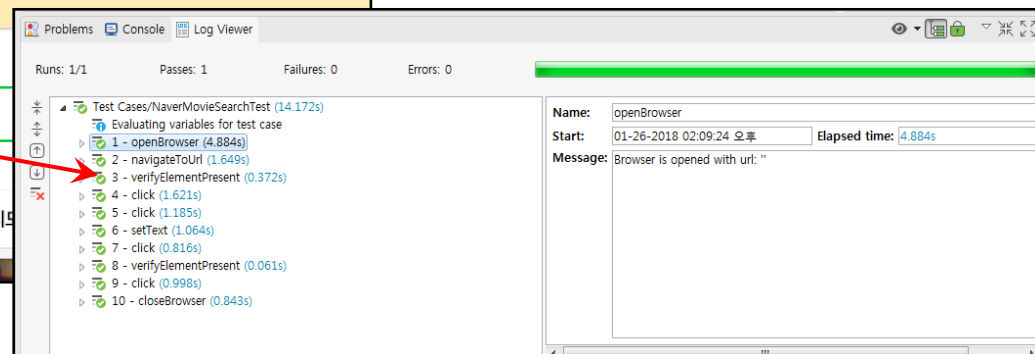
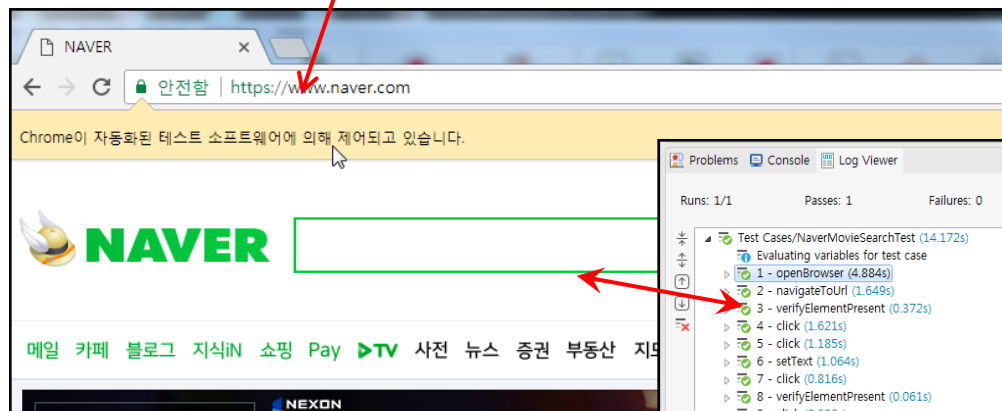
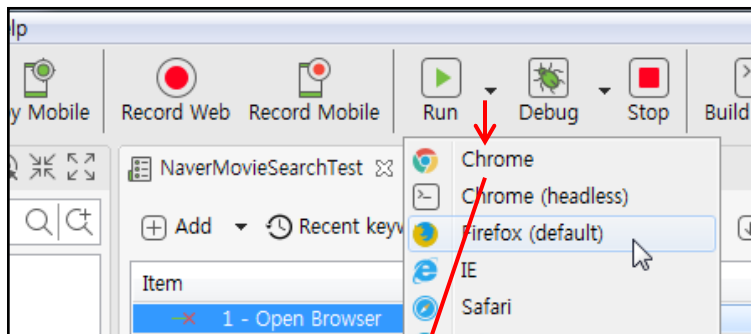
- ① 네이버 메인 페이지에서 '영화' 메뉴 선택
- ② 영화서비스 홈 이동 검증 - 좌상단 'Naver 영화' 표시되는지 검증
- ③ 검색어 필드에 검색어 'avengers infinitywar'를 입력하고 '검색' 버튼을 클릭한다
- ④ 검색 결과에 어벤저스 영화포스터, 영화명 텍스트가 존재하는지 검증한다
- ⑤ 좌상단 Naver 클릭하여 메인페이지로 돌아간다

(네이버 영화검색 기본 흐름 작성2)

별도의 테스트 케이스에서 ①~⑤의 흐름을 따라가되, 존재하지 않는 영화 검색어를 입력하고 " ~ 검색 결과가 없습니다."가 표시되는지 검증하는 테스트를 작성한다

3.2.3 Katalon Studio 작성

7) 작성된 테스트에 대해 Run > 브라우저 선택하여 테스트를 수행합니다



3.2.3 Katalon Studio 작성

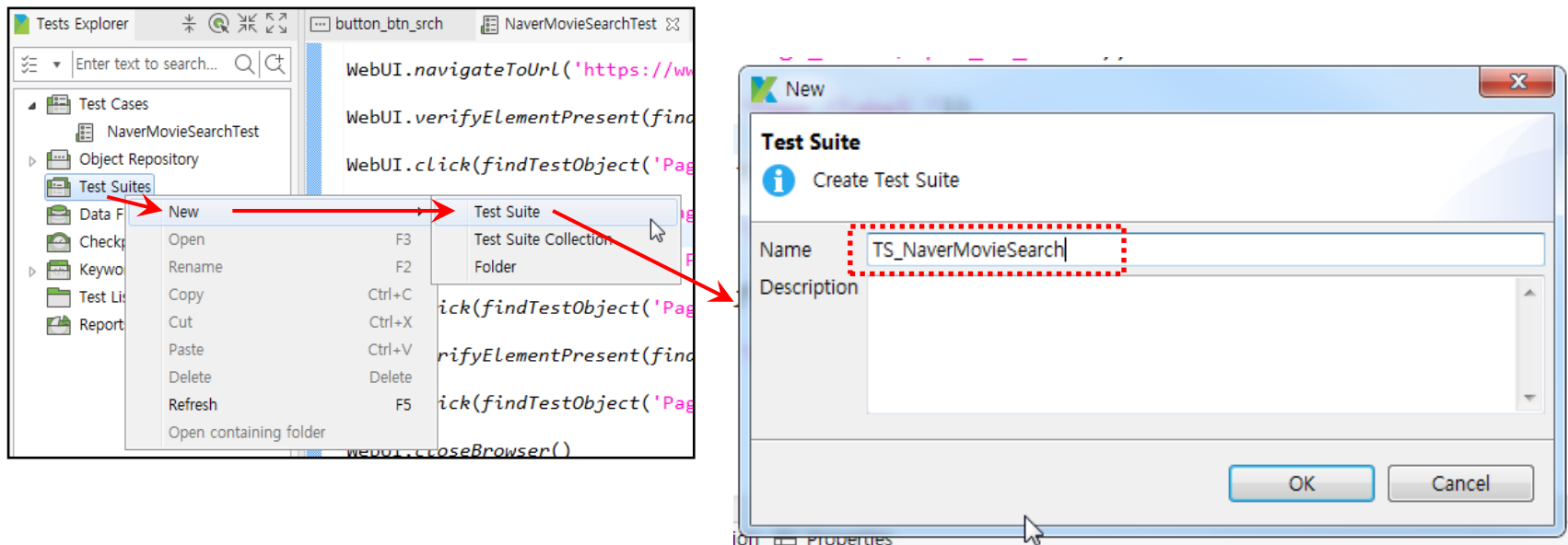
(2) 테스트 묶음(TestSuite) “TS_NaverMovieSearch” 구성

