



# Studio V2.2 Event Item

## 사용자 설명서

본 문서는 A.WORKS Studio V2.2 Event Item 사용자를 위한 설명서입니다.



# 1. 사용하기에 앞서

이 문서는 포스코ICT의 Cloud RPA(Robotic Process Automation) 솔루션인 A.WORKS(에이웍스)의 Studio V2.2 Event Item 사용자 매뉴얼입니다. 이 매뉴얼을 사용하기에 앞서 다음 사항을 읽어주십시오.

## 1.1. 사용 대상

이 사용자 매뉴얼은 A.WORKS RPA 솔루션을 활용하여, 기존의 단순 반복업무를 자동화하여 컴퓨터가 수행할 수 있는 프로세스를 직접 만들어 실무에 적용하는 사용자를 대상으로 합니다. .

RPA에 대한 전문 지식이 없는 일반 사용자도 기존의 업무를 자동화하여 처리하는 프로세스를 쉽고 빠르게 구현하여 적용할 수 있으며, A.WORKS RPA 솔루션에서 사전에 제공되는 다양한 기능 및 단위 업무 또는 조직 내에서 제작한 업무 자동화 프로세스를 템플릿 형태로 공유하며 손쉽게 제작하고 실행할 수 있습니다.

사용자는 A.WORKS RPA 솔루션을 효과적으로 사용하기 위해 다음과 같은 사전 지식이 필요합니다.

- 업무 프로세스 자동화에 대한 기본적인 지식
- Microsoft Windows 기반 컴퓨터 관련 하드웨어 및 소프트웨어, 프로그램에 대한 지식

## 1.2. 법적 고지 사항

이 사용자 매뉴얼의 내용은 포스코ICT가 제공하는 신뢰할 수 있는 정보입니다. 그러나 부정확한 내용이나 잘못 입력된 글자, 사용자 부주의로 인해 발생한 문제 등에 대해서는 포스코ICT에서 책임지지 않습니다.

이 사용자 매뉴얼의 내용과 제품의 사양은 사전 예고 없이 변경될 수 있습니다. 개정에 관한 상세한 정보는 포스코ICT의 A.WORKS RPA 인터넷 홈페이지(<http://aworksdp.com>)에서 확인하실 수 있습니다.

제품명	A.WORKS Cloud RPA
버전	V2.2
구성요소	Studio V2.2 Event Item
출시일	2021년 06월 30일
공급업체	포스코 ICT
공급업체 주소	경기도 성남시 분당구 판교로 255 판교 이노밸리
대표 전화	031-723-2222
전자 메일	<a href="mailto:aworksdp@aworksdp.com">aworksdp@aworksdp.com</a>
홈페이지	<a href="http://aworksdp.com">http://aworksdp.com</a>

※ A.WORKS에 적용된 표준 규격 및 법령이 존재하지 않습니다.

이 사용자 매뉴얼에 대한 저작권을 포함한 지식재산권은 포스코ICT에 있습니다. 포스코ICT의 사전 허가 없이 내용의 일부 또는 전부를 무단 사용하거나 복제하는 것은 금지되어 있으며, 이는 포스코ICT의 지식재산권 침해하는 것입니다.

또한, 이 사용자 매뉴얼에서 제공되는 모든 콘텐츠 즉, 이미지, DB정보 등은 저작권법에 의하여 보호받는 저작물로 별도 저작권 표시 또는 출처를 명시한 경우를 제외하고는 원칙적으로 포스코ICT에 저작권이 있습니다. A.WORKS의 자료 및 UI(User Interface)를 포스코ICT의 허가 없이 복제, 배포하는 경우에는 저작권법 제136조 제1항 제1호 위반에 해당 되며 관련 법규에 의해 처벌 받을 수 있습니다.

Microsoft사의 로고, Microsoft(MS) 및 Windows는 미국Microsoft Corporation의 등록상표입니다.

이 사용자 매뉴얼에 기재되어 있는 각 사의 사명, 제품명 및 서비스 명은 각 사의 상표 또는 등록 상표입니다.

## 1.3. 사용자 매뉴얼 구성

이 사용자 매뉴얼의 구성은 다음과 같습니다.

### 1. 사용하기에 앞서

A.WORKS RPA 솔루션을 사용하기에 앞서 사용대상 및 법적 고지사항 등을 알려드립니다.

### 2. RPA의 개요

A.WORKS RPA 솔루션의 전체 구성 요소와 주요 특징 등을 소개합니다.

### 3. Event Item 가져오기

A.WORKS Studio에 Event Item들을 가져와서 스크립트 작성하는 방법을 간단히 설명합니다.

### 4. Flow Control

A.WORKS Studio에서 프로세스 및 태스크 스크립트 작성시 사용하는 Flow Control에 대하여 설명합니다.

### 5. Event Item

A.WORKS Studio에서 태스크 스크립트 작성시 사용하는 Event Item에 대하여 설명합니다.

### 부록 A. 용어설명

## 1.4. 서체 표기

이 사용자 매뉴얼 내용의 이해를 돋기 위해 다음과 같은 서체를 사용합니다.

굵은 활자체	굵은 활자체는 화면사용과 관련한 요소와 메뉴와 파일 보존할 위치 등을 표기할 때 사용합니다. 다만, 화면사용 요소가운데 포털, 창, 팝업 창, 마법사, 페이지는 굵은 활자체가 아닌 큰따옴표("")로 표기합니다.
이탤릭체	이탤릭체는 사용자가 입력해야 하는 변수나 파라미터 값을 표기할 때 사용합니다.
고정폭 활자	고정폭 활자는 프로그래밍과 관련된 용어나 코드를 표기할 때 사용합니다. 본문에서 언급할 때는 구분을 위해 명령어, 파라미터명, 태스크명, 프로세스명 등은 굵은 고정폭 활자로 표기하고, 변수와 파라미터값은 고정폭 활자 이탤릭체로 표기합니다.

## 1.5. 사용 기호 및 아이콘 설명

A.WORKS RPA 솔루션을 사용할 때 필요한 컴퓨터 하드웨어 및 소프트웨어, 프로그래밍과 관련한 지식은 Info 아이콘 으로 설명합니다.

VDI(Virtual Desktop Infrastructure, 데스크톱 가상화)

 물리적으로 존재하지 않지만 실제 작동하는 컴퓨터 안에서 작동하는 또 하나의 컴퓨터를 만들 수 있는 기술)

- Studio에서 프로세스나 작업항목(Task)을 만들 때 알아두면 편리한 팁이나 예외사항, 제한사항 등 추가적인 정보를 제공할 때 TIP 아이콘 을 사용합니다.



프로세스를 생성하면 자동으로 빈 프로세스가 생성됩니다.

- 시스템, 데이터, 프로세스, Item 등을 사용할 경우 스크립트 구성에 오류나 영향이 있을 경우, 주의 아이콘 을 사용합니다.



Code Expression 화면의 입력 값은 실행 시 C#코드로 변환되어 실행되므로 C#코드의 문법에 맞게 입력해야 합니다.

## 1.6. 개정이력

이 사용자 매뉴얼의 개정이력입니다. 현재 사용하고 계시는 A.WORKS Studio 버전을 확인 후 사용하시기 바랍니다.

Studio 버전	매뉴얼 버전	변경 일자	변경 사항
Studio V2.0	1.0.0	2020. 9. 1.	최초 작성
<b>Studio V2.1</b>	1.0.0	2020.12.30	
<b>Studio V2.1.3</b>	1.0.0	2020.04.21	
<b>Studio V2.2</b>	1.0.0	2021.05.11	

## 2. RPA 의 개요

A.WORKS RPA 솔루션은 사람이 수행하던 기존의 단순 반복업무를 자동화할 수 있도록 일련의 프로세스로 만들어, 소프트웨어 로봇(이하 A.WORKS Bot)에게 대신하여 수행하도록 하고, A.WORKS Bot이 해당 업무를 올바르게 수행하는지 확인하고, 추가적인 업무를 수행시키거나 다른 A.WORKS Bot을 실행시켜 자동화 대상 업무가 정상적으로 이루어질 수 있도록 하는 여러 가지 소프트웨어들로 구성됩니다.

"RPA의 개요"에서 다루는 내용은 다음과 같습니다.

- RPA 솔루션 소개
- RPA 솔루션 구성요소
  - Studio
  - WorkCenter
  - Mini
- RPA 솔루션 전체구성
- 주요특징
- 활용사례

### 2.1. RPA 솔루션 소개

A.WORKS RPA 솔루션을 활용하면, 기존의 단순 반복업무를 자동화하여 A.WORKS Bot에게 수행시키는 동안, 사람은 보다 창의적인 업무에 집중하며 고부가가치 업무를 수행할 수 있습니다.

A.WORKS Bot은 사람보다 더 정확하게 그리고 24시간 365일 쉬지 않고 업무를 자동으로 수행하여 업무 생산성 향상과 비용을 절감할 수 있습니다.

업무 프로세스의 복잡성이나 규모에 관계없이 모든 조직의 비즈니스 프로세스를 단순화하고 개선하여 자동화하면 더 많은 단순 반복업무를 A.WORKS Bot에게 수행시킬 수 있습니다.

A.WORKS RPA 솔루션은 포스코ICT가 다년간 정보통신기술분야에서 많은 업무시스템을 개발하고 구축, 운영하며 축적된 경험과 기술을 집약하여 만든 제품으로, 국내에서 반드시 필요하고 실제로 RPA와 소프트웨어 로봇의 운영, 확산에 도움이 되는 핵심기능 위주의 기업형 RPA 솔루션을 지향합니다.

## 2.2. RPA 솔루션 구성

A.WORKS RPA 솔루션은, 기존 단순 반복업무 가운데 로봇을 이용하여 자동화할 업무를 컴퓨터가 이해할 수 있는 스크립트(Script, 프로그램에 의해 번역되고 수행되는 명령어 집합)로 만들어 A.WORKS Bot를 만드는 **A.WORKS Studio**, A.WORKS Bot들이 잘 수행할 수 있도록 서버에서 일정을 관리하고 수행하는 것을 모니터링하고 수행결과를 확인하는 **A.WORKS WorkCenter**, 사람을 대신하여 사용자의 로컬PC에서 A.WORKS Bot을 실제 실행시키고 모니터링하며 관리하여 주는 **A.WORKS (WorkCenter) Mini**로 구성되어 있습니다.

각 제품의 주요 특징은 다음과 같습니다.

### ■ Studio

사용자가 A.WORKS Bot에게 수행시킬 업무를 컴퓨터가 이해할 수 있는 프로세스로 만드는 업무 자동화 소프트웨어입니다. 기존 단순 반복업무를 프로세스로 나누고, 다시 기능 위주의 단위 작업항목으로 세분화하고, 전문 프로그래머들이 사전에 만들어 놓은 다양한 기능함수들을 이용하여 실제 컴퓨터가 수행할 수 있는 스크립트를 만들고, 정상 작동하는지 검증(디버깅, 스크립트의 오류를 없애는 일)하여 A.WORKS Bot을 만듭니다.

#### ● Easy 사용하기 쉬움

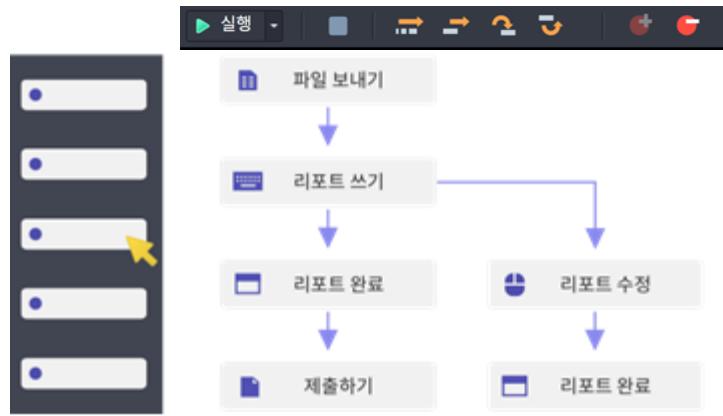
전문지식이 없는 일반 사용자도 Studio에서 제공되는 업무 템플릿(프로세스나 작업항목의 유형들)을 사용하여 해당 업무 프로세스의 수행에 필요한 단위 작업항목인 태스크나 Flow Control, Event Item 등을 간단하게 Drag & Drop(마우스로 클릭하여 끌어다 놓기)방식으로 연결하여 쉽고 빠르게 프로세스 자동화 스크립트를 만들 수 있습니다.

#### ● Productive 높은 생산성

자주 사용하는 기능을 쉽고 빠르게 활용할 수 있는 사용자 인터페이스를 제공하고 있어, 자동화 스크립트를 만드는데 필요한 태스크나 Event Item을 쉽게 검색하여 사용가능하고 프로세스를 단계별로 실행하여 오류를 검증하는 디버깅 기능과 전체 수행을 하기 전에 스크립트의 문법적인 오류를 미리 예방할 수 있는 검증기능(Validation Check)을 제공하여 스크립트 제작 생산성이 향상됩니다.

#### ● Recycle 재활용

전사가 공통적으로 활용 가능한 프로세스나 태스크는 WorkCenter를 통하여 등록하면 공유 태스크 형태나 템플릿의 형태로 공유할 수 있습니다.  
공유된 태스크의 연결만으로도 프로세스의 구현이 가능하며, 공유된 템플릿의 프로세스나 태스크를 재활용하면 스크립트 제작 시간과 비용을 절약할 수 있습니다.



< Studio 스크립트 작성 화면 >

## ■ WorkCenter

WorkCenter는 A.WORKS RPA 솔루션을 이용하는데 필요한 자동화된 프로세스, 라이선스, WorkCenter에서 스케줄에 따라 수행시키는 A.WORKS Bot(이하 S.Bot) 등 관련된 모든 자원을 종합적으로 제어하고 관리하는 자원관리시스템(Resource Management System)입니다.

Studio에서 만들어진 프로세스를 등록하여 놓고, 정해진 일정에 따라 Bot들을 구동시킬 수 있으며, 대규모 Bot의 운영에 최적화되어 있어 Bot들의 업무 진행사항 및 결과에 대한 모니터링과 분석 등을 수행합니다. 로봇의 수행 상태나 결과를 사용자 친화적인 다양한 형태의 이미지나 그래프 등으로 표현한 Dashboard (종합상황판) 형태로 실시간으로 제공합니다..

- **운영 최적화**

실행 우선 순위와 로봇 작동 상태를 확인하여 로봇을 할당하고 실행함으로써, 전체적인 작업 부하에 따라 로봇의 수행일정을 관리하고, 여유 있는 로봇에 대한 정보를 가시적이고 직관적으로 제공합니다.

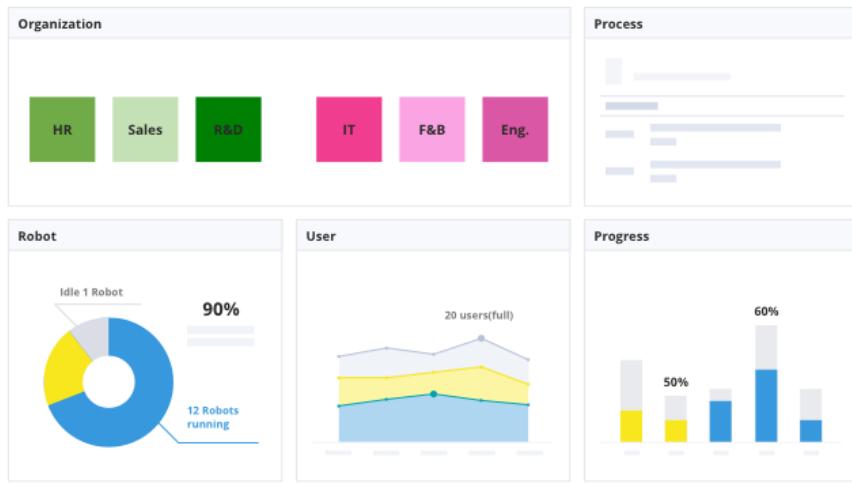
또한, RPA 운영관리의 효율을 높이는 데이터와 부가기능 등을 제공합니다.

- **실시간 Dashboard**

정보의 중요도에 따라 구성된 화면과 우수한 정보 요약 및 사용자 친화적인 다양한 형태의 이미지나 그래프 등으로 시각화하여 제공하여, 로봇의 수행 관리와 확인에 필요한 데이터를 빠르고 직관적으로 파악하고 대응할 수 있습니다.

- **강력한 분석기능**

누적된 운영 데이터는 시스템 효율 개선에 활용할 수 있습니다. 업무 수행 소요시간 변화, RPA 처리 업무량 등 ROI(Return On Investment, 투자대비 수익률)를 체감할 수 있는 정보와 로봇 평균 이용시간과 처리된 로봇 정보, 오류 정보 등 다양한 분석 데이터를 제공합니다.



< WorkCenter Dashboard 화면 >

## ■ Mini(Bot)

A.WORKS Mini(Bot)는 Studio에서 작성되고 검증된 프로세스를 사용자 로컬PC에서 Bot으로 수행하는 역할을 하며, Unattended Robot(자동수행 로봇)과 Attended Robot(반자동수행 로봇, 수행에 사용자가 개입되는 형태)이 있습니다.

WorkCenter에 등록되어 중앙 관리되고 스케줄에 따라 실행되는 로봇은 S.Bot이라고 하며, A.WORKS Mini를 통해 로컬PC에서 사용자가 직접 실행시키고 모니터링 되는 로봇은 A.Bot이라고 합니다.

- **작업 정확성**

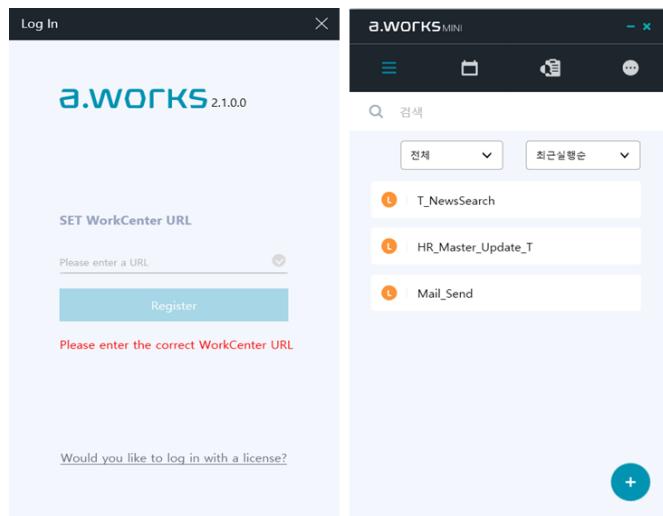
업무에 필요한 Object, 텍스트, 이미지 등을 정확하게 인식하고 정의된 프로세스와 개별 단위 업무를 충실히 정확하게 진행하여 전체 프로세스를 완료합니다.

- **높은 판단력**

자동화된 프로세스를 수행하는데 있어 사용자의 하드웨어 환경 또는 개인 환경 설정을 로봇이 스스로 판단하고 작업을 수행합니다.

- **친숙한 조작성**

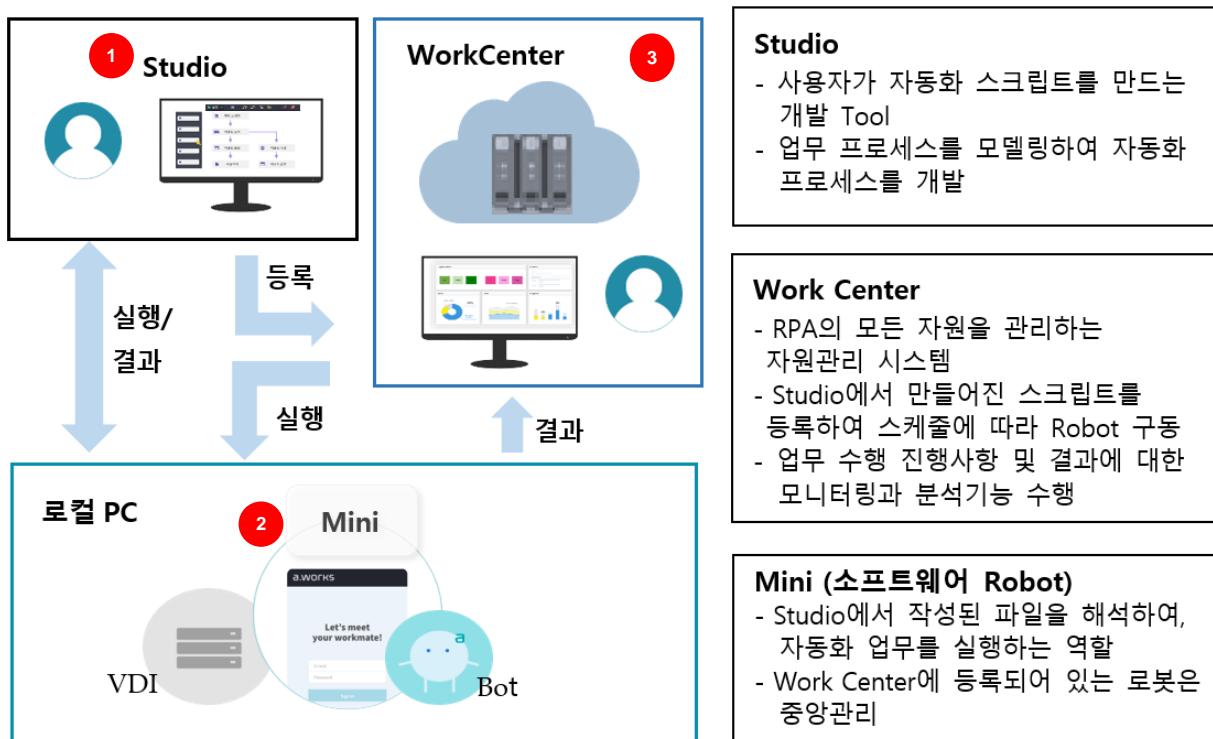
실행되고 있는 로봇의 상태를 직관적으로 모니터링 할 수 있고, 로봇을 실행 및 정지시키는 것이 쉽고 빠릅니다.



< A.WORKS Mini/A.WORKS Bot 화면 >

## 2.3. RPA 솔루션 전체 구성

A.WORKS RPA는 사람이 수행하는 업무 프로세스를 ① Studio를 이용하여 컴퓨터가 이해할 수 있는 스크립트로 만들고, 소프트웨어 로봇인 ② Mini(Bot)가 로컬PC에서 사람을 대신하여 해당 스크립트를 수행합니다. 다수의 Mini(Bot)를 운영하는 경우에는 ③ WorkCenter에 등록하고, 정해진 스케줄에 따라 수행시키고, 모니터링하고 결과를 관리할 수 있습니다.



<A.WORKS RPA 전체 구성 및 개념도 >

## 2.4. 주요 특징

A.WORKS RPA 솔루션은 업무 프로세스를 자동화하고 로봇에 수행시키는데 필요한 사용자의 편의를 최대화하기 위해 다음과 같은 기능을 지원합니다.

- ◆ **확장 가능한 사용자 인터페이스**

항시 화면에 표시되어야 하는 기능을 모아 간결한 형태로 제공하고, 업무처리 형태나 단위 별로 필요에 따라서 제공되어야 할 기능을 확장하여 이용할 수 있도록 Dropdown(메뉴나 리스트 항목의 화살표 등을 클릭 시 리스트로 표시되는 방식)이나 Drawer(서랍, 탭을 클릭 시 좌우로 선택 대상을 펼치거나 접는 처리방식)형태로 제공합니다.

- ◆ **넓은 작업영역**

Studio를 주요 기능위주로 간결하게 구성해 넓은 스크립트 작업영역을 확보했습니다.  
심플한 디자인으로 기존 및 타사의 RPA 프로그램보다 쉽고 편안함을 제공합니다.

- ◆ **Drag & Drop + α**

일반 사용자들도 해당 Object를 쉽게 마우스로 끌어 다 놓는 Drag & Drop 방식의 직관적 사용자 인터페이스를 제공하며, 숙련도가 높은 전문 개발자의 생산성을 확보하기 위하여 필요 시 직접 프로그램 코드도 입력하여 기능을 수행하는 추가기능도 제공합니다.

- ◆ **단순한 흐름제어와 블록단위 표현**

기업의 업무 프로세스에 빈번히 활용되는 순차처리, 분기처리, 반복 등의 흐름제어를 반영한 Flow Chart 방식의 Flow Control과 블록단위의 다이어그램 구조를 제공하여 프로세스를 구조화하며 전체 흐름의 파악이 용이하도록 설계하였습니다.

- ◆ **핵심 Property 탐색**

프로세스를 구성하는 태스크 들에 대하여 Event Item 적용 시 필요한 기본적인 속성값 들 중에 자주 확인이 필요한 Property항목을 개별적으로 명기하도록 하여 프로세스 전체를 확인할 때 바로 속성을 파악하고 검색을 쉽고 빠르게 하였습니다.

- ◆ **Mini Map**

넓은 시야에서의 Canvas 내의 프로세스 위치 파악과 원하는 위치로의 빠른 이동을 위하여 Mini Map(Canvas의 스크립트 축약화면) 기능을 제공합니다.

- ◆ **통합 검색**

필요 시 스크립트 작업영역(Canvas)을 확장하여 프로세스의 확인 및 접근이 필요한 속성값(Property), 변수(Variable), 중단점(Breakpoint) 등 주요 요소들을 Property 영역에 모아 놓아서 빠르게 확인하고 접근할 수 있도록 배치했습니다.

- ◆ **프로세스 검증기능**

사용자가 스크립트를 작성하고 실행할 때 프로세스와 태스크내의 각 Item의 사용 시 발생할 수 있는 오류사항 들을 자동으로 해석하고 분석하는 검증기능 (Validation Check)을 제공합니다.

- ◆ **스크린 이미지처리 고도화**

사용자 화면 위주로 업무처리가 구성되는 A.WORKS Bot의 특성을 고려하여, 화면처리 흐름 및 스크린 인식 기능을 고도화하고 해당 기능 위주로 모아서 탐색할 수 있는 기능을 제공하여 스크립트를 만드는 효율을 높였습니다.

- ◆ **모듈 및 블록단위 그룹화 기능**

타인이 작성한 프로세스를 열었을 때, 전체 구조를 빠르게 파악하고 원하는 수정 위치와 방법을 파악할 수 있도록 블록과 그룹 표현 방식을 활용한 그룹화 기능을 제공합니다.

- ◆ **디버깅 지원**

디버깅 시 단계별 실행, 부분 실행, 전체 실행 등 여러 단계로 오류정보 확인이 가능하며, 동시에 실시간으로 변수 값을 변경하거나 내용을 확인할 수 있습니다.

또한, 잘못된 변수 값의 발생 지점을 즉시 추적할 수 있고, 중단점을 미리 지정하여 그곳까지 수행하거나 필요 시 해제하며 실행을 건너 뛸 수 있습니다.

- ◆ **사용자 정의 코드 지원**

숙련도가 높은 전문 개발자는 프로세스 구성 시 필요에 따라 특정 위치에 직접 프로그램 코드도 입력하여 A.WORKS Bot가 특정기능을 수행할 수 있는 사용자 정의코드를 작성할 수 있습니다.

사용자 정의 코드는 C# 프로그래밍 코드를 자동화 스크립트에 직접 입력하고 활용할 수 있어 보다 복잡하고 전문적인 업무도 자동화 프로세스로 구현할 수 있습니다.

## 2.5. 활용사례

RPA는 초기 금융권 계정처리 등 기간 시스템의 업무에서 시작되어, 건설, 제조, 유통 및 서비스 등 전 업종으로 확산되고 있으며, 금융, 인사, 노무, 회계, 구매, 생산, 공급관리 등 전사 업무에 폭넓게 적용되고 있습니다.

다음은 A.WORKS RPA 솔루션을 활용하여 구축된 사례 중 일부를 소개합니다.

- **재무/회계 분야 - 가수금 정산 사례**

- 고객계좌정보 > 엑셀파일 확인 > Firm Banking > 입금내역 조회 > ERP 채권정보조회 > ERP 정산전표 생성
- 채권반제, 지로지불 자료 생성, 부가세 대사 검증 등 재무, 자금, 세무 부서의 각종 업무 처리 및 대사, 검증에 활용합니다.

- **인사/교육 분야 - 교육수료 처리 사례**

- 교육이수 엑셀파일 확인 > ERP 교육등록여부 확인 > 교육수료 업로드 파일 생성 > ERP 교육실적 업로드
- 인사발령 등록, 교육비 정산, 출장비 점검, 장학금 이중지원 모니터링 등에 활용합니다.

- **구매 분야 - 시황정보 수집 사례**

- 정보제공 웹사이트 접속 > 품목별 시황정보 수집 > 엑셀문서 정리 > 구매담당에 송부
- 원부자재 시황 분석, 구매품목 단가 분석, PO 발행, 납기변경 등에 활용합니다.

- **생산관리 분야 - 납품/출고지시 사례**

- 업무시스템에서 납품정보 추출 > 제품별 납품정보 파일 작성 > 도급사별 홈페이지 접속 > 납품정보 파일 등록 > ERP전표생성 및 출고지시
- 생산실적 분석, 납품/출하 지시, 생산/운송/출하/관제 일보 작성 등에 활용합니다.

- **유통 분야 - 유통점 매출집계 사례**

- 유통점 사이트 접속 > 일일 매출자료 다운로드 > 업로드용 엑셀파일 작성 > ERP업로드
- 판매실적 집계/등록, 재고관리, 수출입 선적처리, 일일 업무보고 작성 등에 활용합니다.

- **금융 분야 - 금융거래 정보제공 사례**

- 금융거래 정보제공 요청서 접수 > 금융거래 조회 및 다운로드 > 승인원자 승인 > 발송문 작성 > 메일 송부
- 금융거래 정보제공, 기업신용등급 조회, 자금이체 보고서 등 은행, 증권사, 카드사, 보험사 Back Office 및 창구업무에도 활용합니다.

### 3. Event Items 가져오기

Event Items은 업무 자동화를 위하여 컴퓨터가 이해할 수 있도록 전문 프로그래머가 제작해 놓은 기능함수들로 라이브러리 형태로 제공합니다.

Event Items에는 프로세스나 태스크 스크립트의 흐름을 제어하는 Flow Control들과 자동화를 위한 단위 작업을 수행하는 Event Item들이 있습니다.

Event items을 가져오기에 앞서, Canvas는 Studio로 업무자동화를 위한 프로세스와 태스크 스크립트를 만들고 필요한 Item들을 가져와서 서로 연결하는 작업공간입니다. Resource 영역에서 원하는 Item들을 Canvas로 마우스로 손쉽게 Drag & Drop하여 사용할 수 있습니다.

하지만, 업무자동화를 위한 가장 기본이 되는 스크립트를 만드는 만큼, Canvas에서 만드는 스크립트에 대한 기본 구성과 상태 그리고 제약사항을 우선 이해하는 것이 중요합니다.

#### @@ Canvas 사용상 제약사항 @@

1. Canvas에는 반드시 하나의 프로세스가 존재하여야 하며, 상위 프로세스는 전체 업무 흐름을 기술하고, 하위 태스크는 단위 업무를 수행하도록 구성해야 합니다.  
따라서, Root에 해당하는 프로세스는 항상 Canvas 상에 맨 왼쪽 첫 번째 탭에 노출되어야 합니다.  
Task 탭은 사용자가 생성하는데 따라서 해당 Task 명으로 오른쪽에 추가되고, 마우스로 탭을 자유롭게 이동할 수 있으나, Process 탭의 왼쪽에 들 수 없습니다.
2. 화면 상에 Process 탭과 Task 탭은 탭의 색과 크기로 구분되어 있으니 반드시 해당 탭 내에서 해당 탭에 필요한 Item을 사용하여 스크립트를 작성해야 합니다.  
Process 명은 Title Bar에 표시되며, Task 명은 Task 탭에 표시합니다.
3. Process 탭과 Task 탭은 모두 Start Item로 시작하여야 하며, 기본적으로 Canvas에 제공되며, 편집을 위하여 모두 선택 시는 제외되고, 삭제, 이동할 수 없습니다.
4. Process 탭에서는 일부 Event Item(Write Log, Assign, Code Expression, Message YesNo, Start Work)만을 사용할 수 있고, 이들은 프로세스와 태스크간 이동도 가능합니다.  
이들은 제외한 다른 모든 Event Item은 사용할 수 없습니다.
5. Task 탭에서는 이미 태스크로 생성된 태스크(Repository 내 생성된 태스크, Shared Task 포함)는 불러올 수 없습니다. 대신 "CallTask"를 이용한 일부 Event Item(Write Log, Assign, Code Expression, Message YesNo, Start Work)은 프로세스와 태스크간 이동이 가능합니다.
6. Flow Control은 프로세스 Canvas, 태스크 Canvas 양쪽 모두 불러와 사용할 수 있습니다.

Studio 사용과 관련한 상세한 내용은 "A.WORKS Studio V2.1.3 사용자 설명서"를 참조하십시오.

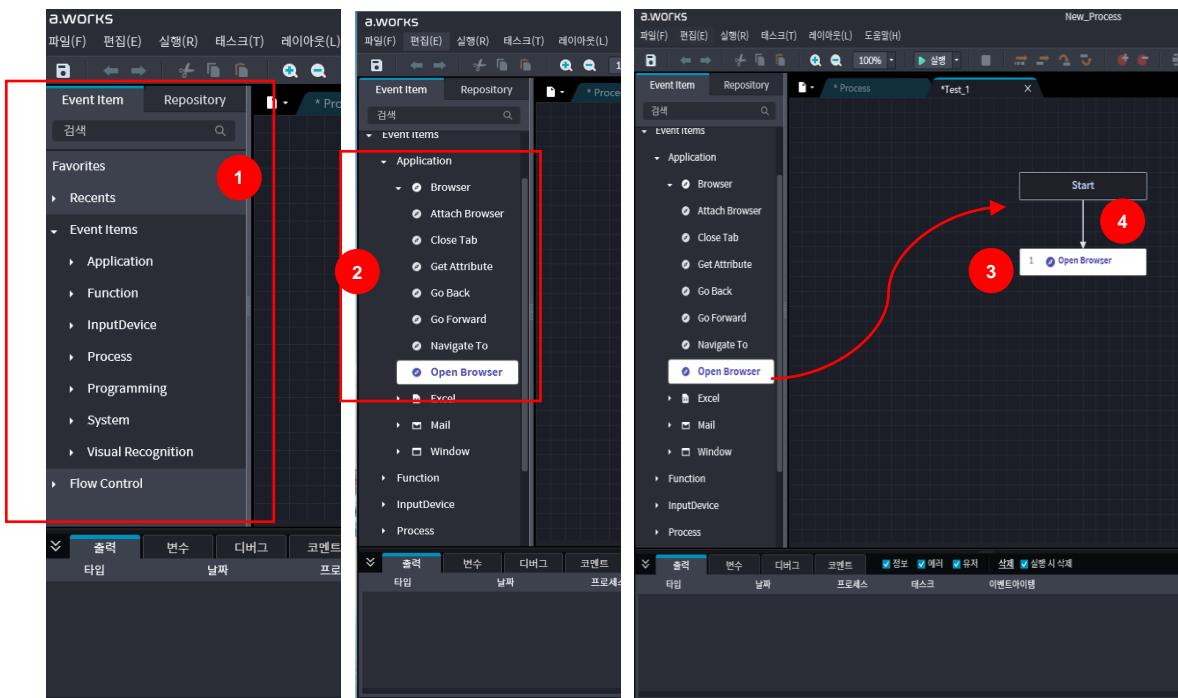
### 3.1. Event Item 가져오기

Canvas의 태스크 탭에 이 태스크에서 사용할 Item들을 가져옵니다. Event Item은 컴퓨터가 이해할 수 있는 명령어로 미리 제작해 놓은 라이브리 형태의 기능함수들입니다.