※ **TestSuite** : 테스트 케이스들을 묶는 논리적인 단위, 하나의 TestSuite 하위에 여러 TestCase구성

예) 회원가입 TestSuite – 기본회원가입 TestCase,외국인회원가입 TestCase, 미성년자회원가입 TestCase

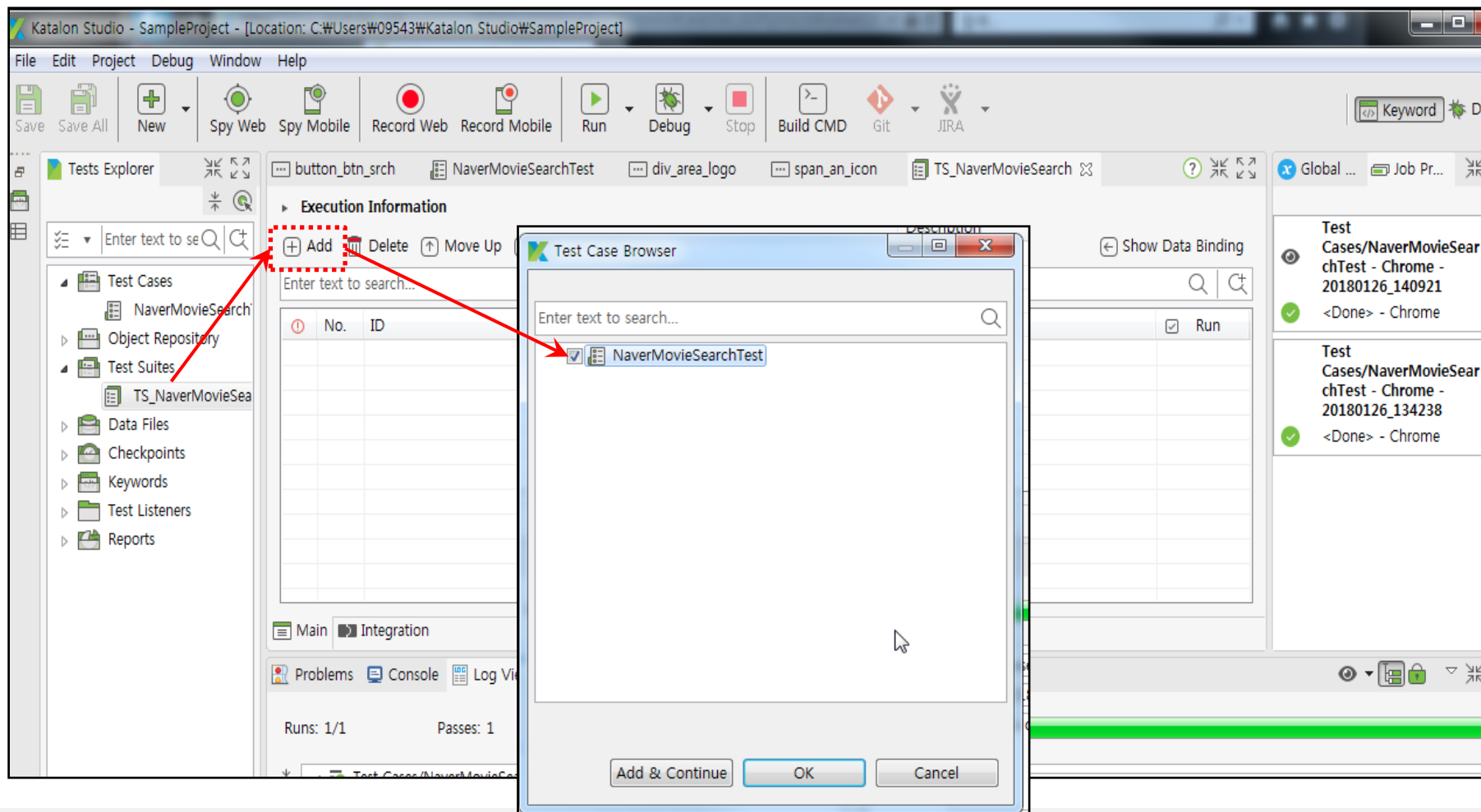
1) 좌측 Test Explorer에서 Test Suites 아이템을 선택하고 New>Test Suite을 선택하고 TestSuite이름을 입력합니다

※ 테스트 케이스 관리에 있어 구조화가 필요한 경우 하위에 'Folder'를 생성하여 관리합니다



3.2.3 Katalon Studio 작성

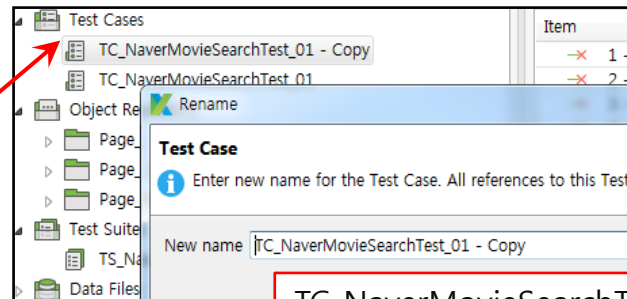
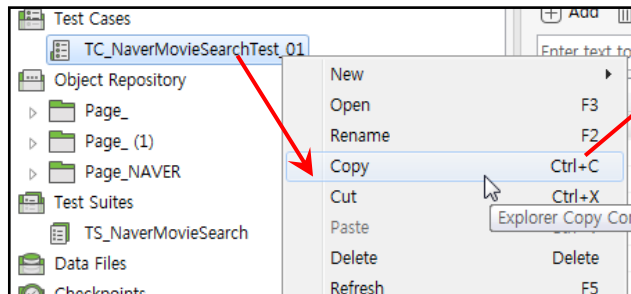
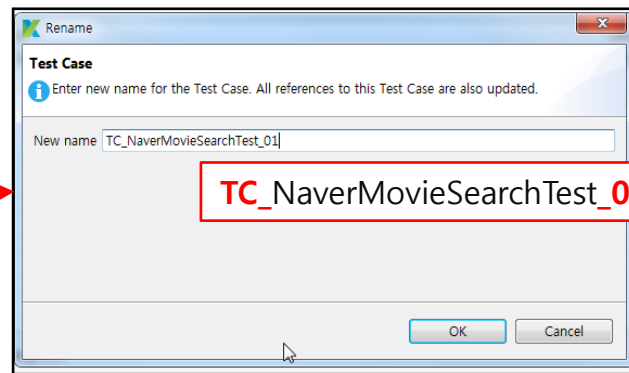
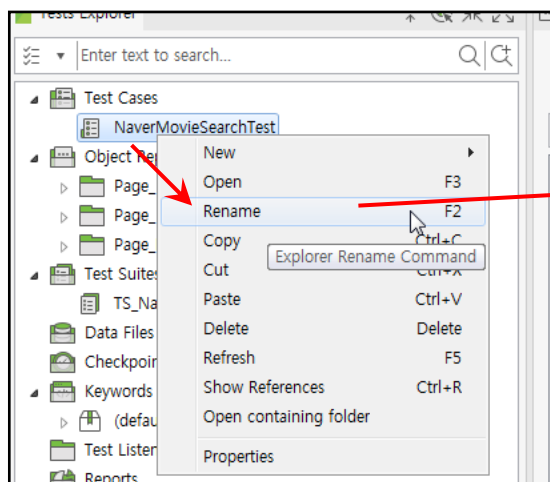
2) 생성된 TestSuite을 더블클릭하면 상세 정보가 표시됩니다. Add 아이콘을 선택하여 TestSuite에 해당하는 TestCase를 선택합니다