Event Item에는 Event Items과 Flow Control이 있으며, Item을 가져오는 방법에 대해 간단히 설명합니다.

프로세스 또는 태스크에서 사용할 Event Item들을 Canvas로 가져오기 위해서는 1) 화면의 Resource 영역에서 직접 가져오는 방법과 2) Resource 영역의 검색 창을 이용하는 방법, 3) Canvas 화면의 Context 메뉴를 이용하여 가져오는 방법 등 세 가지가 있습니다.

먼저, 화면의 Resource 영역에서 직접 가져오는 방법은, 화면 왼쪽의 Resource 영역의 Event Item 탭을 클릭합니다. 해당 항목의 Dropdown 화살표(▶)를 클릭하면 하위 리스트가 표시됩니다.

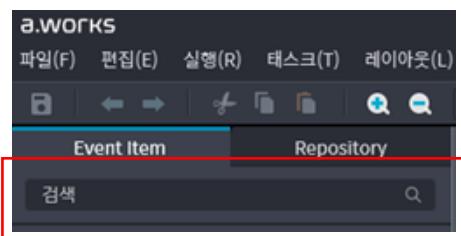


Event Items의 Dropdown 화살표(▶)를 클릭하면 태스크에 사용 가능한 여러 가지 Event Item들을 리스트로 표시하고, 펼쳐진 Event Items의 Dropdown 화살표(▼)를 한번 더 클릭하면 펼쳐진 리스트를 접고 Dropdown 화살표(▶)로 표시합니다.

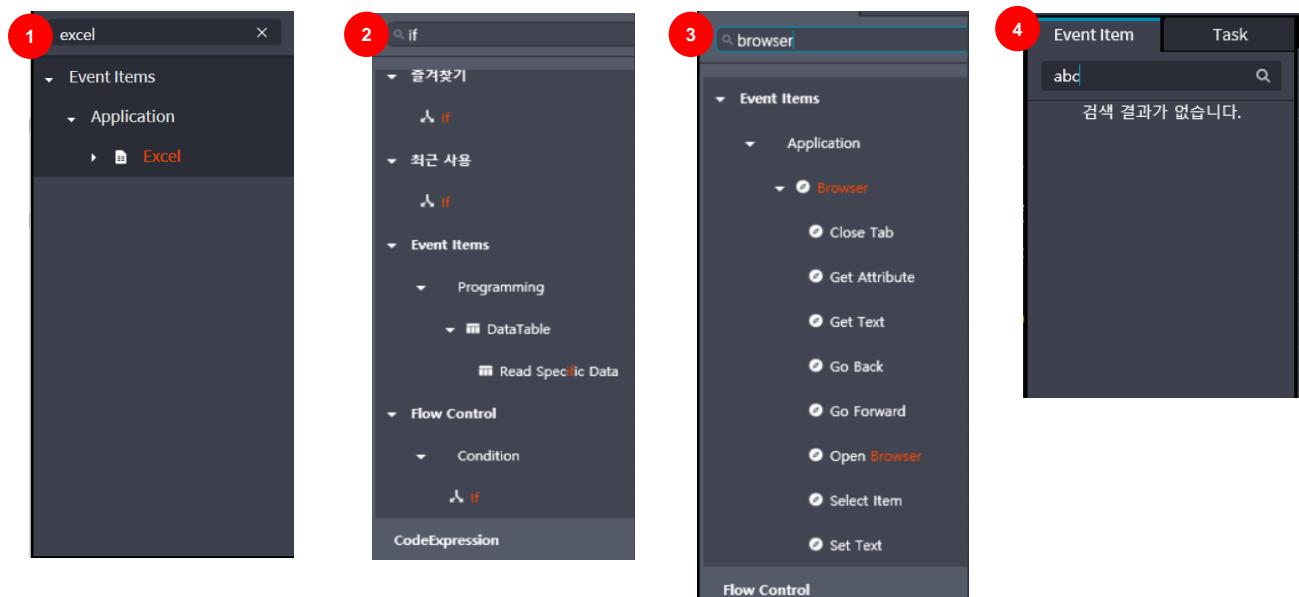
No.	명칭	설명
1	Event Item 탭	Favorites, Recents, Event Items, Flow Control로 구성하고 있습니다..
2	Event Items	Event Items의 Item들을 리스트로 표시합니다. 리스트 가운데 관련 항목의 Dropdown 화살표(▶)를 클릭하면 실제로

		<p>사용 가능한 Event Item들을 표시합니다. 화면에서는 <b>Application &gt; Browser &gt; Open Browser</b>를 선택하였습니다. Event Item은 프로세스 Canvas에 사용 가능한 일부 Event Item(Write Log, Assign, Code Expression, Message YesNo, Start Work)을 제외하고는 태스크 스크립트 작성시에만 사용할 수 있습니다.</p>
3	Item 가져오기	<p>Event Item 리스트에서 마우스로 해당 Item의 리스트를 클릭하면 하이라이트 아이콘으로 바뀌며 마우스의 왼쪽 버튼을 누른 상태에서 Canvas로 이동하여 놓습니다. 이를 Drag &amp; Drop이라고 합니다.</p>
4	연결선 그리기	<p>Canvas에서 가져온 Item을 화살표로 서로 연결합니다. Studio 2.0에서는 Event Item은 기존 Canvas에 있는 Item의 중간 하단부에 걸쳐 놓으면(Drag Over) 다음처리 Item으로 인식하고 자동으로 화살표를 표시합니다. 화살표는 사용자가 직접 그려 넣을 수도 있습니다. <b>※ Item 연결선 그리기 및 자동생성을 참고하시기 바랍니다.</b></p>

두 번째는, Resource 영역의 Event Item 검색 창을 이용하는 방법입니다. 해당 검색 창에 원하는 Event Item과 관련된 Keyword를 입력하면 다음과 같은 검색 유형별로 표시되고 해당 Event Item을 Canvas로 가져옵니다.



### ■ Resource 검색 유형별 표시 형식

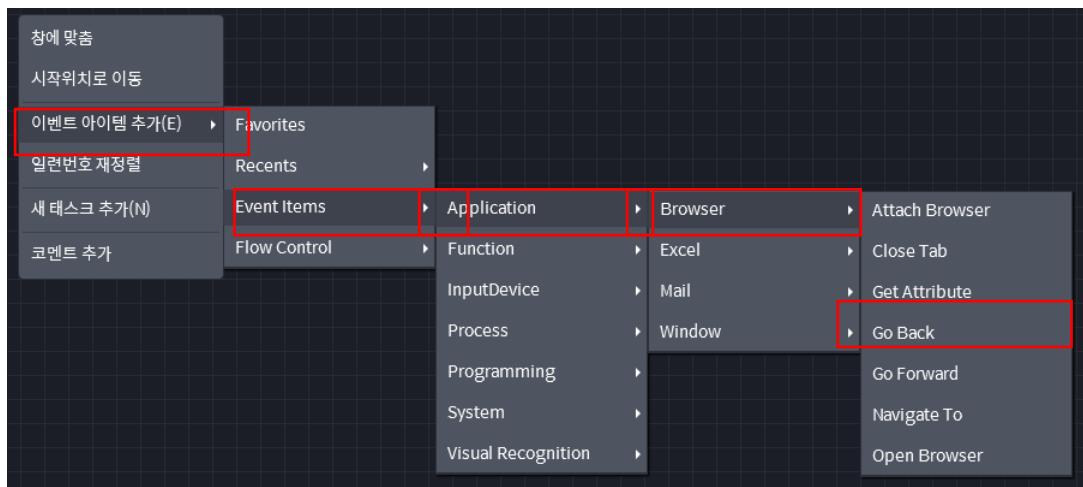


No	명칭	상세
1	검색유형 1	- Category 이름에 검색 결과가 있고, 하위항목에 검색 결과가 없는 경우 > 해당 Category는 확장되어 표시하고 검색 항목 이름을 표시(예: excel) > 검색된 Category나 항목의 단어는 빨강색으로 표시(공통)
2	검색유형 2	- Category 이름에 검색 결과는 없고, 하위항목에 검색 결과가 있는 경우 > Category는 확장되고, 하위 검색된 항목만 표시(예: if)
3	검색유형 3	- Category 이름에 검색 결과가 있고, 하위항목에 검색 결과가 있는 경우 > Category는 확장되고, 하위 모든 항목을 표시. 검색 항목 색으로 표시(예: browser)
4	검색결과 없음	- Category내 검색 결과도 없고, 하위항목에도 검색 결과가 없는 경우 > "검색 결과가 없습니다." 메시지만 표시

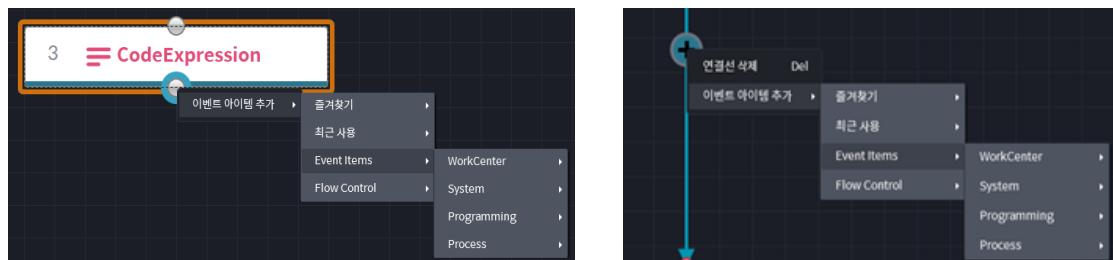
세 번째는, Canvas 화면의 Context 메뉴를 이용하여 가져오는 방법입니다.

Canvas 화면의 빈 공간에 오른쪽 마우스를 클릭하면 다음과 같은 Context 메뉴를 표시합니다.

먼저, Context 메뉴에서 이벤트아이템 추가(E)를 클릭 후 Dropdown 화살표(▶)를 클릭하고, 하위 리스트의 카테고리에서 Event Items 다시 Dropdown 화살표(▶)를 클릭하고, 해당 하위 리스트에서 다시 Dropdown 화살표(▶)로 원하는 Event Item을 찾아가는 방법입니다.



또는, 현재 적용한 Event Item이나 연결선을 한번 클릭하고 오른쪽 마우스를 클릭하면 해당 Item에 대한 Context 메뉴가 다음과 같이 표시되며, 선택 방식은 위와 동일합니다.



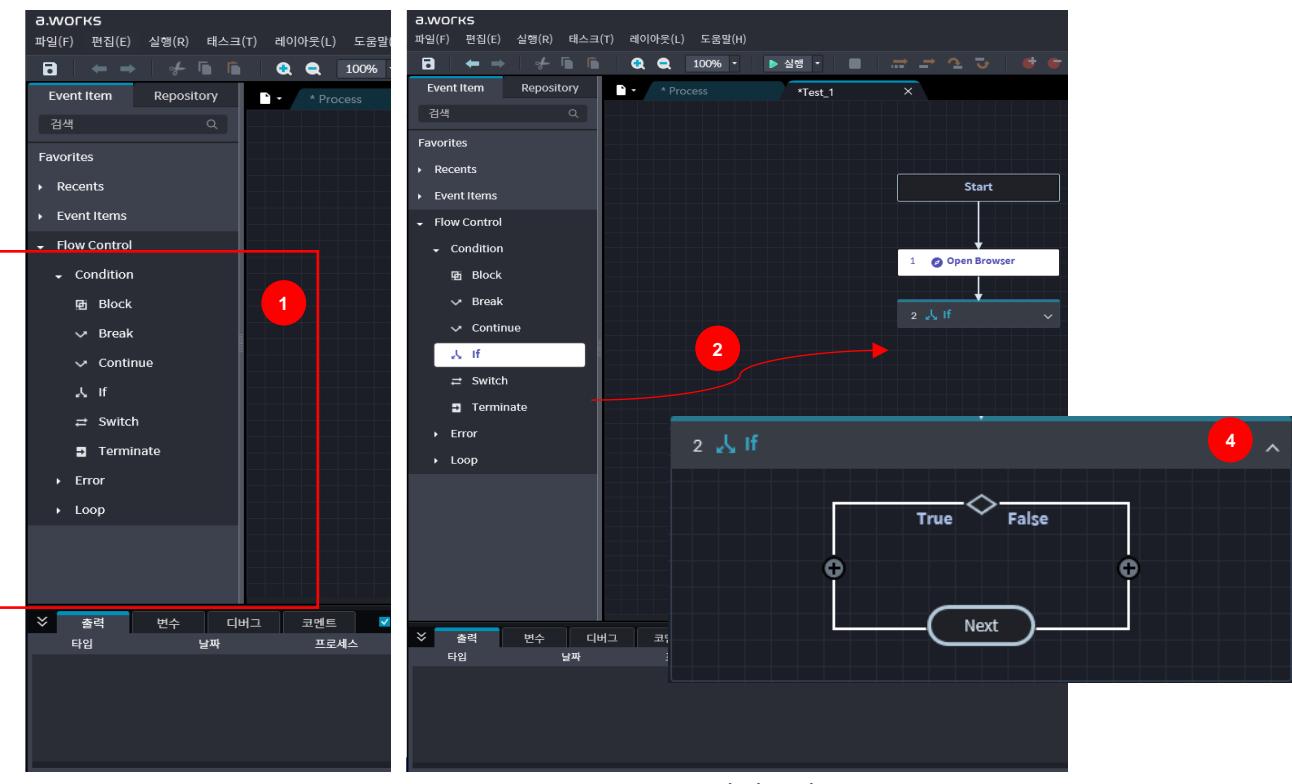
## 3.2. Flow Control 가져오기

프로세스 및 태스크의 흐름을 제어 하는 Item으로 Flow Control 이 있으며, 업무 프로세스를 만드는데 중요한 역할을 하는 Item입니다.

여기에서 Flow Control 을 가져오는 방법에 대해 간단히 설명합니다.

태스크에서 사용할 Flow Control 을 Canvas 로 가져오기 위해서는, 1) 화면의 Resource 영역에서 직접 가져오는 방법과 2) Resource 영역의 검색 창을 이용하는 방법, 3) Canvas 화면의 Context 메뉴를 이용하여 가져오는 방법 등 세 가지가 있습니다.

먼저, 화면의 Resource 영역에서 직접 가져오는 방법은, 화면 왼쪽의 Event Item 탭을 클릭하고 Flow Control 의 Dropdown 화살표를 클릭하여 해당 Item 을 선택하여 Canvas 로 Drag & Drop 합니다.

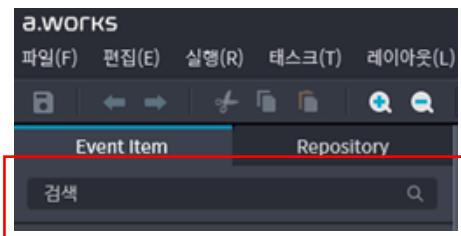


< Flow Control 가져오기 >

No.	명칭	설명
1	Flow Control	Flow Control 항목의 Dropdown 화살표(▶)를 클릭하면 사용 가능한 여러 가지 Flow Control 하위 리스트가 표시됩니다. Condition, Error, Loop 처리를 할 수 있습니다.
2	Flow Control 가져오기	각 그룹별 사용 가능한 하위 Item들은 아이콘으로 표시합니다. 화면에서는 <b>Condition &gt; IF</b> 를 선택하였습니다. Flow Control은 프로세스와 태스크 작성시 모두 사용할 수 있습니다.

		Event Item 과 동일하게 Flow Control 리스트에서 마우스로 해당 Item 을 클릭하면 하이라이트로 바뀌며 마우스의 왼쪽 버튼을 누른 상태에서 Canvas 로 이동하여 놓습니다.
3	연결선 그리기	Canvas에서 가져 온 Item을 화살표로 서로 연결합니다. ※ 화살표 연결선 생성방식은 Event Item 과 동일합니다.
4	접기/펼치기	Flow Control 은 순차처리, 분기처리, 반복처리 등 각 Item 의 처리내용에 따라 접거나 펼치기를 제공하는 경우가 있습니다. Item 의 맨 오른쪽의 아래화살표(▽)로 아래로 펼치고, 위 화살표(▽)로 위로 접을 수 있습니다.

두 번째는, Resource 영역의 Event Item 검색 창을 이용하는 방법입니다. 해당 검색 창에 원하는 Event Item 과 관련된 Keyword 를 입력하면 다음과 같은 검색 유형별로 표시되고 해당 Event Item 을 Canvas 로 가져옵니다.

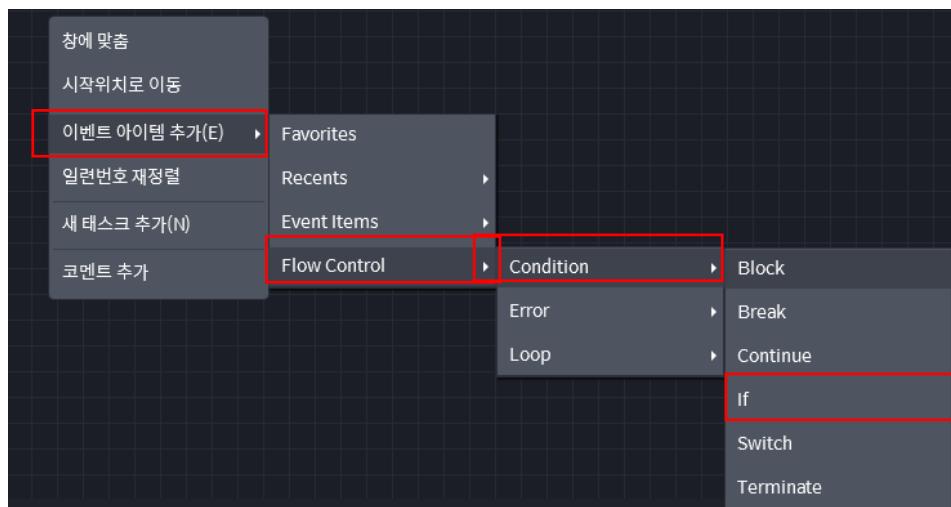


선택된 탭의 항목들을 이름으로 검색 가능하며, 검색어 입력 시 검색어가 포함되어 있는 카테고리 및 하위 Event Item 목록을 모두 보여 주며, 검색어와 일치하는 부분을 하이라이트로 표시합니다. 검색에 대해서는 "5.3.3 Event Item 가져오기의 Resource 영역의 검색 창 설명"을 참조하시기 바랍니다.

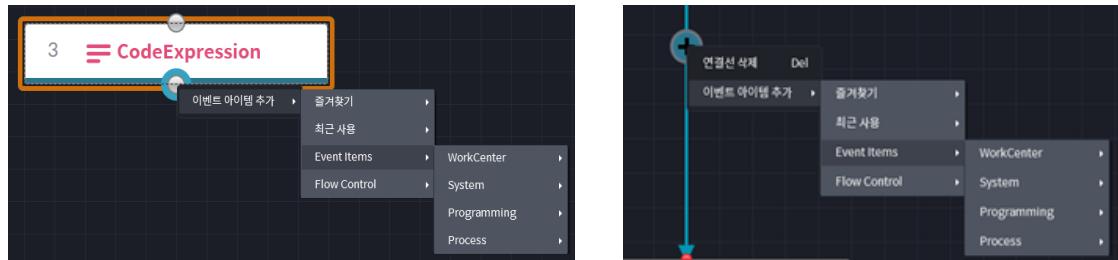
세 번째는, Canvas 화면의 Context 메뉴를 이용하여 가져오는 방법입니다.

Canvas 화면의 빈 공간에 오른쪽 마우스를 클릭하면 다음과 같은 Context 메뉴를 표시합니다.

먼저, Context 메뉴에서 이벤트아이템 추가(E)를 클릭 후 Dropdown 화살표(▶)를 클릭하고, 하위 리스트의 카테고리에서 Flow Control 을 선택하고 다시 Dropdown 화살표(▶)를 클릭하고, 해당 하위 리스트에서 다시 Dropdown 화살표(▶)로 원하는 Flow Control 을 찾아가는 방법입니다.

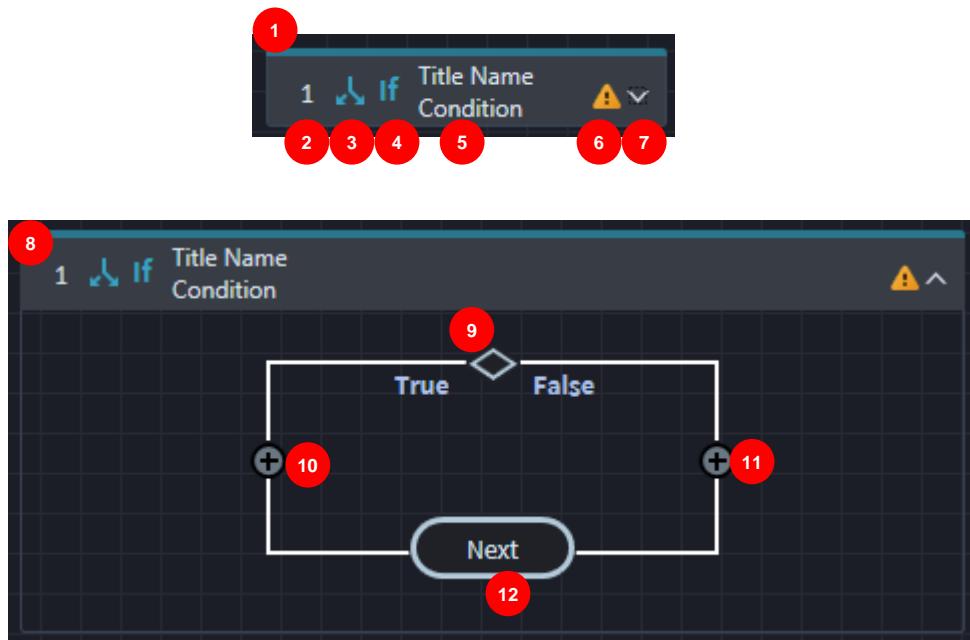


또는, 현재 적용한 Event Item이나 연결선을 한번 클릭하고 오른쪽 마우스를 클릭하면 해당 Item에 대한 Context 메뉴가 다음과 같이 표시되며, 선택 방식은 위와 동일합니다.



### ■ Flow Control의 구성과 상태 표현에 대하여

Flow Control의 구성과 상태 표현에 대하여 설명합니다.

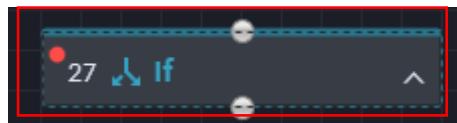


No.	명칭	설명
1	Flow Control Item	Event Item과 구분할 수 있도록 배경색상과 라인을 추가하였습니다. 접었을 경우 통일성을 기하기 위해 Event Item과 같이 가로 너비 고정
2	Item 연결번호	Event Item의 연결 번호
3	Flow Control Icon	Flow Control 을 Icon 으로 표기하였습니다.
4	Flow Control 명	Flow Control 이름을 축약하여 표기하였습니다.
5	Flow Control Header	상단에 Title Name 을 표기합니다. 길어지면 [...]로 축약합니다. Title Name 내용은 Property 에서 편집이 가능합니다. 하단에 Condition 을 표기합니다. 길어지면 [...]로 축약합니다. Condition 내용은 Property 에서 편집이 가능합니다.
6	Validate	Item에 Warning 요소가 있을 경우 우측 끝 부근에 Warning Icon 표시

7	펼치기/접기 버튼	선택 시 펼치기 상태로 표시하고 펼쳐지면 접기 버튼으로 표시합니다.
8	펼쳐진 블록 창	다이어그램 편집의 용이성을 위해 블록 기능을 제공합니다. 블록 구조 Item은 하위에 Child 블록 구조Item을 포함할 수 있고, 블록 내부의 Event Item자동 정렬 및 자동 레이아웃 기능이 적용됩니다. 하위의 항목은 동일 구조 내의 항목들에만 연결됩니다.
9	Condition Icon	Condition 문을 블록 내에서 간단히 Icon으로 표기합니다. 조건에 따라 True/False 등 내용을 표기합니다.
10	Event Item Guide	최소 생성시 Event Item 추가 가능 영역에 Guide Icon 표시합니다.
11	Event Item Guide	마우스 Over 및 Event Item Drag 시 색상이 변화합니다.
12	Next Icon	조건이벤트 실행 후 다음으로 넘어간다는 Icon 표시입니다.

- 접기/펼치기 시의 Flow Control 의 테두리 변화 및 연결점 표기

Flow Control 을 접은 상태에서 Top 및 Bottom 에 연결점을 표기하며,  
중단점(●) 도 표기할 수 있습니다.



펼쳐진 상태에서도 블록의 선택된 Event Item 을 알아 볼 수 있게 테두리 변화,  
연결점을 활성화하여 표시합니다.



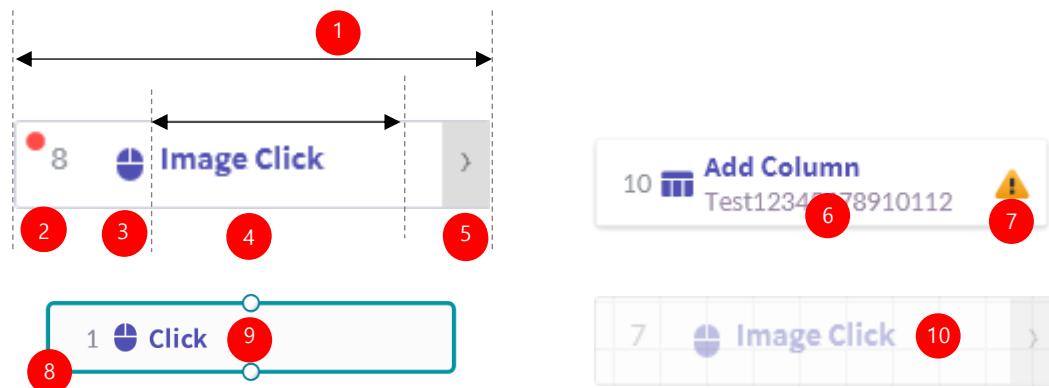
### 3.3. Canvas 레이아웃 정렬

프로세스 템과 태스크 템에서 사용되는 Event Item, Flow Control은 사용자로 하여금 Canvas 화면의 전체적인 내용의 파악과 이해를 돋기 위하여 레이아웃 정렬 기능을 지원합니다.

레이아웃 정렬에 대한 설명에 앞서 Event Item과 연결선에 대하여 설명합니다.

#### ■ Event Item 의 구성과 상태 표현에 대하여

Event Item 의 구성과 상태 표현에 대하여 설명합니다.



No.	명칭	설명
1	Event Item 폭	Event Item의 통일성을 기하기 위하여 Event item의 가로 폭, Event Name의 가로 폭은 고정되어 있습니다.
2	Event Item 번호 및 중단점 표시	Event Items의 번호는 해당 스크립트내에서 자동 부여되는 연속되는 번호입니다. Even Item 왼쪽 구석이 중단점(●)을 설정하면 표시합니다..
3	Event Item Icon	Event Item 의 용도를 바로 알 수 있도록 아이콘으로 표시합니다.
4	Event Item Name	기본적으로 정의되어 있는 이름입니다.
5	Indicator	화면의 객체인식을 위한 도구입니다. 객체인식을 해야 하는 Event Item에 구성되어 있습니다.
6	사용자 정의 Title Name	사용자는 필요 시 Event Name에 따라 정의하는 Title name을 표기하고 추가적인 작업 설명을 포함하여 입력할 수 있습니다.
7	Validate	Event Item을 적용하는데 있어 Warning 요소가 있을 경우 우측 끝에 Warning Icon을 표시합니다.
8	Select	선택된 Item임을 알아 볼 수 있게 테두리와 색상에 변화를 줍니다. 연결점을 활성화하여 표현하며, 연결점은 Top, Bottom에 표시합니다. 연결점에서 마우스로 Drag 시 연결선이 나옵니다.
9	연결점	연결점은 Target(Top, Prev), Source(Bottom, Next)를 의미합니다.
10	비활성화 표시	해당 Item 을 Property 에서 Disable에 V 체크 한 경우, 비활성화 처리된 상태를 알 수 있도록 아이콘, 글자 색을 변경하여 표시합니다.

## ■ Item의 정렬 및 일련번호에 대하여

### 1. Item의 정렬 기준

- 1) Start Item과 연결된 Item들을 우선 정렬합니다.
- 2) Y 좌표를 우선 순위로 정렬합니다.
- 3) Y 좌표가 같은 경우 X 좌표로 정렬합니다.
- 4) X와 Y 좌표가 모두 같은 경우 임의로 정렬합니다.  
(우선 순위 없으므로 내부 리스트 순서로 정렬합니다.)

### 2. 일련번호 재정렬 기준

- 1) Canvas내 Item은 숫자를 지정된 기준으로 재정렬합니다.
- 2) Comment는 별도의 번호 리스트를 가지므로 재정렬하지 않습니다.
- 3) Start Item은 번호가 없으며, 재정렬 되지 않습니다.
- 4) Start에 연결된 Item을 위에서부터 아래로 우선 정렬합니다.  
Start Item을 시작으로 그 다음 Item부터 1번으로 합니다.
- 5) Start Item에 연결되지 않은 경우 서로 연결된 Item을 하나의 묶음으로 보고,  
연결 중 첫 번째 Item을 기준으로 연속된 번호로 정렬합니다.
- 6) Item은 Y 좌표를 우선으로 정렬하고, X 좌표를 차선으로 정렬합니다.

### 3. X, Y 좌표 정렬기준

#### 1) X좌표 정렬

- X 좌표를 기준으로 정렬합니다.
- X 값이 작은 순서대로 우선 정렬합니다.
- X 값 정렬이 차선 순위입니다.

#### 2) Y좌표 정렬

- Y 좌표를 기준으로 정렬합니다.
- Y 값이 작은 순서대로 우선 정렬합니다.
- Y 값 정렬이 우선 순위입니다.

### 4. 연속된 연결의 특수경우 기준

- 1) Item이 연결선으로 연속되어 있을 경우, 연결의 첫 Item을 기준으로 번호를 정렬합니다.  
예) 첫 Item이 2번으로 정렬되는 경우, 그 다음 연결된 Item의 위치와 상관없이 3번이 됩니다.
- 2) 가로로 나열되는 그룹의 경우, 왼쪽부터 번호를 정렬합니다. 예) If
- 3) 세로로 나열되는 그룹인 경우, 위에서부터 번호를 정렬합니다. 예) switch, try catch

## @@ 주의 @@

※ 일련번호를 재정렬 했을 경우, 저장 버튼을 활성화하고 저장하기 전까지 활성화 상태를 유지합니다. (Undo, Redo 개수 혹은 저장 상태와 상관없이 활성화 되어 있습니다.)

## ■ Item 연결선 그리기 및 자동생성에 대하여

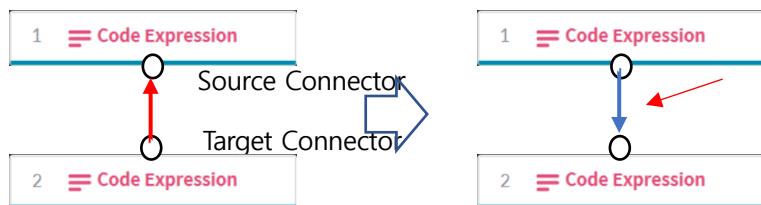
Studio 2.0에서는 기존 Studio 1.X 버전의 Item 간 화살표 그리기 기능의 불편함을 대폭 개선하여 Item 간 연결선을 직접 그리기는 물론 자동 생성 기능을 추가 하였습니다.

### ● Item 연결선에 대하여

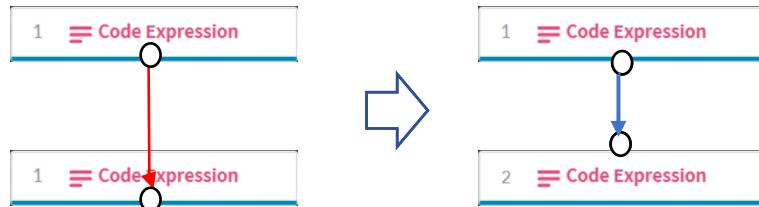
Item에는 Target이 되는 Connector (Top, Prev)와 Source Connector(Bottom, Next)를 갖고 있으며, 상하연결이 기본이고, 단일 연결이 기본이고, 연결선은 Source Connector에서 시작하여 Target Connector 연결합니다.

기존 Studio 1.X에서 제공되지 않던 반대방향 연결도 지원합니다.

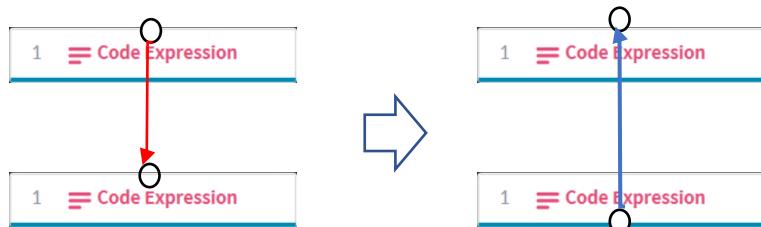
2 번 Item의 Target Connector로 1 번 Item의 Source Connector를 연결할 경우 상하로 자동으로 연결을 합니다.



또한, 다음과 같이 잘못 시도된 경우에는 자동으로 연결을 수정하여 연결합니다.



1 번 Item의 Bottom에서 2 번 Item의 Bottom으로 연결은 잘못된 연결을 시도하고 있으므로, 1 번 Item의 Bottom에서 2 번 Item의 Top으로 연결합니다.