3.2.3 Katalon Studio 작성

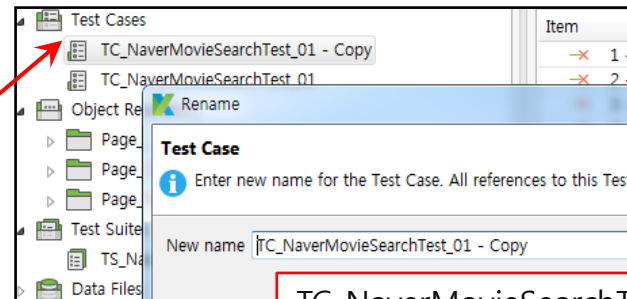
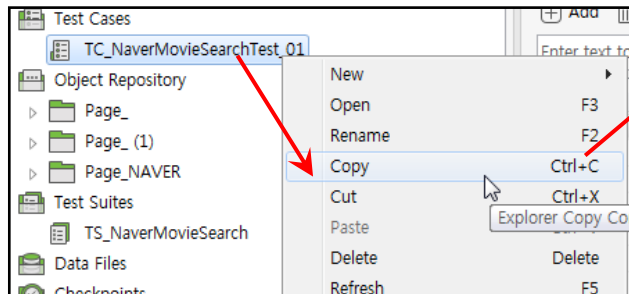
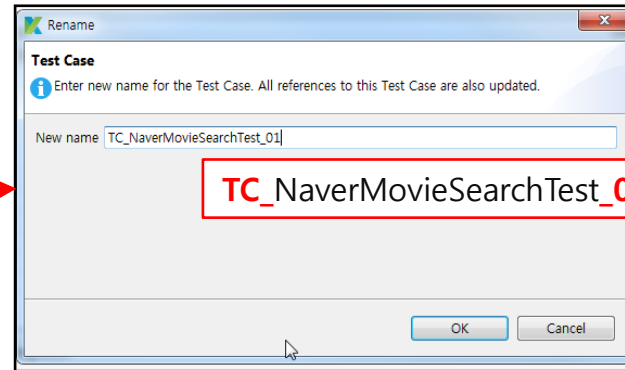
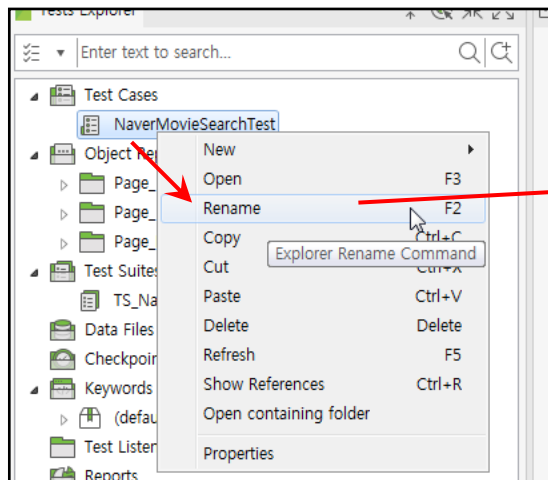
3) TestSuite 하위에 다양한 TestCase를 추가하기 위해, 기존 TestCase의 이름을 "~01"로 변경합니다

4) 기존 TestCase를 복사하여 "~02"를 추가합니다



3.2.3 Katalon Studio 작성

5) 새로 추가한 TestCase는 검색 결과가 없는 검색어 "예)NotExist#(!)\$<#!@>" 를 입력하고 "검색결과가 존재하지 않습니다"를 검증하는 테스트를 작성한다



3.2.3 Katalon Studio 작성

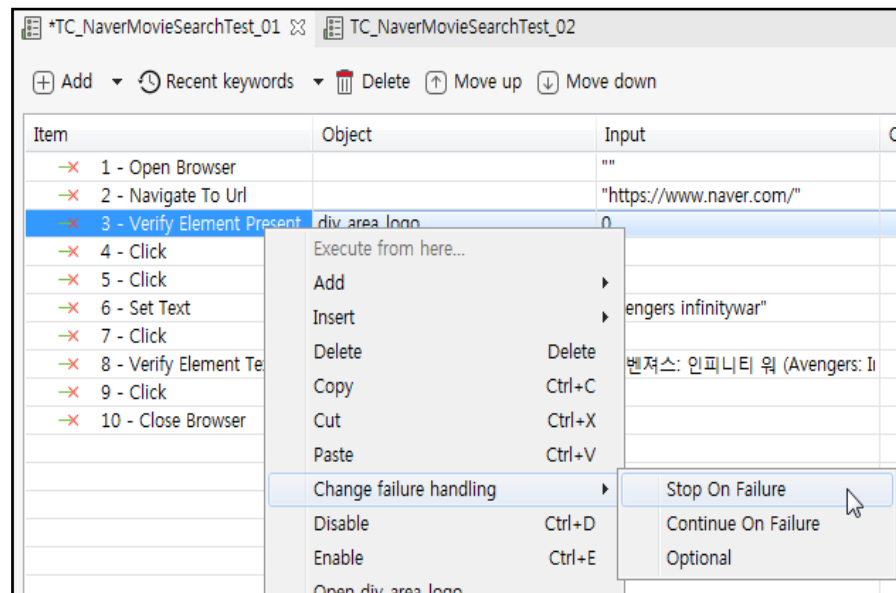
※ Tip. 테스트가 실패(verifyXXX)했을 때의 처리 고려

특정 단계에서 테스트가 실패(verifyXXX)한 경우 남은 단계를 계속 수행할지, 테스트를 바로 실패로 처리하고 멈출지를 정할 수 있다

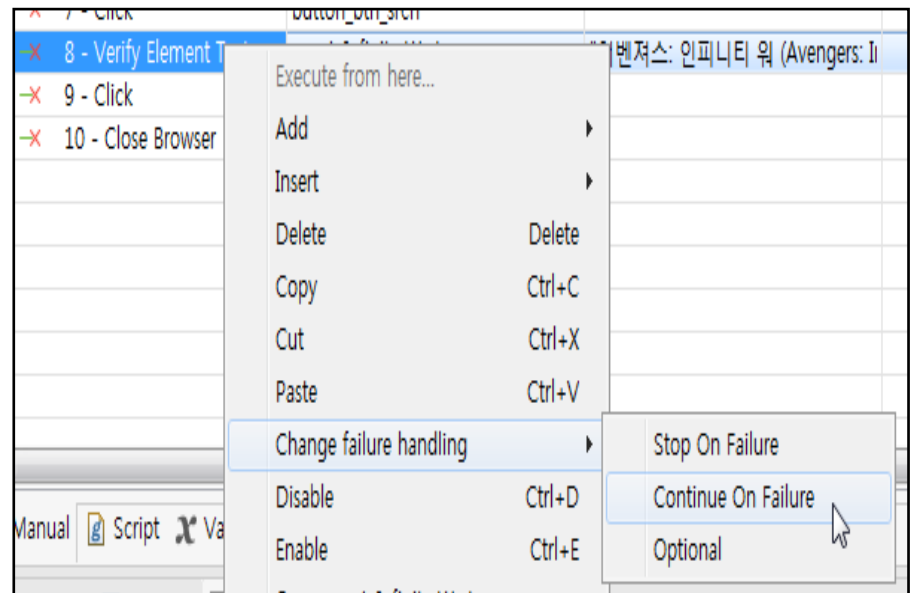
(예1) **Stop On Failure** - 네이버 메인 페이지 확인에서 테스트가 실패했을 때 - 아예 추가 테스트를 수행하는 의미가 없으므로 바로 테스트 실패

(예2) **Continue On Failure** - 네이버 영화검색 페이지의 테스트 수행 결과가 맞지 않은 경우, 다시 네이버 메인 페이지로 돌아가는 나머지 단계 수행을 위해 Continue On Failure 설정