1 번 Item의 Top에서 2 번 Item의 Top으로 연결은 잘못된 연결을 시도하고 있으나 반대방향 연결이 가능성이 있으므로, 2 번 Item의 Bottom에서 1 번 Item의 Top으로 연결합니다.

## @@ 주의 @@

### ※ Item 연결선 그리기의 제약

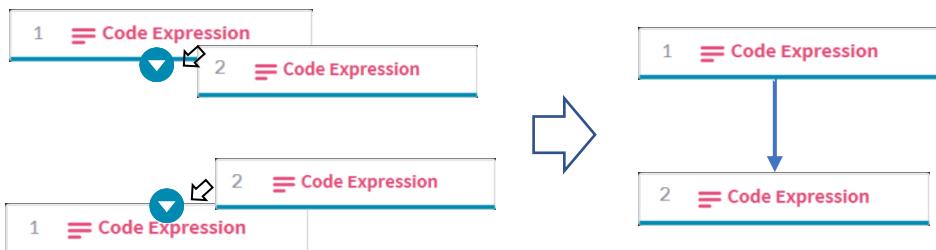
Studio2.0에서는 연결선 그리기 및 자동생성 등 편리함이 추가 되었지만 이에 대한 제약도 존재합니다. 다음의 경우에는 연결선이 자동 연결되지 않습니다. 스크립트 작성시 참고하셔서 연결선의 오류가 발생하지 않도록 주의하여 주십시오.

- 1) 연결을 시도하려는 Target Item에 기존 연결선이 존재할 경우
- 2) 연결을 시도하려는 Source Item에 기존 연결이 존재할 경우
- 3) 여러 개의 연결된 Item 중 Target Item의 출력 화살표가 Source Item의 입력이 되는 순환구조의 경우
- 4) Try Catch 등 블록구조에서 같은 블록이 아닌 외부 Item에서 연결하는 경우

### ● Item 연결선 자동생성에 대하여

Case 1) Item을 Drag Over 시키면 자동 생성

Canvas의 기존 Item에 추가로 Item을 연결할 경우, 중하단에 Drag Over 시키면 아래쪽으로, 중상단에 Drag Over 시키면 위쪽으로 연결선을 자동 생성합니다.

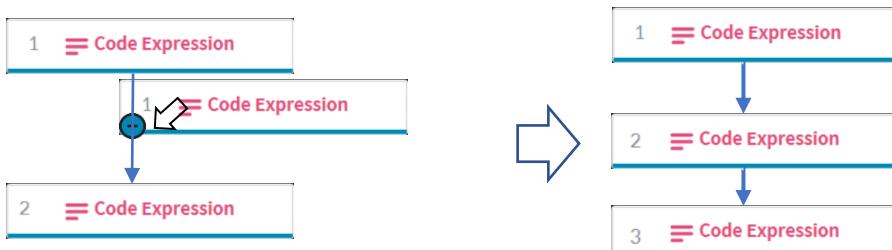


## @@ 주의 @@

※ 단, 연결하고자 하는 방향에 연결선이 없어야 합니다.

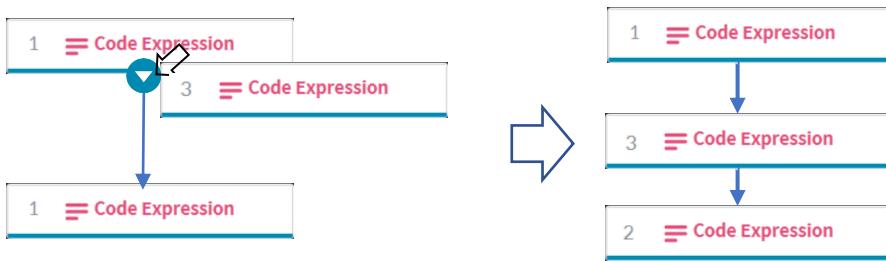
Case 2) Item을 중간에 Drag Over시키면 자동 생성

기존 Item과 Item 사이에 연결선에 새로운 Item을 Drag Over시키면 사이에 (+) 그림이 만들어 지고 해당 위치에 Item을 Drag Over시키면 연결선이 자동으로 연결됩니다.



Case3) 기존 연결 Item에 Drag Over시키면 자동 연결

기존 Item과 Item 사이에 연결선이 존재하는 상황에서 새로운 Item을 Drag Over시키면 해당 위치에 Item을 자동 연결합니다.



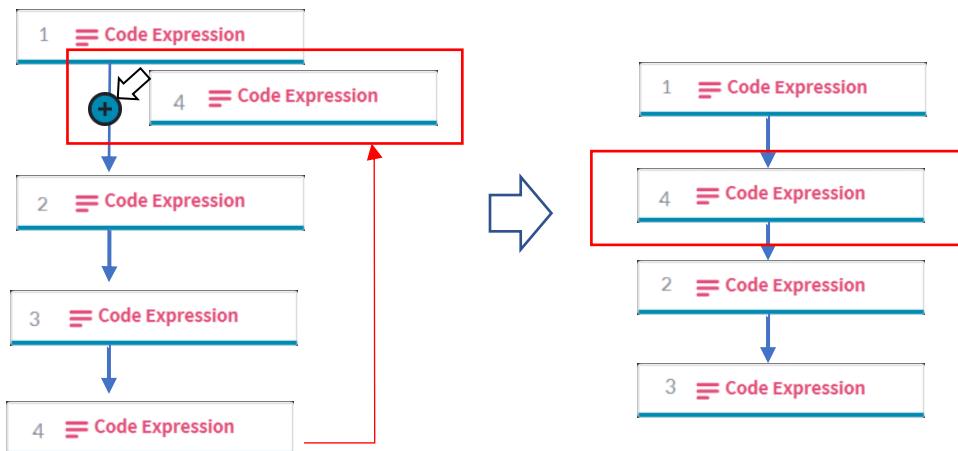
### @@ 주의 @@

\* 단, 연결은 되어 있으나 서로 연속이 아닌 Item을 선택하여 자동연결을 시도하거나, 연결이 되어 있지 않은 Item을 선택하여 자동연결을 시도할 경우에는 연결되지 않습니다.



### Case4) 기존 Item 연결선 자동 생성 및 순서 변경

같은 연결선 안에서는 간단히 Drag & drop 만으로도 Item 의 순서를 변경할 수 있습니다.



## ■ Item의 자동 정렬에 대하여

모든 [Start] Item과 연결된 Item은 기본적으로 가운데 정렬하고 아래 방향으로 연결하는 것으로 레이아웃 정렬을 지원합니다. 태스크 템은 모든 Item을 자동정렬을 지원합니다.

단, 프로세스 템은 Main Menu >레이아웃(L) > 레이아웃 정렬(A)을 클릭하여 Item의 레이아웃을 수동으로 정렬할 수 있습니다. 레이아웃 정렬은 한번 실행 후 다른 동작이 발생하기까지 비활성화 되며 연속해서 사용할 수 없습니다.

또한, Item에 대해서도 Item 정렬을 통하여 수동정렬을 할 수 있으며, 수동정렬의 활성화 조건은, 기준 레이아웃을 정렬 후 수정이 있는 경우에 선택된 Item이 두 개 이상이어야 정렬여부가 활성화 하며, 수정이 없는 경우 오른쪽 그림과 같이 정렬여부가 비활성화되어 표시합니다.

Item 정렬은, 선택한 Item을 기준으로 가장 왼쪽에 있는 Item의 왼쪽을 기준으로 정렬하는 **왼쪽 정렬(L)**, 선택한 Item을 도두 가운데 위치로 정렬하는 **가운데 정렬(C)**, 선택한 Item중 가장 오른쪽 Item의 오른쪽에 맞추는 **오른쪽 정렬(R)**이 있습니다.

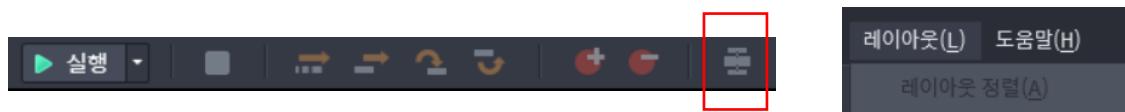


또한, 다음과 같은 Tool Bar 아이콘으로도 레이아웃 정렬(A)의 가능 여부를 확인할 수 있습니다.

- 프로세스 템의 경우, Event Item 추가 등 변경이 있는 경우 아이콘과 메뉴가 활성화합니다.

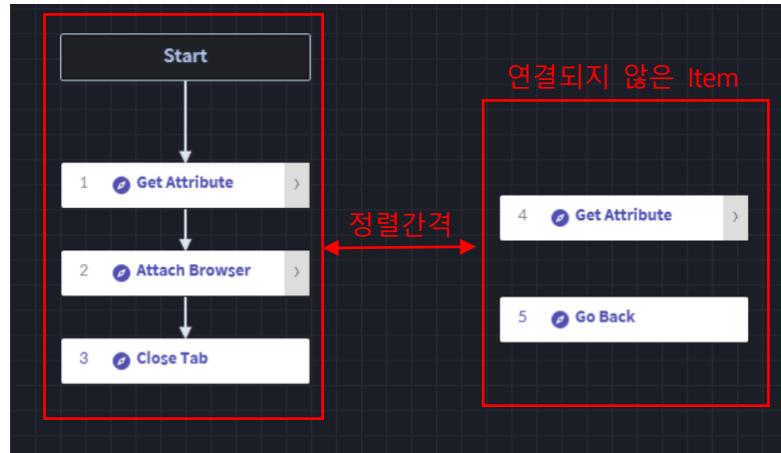


- 태스크 템의 경우에는 사용하는 Even Item은 기본적으로 자동정렬하고 있어 의미는 없습니다.



### ■ Start Item과 연결되어 있지 않은 Item의 정렬에 대하여

프로세스 탭에서 스크립트 작성시 연결을 해 놓고 있지 않은 Item이 발생할 수 있습니다.  
프로세스에서 제외 또는 작성 보류 태스크 등



< Start Item에 연결되지 않은 Item의 레이아웃 정렬 화면 >

이 경우, Start Item과 연결되지 않은 Item들은 Start Item과 연결되어 정렬된 Item들과 일정 간격 우측으로 벗어난 위치에 자동 정렬합니다.

간격은 Event Item 가로 길이의  $\frac{3}{4}$  (Event Item 너비 200, 간격 150)입니다.

또한, Start Item과 연결되지 않은 Item들은 해당 위치에서 X좌표만 정렬되어 표시합니다.

Y 좌표는 임의로 변경할 수 있지만 X 좌표는 항상 정렬되어 변경할 수 없습니다.

### @@ 중요 @@

#### ■ 태스크 탭의 자동정렬 시 제약사항

Studio의 Canvas상의 Process탭과 Task탭의 정렬 방식은 다음과 같습니다.

- Process 탭: Start Item이 항상 존재합니다.

(\*예외: 편집(E)에서 모두 선택(A)을 하면 Start Node는 자동으로 선택이 제외, 이동불가)

- Task 탭: 모든 Event Item이 항상 자동 정렬이 됩니다.

Start Item이 연결된 Item은 왼쪽 정렬이며, 연결되지 않은 Item은 오른쪽 정렬입니다.

- 태스크 탭에서도 자동정렬(아이콘) 기능 적용 예정입니다.

연결선 삭제 시 오른쪽 정렬라인 최 상단으로 위치하며 "Start(시작)"를 넘지 않도록 아래로 겹치지 않게 정렬합니다.

(\*제약: 자동 정렬된 X축상에서 Y축만 이동이 가능합니다. Start Item과 연결되지 않은 오른쪽 정렬된 Item은 Y축의 이동이 자유롭기 때문에 겹쳐 질 수 있습니다.)

예외: 편집(E)에서 모두 선택(A)을 하면 Start Node는 자동으로 선택이 제외, 이동불가)

## 3.4. Property 영역

Property 영역은 기본적으로 이벤트 실행, ScreenShot, Common, Property의 구성 항목을 가지며, 각 Flow Control, Event Item의 특성에 맞는 탭이 추가적으로 생성됩니다.  
여기에서는 Key Input 이라는 Event Item의 Property 패널을 예로 들었습니다.



No.	설명
1	해당 Event Item을 실행합니다.
2	해당 Event Item에 사용한 Screenshot 확인할 수 있습니다. (Indicator가 사용 가능한 Event Item에만 적용됩니다.)
3	Event Item의 실행 전 Wait Time을 입력합니다. (Millisecond 단위, 숫자)
4	Event Item의 실행 후 Wait Time을 입력합니다. (Millisecond 단위, 숫자)
5	해당 Event Item의 Title에 표시되는 이름을 설정합니다. (최대 입력길이: 256자)
6	해당 Event Item의 부가적인 설명을 입력합니다. (최대 입력길이: 256자)
7	Event Item의 실패 여부와 관계없이 실행할지 선택할 수 있습니다. (V 체크 시 계속 실행)
8	Event Item의 실행 여부를 선택 가능합니다. (V 체크 시 실행하지 않습니다.)

※ Item 의 Property 영역의 Common 항목에는 Title Name 과 Description 을 입력하게 되어 있습니다.  
이하 공통사항으로 해당 항목의 설명은 생략합니다.

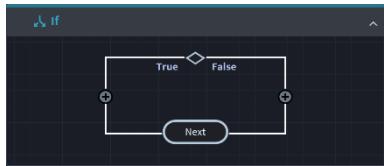
※ 속성 탭의 필수 입력항목은 굵은 글씨로 표시하며, 필수 입력항목이 미입력된 상태에서 실행 시 Output 로그에 오류 내역("해당 이벤트 아이템 이름 + 필수 항목의 값이 없습니다")이 표시됩니다.

### 3.5. Event Items 설명방법

이 사용자 매뉴얼에서 소개하는 Event Item의 표기 방법, 속성값 및 항목설명은 다음과 같습니다.

#### ■ Attach Browser

Browser를 Attach하는데 사용합니다.



또는



Event Item 이름  
Event Item 설명

Event Item 형태

Event Item의 속성값 패널  
항목의 사용방법에 대해  
설명합니다.



Event Item의  
속성값 패널  
번호는 하단의  
No의 설명과  
대응

No.	설명
1	Open하고자 하는 Browser 타입을 선택합니다. (지원Type: IE, Chrome)
2	Indicator로 Browser Title을 가져오거나 Title을 직접 입력합니다. Attach할 Browser 선택.
3	Browser 변수 명을 입력합니다.
4	입력 창을 마우스로 클릭하면 입력 창의 입력방법에 대한 주의가 필요한 경우 TIP을 기능을 두고 Property 패널의 하단에 표시합니다.

**TIP** 실행한 Browser의 정보를 저장할 변수명  
을 입력합니다. Browser는 Browser형식  
의 변수입니다.

# 4. Flow Control

업무 프로세스를 만드는 프로세스 탭과 태스크 탭에서 사용하는 흐름제어를 위한 Item들입니다.  
Condition, Error, Loop 등 세 개의 카테고리로 구분됩니다.

※ Flow Control의 입력값에 대한 제약사항은 다음과 같습니다.

- > 입력값의 검증 결과는 디버그 시 또는 실행 시 하단의 Output 패널에 표시됩니다.
- > 입력 필드에서 지정한 형식대로 입력 되어야 합니다.

※ Flow Control 아이템의 조건문 입력하는 Code Editor 창의 크기 조정이 가능합니다.

## 4.1. Condition

Condition은 If, Switch, Loop 등과 같은 특정 조건에 의해 분기되는 Flow Item들을 모아 놓은 카테고리입니다.  
Condition은 총 6개의 Item으로 구성되어 있습니다.

- Block
- Break
- Continue
- If
- Switch
- Terminate

각 Item의 상세는 다음과 같습니다.

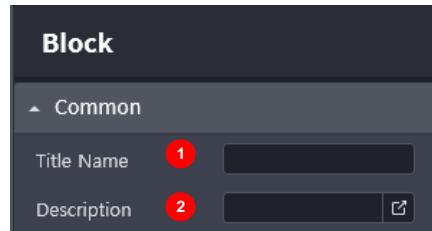
### ■ Block

Block은 Condition이 없는 구조로, 접기/펼치기 기능을 갖고 있으면서 연결된 Item을 담아 놓는 기능만을 하는 Item입니다.



우측 상단의 접기(^), 펼치기(▽) 버튼으로 Block 내용을 펴고 접을 수 있습니다.

※ 이후 Item에 접기(^), 펼치기(▽) 버튼이 있는 경우, 해당 기능 설명은 생략합니다.



No.	설명
1	해당 Item의 Title Name을 String(문자열)으로 입력합니다.
2	해당 Item의 Description을 String(문자열)으로 입력합니다.

※ Item 의 Property 탭의 Common 항목에는 Title Name 과 Description 을 입력하게 되어 있습니다.

이하 공통사항으로 해당 항목의 설명은 생략합니다.

## ■ Break

For/For Each 등의 반복하는 Item문의 수행을 멈추고 다음 Item을 실행 할 수 있게 하는 Item입니다.



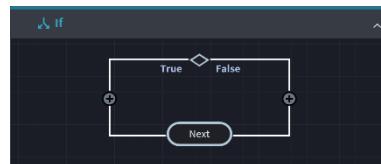
## ■ Continue

For/For Each 등의 반복하는 Item문 내부에서 If/Switch 등의 조건문을 사용하여 특정 조건을 만족하는 경우 반복문의 최상위 Loop로 회귀하는 Item입니다.



## ■ If

Condition의 연산 결과에 따라 True, False 다음에 따라오는 기능을 수행할 수 있는 Item입니다.

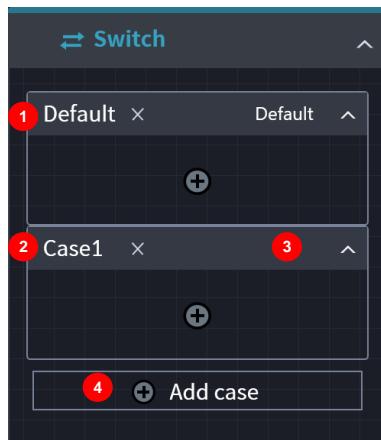


No.	설명
1	결과 값이 Boolean(True/False)로 출력되는 판단문을 입력합니다. (Type: 조건문)

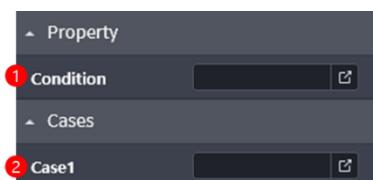
- 정상 동작: 판단문 입력은 결과 값이 Boolean 인 변수 혹은 코드 입력.
- 오류 유발 조건: 입력문 결과 값이 null 혹은 Boolean 이 아닌 모든 경우 Ex) var i = 0)

## ■ Switch

특정 변수의 값에 따라서 실행 할 기능을 지정 하는 Item입니다.



No.	설명
1	Default Block에는 Drag & Drop으로 Event Item을 추가할 수 있습니다. Default Block은 삭제가 불가합니다.
2	Case Block에도 Drag & Drop으로 Event Item을 추가할 수 있습니다. 마우스 Over 시 각 Case마다 삭제(X) 아이콘이 활성화되고 삭제할 수 있습니다.
3	Case Block의 각 Case마다 조건문을 설정 및 표시할 수 있습니다.
4	'Add case' 버튼을 통해 Case Block을 추가할 수 있습니다.



No.	설명
1	조건판단 대상이 되는 변수 명을 입력합니다. (Type: String / Int / Boolean / Enum)합니다.
2	Case 입력값에 따른 동작을 정의(Type: Boolean)합니다. Case에 포함된 값이 없을 경우 'Default'를 수행합니다.

## ■ Terminate

스크립트 진행을 멈추기 위해 사용하는 Item입니다.



## 4.2 Error

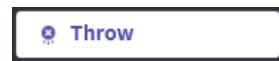
Error는 Error 발생시 예외처리 기능을 가진 Item을 모아 놓은 카테고리입니다.

Error는 총 2개의 Item으로 구성되어 있습니다.

- Throw
- Try Catch

### ■ Throw

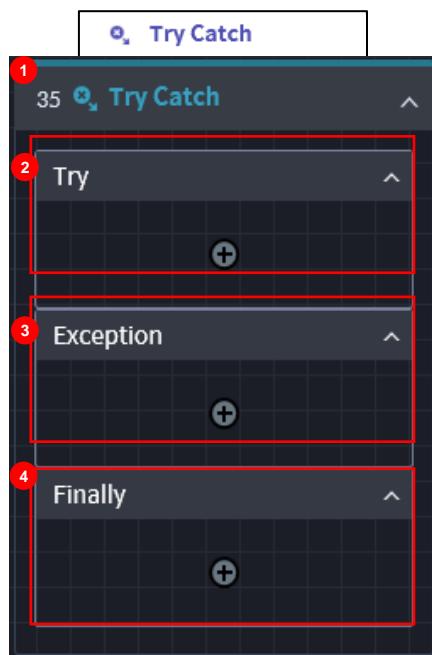
사용자가 Exception을 발생시킬 수 있게 합니다.



No.	설명
1	발생시킬 Exception을 입력합니다. (Type: AwException) · Exception Type은 Try Catch의 Exception Cclass의 종류 참조

### ■ Try Catch (v2.0)

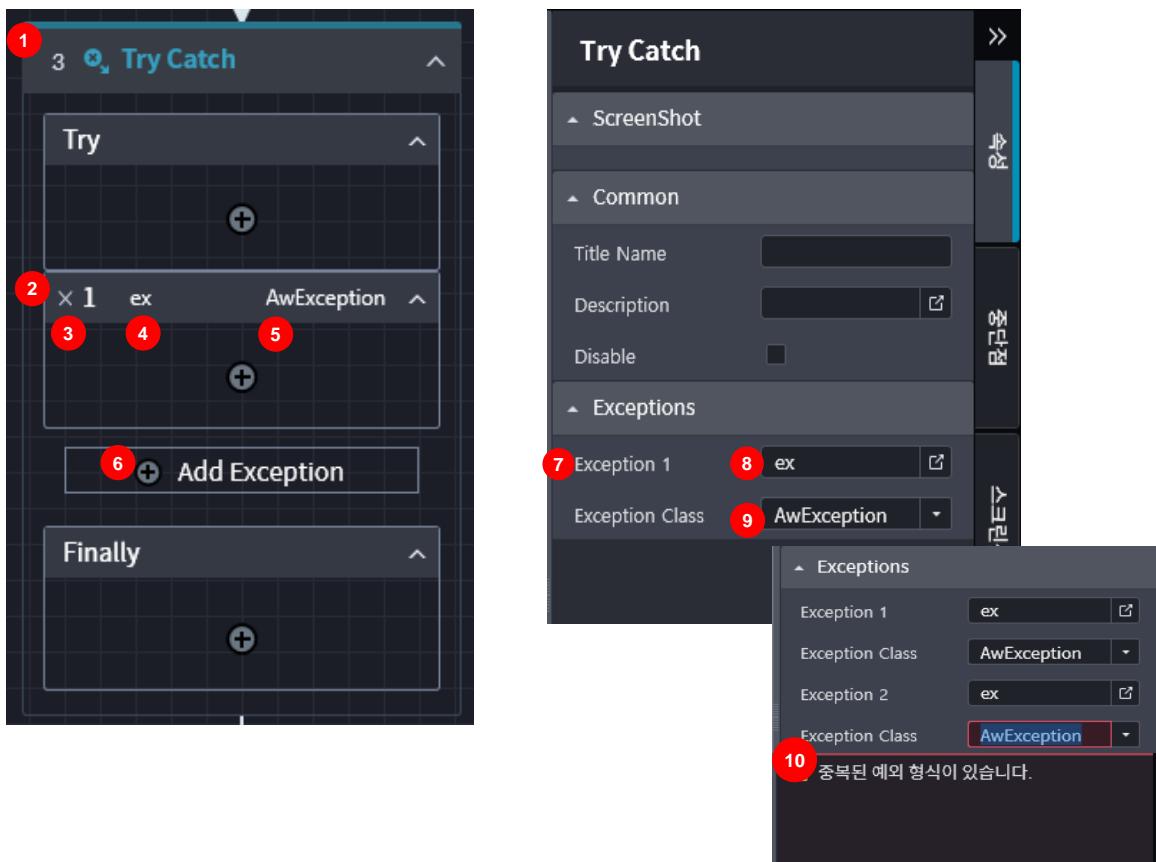
예외사항을 처리할 수 있도록 하는 Try Catch를 적용합니다. Try에 실행하고 싶은 Item을 배열하고 Exception에는 예외상황 발생 시 실행하고 싶은 Item을 배열합니다.



No.	설명
1	Try & Exception & finally의 하위 Block으로 구성합니다. 각 내부 Block에 Event Item을 Drag & Drop하여 추가 가능합니다.
2	Try는 Default Block으로 Event Item을 Drag & Drop하여 추가 가능합니다. Default Block으로 삭제는 불가능합니다.
3	예외처리 Block으로 Event Item을 Drag & Drop하여 추가 가능합니다.
4	최종처리를 위한 Block으로 (+) 선택 시 Case Block을 하나씩 추가합니다.

## ■ Try Catch (v2.1.3)

기존 Try Catch가 가지고 있던 구조상의 문제점이 v2.1.3 부터 개선되어, Exception 변수 이름을 사용자가 지정 가능하게 되고 Throw가 가능해집니다.



No.	설명
1	Try & Exception & finally의 하위 Block으로 구성합니다. 각 내부 Block에 Event Item을 Drag & Drop하여 추가 가능합니다.
2	Exception Block 영역에 Event Item을 Drag & Drop하여 추가 가능합니다. Collapse 기능을 제공합니다.
3	Exception 블록 삭제가 가능합니다.

4	변수 이름으로 ex가 default입니다. Property에서 변경 가능합니다.
5	변수 타입으로 Property에서 Combo Box로 설정합니다.
6	Exception Block을 추가합니다.
7	Title 명은 생성 순으로 숫자만 표기됩니다.
8	우측 이름 입력 Box에 'ex'가 default로 들어가 있습니다. 변수 명으로 생성 가능한 값을 입력 받을 수 있습니다.
9	Exception Class로 Combo Box로 Exception 타입을 선택합니다.
10	동일한 Exception Class를 선택했을 경우 중복 오류가 나타납니다.

### @@ Info @@

- 모든 블록은 Event Item, Event Item Group, Task, Task Group 등을 입력할 수 있습니다.
- Exception 블록: 단일 블록으로 Switch, If 등을 통해 구분하여 스크립트를 작성해야 합니다.
  - ※ Event Item 분기 또는 Error Code 별 분기가 가능
- 변수는 ex. 변수 명을 선언하고 사용하여야 합니다.
  - > Error Code / Error Message / Event Name /Event Code: ex.ErrorCode.ToString()

※ Try Catch의 Exception Class의 종류

No.	Class Name	설명
1	AssetException	Studio에서 WorkCenter Asset의 내용을 가지고 오거나, Asset의 내용을 Setting 할 때 예외 발생 시 사용됩니다.
2	AssignException	Studio에서 Assign Event Item이 예외 발생 시 사용됩니다.
3	AwException	Studio의 모든 Event Item의 예외 발생 Class이며, 최상위 부모 Exception Class입니다.
4	BrowserException	Studio에서 Browser 관련 Event Item의 예외 발생 시 사용됩니다.
5	BusinessException	Studio에서 사용자 정의 예외 Class를 정의할 때 사용됩니다.
6	ClipboardException	Studio에서 Clipboard 관련 Event Item의 예외 발생 시 사용됩니다.
7	CodeExpressionException	Studio에서 Code Expression 관련 Event Item의 예외 발생 시 사용됩니다.
8	ControlException	Studio에서 Control 관련 Event Item의 예외 발생 시 사용됩니다.
9	CryptographyException	Studio에서 Cryptography 관련 Event Item의 예외 발생 시 사용됩니다.
10	DataTableException	Studio에서 DataTable 관련 Event Item의 예외 발생 시 사용됩니다.
11	DialogException	Studio에서 Dialog 관련 Event Item의 예외 발생 시 사용됩니다.
12	DirectoryException	Studio에서 Directory 관련 Event Item의 예외 발생 시 사용됩니다.
13	DisplayException	Studio에서 Display 관련 Event Item의 예외 발생 시 사용됩니다.
14	ExcelException	Studio에서 Excel 관련 Event Item의 예외 발생 시 사용됩니다.
15	FileException	Studio에서 File 관련 Event Item의 예외 발생 시 사용됩니다.
16	FlowControlException	Studio에서 Flow Control 관련 Event Item의 예외 발생 시 사용됩니다.
17	KeyboardException	Studio에서 Keyboard 관련 Event Item의 예외 발생 시 사용됩니다.
18	MailException	Studio에서 Mail 관련 Event Item의 예외 발생 시 사용됩니다.
19	MouseException	Studio에서 Mouse 관련 Event Item의 예외 발생 시 사용됩니다.
20	OCRException	Studio에서 OCR 관련 Event Item의 예외 발생 시 사용됩니다.
21	ProcessException	Studio에서 Process 관련 Event Item의 예외 발생 시 사용됩니다.
22	QueueException	Studio에서 Queue 관련 Event Item의 예외 발생 시 사용됩니다.
23	ScrapingException	Studio에서 Scraping 관련 Event Item의 예외 발생 시 사용됩니다.
24	WindowException	Studio에서 Window 관련 Event Item의 예외 발생 시 사용됩니다.
25	WindowsCredentialException	Studio에서 Windows Credential 관련 Event Item의 예외 발생 시 사용됩니다.
26	WorkException	Studio에서 Work 관련 Event Item의 예외 발생 시 사용됩니다.
27	ZipException	Studio에서 Zip/Unzip 관련 Event Item의 예외 발생 시 사용됩니다.

## 4.3 Loop

특정 기능들을 반복하여 사용하는 것을 가능하게 하는 Item을 모아 놓은 카테고리입니다.

Loop는 총 5개의 Item으로 구성합니다.

- DoWhile
- For
- For Each
- Retry
- While

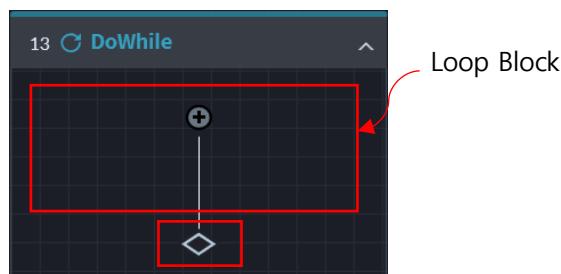
### ■ DoWhile

특정 기능들이 반복적으로 수행될 수 있도록 합니다. 루프 변수의 비교 및 증감을 기준으로 수행됩니다.

C DoWhile

스크립트를 실행 후 종료 조건 판단문을 확인합니다. Loop Block 영역에 Event Item을 Drag & Drop하여 추가하고 조건에 맞는 동안 반복하여 실행할 수 있습니다.

접은 상태에서는 Loop Block은 표시되지 않습니다. 조건 판단문이 Block 하단에 위치하며, 접기/펼치기 기능을 제공합니다.

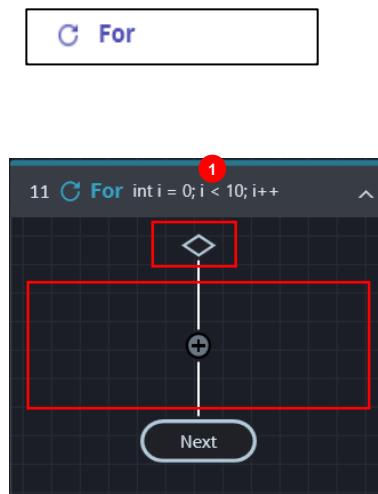


No.	설명
1	프로세스 반복 실행을 멈춰야 하는 조건 판단문을 입력합니다. (Type: 조건문)

## ■ For

특정 기능들이 반복적으로 수행될 수 있도록 합니다. 루프 변수의 비교 및 증감을 기준으로 수행됩니다.

스크립트의 조건문을 확인하고 반복 실행합니다. 조건 판단문이 Block 상단에 위치하며, 접기/펼치기 기능을 제공합니다. Loop Block 영역에 Event Item을 Drag & Drop하여 추가하고 조건에 맞는 동안 반복하여 실행할 수 있습니다. 접은 상태에서는 Loop Block은 표시되지 않습니다.



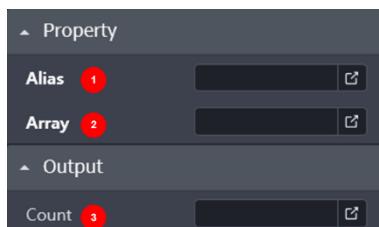
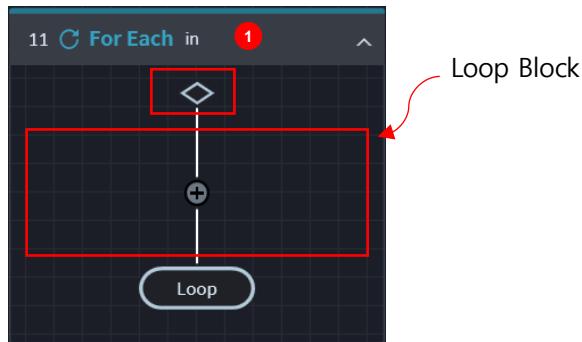
No.	설명
1	변수 선언 및 초기값을 Source Code 형태로 입력합니다. (입력형식: [DataType] [변수 명] = [초기값], Ex) int i = 0) · 오류 유발 조건: 조건문 및 증감문을 사용
2	종료 조건을 입력합니다. (입력형식: Boolean 또는 종료조건 Ex) i < 20) · 오류 유발 조건: 초기화 및 증감문 사용
3	증감식을 설정합니다. (입력형식: 증감문 Ex) i++ 또는 i=i+2 · 오류 유발 조건: 조건문 사용

## ■ For Each

스크립트의 조건문을 확인하고 반복 실행합니다. 조건은 Array(배열)에 저장된 Item 수만큼 반복을 수행합니다. 조건 판단문이 Block 상단에 위치하며, 하단에 Loop 아이콘이 있습니다. 접기/펼치기 기능을 제공합니다.



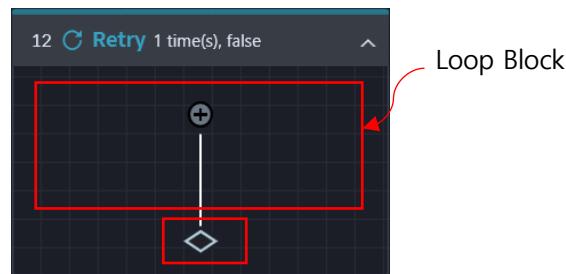
For의 동일하게 Loop Block 영역에 Event Item을 Drag & Drop하여 추가하고 조건에 맞는 동안 반복하여 실행할 수 있습니다. 접은 상태에서는 Loop Block은 표시되지 않습니다.



No.	설명
1	For Each 변수 명을 선언합니다. (입력형식: [DataType] [변수명]) · 정상 동작: Array 및 List 타입으로 선언된 변수의 Type을 선언문 형식으로 입력합니다. > Alias: Object, 수행할 Object 타입을 정의 Ex) string s · 오류 유발 조건: - 선언문 형식의 입력형식이 아닌 경우 - 선언된 DataType이 지정된 Array나 List의 변수유형과 다른 경우
2	Array 변수 명을 지정합니다. (Type: StringArray, IntArray, ObjectArray) > Array: List, 수행할 List 타입을 정의합니다.
3	Loop 카운트 횟수를 지정합니다.

## ■ Retry

특정 기능들이 지정한 조건에 따라 횟수만큼 반복적으로 수행할 수 있도록 합니다.



Loop Block 영역에 Event Item을 Drag & Drop하여 추가하고 조건에 맞는 동안 반복하여 실행할 수 있습니다. 접은 상태에서는 Loop Block은 표시되지 않습니다.

Do While과 동일하지만, Condition, Retry count, Retry interval을 사용할 수 있습니다.

Title Name 옆에 UI에 {count} time(s), {condition} 은 표시하지만 Interval은 표시하지 않습니다.



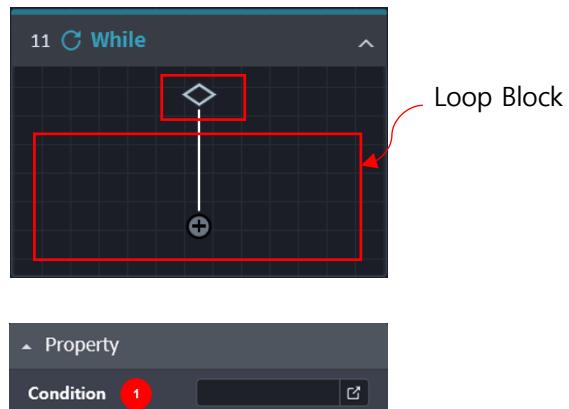
No.	설명
1	프로세스 반복 실행을 멈춰야 하는 판단문을 입력합니다. (Type: Boolean 조건문)
2	반복할 Count를 지정합니다. (Type: Int )
3	Interval을 millisecond 단위로 지정합니다.

## ■ While

특정 조건에 충족되는 동안 반복적으로 프로세스를 수행합니다.



Loop Block 영역에 Event Item을 Drag & Drop하여 추가하고 조건에 맞는 동안 반복하여 실행할 수 있습니다. 접은 상태에서는 Loop Block은 표시되지 않습니다.



No.	설명
1	프로세스 반복 실행을 멈춰야 하는 판단문을 입력합니다. (Type: 조건문)

## 5. Event Item

Event Item들은 업무 자동화를 위하여 컴퓨터가 이해할 수 있도록 전문 프로그래머가 제작해 놓은 기능함수들로 라이브러리 형태로 제공합니다.

여기에서는 프로세스 탭과 태스크 탭에서 사용되는 Event Item에 대하여 설명합니다. Canvas에 대한 사용법에 대해서는 “**A.WORKS Studio V2.1.3 사용자 설명서의 4.3.1 Canvas**”를 참조하시기 바랍니다.

### @@ 주의 @@

Event Item의 입력 값 제약사항은 아래와 같습니다.

- 입력 값의 검증 결과는 디버그 및 실행 시 Output 패널에 표시됩니다.
- 입력 필드에서 지정한 형식대로 입력 되어야 합니다.

### @@ 주의 @@

Event Item은 대부분 태스크를 작성하는 데 사용하지만, Process 탭에서도 스크립트 작성 시 Event Item 탭에서 활성화 되는 다음의 일부 Event Item만은 사용이 가능합니다.

#### 1) Process 의 Control 카테고리 중 Write Log,

- > Write log는 프로세스의 실행 결과로 Output 영역의 출력 패널에 로그를 출력하는 기능입니다.  
상세 사용방법은 “**5.4.1 Control의 Write Log**”를 참조하십시오.

#### 2) Programming 의 Assign 카테고리의 Assign 과 CodeExpression 카테고리의 Code Expression

- > Assign은 사용자가 직접 변수 명에 지정된 값을 할당하는 기능으로 프로세스에서도 사용 가능합니다..  
상세 사용방법은 “**5.5.1 Assign**”을 참조하십시오.  
> Code Expression은 사용자가 직접 C# 코드를 입력할 수 있는 기능으로 프로세스에서도 사용 가능합니다.. 상세 사용방법은 “**5.5.2 Code Expression**”을 참조하십시오.

#### 3) System의 Dialog 카테고리의 Message Yes No

- > Message Yes No는 프로세스 탭에서 Yes/No 버튼을 가진 Message Box를 생성할 때 사용합니다.  
상세 사용방법은 “**5.6.4 Dialog의 Message Yes No**”를 참조하십시오.

#### 4) WorkCenter 의 Work 카테고리의 Start Work

- > Start Work 는 Studio 와 WorkCenter 간의 프로세스 실행을 위한 기능으로 프로세스 탭에서만 사용이 가능한 Event Item 입니다.  
상세한 사용방법은 “**5.8.3 Work 의 Start Work**”를 참조하십시오.

이들은 프로세스와 태스크간 이동도 가능합니다 이들을 제외한 다른 모든 Event Item 은 프로세스 탭에서는 사용할 수 없습니다.

다음은 태스크 탭에서 사용할 수 있는 Event Item을 설명합니다.

## 5.1 Application

A.WORKS와 연동되는 일반 업무용 Application의 사용을 위한 기능들을 제공합니다.

Application은 4개의 카테고리로 구성됩니다.

- Browser
- Excel
- Mail
- Window

### 5.1.1 Browser

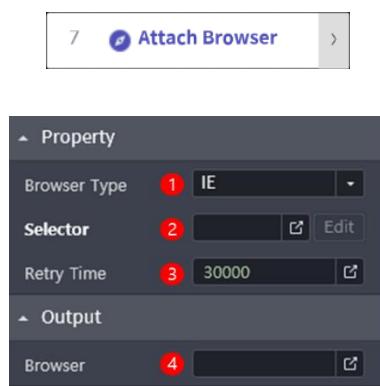
Browser를 제어하는 관련 기능을 제공합니다.

Browser는 다음과 같은 Event Item으로 구성되어 있습니다.

- Attach Browser
- Close Tab
- Get Attribute
- Go Back
- Go Forward
- Navigate To
- Open Browser

#### ■ Attach Browser

Attach할 Browser를 선택합니다. (Indicator 제공).

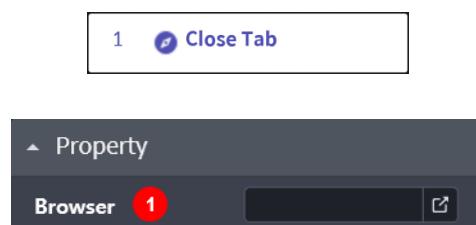


No.	설명
1	열고자 하는 Browser 타입을 선택합니다. (지원Type: IE, Chrome)
2	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집 가능하며 Object Editor 버튼인

	[Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우
3	객체 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)
4	Browser변수 명을 입력합니다.

## ■ Close Tab

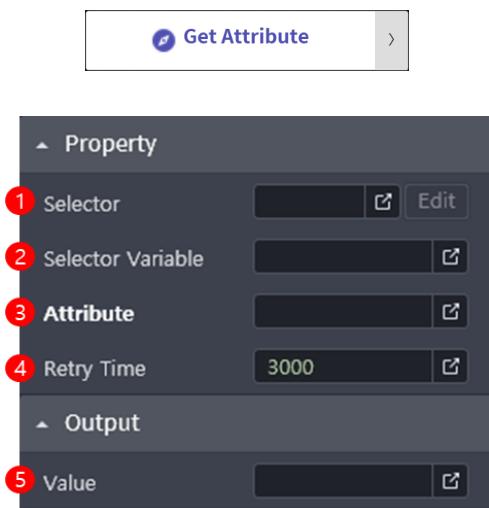
Close Tab에서는 원하는 Browser 내 탭을 닫을 수 있습니다.



No.	설명
1	Browser변수 명을 입력합니다.

## ■ Get Attribute

Get Attribute에서는 선택된 Object의 Attributes 들 중 하나의 값을 얻을 수 있습니다. (Indicator 제공).



No.	설명
1	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우

2	Selector 변수만 입력할 수 있습니다. 단, 1번 또는 2번중 한가지만 입력할 수 있습니다.
3	원하는 Attribute의 이름을 문자로 입력합니다. - 오류 정책: Selector에 기술된 Attribute를 찾지 못했을 경우
4	객체 또는 이미지 인식 시도 Retry Time을 입력합니다. ※ Studio에서 입력하는 Time은 모두 입력 Type: Int에, Millisecond(ms) 단위로 지정합니다. Time 종류: Interval Time, Wait Time, Delay Time, Retry Time, Limit Time이 있습니다.
5	지정한 Attribute를 저장할 변수 명을 선택합니다. (Type: String)

## ■ Go Back

Go Back에서는 선택한 Browser에서 Go Back하고자 하는 Browser로 가기를 실행합니다.



No.	설명
1	Go Back을 실행하고자 하는 Browser 변수 명을 입력합니다.

## ■ Go Forward

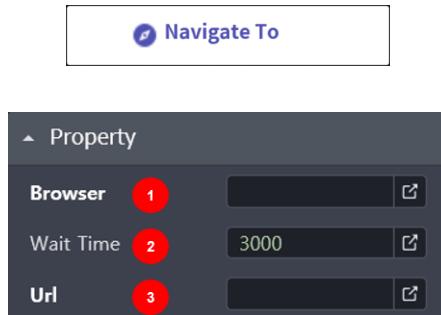
Go Forward에서는 선택한 Browser에서 Go Forward하고자 하는 Browser로 가기를 실행합니다.



No.	설명
1	Go Forward를 실행하고자 하는 Browser 변수 명을 입력합니다.

## ■ Navigate To

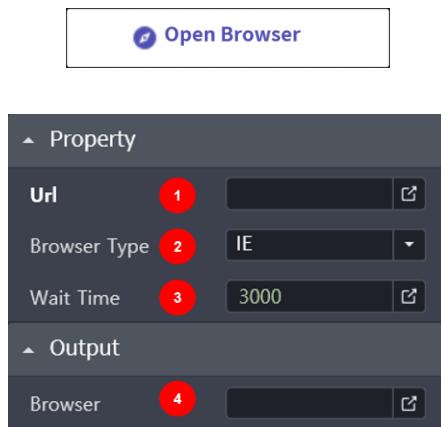
지정하는 Browser 페이지를 열 수 있습니다.



No.	설명
1	Browser 변수 명을 입력합니다.
2	Navigate To 수행 후 페이지 로딩 시까지의 Wait Time입니다. (Type: Int, 단위: ms)
3	Browser 실행 시 접속하고자 하는 URL을 입력합니다. 사용자가 직접 URL을 입력하거나, String 변수 명을 가져올 수도 있습니다.

## ■ Open Browser

Open Browser에서는 원하는 Browser 페이지를 열 수 있습니다.



No.	설명
1	Browser 상에서 열고자 하는 URL을 입력합니다. (Type: String)
2	열고자 하는 Browser 타입을 선택합니다. (지원Type: IE, Chrome)
3	Browser를 열고 페이지 로딩까지 걸리는 Wait Time을 입력합니다. (Type: Int, 단위: ms)
4	Browser 변수 명을 입력합니다.

## 5.1.2 Excel

Excel은 Microsoft사의 Excel을 Handling하기 위한 관련 기능을 모아 놓은 것입니다.

Excel은 다음과 같은 Event Item으로 구성되어 있습니다.

- Activate Cell
- Activate Sheet
- Attach Workbook
- Close Workbook
- Copy Sheet
- Copy Sheet Range
- Delete Sheet
- Excel Scope
- Get All Sheet Names
- Get Row Count
- Get Sheet Count
- Get Sheet Name
- Insert Sheet
- Move Sheet
- Read Cell
- Read Sheet Range
- Rename Sheet
- Run Excel Macro
- Save Workbook
- Start Workbook
- Start Workbook Secure
- Write Cell
- Write Sheet Range

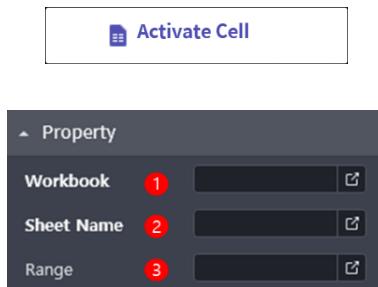
### @@ 주의 @@

※ Excel 관련 Event Item 사용에 대한 제약사항

1. Excel에 관련된 Event Item은 개별 실행을 지원하지 않습니다.
2. Excel을 사용하기 위해서는 반드시 Start Workbook으로 실행한 Workbook이어야 하며, Excel Event Item에서 사용하는 변수 Type도 Workbook이어야 합니다.
3. Start Workbook으로 열 수 있는 Excel 파일의 용량은 100MB로 제한합니다.
4. v2.1.3 기준으로 Workbook이 종료되지 않는 문제를 방지하기 위해 Process가 종료될 때 일괄적으로 Workbook이 종료됩니다.

## ■ Activate Cell

Open된 Excel의 Sheet 내 Cell을 활성화(Activate)합니다.



No.	설명
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	활성화를 실행하고자 하는 Sheet이름을 지정합니다. (Type: String)
3	활성화 하고자 하는 Cell의 범위를 Excel의 Cell 주소형식으로 표기합니다. (Type: String, ex. "A3:D6", "C1", "A:A" 등)

## ■ Activate Sheet

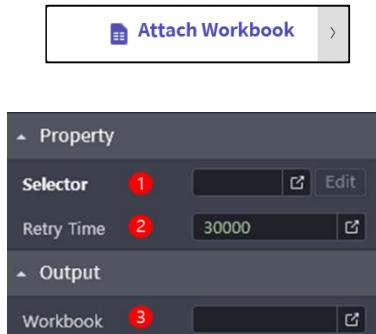
Open된 Excel의 Sheet를 활성화(Activate)합니다



No.	설명
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	활성화를 실행하고자 하는 Sheet이름을 지정합니다. (Type: String)

## ■ Attach Workbook

Start Workbook으로 실행한 Excel 파일을 지정합니다. (Indicator 제공).



No.	설명
1	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집 가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우
2	객체 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)
3	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.

### @@ 주의 @@

※ Attach Workbook의 사용자가 정의하는 Excel 파일명은 다음과 같은 제약이 있습니다.

반드시 확인하시고 파일명을 입력하여 주십시오..

※ Excel 파일명에서 제외되는 규칙 명시: 파일명에 다음의 Keyword가 포함되어 사용은 불가능합니다.

- 1) " - Excel" Ex) AWORKS\_테스트엑셀- Excel.xls
- 2) " - Microsoft Excel"
- 3) "(제품인증실패)"
- 4) "(사용 허가되지 않은 제품)"
- 5) "(Product Activation Failed)"
- 6) " - 호환성 모드
- 7) "[호환모드]"
- 8) "[Compatibility Mode]"
- 9) "[읽기전용]"
- 10) " - 읽기 전용
- 11) "[Read-Only]"

※ Attach Workbook 의 Indicator 로 Excel 의 타이틀을 불러왔으나, 불필요한 값들이 추가 입력되는 경우에는 사용자가 이를 제거하거나, 와일드카드(\*)로 대처해야 합니다.

#### [예시]

열려있는 파일의 타이틀이 아래와 같다고 가정하면,

파일명: doc2.xlsx - 읽기 전용 - Excel

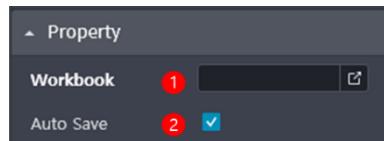
사용자 입력	정규식 변환	Attach 가능여부
doc2*	^doc2.*	가능 (doc2.xlsx 과 정규식(Regexp) 매치)
doc2	^doc2	불가 (doc2 파일이 열려있지 않음.)
doc2.xlsx	^doc2.xlsx	가능 (사용자 입력과 같은 파일이 열려있기 때문)
doc2.xlsx*	^doc2.xlsx.*	가능 (doc2.xlsx 과 정규식(Regexp) 매치)
doc2.xlsx - 읽기 전용 - Excel	^doc2.xlsx - 읽기 전용 - Excel	불가 (doc2.xlsx 과 정규식(Regexp) 언 매치)

※ 상기 파일명을 사용한 경우 다음과 같은 오류 메시지를 출력합니다.

- > 한글: 엑셀 파일이 존재하지 않습니다! 엑셀 파일 이름 제약을 확인하세요!
- > 영어: Excel file does not exist, please check the Excel file name constraints.

## ■ Close Workbook

Start Workbook으로 실행한 Excel 파일을 종료합니다.

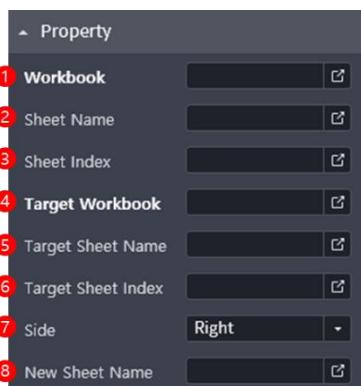


No.	설명
1	종료하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	종료 시 자동으로 저장을 수행하고자 할 경우 선택

## ■ Copy Sheet

Open된 Excel에 존재하는 Sheet를 복사해 삽입합니다.

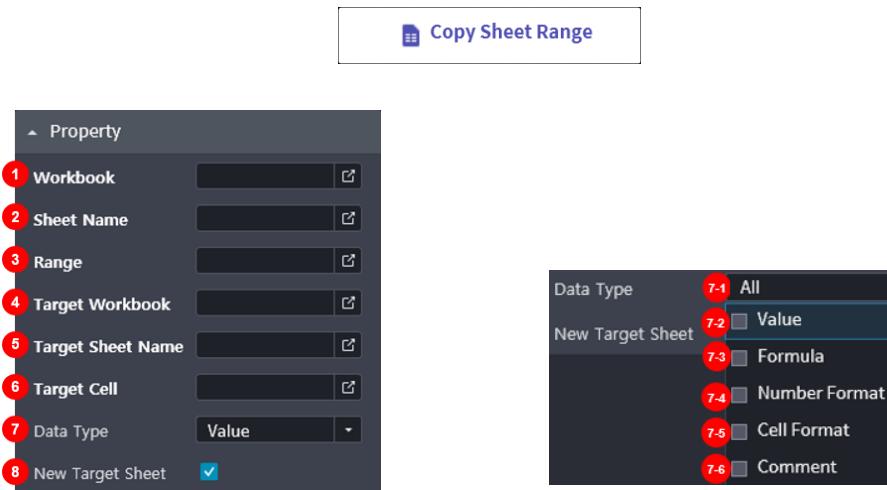
Copy Sheet



No.	설명
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	복사하고자 하는 원본 Sheet 이름을 지정합니다. (Type: String)
3	복사하고자 하는 원본 Sheet Index를 지정합니다. (Type: Int)
4	복사한 시트가 위치할 목적지 Workbook지정합니다.
5	삽입하고자 하는 목적지 Sheet 이름을 지정합니다. (Type: String)
6	삽입하고자 하는 목적지 Sheet Index를 지정합니다. (Type: Int)
7	목적지 Sheet를 기준으로 삽입하고자 하는 위치를 좌측 또는 우측으로 선택합니다.
8	복사된 Sheet를 삽입 시 지정할 새로운 Sheet의 이름을 지정합니다. (Type: String)

## ■ Copy Sheet Range

Excel의 지정된 영역의 데이터를 다른 Excel에 영역을 지정하여 복사하여 붙여 넣는 기능입니다.



No.	설명
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	복사하려는 Cell이 소속된 Sheet 이름을 지정합니다. (Type: String)
3	복사하려는 Cell의 주소를 Excel 주소형식으로 표기합니다. (Type: String, ex. "C1", "A3" 등)
4	대상 Workbook을 지정합니다.
5	대상 워크시트를 지정합니다.
6	대상 데이터 시작 셀을 지정합니다.
7	복사할 데이터의 타입을 지정합니다. > All : 복사할 데이터의 값, 수식, 값 형식, 셀 형식, 메모를 복사 합니다. [7-1] > Value : 복사할 데이터의 값을 복사 합니다. [7-2] > Formula : 복사할 데이터의 수식을 복사합니다. [7-3] > Number Format : 복사할 데이터의 값 형식을 복사합니다. [7-4] > Cell Format : 복사할 데이터의 셀 형식을 복사합니다. [7-5] > Comment : 복사할 데이터의 메모를 복사합니다. [7-6]
8	Excel 문서에 지정된 Sheet가 없을 경우 신규로 생성할 것인지 선택합니다. > Checked: 새로운 Sheet를 생성합니다. > Unchecked: 해당 Sheet 없음으로 Error가 발생합니다.

## ■ Delete Sheet

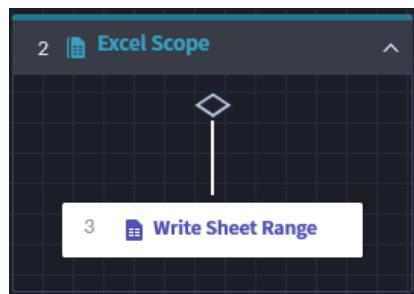
Open된 Excel에 존재하는 Sheet를 삭제합니다

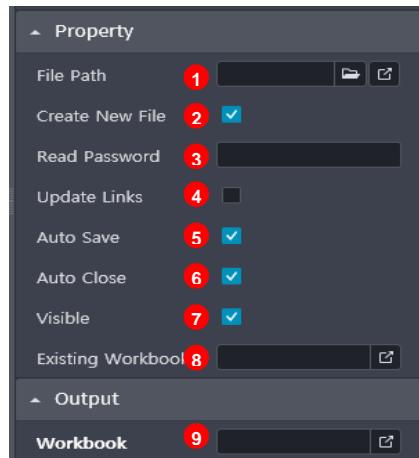


No.	설명	
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.	
2	삭제할 Sheet 이름을 지정합니다. (Type: String)	※ 2 또는 3 항목 중 1개만 선택입력 가능
3	삭제할 Sheet Index를 지정합니다. (Type: Int)	

## ■ Excel Scope

Excel의 열기, 쓰기, 저장, 닫기에 대해서 동일한 Workbook 객체를 사용하도록 범위를 제한하여, Start Workbook, Close Workbook을 하지 않아도 되어 편리합니다.

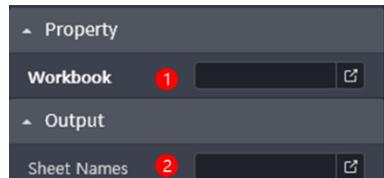




No.	설명
1	실행하고자 하는 Excel 파일의 경로를 입력합니다. (Type: String)
2	File Path에 지정한 위치에 Excel파일이 없을 경우 새 파일 생성여부를 체크합니다. > Checked: 새로운 파일을 생성합니다. > Unchecked: 파일 없음으로 Error가 발생합니다.
3	Excel 파일에 읽기 Password가 설정되어 있을 경우 입력합니다. (Type: String)
4	Excel 문서내에 외부에 연결된 데이터가 있는 경우, 업데이트 여부를 선택합니다. 단, 대상 Excel 문서 (외부 연결된 데이터가 있는 문서) 가 없을 경우 오류가 발생합니다.
5	Scope에서 나갈 때 문서 저장 여부를 나타냅니다.
6	Scope에서 나갈 때 문서 닫기 여부를 나타냅니다.
7	작업을 숨겨서 진행할지 여부를 나타냅니다.
8	Scope 외부에서 이미 열려져있는 Excel 문서를 나타냅니다.
9	실행한 Workbook의 객체를 저장할 Workbook 변수 명을 입력합니다. (Type: Workbook)

## ■ Get All Sheet Names

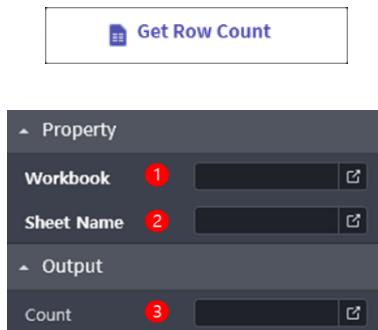
Open된 Excel 내 모든 Sheet이름을 불러옵니다.



No.	설명
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	불러온 Sheet이름을 저장할 변수 명을 지정합니다. (Type: Stringrray 타입 변수)

## ■ Get Row Count

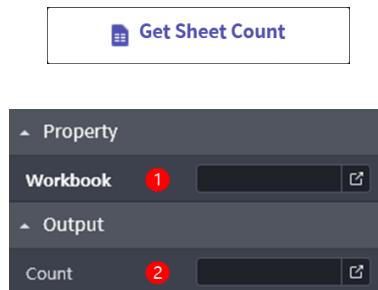
Open된 Excel의 데이터 Row 갯수를 가져옵니다.



No.	설명
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	대상 Cell이 소속된 Sheet 이름을 지정합니다. (Type: String)
3	불러온 Row의 Count값을 저장할 변수 명을 지정합니다. (Type: Int)

## ■ Get Sheet Count

Open된 Excel에 존재하는 Sheet의 개수를 반환합니다.



No.	설명
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	반환되는 Sheet의 개수를 저장할 변수 명을 지정합니다. (Type: Int)

## ■ Get Sheet Name

Open된 Excel 내 특정 위치(Index)의 Sheet이름을 불러옵니다.

Get Sheet Name

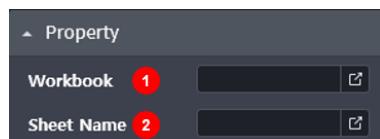


No.	설명
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	불러오고자 하는 Sheet의 Index를 지정합니다. (Type: Int)
3	불러온 Sheet 이름을 저장할 변수 명을 지정합니다. (Type: String)

## ■ Insert Sheet

Open된 Excel에 새로운 Sheet를 추가합니다.

Insert Sheet

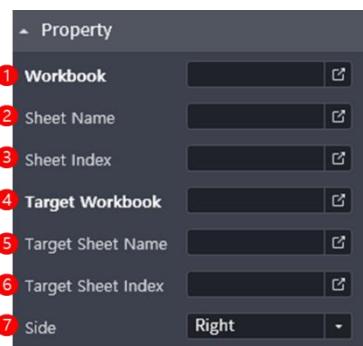


No.	설명
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	추가하고자 하는 Sheet의 이름을 입력합니다. ※ 추가되는 Sheet는 가장 우측 Tab에 생성합니다.

## ■ Move Sheet

Open된 Excel에 존재하는 Sheet의 위치를 변경합니다.

Move Sheet

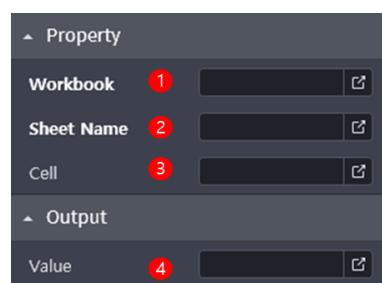


No.	설명	
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.	
2	이동하고자 하는 원본 Sheet 이름을 지정합니다. (Type: String)	2 또는 3 항목
3	이동하고자 하는 원본 Sheet Index를 지정합니다. (Type: Int)	1개만 선택입력 가능
4	이동하고자 하는 목적지 Workbook을 지정합니다.	
5	이동하고자 하는 목적지 Sheet 이름을 지정합니다. (Type: String)	5 또는 6 항목
6	이동하고자 하는 목적지 Sheet Index를 지정합니다. (Type: Int)	1개만 선택입력 가능
7	목적지 Sheet를 기준으로 삽입하고자 하는 위치를 좌측 또는 우측으로 선택합니다.	

## ■ Read Cell

Open된 Excel의 특정 셀에 저장된 값을 불러옵니다.

Read Cell

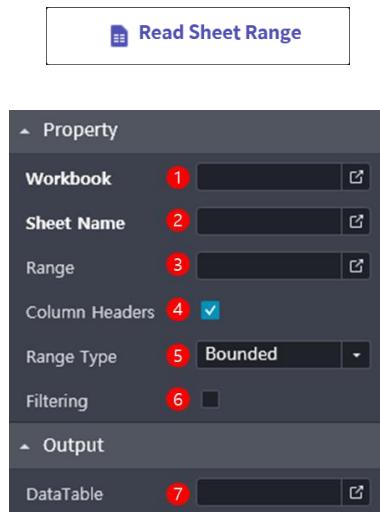


No.	설명
1	실행하고자 하는 Workbook 지정합니다.

	※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	대상 Cell이 소속된 Sheet 이름을 지정합니다. (Type: String)
3	대상 Cell의 주소를 Excel 주소형식으로 표기합니다. (Type: String, ex. "C1", "A3" 등)
4	불러온 Cell의 Data를 저장할 변수 명을 지정합니다. (Type: String)

## ■ Read Sheet Range

Open된 Excel 내 지정된 영역의 Data를 DataTable 변수로 저장합니다.



No.	설명
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	대상 Cell이 소속된 Sheet 이름을 지정합니다. (Type: String)
3*	Cell의 범위를 Excel 주소형식으로 표기합니다. (Type: String, ex. "A3:D6", "C1", "A:A" 등)
4	Header 정보를 1st Row에 포함여부를 지정합니다. > Checked: 첫 번째 Row 정보를 Column Header로 인식 > Unchecked: 첫 번째 Row 정보를 Data로 인식
5	데이터 읽기 방식을 선택합니다. > Bounded: 연속되는 데이터 읽기 > All: 전체 데이터 읽기
6	Filtering 옵션을 지정합니다. > Checked: 숨김 데이터를 제외한 데이터 읽기 > Unchecked: 모든 데이터 읽기
7	불러온 Cell의 Data를 저장할 변수 명을 지정합니다. (Type: DataTable)

\* Cell Range 입력 형식에 따른 동작

- Cell 형식의 주소일 경우: 시작점으로 인식 (Row/Column 크기를 자동으로 인식해 해당 범위를 저장)
- Range 형식의 주소일 경우: 범위로 인식 (해당 영역만 저장)
- Null(공백) or Nothing("")": 시작점을 A1으로 인식

## ■ Rename Sheet

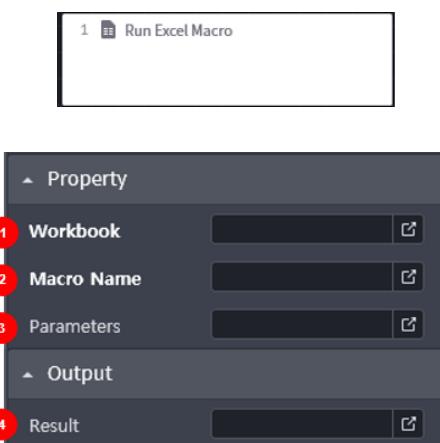
Open된 Excel에 존재하는 Sheet의 이름을 변경합니다.



No.	설명
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	이름을 변경하고자 하는 Sheet 이름을 지정합니다. (Type: String)
3	이름을 변경하고자 하는 Sheet Index를 지정합니다. (Type: Int)
4	지정된 Sheet의 이름을 변경할 새로운 이름을 입력합니다. (Type: String)

## ■ Run Excel Macro

Excel의 Macro의 인자 값을 전달 합니다. Macro 실행 이후 리턴 값을 반환 받습니다.



No.	설명
1	Workbook : 실행하고자 하는 Workbook의 변수명을 입력합니다. Workbook은 Workbook형식의 변수입니다.
2	Macro Name : 실행하고자 하는 Macro의 이름을 입력 합니다. Macro Name은 String형식의 값입니다. 텍스트 또는 String형의 변수가 입력이 가능합니다.
3	Parameters : 실행하고자 하는 Macro의 인자 값을 입력 합니다. Parameters는 ObjectArray형식의 값입니다.
4	Result : Macro를 실행한 후 반환되는 값을 입력 받을 변수 명을 입력합니다. Result는 String, Int, Double, Bool, Date, Object형식의 변수입니다.

## ■ Save Workbook

Start Workbook으로 실행한 Excel 파일을 저장합니다.



No.	설명
1	실행하고자 하는 Workbook 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.

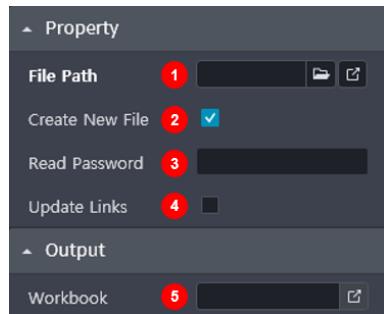
## ■ Start Workbook

A.WORKS에서 작업하고자 하는 Excel 파일을 실행합니다.

**@@ 주의 @@**

※ Start Workbook으로 열 수 있는 Excel 파일의 용량은 100MB로 제한합니다.



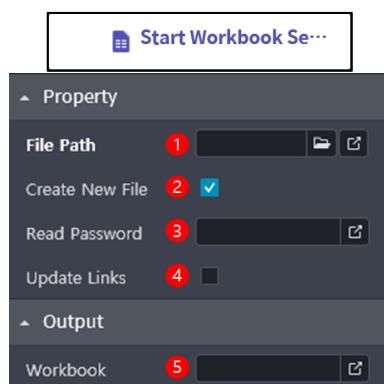


No.	설명
1	실행하고자 하는 Excel 파일의 경로를 입력합니다. (Type: String)
2	File Path에 지정한 위치에 Excel파일이 없을 경우 새 파일 생성여부를 체크합니다. > Checked: 새로운 파일을 생성합니다. > Unchecked: 파일 없음으로 Error가 발생합니다.
3	Excel 파일에 읽기 Password가 설정되어 있을 경우 입력합니다. (Type: String)
4	Excel 문서내에 외부에 연결된 데이터가 있는 경우, 업데이트 여부를 선택합니다. 단, 대상 Excel 문서 (외부 연결된 데이터가 있는 문서) 가 없을 경우 오류가 발생합니다.
5	실행한 Workbook의 객체를 저장할 Workbook 변수 명을 입력합니다. (Type: Workbook) > 이후 Excel관련 Event Item을 실행할 때 대상 Workbook을 지정해야 사용 가능합니다.

## ■ Start Workbook Secure

A.WORKS에서 작업하고자 하는 Excel 파일을 실행합니다.