Stop On Failure : 이후 단계를 실행하지 않고 테스트 실패



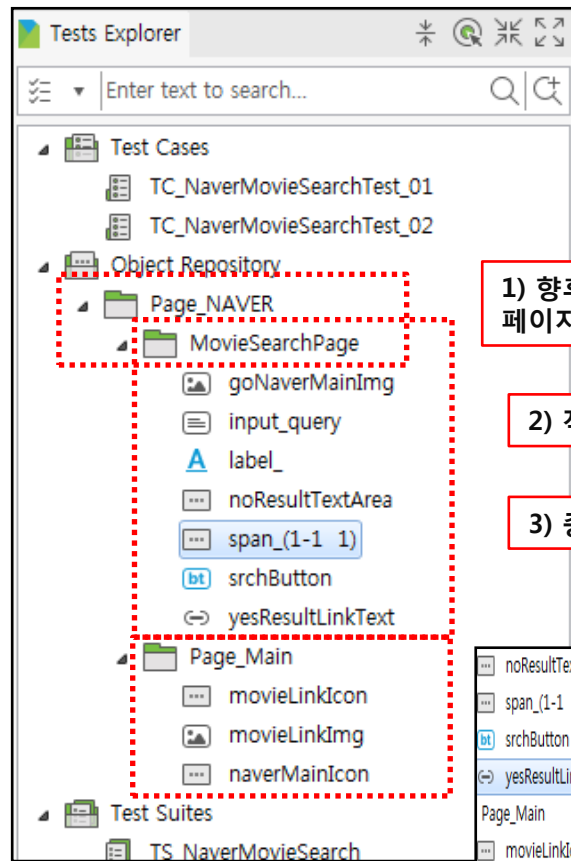
Continue On Failure : 이후 단계를 마저 수행하고 테스트 실패



3.2.3 Katalon Studio 작성

※ Tip. (예) 오브젝트들의 리팩토링

각 레코딩 과정에서 기록된 오브젝트들은 가독성 및 변경이 용이하도록 이름 및 중복된 오브젝트들을 정리한다
해당 오브젝트 상위의 페이지(폴더) 구성, 중복된 오브젝트 제거, 이름 변경, 불필요한 오브젝트 제거



1) 향후 특정 오브젝트가 변경되었을 때 직관적으로 확인/변경이 가능하도록 페이지 레벨로 폴더 구성

2) 직관적으로 이해가 되는 오브젝트 이름으로 변경

3) 중복되는 오브젝트 삭제

noResultTextArea	+	Add	Delete	Clear	
span_(1-1 1)					
srchButton					
yesResultLinkText					
Page_Main					
movieLinkIcon					
movieLinkImg					

Name	Condition	Value	
tag	equals	a	<input type="checkbox"/>
href	equals	/movie/bi/mi/basic.nhn?code=136315	<input type="checkbox"/>
text	equals	어벤져스: 인피니티 워 (: Infinity War)	<input type="checkbox"/>
xpath	equals	id("old_content")/ul[@class="search_list_1"]/li[1]/dl[1]/dt[1]/...	<input checked="" type="checkbox"/>

4) 특정 상황에 의존성이 있는 인식 방식은 체크 해제

3.2.3 Katalon Studio 작성

※ 정리된 TestCase의 예

TC_NaverMovieSearch_01 : 결과가 있는 검색어 테스트

TC_NaverMovieSearch_01 : 결과가 있는 검색어 테스트

Item	Object	Input
1 - Open Browser		""
2 - Navigate To Url		"https://www.naver.com/"
3 - Verify Element Present	naverMainIcon	30
4 - Verify Element Present	movieLinkIcon	1
5 - Click	movieLinkIcon	
6 - Set Text	input_query	"avengers infinitywar"
7 - Click	srchButton	
8 - Verify Element Text	yesResultLinkText	"어벤져스: 인피니티 워 (Avengers: Infinity War)"
9 - Click	goNaverMainImg	
10 - Close Browser		

```
import com.kms.katalon.core.webui.keyword.WebUiBuiltInKeywords as WebUI
import internal.GlobalVariable as GlobalVariable
import org.openqa.selenium.Keys as Keys

WebUI.openBrowser('')
WebUI.navigateToUrl('https://www.naver.com/')
WebUI.verifyElementPresent(findTestObject('Page_NAVER/Page_Main/naverMainIcon'), 30)
WebUI.verifyElementPresent(findTestObject('Page_NAVER/Page_Main/movieLinkIcon'), 1)
WebUI.click(findTestObject('Page_NAVER/Page_Main/movieLinkIcon'))
WebUI.setText(findTestObject('Page_NAVER/MovieSearchPage/input_query'), 'avengers infinitywar')
WebUI.click(findTestObject('Page_NAVER/MovieSearchPage/srcchButton'))
WebUI.verifyElementText(findTestObject('Page_NAVER/MovieSearchPage/yesResultLinkText'), '어벤져스: 인피니티 워 (Avengers: Infinity War)')
WebUI.click(findTestObject('Page_NAVER/MovieSearchPage/goNaverMainImg'))
WebUI.closeBrowser()
```