※ Start Workbook과의 차이점은 암호화된 패스워드를 사용하는 것입니다. (Type: SecureString)

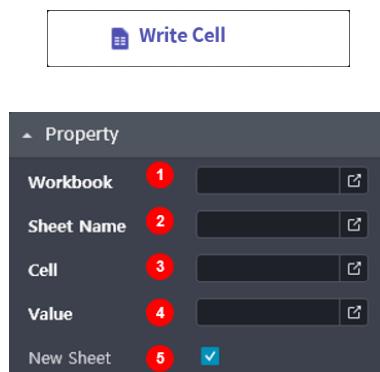


No.	설명
1	실행하고자 하는 Excel 파일의 경로를 입력합니다. (Type: String)
2	File Path에 지정한 위치에 Excel파일이 없을 경우 > Checked: 새로운 파일을 생성합니다. > Unchecked: 파일 없음으로 Error가 발생합니다.
3	Excel 파일에 읽기 Password 입력합니다. (암호화된 패스워드, Type: SecureString)

	<p>※ 정상적으로 작동하기 위해서는 반드시 암호화된 패스워드를 입력해야 합니다.</p> <p>※ SecureString 입력방식에 대해서는 “Text To SecureString” Event Item을 참조</p>
4	<p>Excel 문서내에 외부에 연결된 데이터가 있는 경우, 업데이트 여부를 선택합니다.</p> <p>단, 대상 Excel 문서 (외부 연결된 데이터가 있는 문서) 가 없을 경우 오류가 발생합니다.</p>
5	<p>실행한 Workbook의 객체를 저장할 Workbook 변수 입력합니다. (Type: Workbook)</p> <p>- 이후 Excel관련 Event Item을 실행할 때 대상 Workbook을 지정해야 사용 가능합니다.</p>

## ■ Write Cell

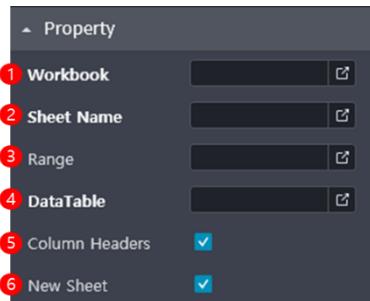
Open된 Excel의 특정 셀에 값을 저장합니다.



No.	설명
1	<p>실행하고자 하는 Workbook 지정합니다.</p> <p>※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.</p>
2	대상 Cell이 소속된 Sheet 이름을 지정합니다. (Type: String)
3	대상 Cell의 주소를 Excel 주소형식으로 표기합니다. (ex. "C1", "A3" 등)
4	Cell에 저장할 값을 지정합니다. (Type: String, "abcd" 형태의 Expression 또는 변수 명)
5	Excel 문서에 지정된 Sheet가 없을 경우 신규로 생성할 것인지 선택합니다.

## ■ Write Sheet Range

Excel에 영역을 지정하여 DataTable 형태의 데이터를 쓰는 기능입니다.



No.	설명
1	실행하고자 하는 Workbook을 지정합니다. ※ Start Workbook으로 실행한 Workbook이어야 하며, Type: Workbook입니다.
2	대상 Cell이 소속된 Sheet 이름을 지정합니다. (Type: String)
3	대상 Cell의 주소를 Excel 주소형식으로 표기합니다. (Type: String, ex. "C1", "A3" 등)
4	저장할 Data가 담긴 DataTable을 지정합니다. (Type: DataTable)
5	Header 정보를 1st Row에 포함여부를 지정합니다. > Checked: 첫 번째 행의 정보를 Column Header로 인식 > Unchecked: 첫 번째 행의 정보를 Data로 인식
6	Excel 문서에 지정된 Sheet가 없을 경우 신규로 생성할 것인지 선택합니다.

### 5.1.3 Mail

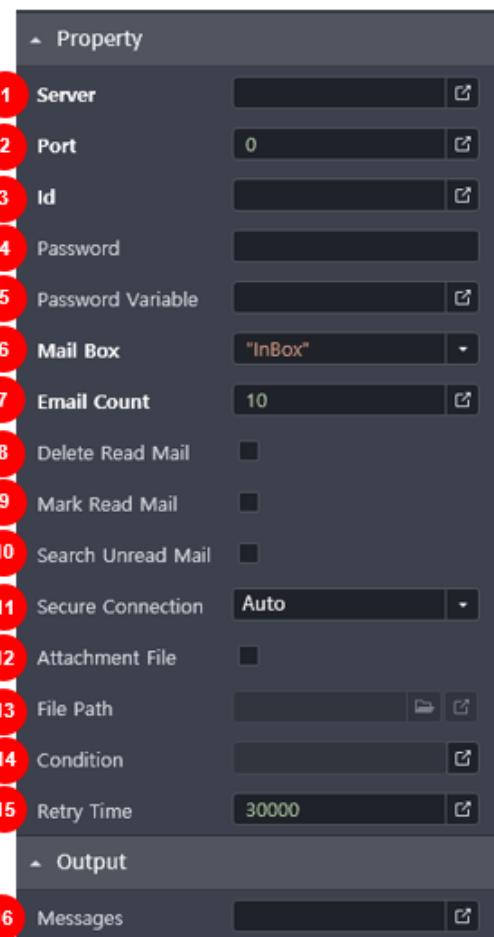
Mail은 Mail기능을 Handling하기 위한 관련된 기능들을 제공합니다.

Mail은 다음과 같은 Event Item으로 구성되어 있습니다.

- Receive IMAP Mail
- Receive POP3 Mail
- Send Mail
- Send Mail Secure

#### ■ Receive IMAP Mail

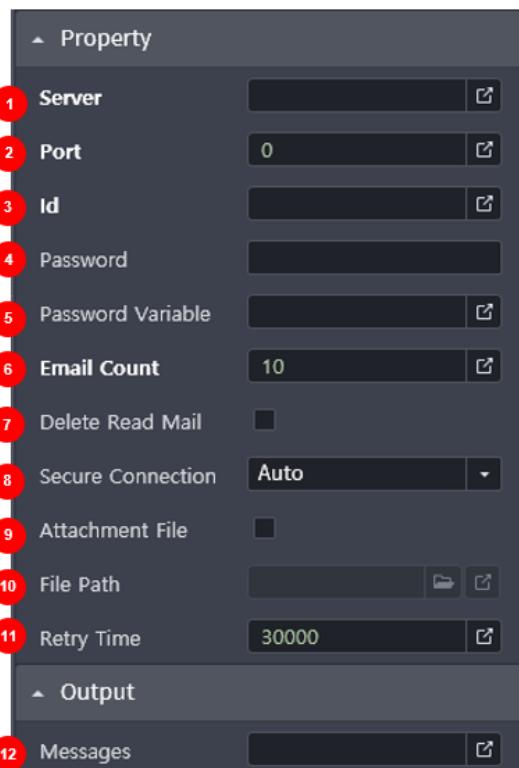
IMAP 프로토콜을 이용하여 메일 메시지를 가져옵니다. 메일의 첨부파일 다운로드 여부 설정에 따라서 지정된 폴더에 첨부파일을 다운로드 가능합니다. 검색 조건을 입력하여 해당 메시지를 검색하여 Messages 객체에 저장이 가능합니다.



No.	설명
1	Server : 메일 서버의 호스트명을 입력합니다. Server는 String형식의 값입니다.
2	Port : 메일 서버의 포트를 입력합니다. Port는 UInt형식의 값입니다.
3	Mail Box : 메일을 읽어 올 메일함을 입력합니다. Mail Box는 Combo, String형식의 값입니다.
4	Id : 로그인 할 메일 계정을 입력합니다. Id는 String형식의 값입니다.
5	Password : 로그인 할 메일 계정의 Password를 입력합니다. Password는 String형식의 값입니다.
6	Password Variable : 읽기 비밀번호 정보를 가지고 있는 변수명을 입력합니다. Password Variable은 String형식의 변수입니다. Password와 Password Variable중 하나만 입력하세요.
7	Email Count : 받아올 메일의 개수를 입력합니다. Email Count는 UInt형식의 값입니다.
8	Delete Read Mail : 읽은 메시지를 삭제할 지 여부를 결정합니다. Delete Read Mail은 Bool형식의 값입니다.
9	Mark Read Mail : 읽은 메일을 읽음 표시할 지 여부를 결정합니다. Mark Read Mail은 Bool형식의 값입니다.
10	Search Unread Mail : 읽지 않은 메일만 검색할 지 여부를 결정합니다. Search Unread Mail은 Bool 형식의 값입니다.
11	Secure Connection : 보안 연결 옵션을 입력합니다. Secure Connection은 Combo형식의 값입니다.
12	Attachment File : 첨부 파일을 다운 받을 지 여부를 결정합니다. 체크를 하면 File Path가 활성화됩니다. Attachment File은 Bool형식의 값입니다.
13	File Path : 첨부 파일을 다운 받을 경로를 입력합니다. File Path는 String형식의 값입니다.
14	Condition : 메일을 검색할 조건을 입력합니다. 메일을 검색할 조건을 입력합니다. 오늘 날짜 메일을 검색할 경우 시작일을 오늘 날짜로 종료일은 내일 날짜로 선택합니다. >보낸사람: 메일을 보낸사람 성함을 입력합니다. >받는사람 : 메일을 받은사람 성함을 입력합니다 >내용 : 제목, 내용, 제목 + 내용 옵션을 선택하여, 검색 내용을 입력합니다. >기간 : 직접 입력, 1주일, 3개월, 6개월, 1년 옵션을 선택하여 내용을 입력 합니다.. ※ 기간을 검색할 때 유의사항을 안내 드립니다. - 21년 6월 14일을 도착한 메일을 검색할 경우 검색일 입력 예제는 아래와 같습니다. 21년 6월 14일 00시부터 21년 6월 15일 00시 까지 검색합니다.
15	Retry Time : 메일 서버에 연결 재시도 할 시간을 입력합니다. Retry Time은 UInt형식의 값입니다.

## ■ Receive POP3 Mail

POP3 프로토콜을 이용하여 메일 메시지를 가져옵니다. 메일의 첨부파일 다운로드 여부 설정에 따라서 지정된 폴더에 첨부파일을 다운로드 가능합니다.



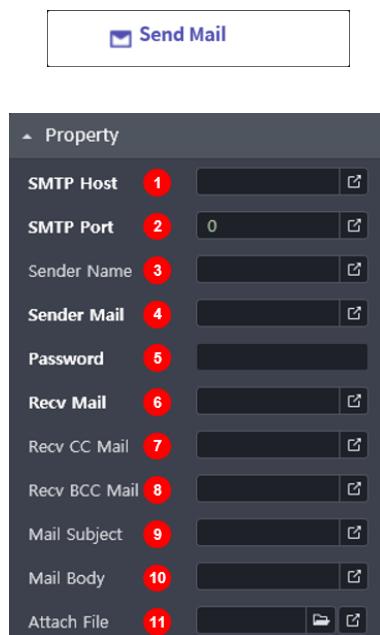
No.	설명
1	Server : 메일 서버의 호스트명을 입력합니다. Server는 String형식의 값입니다.
2	Port : 메일 서버의 포트를 입력합니다. Port는 UInt형식의 값입니다.
3	Id : 로그인 할 메일 계정을 입력합니다. Id는 String형식의 값입니다.
4	Password : 로그인 할 메일 계정의 Password를 입력합니다. Password는 String형식의 값입니다.
5	Password Variable : 읽기 비밀번호 정보를 가지고 있는 변수명을 입력합니다. Password Variable은 String형식의 변수입니다. Password와 Password Variable중 하나만 입력하세요.
6	Email Count : 받아올 메일의 개수를 입력합니다. Email Count는 UInt형식의 값입니다.
7	Delete Read Mail : 읽은 메시지를 삭제할지 여부를 결정합니다. Delete Read Mail은 Bool형식의 값입니다.
8	Secure Connection : 보안 연결 옵션을 입력합니다. Secure Connection은 Combo형식의 값입니다.

9	Attachment File : 첨부 파일을 다운 받을 지 여부를 결정합니다. 체크를 하면 File Path가 활성화됩니다. Attachment File은 Bool형식의 값입니다.
10	File Path : 첨부 파일을 다운 받을 경로를 입력합니다. File Path는 String형식의 값입니다.
11	Retry Time : 메일 서버에 연결 재시도 할 시간을 입력합니다. Retry Time은 UInt형식의 값입니다.
12	Messages : 받아온 메일을 저장할 변수를 입력합니다. Messages는 MailMessage Array형식의 값입니다.

## ■ Send Mail

Send Mail에서는 SMTP(Simple Mail Transfer Protocol)를 활용하여 메일을 보낼 수 있습니다.

※ Studio 2.0에서는 다중 파일을 선택하여 첨부하는 기능이 추가되었습니다.



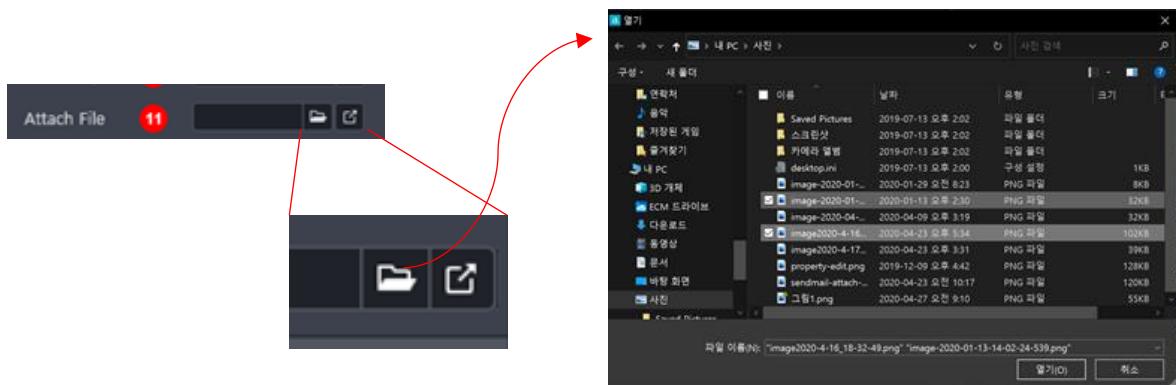
No.	설명
1	메일 서비스의 Host를 입력합니다. (Type: String)
2	메일 서비스의 Port를 입력합니다. (Type: Int)
3	보내는 사람을 입력합니다. (Type: String)
4	보내는 메일 주소를 입력합니다. (Type: String)
5	보내는 사람의 비밀번호를 입력합니다. (Type: String)
6	받는 메일을 입력합니다. (Type: String)
7	받는 참조 메일을 입력합니다. (Type: String)
8	받는 숨김 참조 메일주소를 입력합니다. (Type: String)
9	메일 제목을 입력합니다. (Type: String)
10	메일 내용을 입력합니다. (Type: String)

11 첨부 파일 경로를 입력합니다. (Type: String)

※ Studio 2.0에서는 다중 파일을 첨부하는 기능이 추가되었습니다. (하기 설명 참조)

### @@ 주의 @@

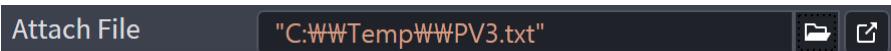
※ 다중 파일 첨부 방법



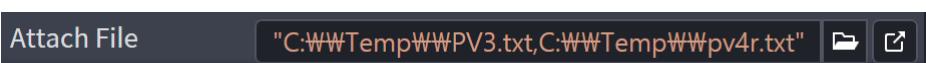
Attach File의 항목에서 직접 입력하거나, 폴더열기 아이콘을 클릭하여 여러 개의 파일 선택하거나 PopupCodeEdit 아이콘을 클릭하여 팝업 창에 파일명을 직접 입력할 수 있습니다.

※ Attach File 입력 필드에 직접 입력 또는 PopupCodeEdit 아이콘을 클릭하여 입력 시에는 반드시 파일 사이를 Comma()로 구분하고, 전체 String에 따옴표 처리하여야 합니다.

> 단일 파일 선택 시



> 복수 파일 선택 시



※ 받는 메일, 참조 메일에서 여러 값을 직접 입력할 때는 구분자로 "\*"를 사용합니다. Ex) aa\*bb

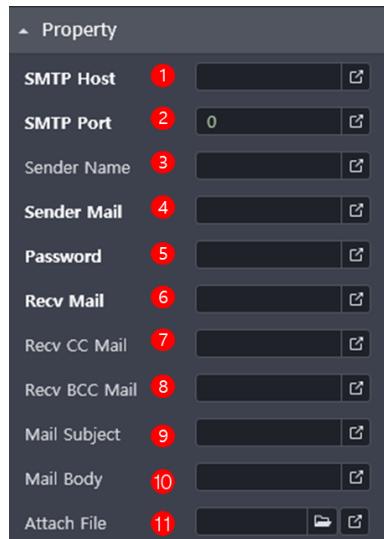
## ■ Send Mail Secure

Send Mail Secure에서는 SMTP(Simple Mail Transfer Protocol)를 활용하여 메일을 보낼 수 있습니다.

※ Send Mail과의 차이점은 암호화된 패스워드를 사용하는 것입니다. (Type: SecureString)

※ Studio 2.0에서는 다중 파일을 선택하여 첨부하는 기능이 추가되었습니다.

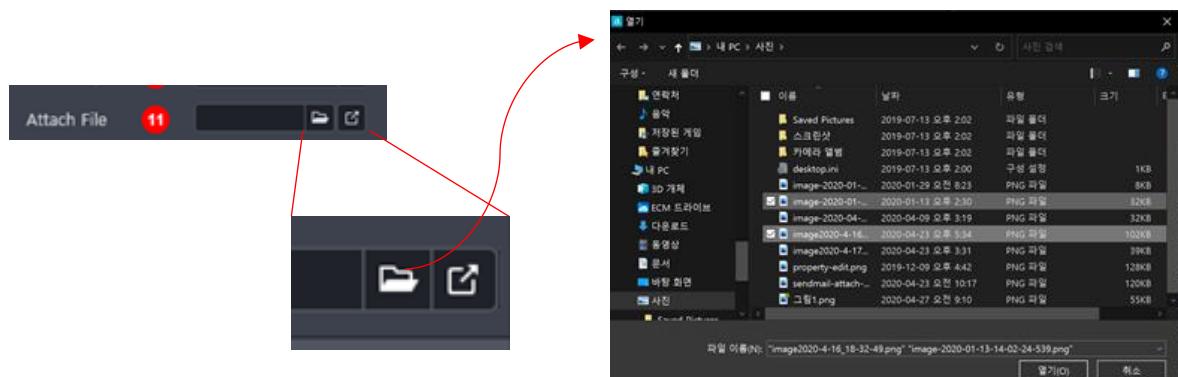
Send Mail Secure



No.	설명
1	메일 서비스의 Host를 입력합니다. (Type: String)
2	메일 서비스의 Port를 입력합니다. (Type: Int)
3	보내는 사람을 입력합니다. (Type: String)
4	보내는 메일 주소를 입력합니다. (Type: String)
5	보내는 사람의 비밀번호를 입력합니다. (Type: SecureString) ※ 정상적으로 작동하기 위해서는 반드시 암호화된 패스워드를 입력해야 합니다. ※ SecureString 입력방식에 대해서는 "Text To SecureString" Event Item 참조하시기 바랍니다.
6	받는 메일을 입력합니다. (Type: String)
7	참조 메일을 입력합니다. (Type: String)
8	숨김 참조 메일주소를 입력합니다. (Type: String)
9	메일 제목을 입력합니다. (Type: String)
10	메일 내용을 입력합니다. (Type: String)
11	첨부 파일 경로를 입력합니다. (Type: String) ※ Studio 2.0에서는 다중 파일을 첨부하는 기능이 추가되었습니다. (하기 설명 참조)

## @@ 주의 @@

※ 다중 파일 첨부 방법



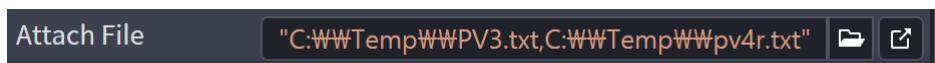
Attach File의 항목에서 직접 입력하거나, 폴더열기 아이콘을 클릭하여 여러 개의 파일 선택하거나 PopupCodeEdit 아이콘을 클릭하여 팝업 창에 파일명을 직접 입력할 수 있습니다.

※ Attach File 입력 필드에 직접 입력 또는 PopupCodeEdit 아이콘을 클릭하여 입력 시에는 반드시 파일 사이를 Comma()로 구분하고, 전체 String에 따옴표 처리하여야 합니다.

> 단일 파일 선택 시



> 복수 파일 선택 시



※ 받는 메일, 참조 메일에서 여러 값을 직접 입력할 때는 구분자로 "\*"를 사용합니다. Ex) aa\*bb

## 5.1.4 Window

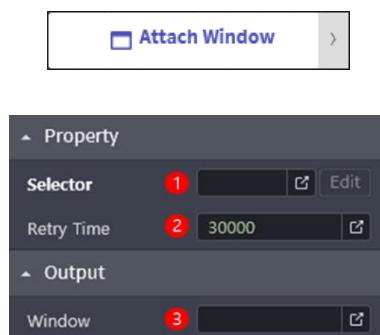
화면의 Window 관련 기능들을 제공합니다.

Window는 다음과 같은 Event Item으로 구성되어 있습니다.

- Attach Window
- Close Window
- Maximize Window
- Minimize Window
- Resize Window

### ■ Attach Window

현재 실행중인 Window중 원하는 Window를 첨부합니다. (Indicator 제공).



No.	설명
1	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집 가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우.
2	객체 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)
3	Attach할 Window 변수 명을 입력합니다. (Type: Window 변수)

## ■ Close Window

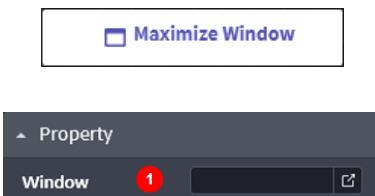
현재 실행중인 Window중 원하는 Window를 닫습니다.



No.	설명
1	닫을 Window 변수 명을 입력합니다. (Type: Window 변수)

## ■ Maximize Window

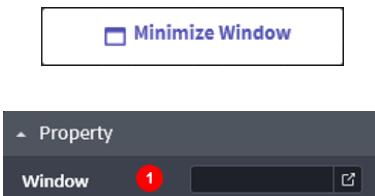
현재 실행중인 Window중 원하는 Window의 크기를 최대화 합니다



No.	설명
1	최대화할 Window 변수 명을 입력합니다. (Type: Window 변수)

## ■ Minimize Window

현재 실행중인 Window중 원하는 Window의 크기를 최소화 합니다.



No.	설명
1	최소화할 Window 변수 명을 입력합니다. (Type: Window 변수)

@@ Info @@

Window 창을 최대화, 최소화 시키는 방법

Ex) Open Browser -> Attach Window(Window 변수 생성) -> Maximize/Minimize Window

## ■ Resize Window

Window의 크기를 조정할 수 있습니다.

39     Resize Window



No.	설명
1	크기를 조정할 Window 변수 명을 입력합니다. (Type: Window 변수)
2	동작을 실행할 위치의 X축 좌표를 지정합니다. (Type : Int)
3	동작을 실행할 위치의 Y축 좌표를 지정합니다. (Type : Int)
4	원하는 창의 너비를 입력합니다. (Type : Int)
5	원하는 창의 높이를 입력합니다. (Type : Int)

## 5.2. Function

A.WORKS와 연동하는 단위 작업을 위한 추가기능을 모아 놓은 것입니다.

Function은 총 3개의 카테고리로 구성되어 있습니다.

- Control
- Display
- Scrap

### 5.2.1 Control

Object를 인식할 경우 Object에서 Text를 읽어 들이거나 Object로 Text를 내보내는 Object 제어와 관련된 기능들을 제공합니다.

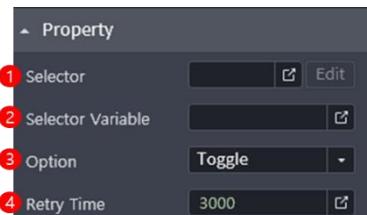
Control은 총 6개의 Event Item으로 구성되어 있습니다.

- Check Box
- Get Item List
- Get Text
- Select Item
- Select Item Text
- Set Text

#### ■ Check Box

체크박스 타입의 객체를 인식할 수 있습니다. (Indicator 제공)

단, WIN32(MSAA, UIA), Internet Explore에서만 사용 가능합니다.

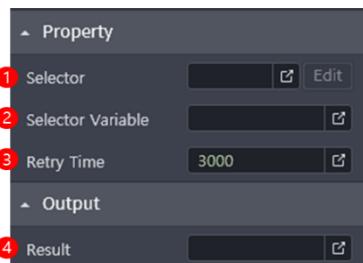


No.	설명
1	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집 가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우

2	Selector 변수만 입력할 수 있습니다. 단, 1번 또는 2번 중 한가지만 입력할 수 있습니다. 원하는 상태 값을 선택합니다. (Toggle, Check, Uncheck)
	객체 또는 이미지 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)

## ■ Get Item List

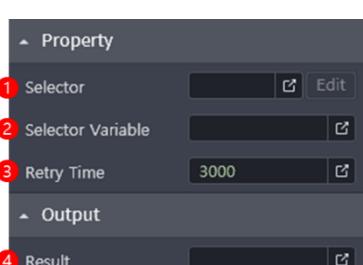
체크박스 타입의 객체를 인식할 수 있습니다. (Indicator 제공)  
단, WIN32(MSAA, UIA), Internet Explore에서만 사용 가능합니다.



No.	설명
1	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우
2	Selector 변수만 입력할 수 있습니다. 단, 1번 또는 2번 중 한가지만 입력할 수 있습니다.
3	객체 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)
4	선택된 아이템 List를 저장할 변수명을 입력합니다. (Type : StringArray)

## ■ Get Text

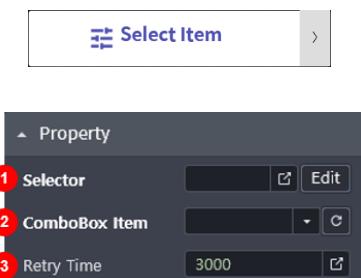
선택된 Object에 있는 Text값을 가져 옵니다. (Indicator 제공)



No.	설명
1	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우
2	Selector 변수만 입력할 수 있습니다. 단, 1번 또는 2번중 한가지만 입력할 수 있습니다.
3	객체 또는 이미지 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)
4	지정한 Text를 저장할 변수 명을 선택합니다. (Type: String)

## ■ Select Item

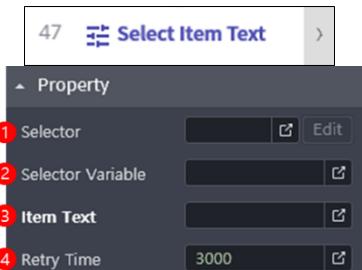
Select Item은 ComboBox Object를 인식하여 목록 가운데 한 값을 선택할 수 있게 합니다. (Indicator 제공)



No.	설명
1	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우
2	선택할 ComboBox의 데이터를 선택합니다. (Type: String) <ul style="list-style-type: none"> <li>- ComboBox의 데이터를 읽을 수 있는 상태 (Dropdown 사용 가능)</li> <li>- ComboBox의 데이터를 읽을 수 없는 상태 (직접입력)</li> </ul>
3	객체 또는 이미지 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)

## ■ Select Item Text

Select Item과 유사하며 ComboBox Object를 인식하여 사용자가 직접 입력 또는 변수 형태로 입력하여 목록 가운데 한 값을 선택할 수 있습니다. (Indicator 제공)

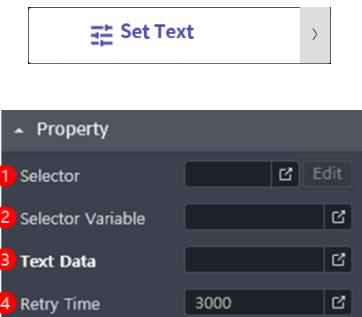


No.	설명
1	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우
2	Selector 변수만 입력할 수 있습니다. 단, 1번 또는 2번 중 한가지만 입력할 수 있습니다.
3	아이템으로 선택할 Text를 입력합니다. (Type : String)
4	객체 또는 이미지 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)

## ■ Set Text

Set Text는 선택된 Object에 미리 입력된 텍스트를 설정합니다 (Indicator 제공).

※ Key Input은 해당 Object에 직접 keyInput을 통하여 Text를 설정하는 반면, Set Text는 사전에 지정해 놓은 Text를 설정하는 것이 차이가 있습니다.



No.	설명
1	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우

2	Selector 변수만 입력할 수 있습니다. 단, 1번 또는 2번중 한가지만 입력할 수 있습니다.
3	Set할 Text를 입력합니다. (Type: String)
4	객체 또는 이미지 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)

## 5.2.2 Display

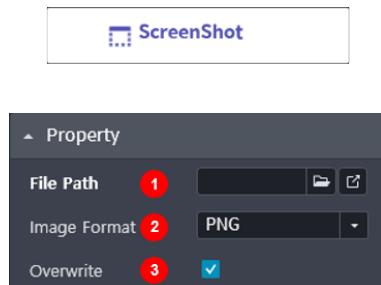
Display에 관련된 기능들을 제공합니다.

Display는 다음과 같은 Event Item으로 구성되어 있습니다.

- ScreenShot

### ■ ScreenShot

Screen Shot을 만들어 저장하는 기능입니다.



No.	설명
1	Screen Shot을 저장할 파일 경로를 입력합니다. (Type: String) File Path: 파일명을 포함한 파일 경로를 입력합니다. (변수 입력 또는 파일 탐색기를 활용하여 파일명 직접 입력)
2	저장할 이미지의 파일형식을 선택합니다. > 파일형식: PNG, JPG, BMP, TIFF
3	1번의 파일 경로에 파일이 존재할 경우 덮어쓰기를 할 것인지 지정합니다. > Checked: 덮어쓰기를 진행합니다. > Unchecked: 동일한 파일 경로가 존재할 경우 Error가 표시됩니다.

## 5.2.3 Scrap

Scrap과 관련 기능들을 모아 놓은 것입니다.

Scrap에는 다음과 같은 Event Item으로 구성되어 있습니다.

- Grab Screen Data
- Grab Table Data

### ■ Grab Screen Data

Object에 의해 지정된 Screen에서 지정된 Data를 가져 옵니다. (Indicator 제공).

## @@ 주의 @@

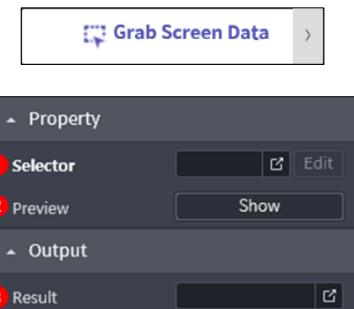
※ 이 Event Item을 실행 시에는 해당 Object가 표시되는 화면이 실행되고 있어야 합니다.

※ Grab Screen Data' 이벤트아이템의 사용상 제약사항에 대하여

본 Event Item의 기술적 특성 및 환경적인 영향에 따라서 100% 인식률을 보장할 수 없습니다.

프로젝트에 적용하기 전에 "Grab Screen Data" 사용에 대한 기술적인 검토가 반드시 필요합니다.

또한, 환경적인 특성에 따라서는 기술적으로 지원할 수 없는 경우가 존재할 수 있습니다.



No.	설명
1	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우
2	지정된 Object의 스크래핑한 결과값을 Preview할 수 있습니다.
3	DataTable을 저장할 변수 명을 지정합니다. (Type: DataTable, String)

※ Grab Screen Data로 스크래핑한 값들은 상→하, 좌→우 순서로 정렬되어 표기됩니다.

## @@ 주의 @@

※ Object나 이미지 인식이 어려운 특정 소프트웨어 환경에서 사용 가능한 기능으로 Studio V2.0에서는 X-Internet 기반의 UI 통합 솔루션인 x-Frame에서만 적용이 가능합니다.

## @@ Info @@

Grab Screen Data 의 상세 기능에 대해서는 "A.WORKS Studio V2.1.3 사용자 설명서의 4.4.3 Screen Scraping"을 참조하십시오.

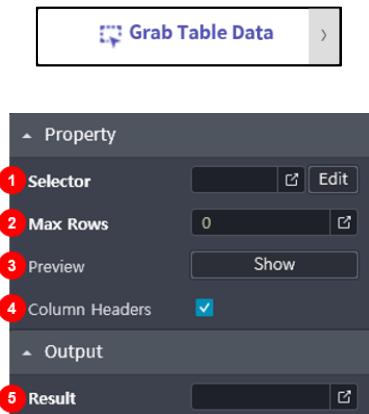
## ■ Grab Table Data

지정된 Object를 포함하는 Table 형식의 데이터를 DataTable로 반환합니다. (Indicator 제공)

웹 환경(Internet Explorer 및 Chrome - Extension 설치 필요)에서 Table이나 Non-Table 형태의 데이터를 인식하고 관련 데이터를 모두 가져오는 기능입니다.

## @@ 주의 @@

※ 이 Event Item을 실행 시에는 해당 Object가 표시되는 웹 페이지가 실행되고 있어야 합니다.



No.	설명
1	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집 가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류 발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우
2	추출할 DataTable에 최대 Row 개수를 지정합니다. (Type: Int, Default: 0 - Unlimited)
3	지정된 Object의 Table이 Grab이 가능한지 Preview 할 수 있습니다.
4	추출한 DataTable의 첫 번째 행을 Column Headers로 사용할 것인지 선택합니다. - Checked: 첫 번째 행을 Column Headers Name으로 사용 - Unchecked: 첫 번째 행을 Column Headers Name으로 사용하지 않음
5	반환할 Data를 저장할 DataTable 변수 명을 지정합니다. (Type: DataTable)

## @@ 주의 @@

※ Scrap 대상이 Table 형식인 경우에는 해당 Table 일괄하여 가져와서 Show 버튼을 통하여 Preview로 확인하여 필요한 Data를 Scrap할 수 있습니다. 하지만, Non-Table 형식의 Data는 순차적으로 읽어 와서 확인하고 추가적으로 Data를 가져오는 방식입니다. 사용방법에 대한 숙지가 필요합니다.

## @@ Info @@

Grab Table Data의 상세 기능에 대해서는 “A.WORKS Studio V2.0 사용자 설명서의 4.4.4 Data Scrapping” 을 참조하십시오.

## 5.3. Input Device

화면에서 Device를 이용한 Data 입력과 관련된 기능들을 모아 놓은 것입니다.

Input Device는 총 2개의 카테고리로 구성되어 있습니다.

- Keyboard
- Mouse

### 5.3.1 Keyboard

Keyboard를 이용한 입력 관련 기능들을 모아 놓은 것입니다.

Keyboard는 다음과 같은 Event Item으로 구성되어 있습니다.

- Key Input
- Key Input Secure

#### ■ Key Input

Keyboard를 입력하는 기능을 수행합니다. Text 입력을 원하는 Object를 지정해 입력할 수 있습니다. (Indicator 제공).

※ Studio 2.0에서는 사용자 편의성 향상을 위해 가상 Keyboard를 추가하였습니다.



No.	설명
1	<p>입력할 내용을 입력합니다. (Type: String, 일반 문자열은 ""를 사용)</p> <p>※ Studio 2.0에서는 특수기 및 Hotkey 입력 가상화 기능을 추가하였습니다.</p> <p>(Value Editor Icon을 클릭하면 특수기 및 Hotkey 입력 팝업 창을 표시. 아래 설명 참조)</p>
2	<p>Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용)</p> <p>Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다</p> <p>※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우</p>
3	Selector 변수만 입력할 수 있습니다. 단, 2번 또는 3번중에 한가지만 입력할 수 있습니다.
4	<p>Keyboard 입력 모드를 선택합니다.</p> <p>각 입력모드에 따라서 Verify 설정 &amp; Empty Field 조합이 다르니 주의가 필요합니다.</p> <p>아래의 <b>"Input 값 입력방법 상세"</b>를 참조하시기 바랍니다.</p> <ul style="list-style-type: none"> <li>&gt; Hardware: Hardware Driver를 이용한 입력 방식 <ul style="list-style-type: none"> <li>- 마우스와 키보드 드라이버를 사용하여 동작</li> <li>- 해당 프로그램이 항상 Active 상태여야 함</li> <li>- 3가지 방식 중 속도가 가장 느리며, 사용자가 자동화 도중에 끼어들 수 있는 단점이 있음</li> </ul> </li> <li>※ Hardware 모드는 클릭 이벤트는 실제 마우스와 동일하게 동작하며, Button Down이 선동작하지 않으면 Button Up이 수행되지 않음</li> <li>&gt; Software: Software를 이용한 입력방식 <ul style="list-style-type: none"> <li>- Studio가 이벤트 상세내역을 프로그램의 메시지 루프에 보내어 프로그램의 윈도우 프로시저가 대상 UI 요소로 하여금 이벤트를 처리</li> <li>- 백그라운드 작업 필요 시 사용</li> </ul> </li> <li>※ Software 모드는 Button Down이 선동작하지 않더라도 Button Up이 수행 됨</li> <li>&gt; Simulate: 팝업 창의 가상키보드 방식을 이용한 입력방식 <ul style="list-style-type: none"> <li>- Studio가 프로그램을 가로채서 해당 UI 요소 (버튼, 입력 창 등)의 해당 이벤트 핸들러를 트리거 시킴</li> <li>- 백그라운드 작업 필요 시 사용</li> <li>- 3가지 방식 중 속도가 가장 빠르나, 키보드 단축키 사용할 수 없음</li> </ul> <p>(내부에 Empty Field 기능이 포함되어 있음)</p> </li> <li>&gt; Scancode : 키보드 보안 프로그램 사용시 입력방식 <ul style="list-style-type: none"> <li>- 키보드의 키마다 특정한 고유의 값을 부여해 놓은 스캔 코드를 이용하는 방식으로, 키보드 보안 프로그램이 동작하여 Software 방식으로 입력이 제한되는 곳에 사용</li> </ul> </li> </ul>
5	<p>입력이 잘 들어갔는지 검증하는 옵션입니다. 입력 값과 실제 입력 값을 비교합니다.</p> <p>값이 다른 경우 한번 더 Key 입력을 시도합니다.</p> <p>아래의 <b>"Input 값 입력방법 상세"</b>를 참조하시기 바랍니다.</p> <ul style="list-style-type: none"> <li>&gt; Checked: 값이 다른 경우 한번 더 Key 입력을 시도</li> <li>&gt; Unchecked: 값이 달라도 무시하고 입력 값으로 처리</li> </ul> <p>※ V2.0 에서는 지원되지 않는 기능입니다.</p>
6	<p>Object가 설정된 경우, Object에 입력된 기준 값을 삭제 후 Key 입력을 다시 시도합니다.</p> <p>아래의 <b>"Input 값 입력방법 상세"</b>를 참조하시기 바랍니다.</p>

	<p><b>@@ 주의 @@</b> Java 환경에서 Empty Field 체크 옵션 동작하지 않습니다.</p> <ul style="list-style-type: none"> <li>&gt; Checked: 값이 다른 경우 한번 더 Key 입력을 시도</li> <li>&gt; Unchecked: 값이 없어도 무시하고 입력 값으로 처리</li> </ul> <p>※ 4번에서 Simulate 선택 시 비활성화합니다.</p>
7	<p>인식된 객체에 대하여 Focus On 기능을 선택합니다.</p> <p>Selector를 Indicate 했을 시 해당 Object에 강제로 Focus On 기능을 선택합니다.</p> <ul style="list-style-type: none"> <li>&gt; Checked: 강제로 Focus On 함</li> <li>&gt; Unchecked: (Default) Focus Off하지 않거나 Disable 됨</li> </ul>
8	Key 입력과 Key 입력 간의 간격을 속도로 설정합니다. (Type: Int, 단위: ms)
9	객체 또는 이미지 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)
10	목표한 Object로부터의 X축 상대좌표를 입력합니다. (Offset 값: Int)
11	목표한 Object로부터의 Y축 상대좌표를 입력합니다. (Offset 값: Int)
12	결과를 저장할 변수 명을 설정합니다. (Type: Boolean)

## @@ Info @@

### ■ Input 값 입력방법 상세

#### 1. 입력 값에 대하여

- 1) 쌍따옴표(" ") 안에 입력된 값은 String으로 간주하여 Text 형태로 출력합니다.
- 2) 쌍따옴표 안에 '제어문자'가 있는 경우(Hardware, Software 모드 동일)
  - 입력: "aa + Enter + bb", 출력: aa + Enter + bb
- 3) 쌍따옴표 안에 'Escape Sequence'가 있는 경우(Hardware, Software 모드 동일)
  - 입력: "aa + \n + bb", 출력: aa + \n +bb
- 4) 사용자가 키보드를 이용하여 직접 Enter, Tab 키를 적용한 경우
  - > Hardware 모드: 입력: aa bb, 출력: aa bb
  - > Software 모드: 입력: aabb, 출력: aabb
- 5) 여러 문자를 조합하여 입력하는 경우
 

Ex.) "aa" + " " + sTemp를 입력

  - sTemp가 String Type으로 선언된 경우 "aa sTemp내용"으로 입력
  - sTemp가 String Type으로 선언 안 된 경우는 오류 발생

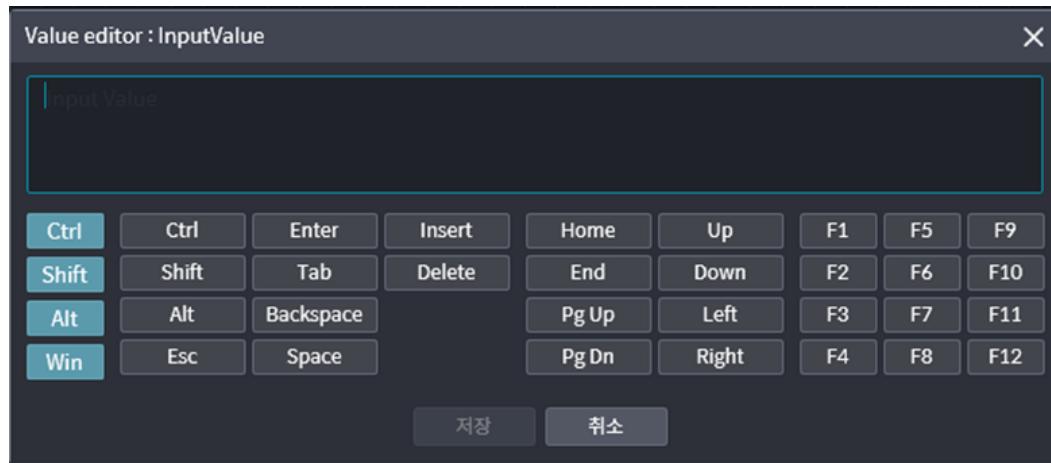
#### 2. 특수문자 입력 기준은 다음과 같습니다.

입력	출력
"string"	string
""""	"
"\\\"	\\"
"!@#"	!@#
"\\n+"	\n+

단, + 기호는 쌍따옴표(")와 복합으로 사용은 불가합니다.

## 2. 특수키 및 Function key 입력

Value Editor Icon을 클릭 시 다음과 같은 팝업 창이 표시되고 해당 키를 입력 가능합니다.



- 특수키 입력은 다음과 같이 표시합니다.

Ex) [Enter] + [Ctrl[A]] + [Ctrl[Alt[F1]]] + [F12] + [Home]

- [Ctrl], [Shift], [Alt], [Win] 버튼은 토글 방식의 버튼으로 누르면 활성화합니다.

## 3. 사용가능 특수키

특수키	표현법
Ctrl	[ctrl]
Shift	[shift]
Alt	[alt]
Windows	[windows]
Enter	[enter]
ESC	[esc]
BackSpace	[backspace]
Tab	[tab]
Space	[space]
ScrollLock	[scrolllock]
Pause	[pause]
Insert	[insert]

특수키	표현법
Home	[home]
PageUp	[pageup]
Delete	[delete]
End	[end]
PageDown	[pagedown]
Right	[right]
Left	[left]
Down	[down]
Up	[up]
NumLock	[numlock]
한/영	[hangul]

특수키	표현법
F1	[f1]
F2	[f2]
F3	[f3]
F4	[f4]
F5	[f5]
F6	[f6]
F7	[f7]
F8	[f8]
F9	[f9]
F10	[f10]
F11	[f11]
F12	[f12]

## 4. Key Input의 Verify & Empty Field 사양

### 1) 모드 & Verify 조합

> Software: Checked/Unchecked 모두 가능

- > Hardware: Checked/Unchecked 모두 가능
- > Simulate: Unchecked 만 가능
- 2) 모드 & Empty Fields 조합
  - > Software: Checked/Unchecked 모두 가능
  - > Hardware: Checked/Unchecked 모두 가능
  - > Simulate: Checked 만 가능
- 3) Verify 실패시의 Key Input 정상/실패 처리 기준
  - > Studio V2.0 에서는 Verify 실패 시 정상 실행 처리

## 5. 기타 제약사항

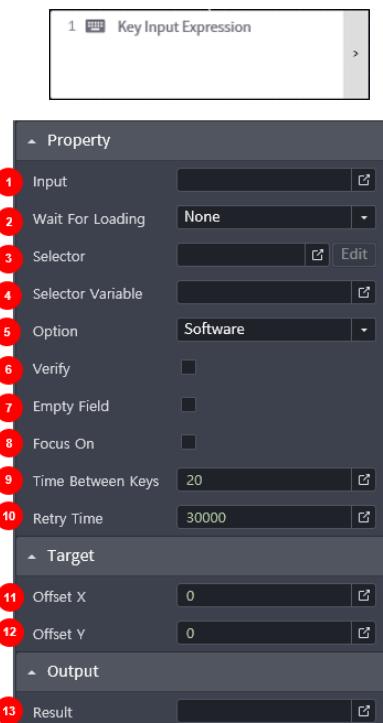
- ※ 사용자가 키보드를 이용하여 직접 Enter, Tab키를 적용한 경우에는 적용 Application(HTML, Word, Notepad)에 따라 출력되는 방식이 다를 수 있음.
- ※ 정확한 제어문 처리를 위해서는 [Enter], [Tab]등은 Property 창에 보여지는 특수키 입력 창에서 제공하는 제어문을 사용해야 함.

@@ Info @@

특수키 및 Hotkey 입력의 상세 기능에 대해서는 "A.WORKS Studio V2.0 사용자 설명서의 4.4.5 특수키 입력"을 참조하십시오.

## ■ Key Input Expression

Keyboard를 입력하는 기능을 수행합니다. Text 입력을 원하는 Object를 지정해 입력할 수 있습니다. (Indicator 제공). Key Input Expression은 Input에 StringArray 또는 C# Code Expression을 수행하여 입력이 가능합니다. Key Input은 Input에 사용자가 입력하는 텍스트 또는 String 형의 변수가 입력이 가능합니다.



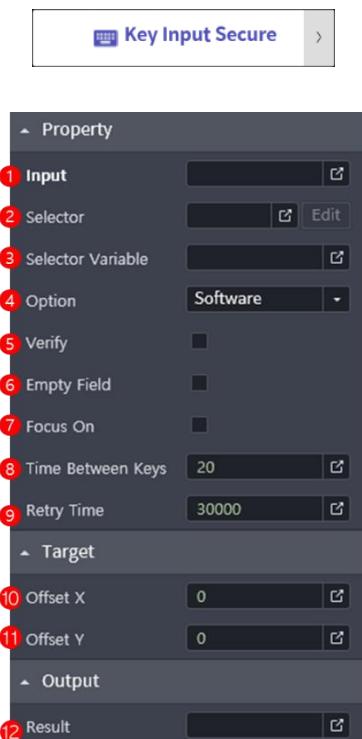
No.	설명
1	<p>입력할 내용을 입력합니다. Input은 C# Code Expression 형식의 값입니다.</p> <p>※ Studio 2.0에서는 특수키 및 Hotkey 입력 가상화 기능을 추가하였습니다.</p> <p>(Value Editor Icon을 클릭하면 특수키 및 Hotkey 입력 팝업 창을 표시. Key Input과 동일하며, 설명 참조)</p>
2	<p>Wait For Loading: Object Detection 실행하기 위한 문서의 로딩 상태를 결정합니다.</p> <p>이 옵션은 Internet Explorer에서만 적용됩니다.</p>
3	<p>Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용)</p> <p>Value Editor 아이콘을 클릭하여 Object 확인, 편집 가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다</p> <p>※ 오류 발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우</p>
4	<p>Selector 변수만 입력할 수 있습니다. 단, 2번 또는 3번 중에 한가지만 입력할 수 있습니다.</p>
5	<p>Keyboard 입력 모드를 선택합니다.</p> <p>각 입력 모드에 따라서 Verify 설정 &amp; Empty Field 조합이 다르니 주의가 필요합니다.</p> <p>아래의 “<b>Input 값 입력 방법 상세</b>”를 참조하시기 바랍니다.</p> <ul style="list-style-type: none"> <li>&gt; Hardware: Hardware Driver를 이용한 입력 방식 <ul style="list-style-type: none"> <li>- 마우스와 키보드 드라이버를 사용하여 동작</li> <li>- 해당 프로그램이 항상 Active 상태여야 함</li> <li>- 3가지 방식 중 속도가 가장 느리며, 사용자가 자동화 도중에 끼어들 수 있는 단점이 있음</li> </ul> <p>※ Hardware 모드는 클릭 이벤트는 실제 마우스와 동일하게 동작하며, Button Down이 선동작하지 않으면 Button Up이 수행되지 않음</p> </li> <li>&gt; Software: Software를 이용한 입력 방식 <ul style="list-style-type: none"> <li>- Studio가 이벤트 상세내역을 프로그램의 메시지 루프에 보내어 프로그램의 윈도우 프로세서가 대상 UI 요소로 하여금 이벤트를 처리</li> <li>- 백그라운드 작업 필요 시 사용</li> </ul> <p>※ Software 모드는 Button Down이 선동작하지 않더라도 Button Up이 수행 됨</p> </li> <li>&gt; Simulate: 팝업 창의 가상키보드 방식을 이용한 입력방식 <ul style="list-style-type: none"> <li>- Studio가 프로그램을 가로채서 해당 UI 요소 (버튼, 입력 창 등)의 해당 이벤트 핸들러를 트리거 시킴</li> <li>- 백그라운드 작업 필요 시 사용</li> <li>- 3가지 방식 중 속도가 가장 빠르나, 키보드 단축키 사용할 수 없음 (내부에 Empty Field 기능이 포함되어 있음)</li> </ul> </li> <li>&gt; Scancode : 키보드 보안 프로그램 사용시 입력방식 <ul style="list-style-type: none"> <li>- 키보드의 키마다 특정한 고유의 값을 부여해 놓은 스캔 코드를 이용하는 방식으로, 키보드 보안 프로그램이 동작하여 Software 방식으로 입력이 제한되는 곳에 사용</li> </ul> </li> </ul>
6	<p>입력이 잘 들어갔는지 검증하는 옵션입니다. 입력 값과 실제 입력 값을 비교합니다.</p> <p>값이 다른 경우 한번 더 Key 입력을 시도합니다.</p> <p>아래의 “<b>Input 값 입력 방법 상세</b>”를 참조하시기 바랍니다.</p> <ul style="list-style-type: none"> <li>&gt; Checked: 값이 다른 경우 한번 더 Key 입력을 시도</li> <li>&gt; Unchecked: 값이 달라도 무시하고 입력 값으로 처리</li> </ul>

	※ V2.0에서는 지원되지 않는 기능입니다.
7	<p>Object가 설정된 경우, Object에 입력된 기존 값을 삭제 후 Key 입력을 다시 시도합니다.</p> <p>아래의 “<b>Input 값 입력방법 상세</b>”를 참조하시기 바랍니다.</p> <p><b>@@ 주의 @@</b> Java 환경에서 Empty Field 체크 옵션 동작하지 않습니다.</p> <ul style="list-style-type: none"> <li>&gt; Checked: 값이 다른 경우 한번 더 Key 입력을 시도</li> <li>&gt; Unchecked: 값이 없어도 무시하고 입력 값으로 처리</li> </ul> <p>※ 4번에서 Simulate 선택 시 비활성화합니다.</p>
8	<p>인식된 객체에 대하여 Focus On 기능을 선택합니다.</p> <p>Selector를 Indicate 했을 시 해당 Object에 강제로 Focus On 기능을 선택합니다. .</p> <ul style="list-style-type: none"> <li>&gt; Checked: 강제로 Focus On 함</li> <li>&gt; Unchecked: (Default) Focus Off하지 않거나 Disable 됨</li> </ul>
9	Key 입력과 Key 입력 간의 간격을 속도로 설정합니다. (Type: Int, 단위: ms)
10	객체 또는 이미지 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)
11	목표한 Object로부터의 X축 상대좌표를 입력합니다. (Offset 값: Int)
12	목표한 Object로부터의 Y축 상대좌표를 입력합니다. (Offset 값: Int)
13	결과를 저장할 변수 명을 설정합니다. (Type: Boolean)

## ■ Key Input Secure

Keyboard를 입력하는 기능을 수행합니다. Text 입력을 원하는 Object를 지정해 입력할 수 있습니다. (Indicator 제공).

Key Input Secure는 Input항목의 Key Fields에 대하여 암호화 처리를 하여 수행합니다.



No.	설명
1	<p>입력할 내용을 SecureString으로 변환하여 지정해야 합니다. (Type: SecureString 사용)</p> <p>※ SecureString 입력방식에 대해서는 "Text To SecureString" Event Item을 참조하시기 바랍니다.</p>
2	<p>Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용)</p> <p>Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다</p> <p>※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우</p>
3	Selector 변수만 입력할 수 있습니다. 단, 2번 또는 3번중 한가지만 입력할 수 있습니다.
4	<p>Keyboard 입력 모드를 선택합니다.</p> <p>각 입력모드에 따라서 Verify 설정 &amp; Empty Field 조합이 다르니 주의가 필요합니다.</p> <p>아래의 "<b>Input 값 입력방법 상세</b>"를 참조하시기 바랍니다.</p> <ul style="list-style-type: none"> <li>&gt; Hardware: Hardware Driver를 이용한 입력 방식           <ul style="list-style-type: none"> <li>※ Hardware 모드는 클릭 이벤트는 실제 마우스와 동일하게 동작하며, Button Down이 선동작하지 않으면 Button Up이 수행되지 않음</li> </ul> </li> <li>&gt; Software: Software를 이용한 입력방식           <ul style="list-style-type: none"> <li>※ Software 모드는 Button Down이 선동작하지 않더라도 Button Up이 수행 됨</li> </ul> </li> <li>&gt; Simulate: 팝업 창의 가상키보드 방식을 이용한 입력방식</li> <li>&gt; Scancode : 키보드 보안 프로그램 사용시 입력방식           <ul style="list-style-type: none"> <li>- 키보드의 키마다 특정한 고유의 값을 부여해 놓은 스캔 코드를 이용하는 방식으로, 키보드 보안 프로그램이 동작하여 Software 방식으로 입력이 제한되는 곳에 사용</li> </ul> </li> </ul>
5	<p>입력이 잘 들어갔는지 검증하는 옵션입니다. 입력 값과 실제 입력 값을 비교합니다.</p> <p>값이 다른 경우 한번 더 Key 입력을 시도합니다.</p> <p>아래의 "<b>Input 값 입력방법 상세</b>"를 참조하시기 바랍니다.</p> <ul style="list-style-type: none"> <li>&gt; Checked: 값이 다른 경우 한번 더 Key 입력을 시도</li> <li>&gt; Unchecked: 값이 달라도 무시하고 입력 값으로 처리</li> </ul>
6	<p>Object가 설정된 경우, Object에 입력된 기존 값을 삭제 후 Key 입력을 다시 시도합니다.</p> <p>아래의 "<b>Input 값 입력방법 상세</b>"를 참조하시기 바랍니다.</p> <p><b>@@ 주의 @@</b> Java 환경에서 Empty Field 체크 옵션 동작하지 않습니다.</p> <ul style="list-style-type: none"> <li>&gt; Checked: 값이 다른 경우 한번 더 Key 입력을 시도</li> <li>&gt; Unchecked: 값이 없어도 무시하고 입력 값으로 처리</li> </ul>
7	<p>인식된 객체에 대하여 Focus On 기능을 선택합니다.</p> <p>Selector를 Indicate 했을 경우 해당 Object에 강제로 Focus On 기능을 선택합니다. .</p> <ul style="list-style-type: none"> <li>&gt; Checked: 강제로 Focus On 함</li> <li>&gt; Unchecked: (Default) Focus Off하지 않거나 Disable 됨</li> </ul>
8	Key 입력과 Key 입력 간의 간격을 속도로 설정합니다. (Type: Int, 단위: ms)
9	객체 또는 이미지 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)
10	목표한 Object로부터의 X축 상대좌표를 입력합니다. (Offset 값: Int)
11	목표한 Object로부터의 Y축 상대좌표를 입력합니다. (Offset 값: Int)

@@ Info @@

### ■ Input 값 입력방법 상세

1. SecureString 입력방식에 대해서는 "Text To SecureString" Event Item을 참조하시기 바랍니다.

2. Key Input의 Verify & Empty Field 사양

1) 모드 & Verify 조합

- > Software: Checked/Unchecked 모두 가능
- > Hardware: Checked/Unchecked 모두 가능
- > Simulate: Unchecked 만 가능

2) 모드 & Empty Fields 조합

- > Software: Checked/Unchecked 모두 가능
- > Hardware: Checked/Unchecked 모두 가능
- > Simulate: Checked 만 가능

3) Verify 실패시의 Key Input 정상/실패 처리 기준

- > Studio V2.0 에서는 Verify 실패 시 정상 실행 처리

@@ Info @@

특수키 및 Hotkey 입력의 상세 기능에 대해서는 "A.WORKS Studio V2.0 사용자 설명서의 4.4.5 특수키 입력"을 참조하십시오.

### 5.3.2 Mouse

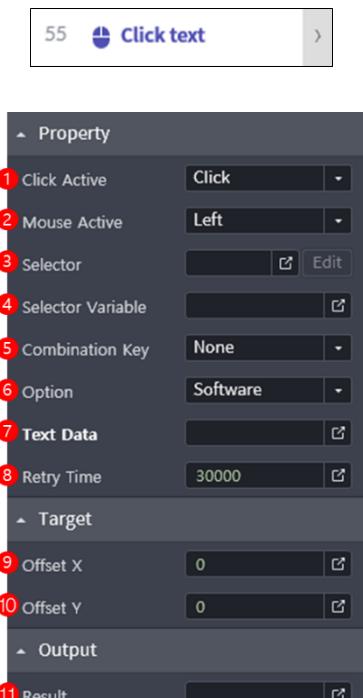
Mouse와 관련된 기능들을 모아 놓은 것입니다.

Mouse는 다음과 같은 Event Item으로 구성되어 있습니다.

- Click Text
- Drag
- Get Position
- Image Click
- Image Match
- Invisible Area Click
- Object Click
- Object Match
- Position Click
- Wait Object Hide
- Wait Object See

#### ■ Click Text

지정한 Object 내에서 Text 값을 찾아 클릭할 수 있습니다. (Indicator 제공)



No.	설명
1	마우스 클릭방식을 선택합니다. > Click: 마우스 한번 클릭

	<ul style="list-style-type: none"> <li>&gt; Double Click: 연달아 2번 클릭</li> <li>&gt; Down: 마우스 버튼을 누른 상태</li> <li>&gt; Up: 마우스 버튼의 누름을 해제(Down 상태에서만 가능)</li> <li>&gt; Over: 해당 마우스 위치로 옮겨짐.</li> </ul>
2	<p>동작을 실행할 마우스 버튼을 선택합니다.</p> <ul style="list-style-type: none"> <li>&gt; Left: 왼쪽 버튼</li> <li>&gt; Right: 오른쪽 버튼</li> <li>&gt; Wheel: 휠 버튼(중간 버튼)</li> </ul>
3	<p>Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용)</p> <p>Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다</p> <p>※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우</p>
4	Selector 변수만 입력할 수 있습니다. 단, 3번 또는 4번중 한가지만 입력할 수 있습니다.
5	<p>Click 동작 시 조합할 Key를 지정합니다.</p> <ul style="list-style-type: none"> <li>&gt; None: 사용 안 함(Default)</li> <li>&gt; Alt: Alt Key</li> <li>&gt; Ctrl: Ctrl Key</li> <li>&gt; Shift: Shift Key</li> <li>&gt; Win: Windows Key</li> </ul>
6	<p>Mouse Click 방식 선택합니다. (Key Input과 동일함)</p> <ul style="list-style-type: none"> <li>&gt; Software(Default): Software를 이용한 입력방식           <ul style="list-style-type: none"> <li>※ Software 모드는 Button Down이 선동작하지 않더라도 Button Up이 수행 됨</li> </ul> </li> <li>&gt; Hardware: Hardware Driver를 이용한 입력 방식           <ul style="list-style-type: none"> <li>※ Hardware 모드는 클릭 이벤트는 실제 마우스와 동일하게 동작하며, Button Down이 선동작하지 않으면 Button Up이 수행되지 않음</li> </ul> </li> </ul>
7	클릭으로 선택할 Text를 입력합니다. (Type : String)
8	객체 또는 이미지 인식시도 시 Retry Time을 입력합니다. (Type: Int, 단위: ms)
9	선택한 Object로부터 입력된 값만큼 X축 상대좌표인 Offset을 입력합니다. (Type: Int)
10	선택한 Object로부터 입력된 값만큼 Y축 상대좌표인 Offset을 입력합니다. (Type: Int)
11	결과를 저장할 변수 명을 선택합니다. (Type: Boolean)

## ■ Drag

Object를 원하는 위치로 Drag 할 수 있습니다. (Indicator 제공)



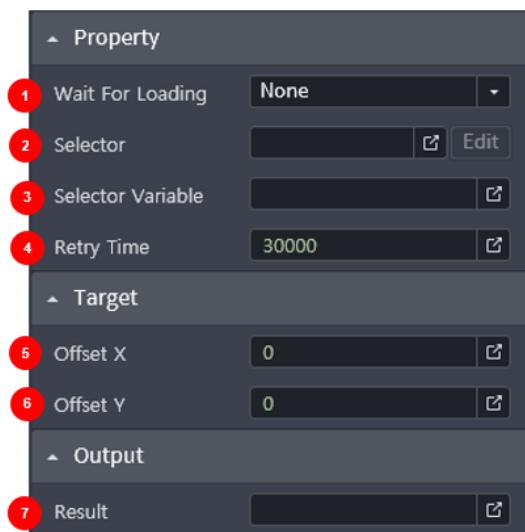


No.	설명
1	Indicator의 Drag 시작 점의 X값을 가져옵니다. (Type: Int) ※ PopupCodeEdit로 문자 및 특수문자 입력 시 오류가 발생합니다.
2	Indicator의 Drag 시작 점의 Y값을 가져옵니다. (Type: Int) ※ PopupCodeEdit로 문자 및 특수문자 입력 시 오류가 발생합니다.
3	Indicator의 Drag 완료 점의 X값을 가져옵니다. (Type: Int) ※ PopupCodeEdit로 문자 및 특수문자 입력 시 오류가 발생합니다.
4	Indicator의 Drag 완료 점의 Y값을 가져옵니다. (Type: Int) ※ PopupCodeEdit로 문자 및 특수문자 입력 시 오류가 발생합니다.
5	마우스 Down 후 마우스 Up까지의 시간을 지정합니다. (Type: Int, 단위: ms)

## ■ Get Position

선택 된 Object의 위치 좌표를 Point변수(Point.X, Point.Y)에 저장 하여 사용할 수 있습니다.

선택 된 Object로부터 떨어진 지점의 좌표를 구하기 위해서는 F3을 누른 후 해당 객체를 선택하면 Point 변수에 저장하여 사용할 수 있습니다. Get Position의 결과 Point값은 Position Click 또는 Drag 이벤트 아이템의 X좌표, Y좌표의 입력 값으로 활용 할 수 있습니다.



No.	설명
1	Wait For Loading: Object Detection 실행하기 위한 문서의 로딩 상태를 결정합니다. 이 옵션은 Internet Explorer에서만 적용됩니다. >None : 객체가 웹페이지에서 로드될 때 까지 기다리지 않고 객체를 찾습니다. >Complete : 객체가 웹페이지에서 완벽하게 로드될 때 까지 기다린 후 객체를 찾습니다.
2	Selector: 원하는 객체를 Indicator로 선택하거나 직접 입력합니다. Selector는 String형식의 값입니다. Selector와 Selector Variable중 하나만 입력하세요.
3	Selector Variable: 원하는 객체 정보가 있는 Selector변수명을 입력합니다. Selector Variable은 Selector, String형식의 변수입니다. Selector와 Selector Variable중 하나만 입력하세요.
4	Retry Time: 실패한 경우 재시도 하는 시간을 설정합니다. Retry Time은 UInt형식의 값입니다.
5	Offset X: 목표한 Object로부터의 x축 상대좌표를 입력합니다. Offset X는 Int형식의 값입니다.
6	Offset Y: 목표한 Object로부터의 y축 상대좌표를 입력합니다. Offset Y는 Int형식의 값입니다.
7	Result: 결과값을 받아올 변수명을 입력합니다. Result는 Point형식의 변수입니다.

## ■ Image Click

원하는 Image에서 수행할 마우스 동작을 지정합니다. (Indicator제공)



No.	설명
1	Indicator로 지정한 마우스 동작 대상의 캡처 이미지 저장경로입니다. (Type: String)
2	Detection Mode를 지정합니다. > Single: 저장된 좌표를 중심으로 가장 비슷한 이미지 검색하며, 화면 배율을 적용한 방식으로, 아래의 Scale Invariant 옵션이 사용 가능합니다.

	<ul style="list-style-type: none"> <li>&gt; Multiple: 유사한 이미지를 전체 검색하며, 아래의 Image Index를 이용하여 사용할 이미지를 지정할 수 있습니다.</li> </ul> <p>※ Detection Mode에 따라 사용 가능한 옵션이 활성화 됩니다.</p>
3	위의 2번에서 Detection Mode를 Multiple로 지정한 경우 여러 이미지 가운데 번호를 지정하여 사용할 수 있습니다.
4	<ul style="list-style-type: none"> <li>위의 2번에서 Detection Mode를 Single로 지정한 경우 사용 가능합니다.</li> <li>&gt; Checked: 스케일이 변형된 객체의 이미지를 탐지 Ex) 크기변화</li> <li>&gt; Unchecked: 변형된 것을 무시(감지하지 못함)</li> </ul>
5	<ul style="list-style-type: none"> <li>듀얼 모니터 사용시 이미지 인식이 가능하도록 설정</li> <li>&gt; Checked : 주 모니터에서만 이미지 인식</li> <li>&gt; Unchecked : 듀얼 모니터(주/부) 모두 이미지 인식</li> </ul>
6	<ul style="list-style-type: none"> <li>마우스 클릭방식을 선택합니다.</li> <li>&gt; Click: 마우스 한번 클릭</li> <li>&gt; Double Click: 연달아 2번 클릭</li> <li>&gt; Down: 마우스 버튼을 누른 상태</li> <li>&gt; Up: 마우스 버튼의 누름을 해제(Down 상태에서만 가능)</li> <li>&gt; Over: 해당 마우스 위치로 옮겨짐.</li> </ul>
7	<ul style="list-style-type: none"> <li>동작을 실행할 마우스 버튼을 선택합니다.</li> <li>&gt; Left: 왼쪽 버튼</li> <li>&gt; Right: 오른쪽 버튼</li> <li>&gt; Wheel: 휠 버튼(중간 버튼)</li> </ul>
8	<ul style="list-style-type: none"> <li>이미지 비교 정확도를 지정합니다.</li> </ul> <p>※Type: Double(소수점 1자리 숫자), Default: 0.8, 최대: 1.0/최소: 0.1</p>
9	<ul style="list-style-type: none"> <li>Click 동작 시 조합할 Key를 지정합니다.</li> <li>&gt; None: 사용 안 함(Default)</li> <li>&gt; Alt: Alt Key</li> <li>&gt; Ctrl: Ctrl Key</li> <li>&gt; Shift: Shift Key</li> <li>&gt; Win: Windows Key</li> </ul>
10	<ul style="list-style-type: none"> <li>Mouse Click 방식 선택합니다. (Key Input과 동일함)</li> <li>&gt; Software(Default): Software를 이용한 입력방식</li> </ul> <p>※ Software 모드는 Button Down이 선동작하지 않더라도 Button Up이 수행 됨</p> <p>&gt; Hardware: Hardware Driver를 이용한 입력 방식</p> <p>※ Hardware 모드는 클릭 이벤트는 실제 마우스와 동일하게 동작하며, Button Down이 선동작하지 않으면 Button Up이 수행되지 않음</p>
11	객체 또는 이미지 인식시도 시 Retry Time을 입력합니다. (Type: Int, 단위: ms)
12	선택한 Object로부터 입력된 값만큼 X축 상대좌표인 Offset을 입력합니다. (Type: Int)
13	선택한 Object로부터 입력된 값만큼 Y축 상대좌표인 Offset을 입력합니다. (Type: Int)
14	결과를 저장할 변수 명을 선택합니다. (Type: Boolean)

## @@ 주의 @@

※ Image Click Indicator 동작 시에 아래와 같은 설정 정보를 따릅니다.