TC_NaverMovieSearch_02 : 결과가 없는 검색어 테스트

TC_NaverMovieSearch_02 : 결과가 없는 검색어 테스트

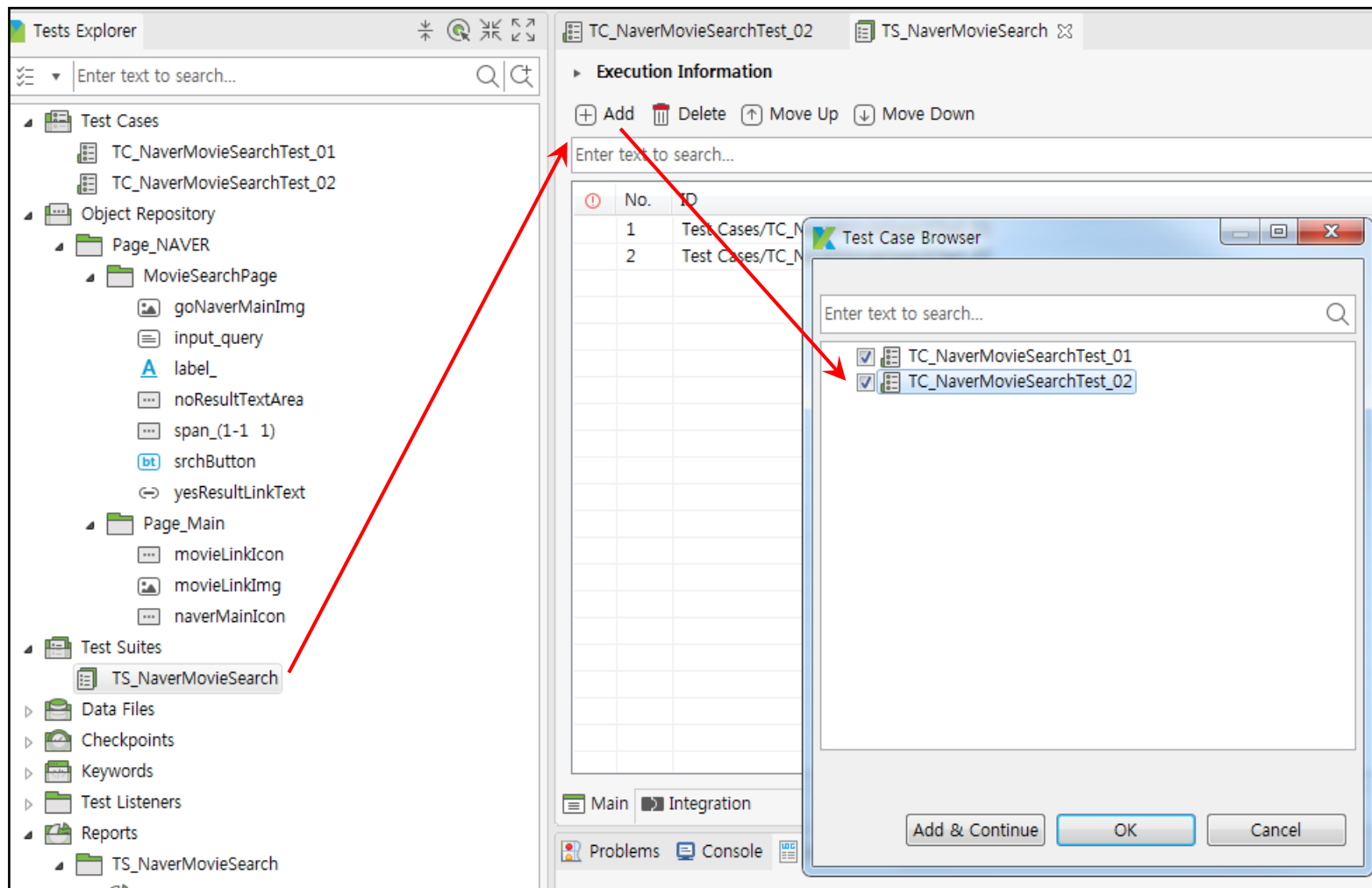
Item	Object	Input
1 - Open Browser		""
2 - Navigate To Url		"https://www.naver.com/"
3 - Verify Element Present	naverMainIcon	0
4 - Click	movieLinkIcon	
5 - Set Text	input_query	"NOTEXIST!@#@#"
6 - Click	srchButton	
7 - Verify Element Present	noResultTextArea	0
8 - Click	goNaverMainImg	
9 - Close Browser		

```
import com.kms.katalon.core.webui.keyword.WebUiBuiltInKeywords as WebUI
import internal.GlobalVariable as GlobalVariable
import org.openqa.selenium.Keys as Keys

WebUI.openBrowser('')
WebUI.navigateToUrl('https://www.naver.com/')
WebUI.verifyElementPresent(findTestObject('Page_NAVER/Page_Main/naverMainIcon'), 0)
WebUI.click(findTestObject('Page_NAVER/Page_Main/movieLinkIcon'))
WebUI.setText(findTestObject('Page_NAVER/MovieSearchPage/input_query'), 'NOTEXIST!@#@#')
WebUI.click(findTestObject('Page_NAVER/MovieSearchPage/srcchButton'))
WebUI.verifyElementPresent(findTestObject('Page_NAVER/MovieSearchPage/noResultTextArea'), 0)
WebUI.click(findTestObject('Page_NAVER/MovieSearchPage/goNaverMainImg'))
WebUI.closeBrowser()
```

3.2.3 Katalon Studio 작성

6) 새로 추가한 TestCase를 TestSuite 뷰에서 Add>체크 해서 추가한다



3.2.3 Katalon Studio 작성

(3) 테스트 실행&리포트 확인

1) Test Suite을 선택하고 Run>수행할 브라우저를 선택하여 테스트를 실행합니다

The screenshot displays the Katalon Studio interface. On the left, the 'Tests Explorer' pane shows a tree structure with 'Test Suites' expanded, highlighting 'TS_NaverMovieSearch'. A red arrow points from this suite to the 'Run' button in the top toolbar. Another red arrow points from the 'Run' button to the 'Chrome' option in the dropdown menu. Below the toolbar, the 'Execution Information' table lists two test cases:

No.	ID
1	Test Cases/TC_NaverMovieS
2	Test Cases/TC_NaverMovieS

On the right, a browser window shows the Naver homepage. A yellow banner at the top reads: 'Chrome이 자동화된 테스트 소프트웨어에 의해 제어되고 있습니다.' (Chrome is controlled by automated test software). The browser address bar shows 'https://www.naver.com'.

3.2.3 Katalon Studio 작성

- 2) TestExplorer의 Reports 하위에 생성된 테스트 리포트를 더블클릭하여 오픈한다
- 테스트 리포트는 툴 상으로도 제공되고, 별도 파일로도 저장된다

The screenshot displays the Katalon Studio interface with three main panels highlighted by red dashed boxes and labels:

- 2) 테스트케이스별 테스트 결과** (Test Cases Table): This panel shows a table of test cases. The first two rows are highlighted: TC_NaverMovieSearchTest_01 (11.661s) and TC_NaverMovieSearchTest_02 (39.444s). A red arrow points from the 'Reports' folder in the Test Explorer to this panel.
- 3) 현재 선택한 테스트케이스내 단계별 상세 결과** (Test Case's Log): This panel shows the detailed log for the selected test case, TC_NaverMovieSearchTest_02. It includes a table of test steps with their descriptions and elapsed times. The last step, 'verifyElementNotPresent', is highlighted in red, indicating a failure.
- 1) 전체 테스트 결과 요약, 수행환경 정보** (Summary): This panel shows the summary of the test execution, including the test suite ID, host name, Katalon version, OS, platform, and execution environment. It also displays the total number of test cases, passed, failed, and incomplete results.

The Test Explorer on the left shows the hierarchy of test cases, test suites, and reports. The 'Reports' folder is expanded, showing the test results for the selected test case.

Business

Playing by Kennerly's Rules

1-800-242-7522
 1-800-242-7522
 1-800-242-7522

3) 별도로 생성된 테스트 리포트 결과를 확인한다

Katalon Studio ▶ SampleProject ▶ Reports ▶ TS_NaverMovieSearch ▶ 20180129_151907

보기(V) 도구(T) 도움말(H)

열기(○) 공유 대상(□) 새 폴더(+) 새 폴더

이름	수정한 날짜	유형	크기
20180129_151907.csv	2018-01-29 오후...	Microsoft Excel ...	
20180129_151907.html	2018-01-29 오후...	HTML 문서	
1517206800781.png	2018-01-29 오후...	PNG 이미지	
1517206801154.png	2018-01-29 오후...	PNG 이미지	
execution.properties	2018-01-29 오후...	PROPERTIES 파일	
execution0.log	2018-01-29 오후...	텍스트 문서	
JSON_Report.json	2018-01-29 오후...	JSON 파일	
JUnit_Report.xml	2018-01-29 오후...	XML 문서	
Report.html	2018-01-29 오후...	HTML 문서	

테스트가 실패했을 때의 스크린 캡처

NAVIER 영화

영화순
상영작 예정작
영화평점
예매
시사회 이벤트
평점 리뷰
다운로드
인더극장

'WOTEST7%0A%0A%0A'에 대한 영화 통합검색결과입니다.

실패

'WOTEST7%0A%0A%0A'에 대한 영화 검색 결과가 없습니다.