Nvidia 제어판 -> 3D 설정 관리 -> 기본 그래픽 프로세서에서 '자동선택'을 선택.

(\*설정 정보를 선택하지 않을 경우 오류메시지: "Attempted to read or write protected memory. This is often an indication that other memory is corrupt. ")

## ■ Image Match

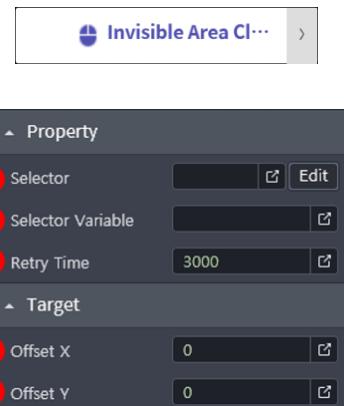
캡처한 이미지와 동일한 이미지의 존재 여부를 반환합니다. (Indicator 제공)



No.	설명
1	Indicator로 지정한 캡처 이미지의 저장경로를 지정합니다. (Type: String)
2	Detection Mode를 지정합니다. > Single: 저장된 좌표를 중심으로 가장 비슷한 이미지 검색하며, 화면 배율을 적용한 방식으로, 아래의 Scale Invariant 옵션이 사용 가능합니다. > Multiple: 유사한 이미지를 전체 검색하며, 아래의 Image Index를 이용하여 사용할 이미지를 지정할 수 있습니다. ※ Detection Mode에 따라 사용 가능한 옵션이 활성화 됩니다.
3	위의 2번에서 Detection Mode를 Multiple로 지정한 경우 매치시킬 이미지의 개수를 지정하여 사용할 수 있습니다.
4	위의 2번에서 Detection Mode를 Single로 지정한 경우 사용 가능합니다. > Checked: 스케일이 변형된 객체의 이미지를 탐지 Ex) 크기변화 > Unchecked: 변형된 것을 무시(감지하지 못함)
5	듀얼 모니터 사용시 이미지 인식이 가능하도록 설정 > Checked : 주 모니터에서만 이미지 인식 > Unchecked : 듀얼 모니터(주/부) 모두 이미지 인식
6	이미지 비교 정확도를 지정합니다. ※ Type: Double(소수점 1자리 숫자), Default: 0.8, 최대: 1.0/최소: 0.1
7	객체 또는 이미지 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)
8	결과를 저장할 변수 명을 선택합니다. (Type: Boolean)

## ■ Invisible Area Click

다른 어플리케이션에 의해 Window에서 가려진 영역의 이미지를 클릭합니다. (Indicator 제공)



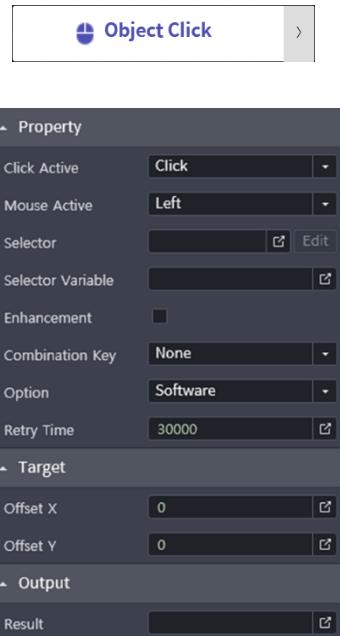
No.	설명
1	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우
2	Selector 변수만 입력할 수 있습니다. 단, 1번 또는 2번중 한가지만 입력할 수 있습니다.
3	객체 또는 이미지 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)
4	선택된 Object로부터 입력된 값만큼 X축 상대좌표인 Offset을 입력합니다. (Type: Int)
5	선택된 Object로부터 입력된 값만큼 Y축 상대좌표인 Offset을 입력합니다. (Type: Int)

### @@ 주의 @@

※ (제약사항) Java 환경에서 해당 기능은 지원되지 않습니다.

## ■ Object Click

선택한 Object를 클릭합니다. (Indicator 제공)

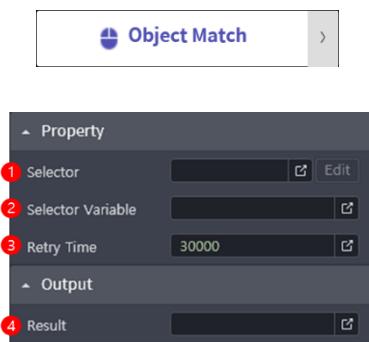


No.	설명
1	마우스 클릭방식을 선택합니다. > Click: 마우스 한번 클릭 > Double Click: 연달아 2번 클릭 > Down: 마우스 버튼을 누른 상태 > Up: 마우스 버튼의 누름을 해제(Down 상태에서만 가능) > Over: 해당 마우스 위치로 옮겨짐.
2	동작을 실행할 마우스 버튼을 선택합니다. > Left: 왼쪽 버튼 > Right: 오른쪽 버튼 > Wheel: 휠 버튼(중간 버튼)
3	Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용) Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다 ※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우
4	Selector 변수만 입력할 수 있습니다. 단, 3번 또는 4번중 한가지만 입력할 수 있습니다.
5	Object 탐색 실패 시, Image 기반 탐색을 수행할 것인지 선택합니다. > Checked: Object Detection 수행 실패 시, Image Detection을 통해 기능을 다시 수행 > Unchecked: Object Detection 만 수행
6	Click 동작 시 조합할 Key를 지정합니다. > None: 사용 안 함(Default) > Alt: Alt Key

	<ul style="list-style-type: none"> <li>&gt; Ctrl: Ctrl Key</li> <li>&gt; Shift: Shift Key</li> <li>&gt; Win: Windows Key</li> </ul>
7	<p>Mouse Click 방식 선택합니다. (Key Input과 동일함)</p> <ul style="list-style-type: none"> <li>&gt; Software(Default): Software를 이용한 입력방식           <ul style="list-style-type: none"> <li>※ Software 모드는 Button Down이 선동작하지 않더라도 Button Up이 수행 됨</li> </ul> </li> <li>&gt; Hardware: Hardware Driver를 이용한 입력 방식           <ul style="list-style-type: none"> <li>※ Hardware 모드는 클릭 이벤트는 실제 마우스와 동일하게 동작하며, Button Down이 선동작하지 않으면 Button Up이 수행되지 않음</li> </ul> </li> </ul>
8	객체 또는 이미지 인식시도 시 Retry Time을 입력합니다. (Type: Int, 단위: ms)
9	선택한 Object로부터 입력된 값만큼 X축 상대좌표인 Offset을 입력합니다. (Type: Int)
10	선택한 Object로부터 입력된 값만큼 Y축 상대좌표인 Offset을 입력합니다. (Type: Int)
11	결과를 저장할 변수 명을 선택합니다. (Type: Boolean)

## ■ Object Match

Object Match에서는 Object를 선택하여 해당 위치에 Object가 있는지 판단합니다. (Indicator 제공)



No.	설명
1	<p>Indicator로 선택한 Object에 대하여 Selector를 가져옵니다. (Indicator 활용)</p> <p>Value Editor 아이콘을 클릭하여 Object 확인, 편집가능하며 Object Editor 버튼인 [Edit]를 클릭하여 Object Selector를 편집하거나 다시 Object를 Detect할 수 있습니다</p> <p>※ 오류발생 기준: Selector에 기술된 Window Object를 찾지 못했을 경우</p>
2	Selector 변수만 입력할 수 있습니다. 단, 1번 또는 2번중 한가지만 입력할 수 있습니다.
3	객체 또는 이미지 인식 시도 Retry Time을 입력합니다. (Type: Int, 단위: ms)
4	결과를 저장할 변수 명을 선택합니다. (Type: Boolean)

## ■ Position Click

원하는 위치 좌표를 지정하여 클릭합니다. (Indicator 제공)

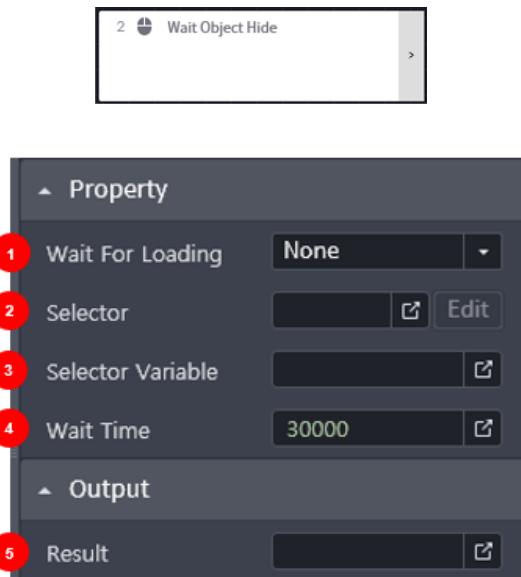


No.	설명
1	동작을 실행할 위치의 X축을 지정합니다. (Type: Int)
2	동작을 실행할 위치의 Y축을 지정합니다. (Type: Int)
3	마우스 클릭방식을 선택합니다. > Click: 마우스 한번 클릭 > Double Click: 연달아 2번 클릭 > Down: 마우스 버튼을 누른 상태 > Up: 마우스 버튼의 누름을 해제(Down 상태에서만 가능) > Over: 해당 마우스 위치로 옮겨짐.
4	동작을 실행할 마우스 버튼을 선택합니다. > Left: 왼쪽 버튼 > Right: 오른쪽 버튼 > Wheel: 휠 버튼(중간 버튼)
5	Click 동작 시 조합할 Key를 지정합니다. > None: 사용 안 함(Default) > Alt: Alt Key > Ctrl: Ctrl Key > Shift: Shift Key > Win: Windows Key
6	Mouse Click 방식 선택합니다. (Key Input과 동일함) > Software(Default): Software를 이용한 입력방식 ※ Software 모드는 Button Down이 선동작하지 않더라도 Button Up이 수행 됨 > Hardware: Hardware Driver를 이용한 입력 방식 ※ Hardware 모드는 클릭 이벤트는 실제 마우스와 동일하게 동작하며, Button Down이 선동작하지 않으면 Button Up이 수행되지 않음

## ■ Wait Object Hide

사용자가 지정한 객체가 사라졌는지 여부를 결과 값을 받아서 알 수 있습니다. (Indicator 제공)

Wait Object See에서 사용자가 지정한 객체가 나타났는지 여부를 확인 한 후, Wait Object Hide로 객체가 사라졌는지 여부를 확인 합니다. 예를 들어 완료 상태를 알리는 팝업 창이 뜬 후 사라졌을 때를 확인하기 위해 사용합니다.

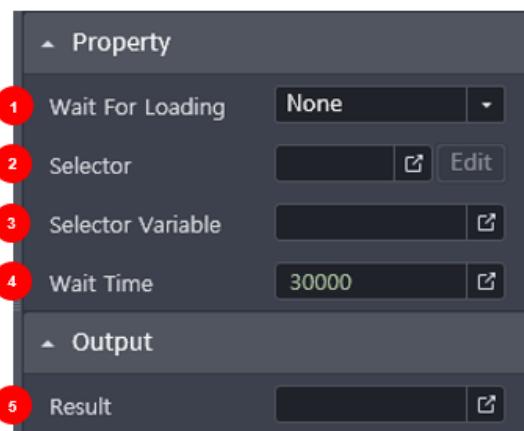
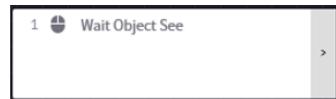


No.	설명
1	Wait For Loading: Object Detection 실행하기 위한 문서의 로딩 상태를 결정합니다. 이 옵션은 Internet Explorer에서만 적용됩니다. >None : 객체가 웹페이지에서 로드 될 때 까지 기다리지 않고 객체를 찾습니다. >Complete : 객체가 웹페이지에서 완벽하게 로드 될 때 까지 기다린 후 객체를 찾습니다.
2	Selector: 원하는 객체를 Indicator로 선택하거나 직접 입력합니다. Selector는 String형식의 값입니다. Selector와 Selector Variable중 하나만 입력하세요.
3	Selector Variable: 원하는 객체 정보가 있는 Selector변수명을 입력합니다. Selector Variable은 Selector, String형식의 변수입니다. Selector와 Selector Variable중 하나만 입력하세요.
4	Wait Time: Object가 사라질 때까지 기다리는 시간을 설정합니다. Wait Time은 UInt형식의 값입니다.
5	Result: 결과값을 받아올 변수명을 입력합니다. Result는 Bool형식의 변수입니다.

## ■ Wait Object See

사용자가 지정한 객체가 나타났는지 여부를 결과 값을 받아서 알 수 있습니다. (Indicator 제공)

Wait Object See에서 사용자가 지정한 객체가 나타났는지 여부를 확인 한 후, Wait Object Hide로 객체가 사라졌는지 여부를 확인 합니다. 예를 들어 완료 상태를 알리는 팝업 창이 뜬 후 사라졌을 때를 확인하기 위해 사용합니다.



No.	설명
1	Wait For Loading: Object Detection 실행하기 위한 문서의 로딩 상태를 결정합니다. 이 옵션은 Internet Explorer에서만 적용됩니다. >None : 객체가 웹페이지에서 로드 될 때 까지 기다리지 않고 객체를 찾습니다. >Complete : 객체가 웹페이지에서 완벽하게 로드 될 때 까지 기다린 후 객체를 찾습니다.
2	Selector: 원하는 객체를 Indicator로 선택하거나 직접 입력합니다. Selector는 String형식의 값입니다. Selector와 Selector Variable중 하나만 입력하세요.
3	Selector Variable: 원하는 객체 정보가 있는 Selector변수명을 입력합니다. Selector Variable은 Selector, String형식의 변수입니다. Selector와 Selector Variable중 하나만 입력하세요.
4	Wait Time: Object가 나타날 때까지 기다리는 시간을 설정합니다. Wait Time은 UInt형식의 값입니다.
5	Result: 결과값을 받아올 변수명을 입력합니다. Result는 Bool형식의 변수입니다.

## 5.4. Process

Process은 A.WORKS의 프로세스와 관련된 기능들을 모아 놓은 것입니다.

Process는 총 1개의 카테고리로 구성되어 있습니다.

- Control

### 5.4.1 Control

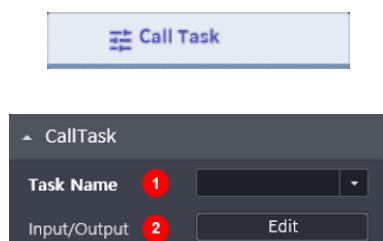
Process를 제어하기 위한 기능들을 모아 놓은 카테고리입니다.

Control에는 다음과 같은 Event Item으로 구성되어 있습니다.

- Call Task
- Delay Time
- Find Process
- Kill Process
- Start Process
- Write Log

#### ■ Call Task

Task 안에서 동일 Process 내의 다른 Task를 호출합니다.



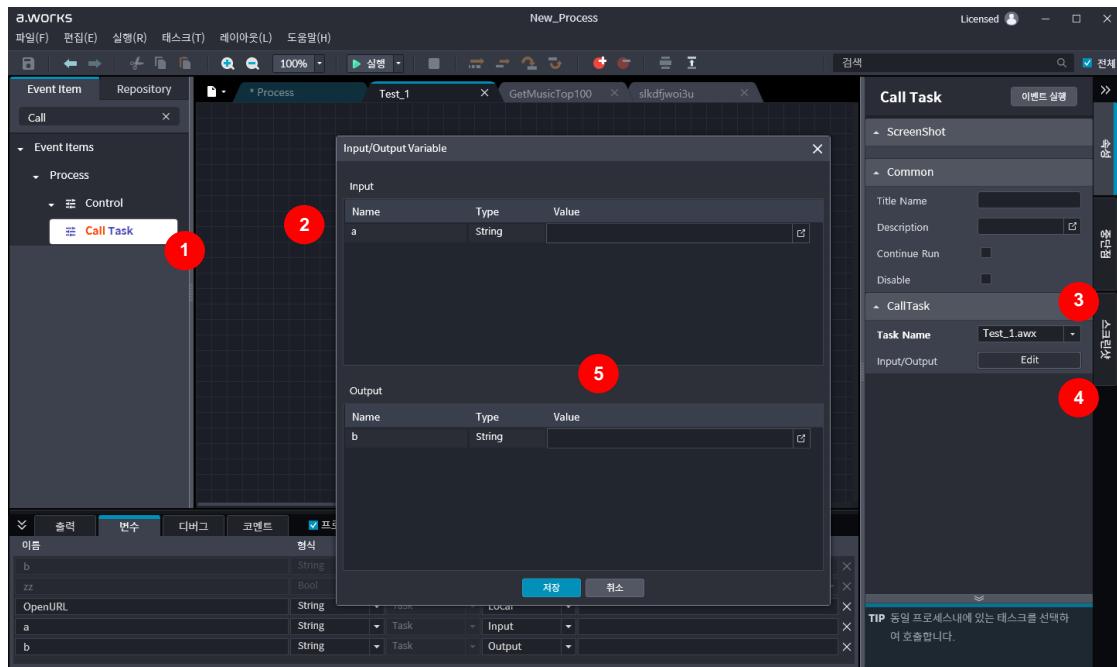
No.	설명
1	Call할 Task를 선택합니다. 클릭 시 Dropdown으로 선택 가능한 Task 목록을 표시합니다. ※ 호출할 Task를 선택할 때, 자기 자신을 호출할 수 없음(선택리스트에 보이지 않음)
2	Task에 Input/Output Variable이 정의되어 있을 경우 Value를 입력합니다.

#### @@ Info @@

※ Call Task Event Item에서 Input/Output Variable 지정

Task Canvas에서 Event Item을 검색하여 "Call Task"를 선택하여 스크립트로 선언하고 클릭 시 속성 탭이 활성화 됩니다. 활성화된 속성 탭의 Call Task항목에서 사용할 Task Name을 지정하고, Input/Output의 [Edit] 버튼을 클릭합니다.

Input/Output Variable 팝업 창이 표시되고 Output 패널의 변수 탭에서 Input/Output으로 선언한 변수 a, b를 연결할 수 있습니다.



< Call Task 를 통한 변수 지정 화면 >

No.	명칭	설명
1	Event Item	Task에서 사용할 Call Task Event Item을 검색합니다.
2	Task Canvas	Call Task Event Item을 Task의 스크립트로 연결하고 클릭합니다.
3	속성 탭	Call Task항목에서 사용할 Task Name을 선택합니다.
4	Edit 버튼	Call Task항목에서 Input/Output의 [Edit] 버튼을 클릭합니다.
5	팝업 창	Input/Output 변수 설정 팝업 창이 표시되고 Output의 변수 탭에서 In-put/Output으로 선언한 a, b를 편집할 수 있습니다.

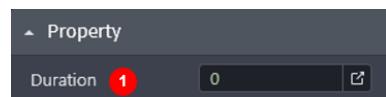
Input/Output Variable 의 상세 기능에 대해서는 “A.WORKS Studio V2.0 사용자 설명서의 5.4.4 변수설정하기”를 참조하십시오.

※ Call Task에서 Shared Task 선택시 ComboBox List에 Task Name 앞에 "(S)사용자 정의 Task 이름"으로 표시하여 일반 Task와 Shared Task를 구분합니다.

※ Call Task의 재귀 호출(즉, A Task → B Task → C Task를 호출 하고, C Task → A Task를 호출)은 허용이 됩니다.

## ■ Delay Time

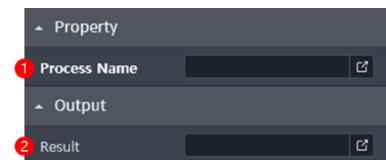
Script의 수행을 일정 시간만큼 지연시킵니다.



No.	설명
1	Delay할 Time 간격을 입력합니다. (Type: Int, 단위: ms)

## ■ Find Process

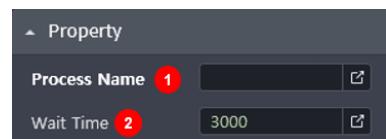
Find Process에서는 원하는 Process를 조회하여 존재여부의 결과를 반환합니다.



No.	설명
1	찾고자 하는 Process명을 입력합니다. (Type: String)
2	Process 존재 여부를 저장할 변수 명을 지정합니다. (Type: Boolean)

## ■ Kill Process

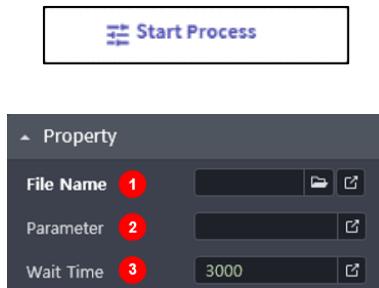
지정한 Process의 수행을 중단합니다.



No.	설명
1	중단하고자 하는 Process명을 입력합니다. (Type: String)
2	Kill Process 수행 후 Process 종료 시까지의 Wait Time을 입력합니다. (Type: Int, 단위: ms)

## ■ Start Process

지정한 Process를 실행합니다.



No.	설명
1	실행할 Process의 경로 입력합니다. (Type: String) ※ 아래의 실행 Process 지정시 주의사항을 참조하여 주십시오.
2	실행 프로그램에 전달이 필요한 Parameter 값이 있을 경우 입력합니다. (Type: String)
3	Start Process 실행 후 프로세스 기동까지의 Wait Time을 입력합니다. (Type: Int, 단위: ms)

### @@ 주의 @@

※ 실행 Process 지정 시 주의사항

- 환경변수의 Path에 등록된 폴더는 파일 이름만 입력해도 동작합니다.
- 환경변수에 포함되지 않은 폴더의 실행파일은 전체경로(Full Path)를 입력해야 동작합니다.
- 지정한 경로에 실행파일이 없을 경우 오류가 발생합니다.

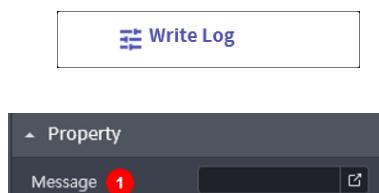
※ Start Process의 지원하는 파일포맷은 다음과 같습니다.

Window환경에서 제공하는 연결프로그램으로 연결이 가능한 모든 파일 형태가 지원된다.

(\* Window환경은 사용자의 구축환경이므로 사용자에 따라 다릅니다.)

## ■ Write Log

Output 영역의 출력 패널에 로그를 출력합니다.



No.	설명
1	출력할 로그의 내용을 입력합니다. (Type: String) 출력되는 Log는 Output 패널에서 로그 Type이 'User'로 지정됩니다.

## 5.5. Programming

일반적인 프로그래밍을 활용할 수 있는 기능들을 모아 놓은 것입니다.

Programming은 총 3개의 카테고리로 구성되어 있습니다.

- Assign
- Code Expression
- DataTable

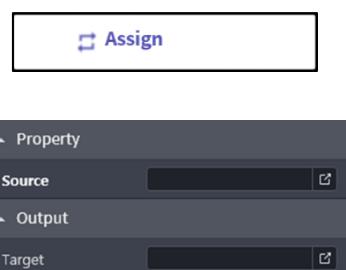
### 5.5.1 Assign

사용자가 직접 변수 명에 지정된 값을 할당하는 기능입니다.

Assign는 다음과 같은 Event Item으로 구성되어 있습니다.

#### ■ Assign

사용자가 직접 변수 명에 지정된 값을 할당합니다.



No.	설명
1	<p>Assign할 입력값을 Code형식으로 입력합니다. ※ PopupCodeEdit에서는 다음사항을 제공합니다.</p> <ul style="list-style-type: none"><li>- 사용가능 함수 또는 변수 명을 Dropdown 표시하여 선택하는 Intelligence 가능</li><li>- Result에 입력된 변수의 Type과 결과값의 비교(Validation)</li><li>- 입력코드의 오류 검출</li></ul> <p><b>@@ 주의 @@</b> 입력값은 String, Int, Boolean, Object형식이어야 합니다.</p>
2	<p>Assign 결과를 입력할 변수 명을 지정합니다. ※ 변수Type은 String, Int, Boolean, Object이어야 합니다.</p>

## 5.5.2 Code Expression

C# 문법을 사용하여 직접 코드를 작성 할 수 있습니다. Syntax Editor를 사용하여 Intelligence 기능을 제공합니다.

Code Expression은 다음과 같은 Event Item으로 구성되어 있습니다.

### ■ Code Expression

Code Expression을 클릭하면 C# 코드를 입력할 수 있는 Syntax Editor 팝업 창이 표시됩니다.

입력한 코드에 오류가 있는 경우, 오류 내용이 하단에 표시됩니다. 제약조건은 Class 와 Function을 선언 또는 정의하여 사용할 수 없으며 Robot Process의 안정적인 동작을 위해 goto 문 등의 사용은 지양해야 합니다. 또한 Code Expression에서는 Process 변수는 사용이 불가합니다.



No.	설명
1	C# Code를 직접 Editor에 입력합니다.
2	C# Code에 오류가 있는 경우 Line, Char, Description 순으로 오류 내용을 표시합니다. [저장] 버튼을 클릭하여 코드를 저장합니다.

#### @@ 주의 @@

※ Code Expression 사용에 따른 제약사항

1. Code Expression에서는 입력 값 길이의 제한을 두지 않습니다.

이로 인하여 입력 값이 많아 질 경우 프로세스 실행 시 시스템부하, 속도저하 등의 문제가 발생할 수 있습니다.

2. 사용자는 Code Expression을 통해 C# 문법을 사용하여 직접 코딩을 할 수 있습니다.

제공하는 DLL 파일 목록은 다음과 같습니다.

기존(1.X)	추가(2.0)
DataBase 관련 - Oracle.ManagedDataAccess.dll - MySql.Data.dll	Internet Explore 구조 - SHDoc.dll
Json 관련 - Newtonsoft.Json.dll	HTML 구조 - MSHTML.dll
Office 관련 - Microsoft.Office.Interop.Excel.dll - System.IO.FileSystem.Primitives.dll - System.IO.Packaging.dll - DocumentFormat.OpenXml.dll - WindowsBase.dll	파일 압축/해제 관련 - ZipFile.dll - ZipArchive.dll, - ZipArchiveEntry.dll  이미지 관련 - Drawing.dll: 이미지 컨트롤  메일 발송 관련 - ImapX.dll - OpenPOP.NET.dll - Outlook.dll

### @@ 주의 @@

※ DataTableArray 변수를 할당 시 문법 오류 발생할 수 있음

DataTable 클래스가 System.Data.DataTable 과 Microsoft.Office.Interop.Excel.DataTable 두 곳에 정의되어 있어, Full Namespace를 작성해야 합니다.

※ Code Expression 내에서 Using문과 Class 선언이 불가합니다.

### 5.5.3. DataTable

DataTable은 DataTable 변수를 생성, 수정, 삽입, 삭제 할 수 있는 관련 기능을 모아 놓은 것입니다.

DataTable은 다음과 같은 Event Item으로 구성되어 있습니다.

- Add Column
- Add Row
- Clear DataTable
- DataTable to Popup
- DataTable to String
- Delete Column
- Delete Row
- Filter DataTable
- Get Data
- Lookup DataTable
- Loop DataTable
- Merge DataTables
- Read DataRow
- Read Specific Data
- Remove Duplicated Rows
- Set Data
- Sort DataTable
- String to DataTable

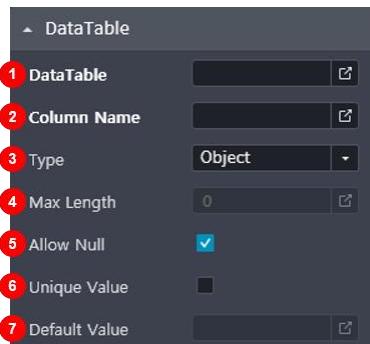
#### @@ 주의 @@

※ DataTable 관련 Event Item 사용에 대한 제약사항

DataTable에 관련된 Event Item은 개별 실행을 지원하지 않습니다.

## ■ Add Column

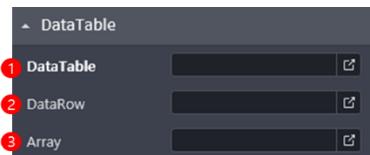
DataTable에 새로운 Column을 추가합니다.



No.	설명
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)
2	새로 추가할 Column의 이름을 지정합니다. (Type: String)
3	새로 추가할 Column의 Data Type을 Dropdown하여 지정합니다.
4	String형식인 경우 Column 입력 값의 최대 길이를 지정합니다. (Type: String, Default: 100)
5	입력 값에 Null을 허용할 것인지 여부를 지정합니다. > Checked: 허용 > Unchecked: 허용하지 않음
6	Column의 값이 유일 값을 갖게 할 것인지 여부를 지정합니다. > Checked: 유일 값 > Unchecked: 중복 허용
7	초기값/기본값을 지정할 것인지 선택적으로 사용합니다. <b>@@ 주의 @@</b> 기본값은 String, Int, Double, Boolean 또는 Data형으로 3번에서 지정한 형식과 호환이 되어야 합니다.

## ■ Add Row

DataTable에 새로운 Row를 추가합니다.



No.	설명	
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)	
2	삽입할 DataRow 변수 명을 지정합니다. (Type: DataRow)	@@ 주의 @@
3	삽입할 Array 형식 및 Array Expression 데이터를 입력합니다. Type: StringArray, IntArray, DoubleArray, BooleanArray, DataArray, ObjectArray 형식입니다. ※ Array Expression은 하기 참조	※ 2 또는 3 항목 중 1개만 선택입력 가능 합니다.

@@ 주의 @@

※ Array Expression: Start = "{", End = "}", 구분자= "," Ex) {"1st Col", "2nd Col", "2019-07-10", 2314}

## ■ Clear DataTable

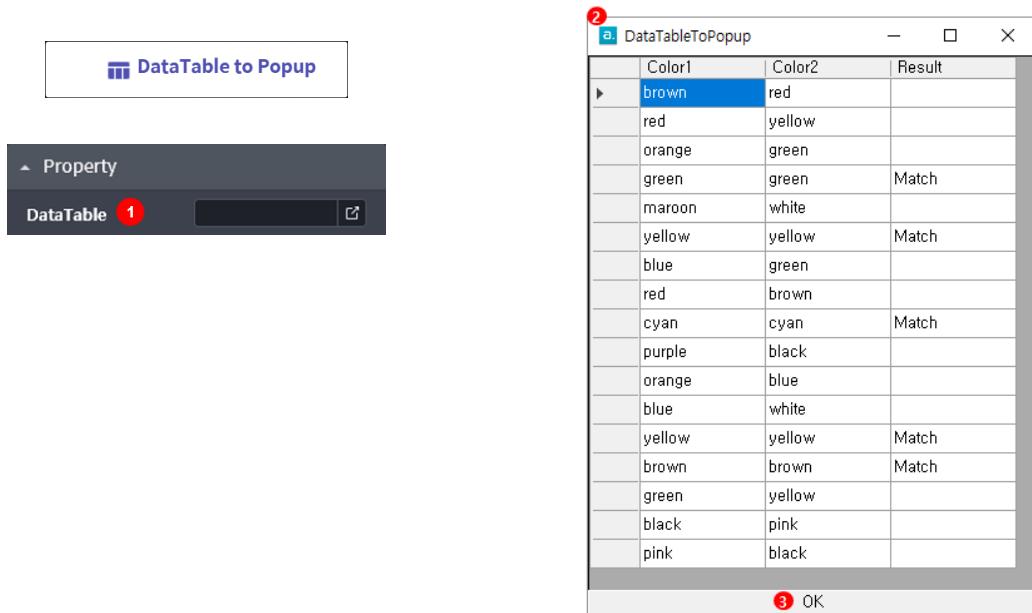
DataTable의 Column 구조는 남겨두고 모든 Row만 삭제합니다.



No.	설명
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)

## ■ DataTable To Popup

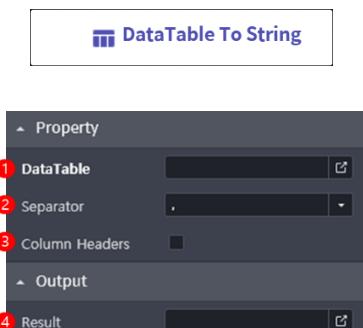
DataTable의 내용을 Popup으로 보여줍니다.



No.	설명
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)
2	Popup 출력 결과
3	OK버튼을 누르면 Popup이 닫히고 후속 Script를 수행합니다.

## ■ DataTable to String

DataTable을 지정된 구분자(Separator)로 분리해 String으로 반환합니다.

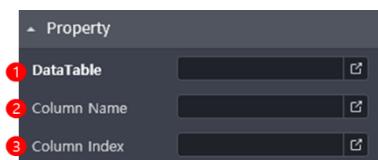


No.	설명
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)
2	Column의 구분자를 지정합니다. > Dropdown으로 "", Tab, "/" 선택 가능

3	출력할 정보에 Column Header를 포함할 것인지 여부를 선택합니다. >Checked: 첫 번째 Row 정보를 Column Header로 인식 >Unchecked: 첫 번째 Row 정보를 Data로 인식
4	수행 결과를 저장할 String 변수 명을 지정합니다. (Type: String)

## ■ Delete Column

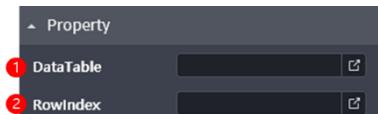
DataTable에 저장된 Column을 삭제합니다.



No.	설명
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)
2	삭제할 Column의 이름을 지정합니다. (Type : String)
3	삭제할 Column의 Index를 지정합니다. (Type : Int)

## ■ Delete Row

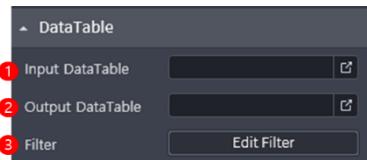
DataTable에 지정된 Index의 Row를 삭제합니다.