영화정보의 수질이나 신규성질이 필요한 경우 고객센터 02-6300-0000로 문의하십시오. [고객센터 02-6300-0000로 문의하기]

영화통합검색

검색

Suite/Test	Browser	Descriptio	Start time	End time	Duration	Status
TS_Naver	Chrome	63.0.3239.13	#####	#####	52.520s	FAILED
csv형태 리포트						
Test Case	Chrome	63.0.3239.13	#####	#####	11.661s	PASSED
openBrow	Chrome	63.0.3239.13	#####	#####	4.421s	PASSED
navigateTr	Chrome	63.0.3239.13	#####	#####	1.451s	PASSED
verifyElem	Chrome	63.0.3239.13	#####	#####	0.319s	PASSED
verifyElem	Chrome	63.0.3239.13	#####	#####	0.067s	PASSED
click	Chrome	63.0.3239.13	#####	#####	1.053s	PASSED
setText	Chrome	63.0.3239.13	#####	#####	1.147s	PASSED
click	Chrome	63.0.3239.13	#####	#####	0.820s	PASSED
verifyElem	Chrome	63.0.3239.13	#####	#####	0.139s	PASSED
click	Chrome	63.0.3239.13	#####	#####	0.731s	PASSED
closeBrow	Chrome	63.0.3239.13	#####	#####	0.933s	PASSED
Test Case	Chrome	6.0.0.124W	#####	#####	39.444s	FAILED
openBrow	Chrome	63.0.3239.13	#####	#####	3.383s	PASSED
navigateTr	Chrome	63.0.3239.13	#####	#####	1.181s	PASSED
verifyElem	Chrome	63.0.3239.13	#####	#####	0.312s	PASSED
click	Chrome	63.0.3239.13	#####	#####	1.003s	PASSED
setText	Chrome	63.0.3239.13	#####	#####	1.417s	PASSED
click	Chrome	63.0.3239.13	#####	#####	0.762s	PASSED
verifyElem	Chrome	63.0.3239.13	#####	#####	31.022s	FAILED

The screenshot shows a web browser window with the address bar displaying 'TS_NaverMovieSearch Test Log'. The page content is a test report titled 'html 형태 리포트' (HTML Form Report) in a yellow box. The report details the test suite 'TS_NaverMovieSearch' and the test case 'TEST CASE: Test Cases/TC_NaverMovieSearchTest_01'. The test results for the test case are shown as 'PASSED'.

Test Suite	Test Case	Start / End / Elapsed	Status
TS_NaverMovieSearch	TEST CASE: Test Cases/TC_NaverMovieSearchTest_01	2018-01-29 15:19:09.956 / 2018-01-29 15:20:02.476 / 00:00:52.520	2 test total, 1 passed, 1 failed, 0 error, 0 incomplete
	Full Name: TS_NaverMovieSearchTest_Cases/TC_NaverMovieSearchTest_01	2018-01-29 15:19:10.350 / 2018-01-29 15:19:22.011 / 00:00:11.661	PASSED

The test steps for the test case are listed below the results:

- TEST STEP: openBrowser
- TEST STEP: navigateToUrl
- TEST STEP: verifyElementPresent
- TEST STEP: verifyElementPresent
- TEST STEP: click

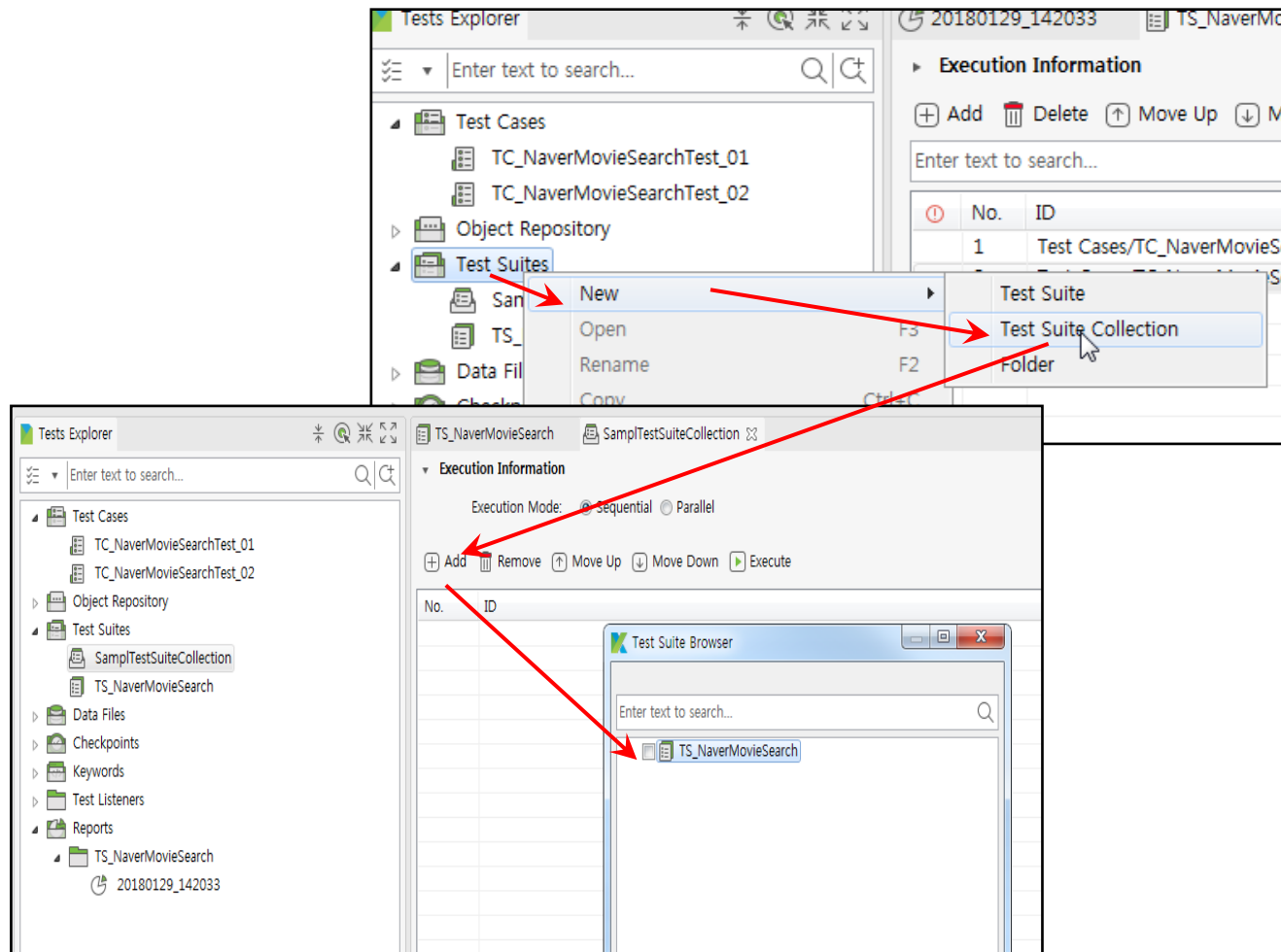
Jenkins 리포트 연동이 가능한

```
0
1  <?xml version="1.0" encoding="UTF-8"?>
2  <testsuite name="TS_NaverMovieSearch" time="52" tests="3"
3    <testsuite name="TS_NaverMovieSearch" tests="1" failure
4      <properties>
5        <property name="deviceName" value=""/>
6        <property name="devicePlatform" value=""/>
7        <property name="logFolder" value="C:\Users\09543\
8          <property name="logFiles" value="C:\Users\09543\
9        <property name="attachments" value="C:\Users\09543\
10       <property name="hostName" value="09543 - SKOC16
11       <property name="os" value="Windows 7 64bit"/>
12       <property name="katalonVersion" value="5.2.0.1"/>
13       <property name="browser" value="Chrome 63.0.3239
14       <property name="hostAddress" value="10.250.46.112
15       <property name="sessionId" value="7c79fb0be068176
16       <property name="seleniumVersion" value="3.7.1"/>
17       <property name="proxyInformation" value=ProxyIn
18       <property name="platform" value="Windows 7"/>
19     </properties>
20     <testcase name="Test Cases/TC_NaverMovieSearchTest
21     <system-out><![CDATA[2018-01-29 15:19:10 - [TEST
```

3.2.3 Katalon Studio 작성

※ 고급Tip. 여러 TestSuite을 동시에 실행시키기 위한 TestSuiteCollection

1) TestSuites > New > Test Suite Collection을 생성하면 미리 생성해 두었던 여러 TestSuite을 묶어서 실행이 가능하다

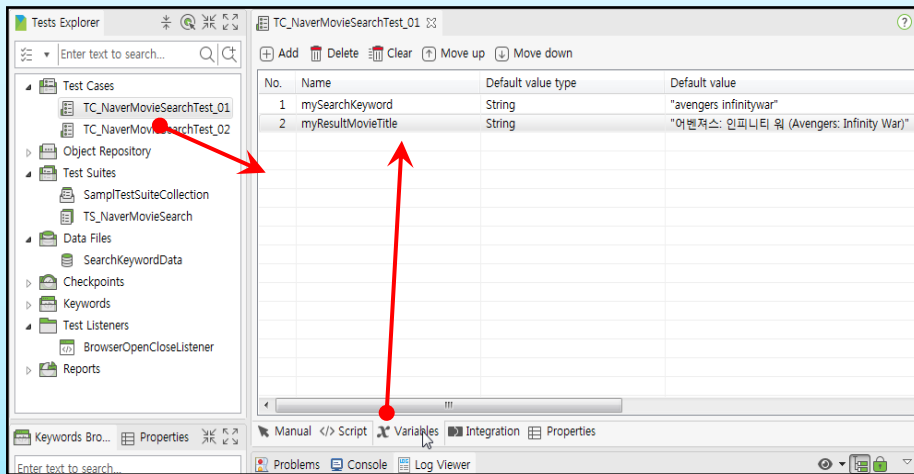


3.2.3 Katalon Studio 작성

※ 고급Tip. 테스트 케이스에서의 변수 처리 (할당/사용)

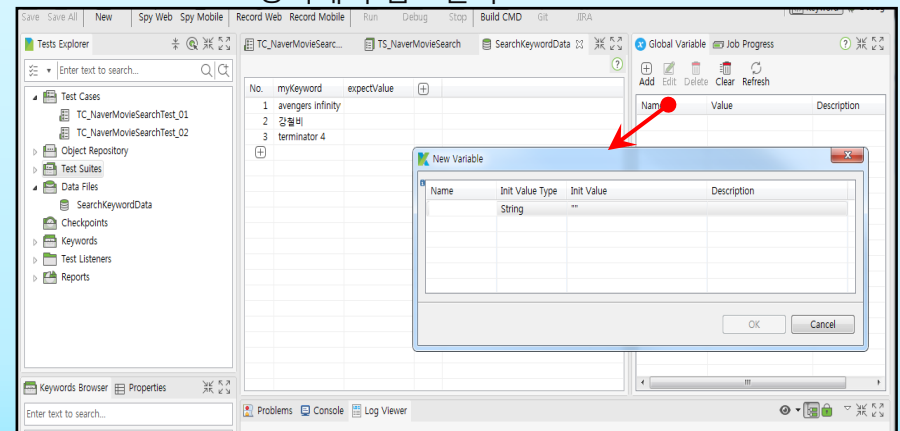
할당1) Local Variable 정의

TestCase에 종속적인 변수(Local variables)의 경우
상세보기>Variables탭에 정의한다



할당2) Global Variable 정의

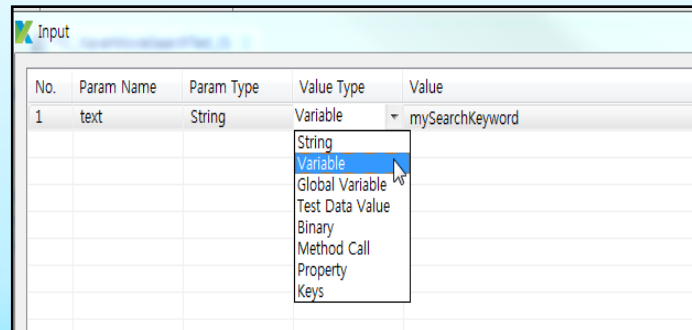
우측 Global Variable 탭 선택 후 변수를 정의한다
<-여러 TestCase에서 공통으로 사용되는 변수의 경우
GlobalVariable로 정의해서 참조한다



사용)

TestCase 상세보기에서 값을 참조하려는 스텝을
더블 클릭 후 **Value Type**을 **Variable** 또는 **Global**
Variable을 선택한다

※ 다음장 - 별도로 정의한 TestData를 사용하는
경우 Test Data Value를 선택할 수 있다

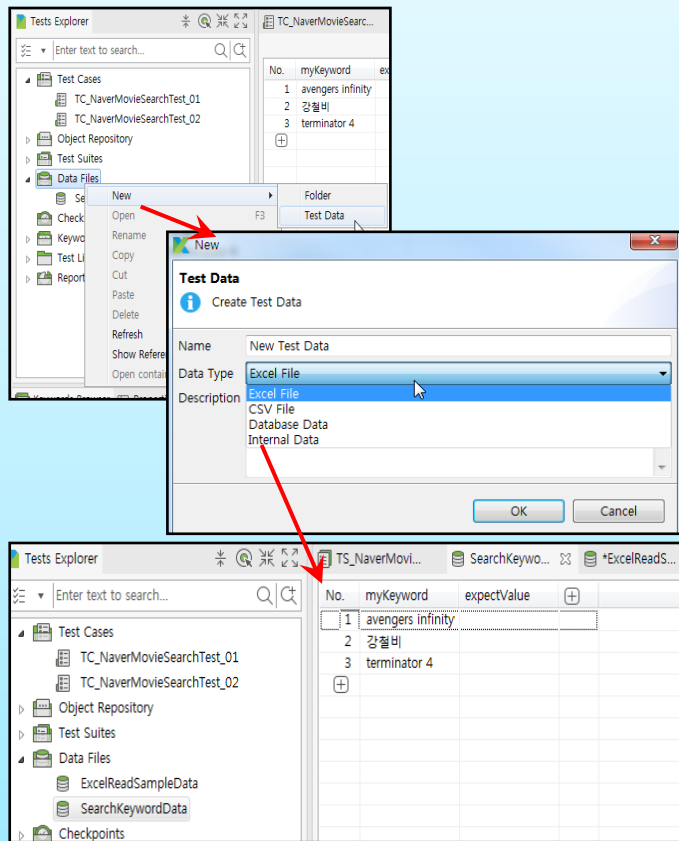


3.2.3 Katalon Studio 작성

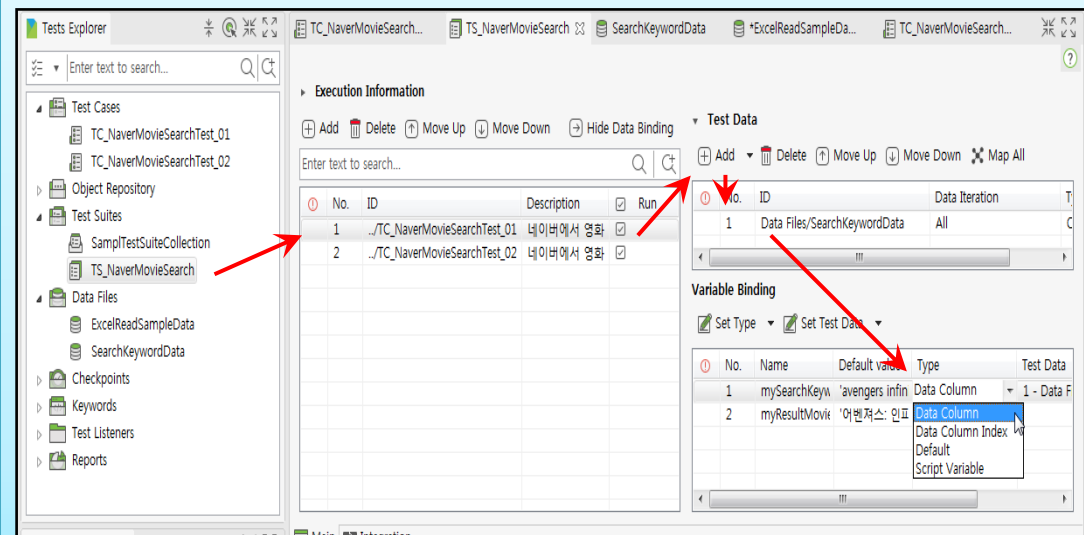
※ 고급Tip. 테스트 데이터의 동적 바인딩 (data-driven testing)

- 1) TestExplorer의 Data Files에서 데이터 정의 (엑셀, csv, 내부 테이블, 데이터베이스)
- 2) TestCase TestSuite 상세보기에서 Show Data Binding 선택 후 연결할 데이터 선택, Variable Binding을 통해 data-driven testing(하나의 테스트에서 데이터 row 만큼 데이터가 변경되면서 반복 테스트 수행)한다

[테스트 데이터 (대량)정의]



[테스트류음 뷰에서 케이스>값에 연결]





3.2.3 Katalon Studio 작성

※ 기타 – GUI 테스트 자동화에서 발생하는 일반적인 이슈와 Katalon의 해법

상세 내용은 다음 url 참조

: https://medium.com/@katalon_studio/how-to-solve-the-common-web-ui-test-automation-problems-using-the-katalon-studio-free-toolset-c36794f52554

#1) Wait-time issues and solution approach (동작과 동작 사이의 대기 시간 이슈)

(해법1) `WebUI.waitForPageLoad(GlobalVariable.G_Timeout_Small)`

(해법2) `WebUI.waitForElementPresent(findTestObject('Page_KatalonLogin/div_LoginMessage'), GlobalVariable.G_Timeout_Small)`

#2) Iframe issues and solution approach (쪼개져 있는 GUI 구분 중 하나인 Iframe 이슈)

(해법 예) `WebUI.switchToFrame(findTestObject('Page_jQuery_Drag and Drop Example/ifr_Demo Frame'), GlobalVariable.G_Timeout_Small)`

#3) Pop-up issues and solution approach (팝업창이 인식 또는 포커스 이동이 안 되는 이슈)

(해법 예) `WebUI.switchToWindowTitle('Share a link on Twitter')`

#4) Issues in identifying deeply nested elements (특정 엘리먼트가 너무 복잡한 계층의 하위에 있는 경우의 이슈)

(해법) Katalon Studio는 해당 엘리먼트의 xpath 구조 상 가장 가까우면서 유일한 엘리먼트를 기준으로 한 상대경로를 제공해 줌

3.2.4 Katalon Studio와 CI툴(Jenkins) 연동

(4) 테스트 자동 실행

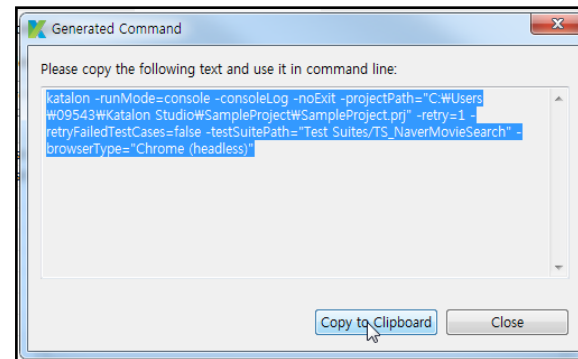
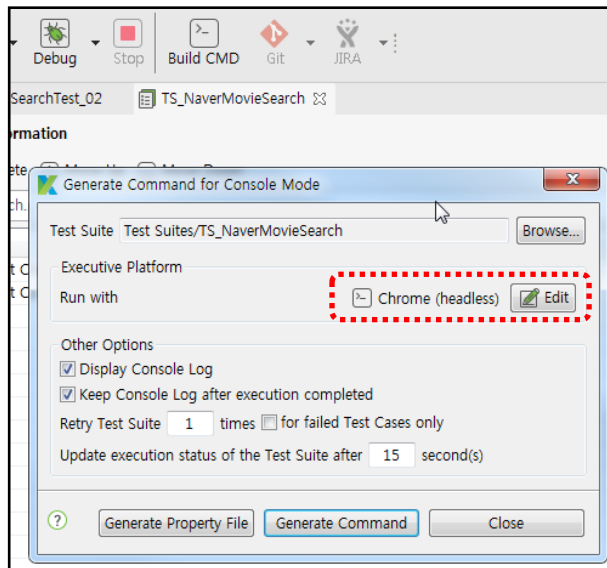
※ 기본적으로 Katalon Studio는 Jenkins 등의 CI툴에서 커맨드라인 명령어로 테스트 수행을 지원한다

(사전조건 : 해당 CI가 구축된 PC에 Katalon 설치 및 명령어 path 지정,

Katalon 프로젝트를 CI 구축된 PC에 내려받는다(참조. Katalon Studio가 git 연동))

- 1) 테스트를 수행할 대상(TestSuite, TestSuite Collection)을 선택한 상태에서 상단 Build CMD 클릭 후 실행 옵션 설정
- 2) 생성된 커맨드를 Jenkins 설정에서 실행 스크립트로 설정하고 테스트를 자동으로 수행한다

※ *Jenkins 빌드 설정, 수행 후 리포트 연결 등은 기존 유사 테스트 자동화와 동일하다*



예) `katalon -runMode=console -consoleLog -noExit -projectPath="C:\Users\W09543\Katalon Studio\SampleProject\SampleProject.prj" -retry=1 -retryFailedTestCases=false -testSuitePath="Test Suites/TS_NaverMovieSearch" -browserType="Chrome (headless)"`

3.2.4 Katalon Studio와 CI툴(Jenkins) 연동

※ 발췌. <https://docs.katalon.com/pages/viewpage.action?pageId=786527>

1) Jenkins 신규 Job 생성

2. Enter job name (e.g "Katalon Studio Tests"), and then choose "Freestyle Project"

Item name: Katalon Studio Test

☒ Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with an other than software build.

☐ Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the c

2) 실행을 shell로 선택하고 앞에서 생성한 커맨드 명령어 입력

3. Add execution step

mac OS

3.1 Add "Execute Shell" step

- Build after other projects are built
- Add a new template to all docker clouds
- Build / Publish Docker Containers
- Download from Azure Blob storage
- Execute Docker command
- Execute Windows batch command
- Execute shell**
- Invoke Ant
- Invoke Gradle script

3) 빌드 후 동작에 Junit 형태의 리포트 생성 선택

Publish JUnit reports

From Katalon Studio 4.7, JUnit report is generated result report' item.

- Aggregate downstream test results
- Archive the artifacts
- Build other projects
- Publish JUnit test result report**
- Publish Javadoc
- Record fingerprints of files to track usage
- Git Publisher

4) Jenkins상에서 테스트 리포트 확인~~

Jenkins > Katalon Studio Test

- Back to Dashboard
- Status
- Changes
- Workspace
- Build Now
- Delete Project
- Configure
- Test Results Analyzer



Post-build Actions

Publish JUnit test result report

Test report XMLs: Reports\TS_RegressionTest\JUnit_Report.xml

Fileset 'includes' setting that specifies the generated raw XML re

1% failing tests scores as 99% health. 5% failing tests scores as

Test report features: Add

Do not fail the build on empty test results

테스트 후 생성된 Junit_Report.xml 위치 입력