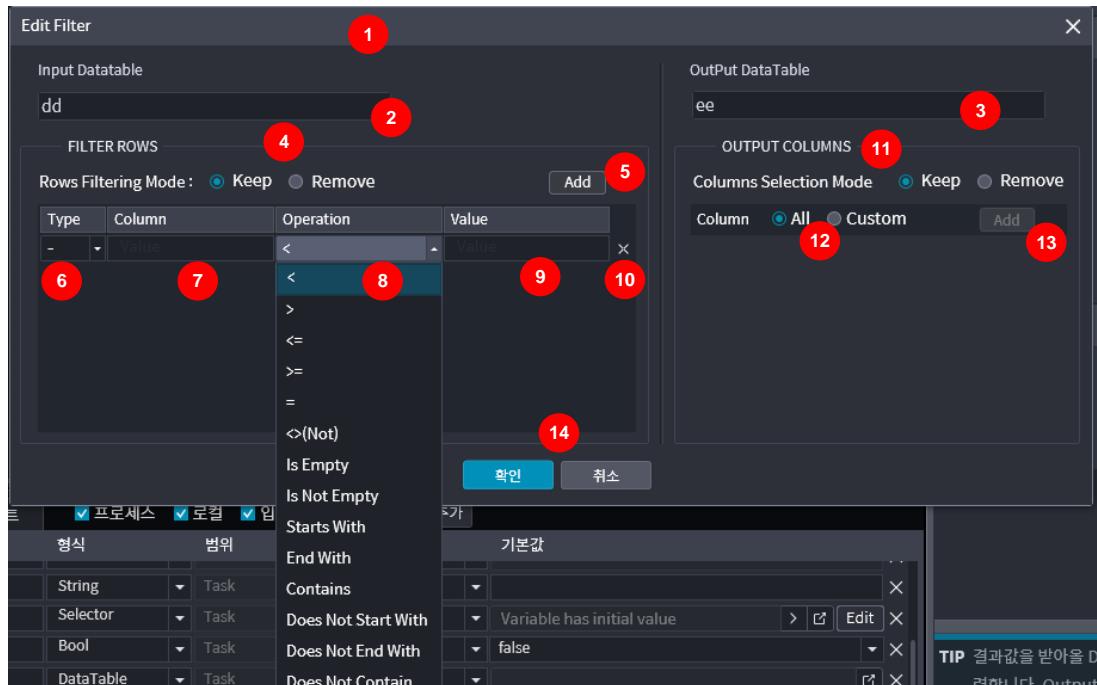
No.	설명
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)
2	삭제하고자 하는 Row의 Index를 지정합니다. (Type: Int)

## ■ Filter DataTable

DataTable의 지정된 Column의 Filter조건을 만족하는 Row와 Column을 유지(Keep) 또는 삭제(Remove)하여 결과를 Output DataTable에 저장합니다.



No.	설명
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)
2	실행 결과를 저장할 DataTable 변수 명을 지정합니다. (Type: DataTable)
3	Filter조건을 지정할 다음과 같은 PopUp 창을 Open합니다.



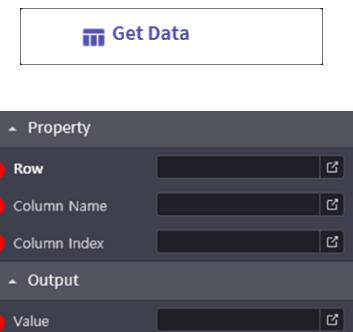
<DATATABLE Filter Edit Popup 화면>

No.	설명
1	DataTable Filter Edit Popup 화면입니다.
2	지정한 Input DataTable 변수 명을 표시합니다. ※ 편집은 불가합니다.
3	지정한 Output DataTable 변수 명을 표시합니다.

	※ 편집은 불가합니다.
4	Row를 Filtering하기 위한 Mode를 선택합니다. > Keep: 유지, Remove: 제거를 Radio 버튼으로 선택
5	[Add] 버튼으로 Filtering할 Row를 추가할 수 있습니다.
6	Row를 Filtering할 조건으로 다중연산이 필요 시 AND 또는 OR를 선택할 수 있습니다. 2번째 행부터 Type(AND/OR)을 선택할 수 있습니다.
7	Row를 Filtering할 조건의 Column을 지정합니다. > 입력Type: String 시 Column Name, Int 시 Column Index
8	Row를 Filtering할 조건 연산자를 선택할 수 있습니다. > 제공연산자: < , > , <= , >= , = , <>(Not) , Is Empty , Is Not Empty , Starts With , Ends With , Contains , Does Not Start With , Does Not End With , Does Not Contain
9	비교대상 Column의 Data와 조건 연산자에 대입할 값을 입력합니다. > 입력 값은 속성에 따른 Type형식을 직접입력 또는 동일한 Type의 변수 명을 선택하여 입력
10	Row를 Filtering할 조건의 해당 Row를 삭제할 수 있습니다.
11	Filter 조건 비교 후 결과로 나올 Column의 선택 Mode입니다. > Keep: 유지, Remove: 제거를 Radio 버튼으로 선택
12	출력 대상 Column을 All(전체), Custom(사용자 지정)으로 선택합니다.
13	대상을 Column을 Custom(사용자 지정)으로 선택 시 활성화하고 추가할 수 있습니다.
14	해당 내용을 확정시 [확인] 버튼으로 저장하고, [취소] 시 Canvas로 되돌아갑니다.

## ■ Get Data

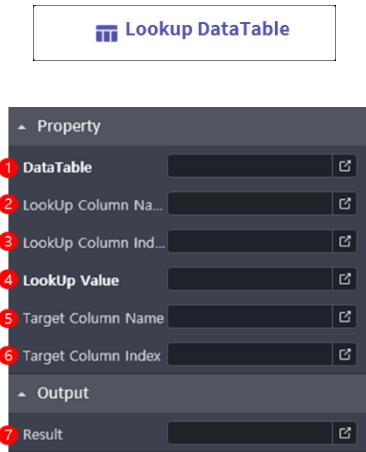
Loop DataTable Event Item내에서 작동하며, 해당 Loop의 Row에 지정된 Column의 Data를 반환합니다.



No.	설명	
1	가져오고자 하는 Column의 Row를 변수로 지정합니다. (Type: DataRow)	
2	해당 Row에서 읽을 Column의 이름을 지정합니다. (Type : String)	※ 2 또는 3 항목 중
3	해당 Row에서 읽을 Column의 Index를 지정합니다. (Type : Int)	1개만 선택입력 가능
4	읽은 Data를 저장하고자 하는 변수 명을 지정합니다. >Type: String, Int, Double, Boolean, Data, Object	

## ■ Lookup DataTable

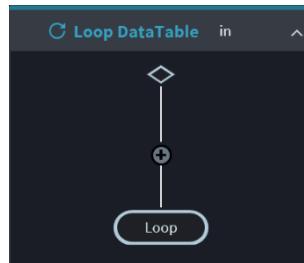
DataTable내 Lookup Column의 Data와 Lookup Value이 같은 Row의 Target Column의 Data 집합을 Array로 반환합니다.

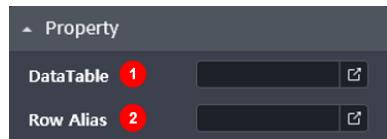


No.	설명
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)
2	검색하고자 하는 기준 Column의 이름을 지정합니다. (Type: String)
3	검색하고자 하는 기준 Column의 Index를 지정합니다. (Type: Int)
4	검색 Column에서 찾고자 하는 값을 지정합니다. (Type: String, Int, Double, Boolean, Data, Object)
5	출력하고자 하는 기준 Column의 이름을 지정합니다. (Type: String)
6	출력하고자 하는 기준 Column의 Index를 지정합니다. (Type: Int)
7	검색 결과를 저장하고자 하는 Array 변수 명을 지정합니다. (Type: Array(String, Int, Double, Boolean, Data, Object))

## ■ Loop DataTable

DataTable에 저장된 Row 개수만큼 반복을 수행합니다. (각 Loop마다 Index 0번부터 차례로 Row Alias에 반환)





No.	설명
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)
2	Loop DataTable 내에서 Get/Set Data와 연결하기 위한 Row Alias를 직접 입력합니다. (Type: String)

## ■ Merge DataTable

Source DataTable과 Destination DataTable을 Merge합니다.



No.	설명
1	실행하고자 하는 원본 DataTable 변수 명을 지정합니다. (Type: DataTable)
2	Merge하고자 하는 대상을 DataTable 변수 명을 지정합니다. (Type: DataTable)
3	Source에는 존재하나 Destination에 없는 Column이 있을 경우 처리방식을 지정합니다. >Add(Default): Destination에 Column을 추가 >Ignore: Destination에 없는 Column은 무시 >Error: Column 정보가 일치하지 않을 경우 Error 발생 >AddWithKey: Primary Key를 기준으로 Column을 추가
4	수행 결과를 저장할 DataTable 변수 명을 지정합니다. (Type: DataTable)

## ■ Read DataRow

DataTable에 지정된 Index의 Row를 DataRow 형식으로 반환합니다.

Read DataRow

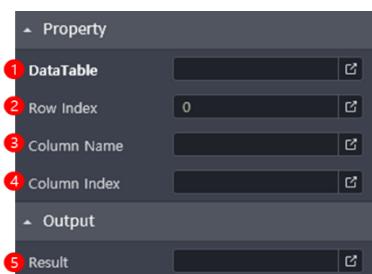


No.	설명
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)
2	저장하고자 하는 Row의 Index를 지정합니다. (Type: Int, Default Value:0)
3	저장하고자 하는 DataRow 변수 명을 지정합니다. (Type: DataRow)

## ■ Read Specific Data

DataTable의 지정된 Index Row에 지정된 Column의 Data를 반환합니다

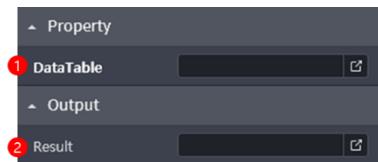
Read Specific Data



No.	설명	
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)	
2	저장하고자 하는 Row의 Index를 지정합니다. (Type: Int, Default Value: 0)	
3	해당 Row에서 저장할 Column의 Name을 지정합니다. (Type: String)	※ 3 또는 4 항목 중
4	해당 Row에서 저장할 Column의 Index를 지정합니다. (Type: Int)	1개만 선택입력 가능
5	저장하고자 하는 변수 명을 지정합니다. (Type: String, Int, Double, Boolean, Data, Object)	

## ■ Remove Duplicated Rows

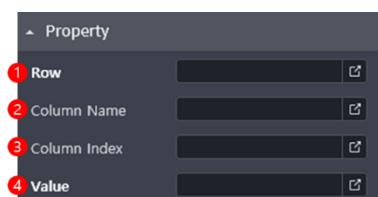
DataTable에 저장된 Row 가운데 중복된 Row를 1개만 남겨두고 삭제합니다.



No.	설명
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)
2	실행 결과를 저장하고자 하는 변수 명을 지정합니다. (Type: DataTable)

## ■ Set Data

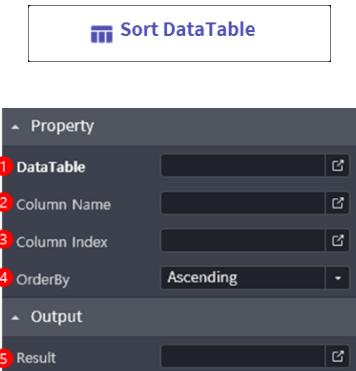
DataTable Loop 내에서 작동하며, 해당 Loop의 Row 내 지정된 Column에 Data를 저장합니다.



No.	설명
1	실행하고자 하는 DataRow 변수 명을 지정합니다. (Type: DataTable)
2	해당 Row에서 저장할 Column의 Name을 지정합니다. (Type: String)
3	해당 Row에서 저장할 Column의 Index를 지정합니다. (Type: Int)
4	저장하고자 하는 Data를 지정합니다. (Type: String, Int, Double, Boolean, Data, Object)

## ■ Sort DataTable

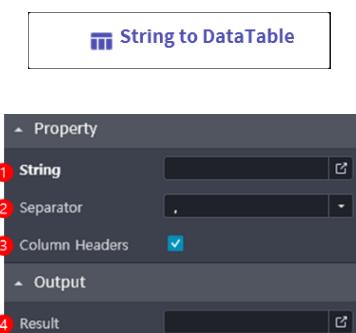
DataTable의 지정된 Column을 기준으로 오름/내림차순 정렬 후 결과를 Result DataTable에 저장합니다.



No.	설명
1	실행하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)
2	정렬하고자 하는 기준 Column의 Name을 지정합니다. (Type: String)
3	정렬하고자 하는 기준 Column의 Index를 지정합니다. (Type: Int) ※ 2 또는 3 항목 중 1개만 선택입력 가능
4	정렬 방식을 오름차순(Ascending)과 내림차순(Descending)으로 지정합니다. >Default: Ascending
5	정렬 결과를 저장하고자 하는 DataTable 변수 명을 지정합니다. (Type: DataTable)

## ■ String to DataTable

구조화된 String을 지정된 구분자(Separator)를 기준으로 구분해 DataTable로 반환합니다.



No.	설명
1	실행하고자 하는 String 변수 명을 지정합니다. (Type: String)
2	Column의 구분자를 지정합니다. >Dropdown으로 "", , Tab, "/" 선택 가능 <b>@@ Info @@</b>

	위 세 가지 구분자외에 사용자가 직접 입력도 가능합니다.
3	입력할 String에 Column Header 포함 여부를 선택합니다. >Checked: 첫 번째 Row 정보를 Column Header로 인식 >Unchecked: 첫 번째 Row 정보를 Data로 인식
4	수행 결과를 저장할 DataTable 변수 명을 지정합니다. (Type: DataTable)

## 5.6. System

Windows에서 제공하는 System 관련 기능을 모아 놓은 것입니다.

System은 총 6개의 카테고리로 구성되어 있습니다.

- Clipboard
- Credential
- Cryptography
- Dialog
- Directory
- File

### 5.6.1 Clipboard

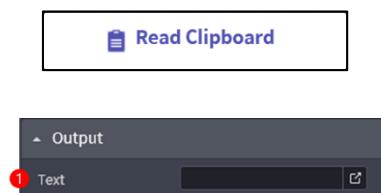
Windows의 Clipboard를 조작할 수 있는 기능을 모아놓은 카테고리입니다.

Clipboard는 다음과 같은 Event Item으로 구성되어 있습니다.

- Read Clipboard
- Write Clipboard

#### ■ Read Clipboard

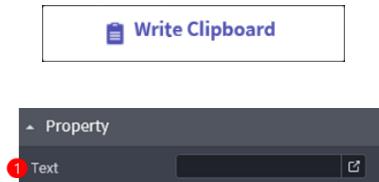
Clipboard에 저장한 내용을 불러올 수 있습니다.



No.	설명
1	불러온 결과를 저장할 변수 명을 입력합니다. (Type: String)

## ■ Write Clipboard

Clipboard에 저장하고자 하는 내용을 입력할 수 있습니다.



No.	설명
1	Clipboard에 입력하고자 하는 내용을 입력합니다. (Type: String)

## 5.6.2 Cryptography

Cryptography는 암호화 관련 기능을 모아놓은 카테고리입니다.

Cryptography는 다음과 같은 Event Item으로 구성되어 있습니다.

- Decrypt File
- Decrypt Text
- Encrypt File
- Encrypt Text

## ■ Decrypt File

암호화된 파일을 복호화 합니다.

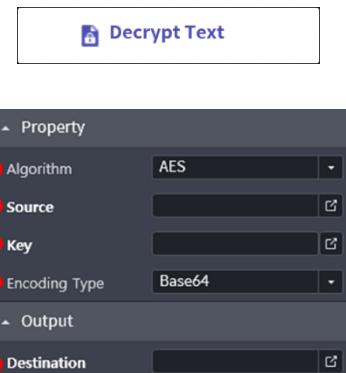


No.	설명
1	파일의 복호화 방식을 선택합니다. >AES / DES / RE2 / Rijndael

2	복호화할 대상 파일의 경로를 입력합니다. (Type: String)
3	파일 복호화에 사용되는 Key 값을 입력합니다. (Type: String, * 처리 안함)
4	복호화된 값을 파일로 저장할 때 사용되는 인코딩 방식을 선택합니다. (Type: Encoding, None/Base64/BinHex)
5	복호화된 파일을 저장할 디렉터리 위치를 지정합니다. (Type: String)
6	기존에 동일한 파일명이 있을 경우 덮어쓰기 여부를 지정합니다. >Checked: 덮어쓰기 >Unchecked 동일한 파일명이 있을 경우 오류가 표시됩니다.

## ■ Decrypt Text

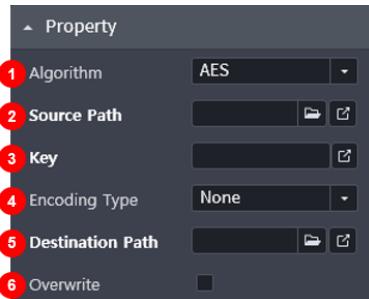
암호화된 Text를 복호화 합니다.



No.	설명
1	Text를 복호화하는 방식을 선택합니다. >AES / DES / RC2 / Rijndael
2	복호화 하려는 원본 Text를 지정합니다.
3	Text를 복호화하는데 사용되는 Key 값을 지정합니다 (Type: String, * 처리 안함)
4	Text를 복호화하는데 사용되는 인코딩 방식을 선택합니다. (Type: Encoding, Base64/BinHex)
5	복호화한 결과를 저장할 변수 명을 입력합니다. (Type : String)

## ■ Encrypt File

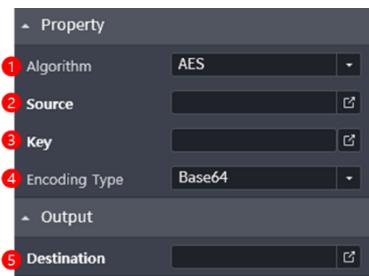
사용자가 지정하는 File을 암호화 합니다.



No.	설명
1	파일의 암호화 방식을 선택합니다. >AES / DES / RE2 / Rijndael
2	암호화 대상 파일의 경로를 입력합니다. (Type: String)
3	파일 암호화에 사용되는 Key 값을 입력합니다. (Type: String, * 처리 안함)
4	암호화된 값을 파일로 저장할 때 사용되는 인코딩 방식을 선택합니다. (Type: Encoding, None/Base64/BinHex)
5	암호화된 파일을 저장할 디렉터리 위치를 지정합니다. (Type: String)
6	기존에 동일한 파일명이 있을 경우 덮어쓰기 여부를 지정합니다. >Checked: 덮어쓰기 >Unchecked: 동일한 파일명이 있을 경우 오류가 표시됩니다.

## ■ Encrypt Text

사용자가 지정하는 Text를 암호화 합니다.



No.	설명
1	Text를 암호화하는 방식을 선택합니다.

	>AES / DES / RC2 / Rijndael
2	암호화 하려는 원본 Text를 지정합니다.
3	Text를 암호화하는데 사용되는 Key 값을 지정합니다 (Type: String, * 처리 안함)
4	Text를 암호화하는데 사용되는 인코딩 방식을 선택합니다. (Type: Encoding, Base64/BinHex)
5	암호화한 결과를 저장할 변수 명을 입력합니다. (Type : String)

### 5.6.3 Dialog

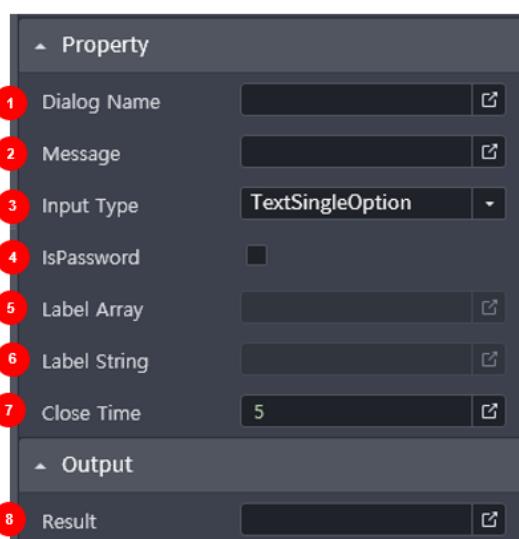
Dialog는 다이얼로그 관련 기능을 모아놓은 카테고리입니다.

Dialog는 다음과 같은 Event Item으로 구성되어 있습니다.

- Input Dialog Box
- Message Dialog Box
- Message Yes No
- Text Input Box

#### ■ Input Dialog Box

사용자에게 메시지창을 통해 질문 내용을 보여주고, 이것에 대한 의견을 텍스트, 라디오, 리스트로 선택할 수 있는 Input Dialog Box를 생성합니다.

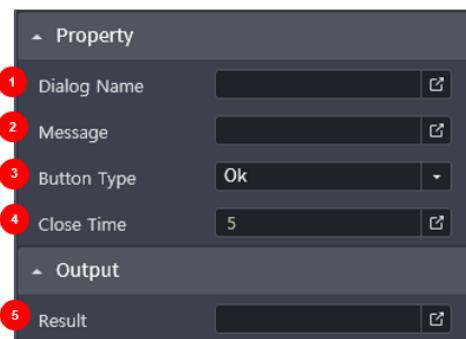


No.	설명
1	<b>Dialog Name :</b> 사용자에게 보여줄ダイアル로그창의 제목을 입력합니다. Dialog Name은 String형식의 값입니다.
2	<b>Message :</b> 사용자에게 보여줄ダイアル로그창의 내용을 입력합니다. Message는 String형식의 값입니다.
3	<b>Input Type :</b> 사용자에게 보여줄ダイアル로그창의 형태를 선택합니다. >Text Single Option은 사용자가 텍스트를 최대 100글자 입력 받을 수 있습니다. >Radio Multi Option은 사용자가 입력한 문자열/배열 값으로 최대 4개의 라디오 버튼을 생성합니다. >List Multi Option은 사용자가 입력한 문자열/배열 값으로 최대 50개의 콤보박스 리스트를 생성합니다.
4	<b>Password :</b> Password 여부를 설정합니다.
5	<b>Label Array :</b> 사용자가 선택 할 옵션의 내용을 입력합니다. Label Array는 String Array형식의 값입니다. Label Array 또는 Label String 둘 중의 하나의 값만 입력 할 수 있습니다.
6	<b>Label String :</b> 사용자가 선택 할 옵션의 내용을 입력합니다. Label String은 String형식의 값입니다. Label String의 값은 콤마로 분리하여 입력 합니다. 예시 : “사과,바나나,파인애플” Label Array 또는 Label String 둘 중의 하나의 값만 입력 할 수 있습니다.
7	<b>Close Time :</b> 사용자가ダイアル로그 창이 사라지게 할 시간을 입력합니다. Close Time은 Unit형식의 값입니다. 단위는 분입니다.
8	<b>Result :</b> 사용자가 입력한 결과값을 받아올 변수 명을 입력합니다. Result는 String형식의 변수입니다.

## ■ Message Dialog Box

사용자의 텍스트 물음에 옵션을 선택 할 수 있는 Message Box를 생성합니다.

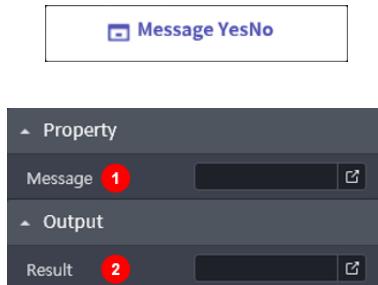
선택 옵션은 확인, 확인/취소, 예/아니요, 예/아니요/취소이며, 사용자는 옵션 중 하나를 선택 할 수 있습니다.



No.	설명
1	<b>Dialog Name</b> : 사용자에게 보여줄 메시지창의 제목을 입력합니다. Dialog Name은 String형식의 값입니다.
2	<b>Message</b> : 사용자에게 보여줄 메시지창의 내용을 입력합니다. Message는 String형식의 값입니다.
3	<b>Button Type</b> : 사용자에게 보여줄 메시지창 버튼의 형태를 선택합니다. >Ok : 확인 버튼 >OkCancel : 확인/취소 버튼 >YesNo : 예/아니요 버튼 >YesNoCancel : 예/아니요/취소 버튼
4	<b>Close Time</b> : 사용자가 메시지창이 사라지게 할 시간을 입력합니다. Close Time은 Unit형식의 값입니다. 단위는 분입니다.
5	<b>Result</b> : 사용자가 입력한 결과값을 받아올 변수 명을 입력합니다. Result는 String형식의 변수입니다.

## ■ Message Yes No

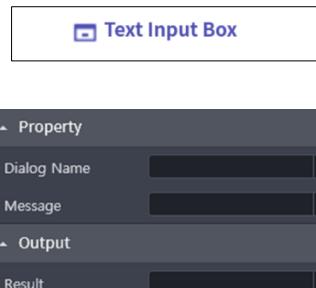
Yes/No 버튼을 가진 Message Box를 생성합니다.



No.	설명
1	Message Box에 출력할 메시지를 입력합니다. (Type: String)
2	결과를 저장할 변수 명을 지정합니다. (Type: Boolean)

## ■ Text Input Box

Dialog 창(Input Box)으로 직접 Text를 입력 받기 위한 기능을 제공합니다.



No.	설명
1	Dialog(Input Box)의 Title Area에 출력할 Name을 입력합니다. (Type: String)
2	Dialog(Input Box)의 Message Area에 표시할 Message를 입력합니다. (Type: String)
3	Dialog(Input Box)의 Result Area에 입력한 값을 저장할 변수 명을 입력합니다. (Type: String)

Event Item을 실행 시 다음과 같은 Dialog(Input Box) 창으로 표시합니다.



참조. 실행결과

## 5.6.4 Direct

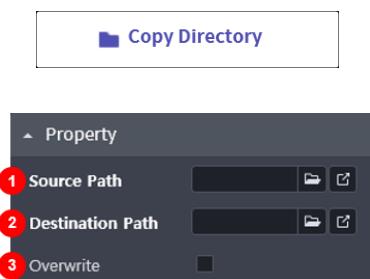
Directory 관련 기능을 모아놓은 카테고리입니다.

Directory는 다음과 같은 Event Item으로 구성되어 있습니다.

- Copy Directory
- Create Directory
- Delete Directory
- Exist Directory
- Move Directory

### ■ Copy Directory

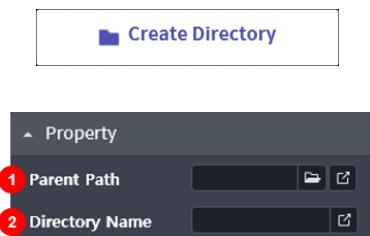
사용자가 지정하는 폴더를 복사합니다.



No.	설명
1	복사할 원본 Directory 경로를 지정합니다. (Type: String)
2	생성할 목적지 경로를 지정합니다. (Type: String)
3	기존에 동일한 Directory명이 있을 경우 덮어쓰기 여부를 지정합니다. >Checked: 덮어쓰기 >Unchecked: 동일한 Directory명이 있을 경우 오류가 표시됩니다.

### ■ Create Directory

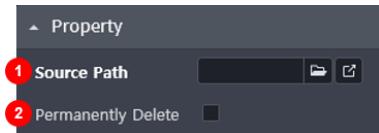
지정된 경로에 새로운 Directory를 생성합니다.



No.	설명
1	Directory가 생성될 경로를 지정합니다. (Type: String)
2	생성할 Directory명을 지정합니다. (Type: String)

## ■ Delete Directory

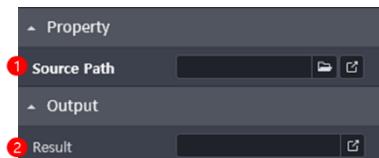
지정된 경로에 있는 Directory를 삭제합니다.



No.	설명
1	삭제할 Directory가 위치한 경로를 지정합니다. (Type: String)
2	Directory를 영구 삭제 유무를 선택합니다. >Checked: 영구 삭제 >Unchecked: 휴지통으로 이동

## ■ Exist Directory

지정된 경로에 Directory의 존재여부를 확인합니다.



No.	설명
1	대상 Directory가 위치한 경로를 지정합니다. (Type: String)
2	존재유무의 결과값을 저장할 변수 명을 입력합니다. (Type: Boolean)

## ■ Move Directory

지정된 경로로 Directory를 이동합니다.



No.	설명
1	이동시킬 원본 Directory의 경로를 지정합니다. (Type: String)
2	이동할 목적지 Directory의 경로를 지정합니다. (Type: String)
3	Directory를 덮어쓰기 여부를 선택합니다. >Checked: 덮어쓰기 >Unchecked: 동일한 Directory가 있을 경우 오류가 표시됩니다.

## 5.6.5 File

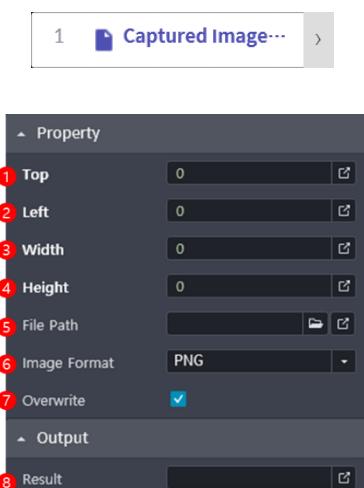
File 관련 기능을 모아놓은 카테고리입니다.

File은 다음과 같은 Event Item으로 구성되어 있습니다.

- Captured Image to File
- Copy File
- Create File
- Delete File
- Exist File
- Move File
- Read File to String
- Unzip
- Wait File Change
- Write File From String
- Zip

### ■ Captured Image to File

캡처된 이미지를 파일로 저장합니다. (Indicator 제공)



No.	설명
1	Indicator로 캡처한 이미지의 Top position을 가져옵니다. (Type: Int)
2	Indicator로 캡처한 이미지의 Left position을 가져옵니다. (Type: Int)
3	Indicator로 캡처한 이미지의 너비를 가져옵니다. (Type: Int)
4	Indicator로 캡처한 이미지의 높이를 가져옵니다. (Type: Int)
5	캡처한 이미지를 저장할 경로를 입력합니다. (Type : String)
6	저장할 이미지의 포맷을 선택합니다. (PNG, JPG, BMP, TIFF)
7	파일이 존재할 경우 덮어쓸 것인지 설정합니다.
8	Indicator로 캡처한 이미지 파일을 저장할 경로를 지정합니다. (Type: String)

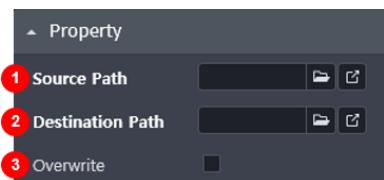
## @@주의@@

※ 정상 입력: Top, Left, Width, Height의 값이 정수(Int)이어야 함.

비정상 입력: Top, Left, Width, Height값이 실수, 문자, 특수 문자 그 외 다수 입력 시.

## ■ Copy File

파일을 복사합니다.



No.	설명
1	복사할 원본파일의 전체 경로명을 지정합니다. (Type: String)
2	사본 파일이 위치할 목적지의 전체 경로명을 지정합니다. (Type: String)
3	복사한 파일이 이미 존재할 경우 덮어쓰기 여부를 선택합니다. >Checked: 덮어쓰기 >Unchecked: 동일한 파일명이 있을 경우 오류가 표시됩니다.

## ■ Create File

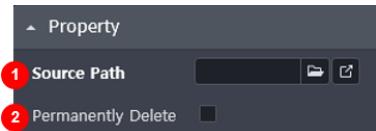
파일을 생성합니다.



No.	설명
1	생성할 파일이 위치할 전체 경로명을 지정합니다. (Type: String)
2	생성할 파일명을 확장자를 포함하여 지정합니다. (Type: String)
3	생성한 파일이 이미 존재할 경우 덮어쓰기 여부를 선택합니다. >Checked: 덮어쓰기 >Unchecked: 동일한 파일명이 있을 경우 오류가 표시됩니다.

## ■ Delete File

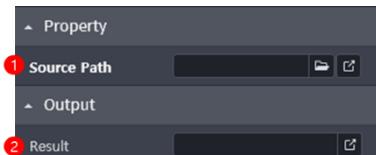
파일을 삭제합니다.



No.	설명
1	삭제할 파일이 위치할 전체 경로명을 지정합니다. (Type: String)
2	삭제할 파일의 영구 삭제 유무를 선택합니다. >Checked: 영구 삭제 >Unchecked: 휴지통으로 이동

## ■ Exist File

파일 존재여부를 확인합니다.



No.	설명
1	존재하는지 알아볼 파일의 전체 경로명을 지정합니다. (Type: String)
2	파일 존재 유무에 대한 결과값을 저장할 변수 명을 지정합니다. (Type : Boolean)

## ■ Move File

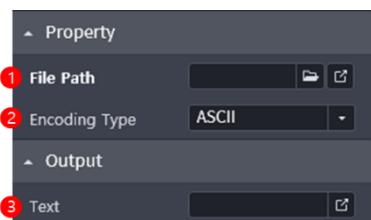
파일을 이동합니다.



No.	설명
1	이동시킬 원본 파일의 전체 경로를 지정합니다. (Type: String)
2	이동할 목적지 파일의 전체 경로를 지정합니다. (Type: String)
3	Directory를 덮어쓰기 여부를 선택합니다. >Checked: 덮어쓰기 >Unchecked: 동일한 Directory가 있을 경우 오류가 표시됩니다.

## ■ Read File To String

지정된 경로의 파일명을 String 변수로 저장합니다.

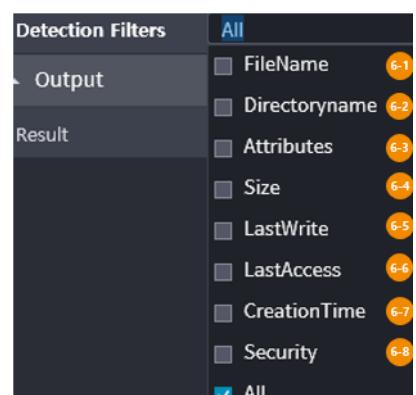
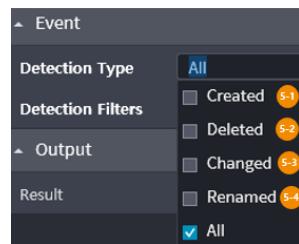
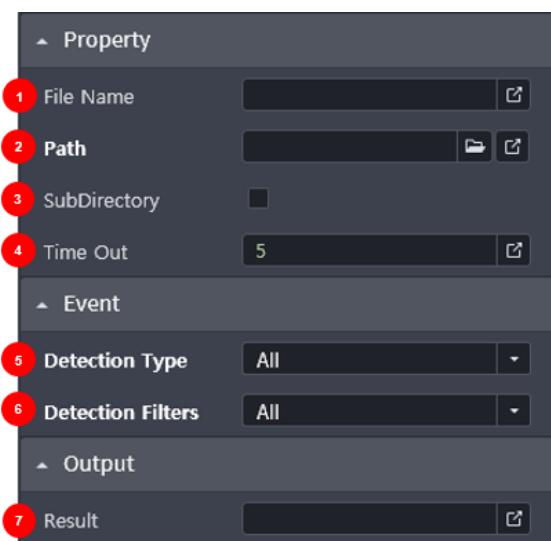
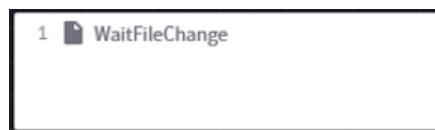


No.	설명
1	읽어 올 파일명의 전체 경로를 지정합니다. (Type: String)
2	저장할 파일명의 Encoding Type을 Dropdown하여 지정합니다. >Encoding Type: Default, ASCII, UTF7, UTF8, UTF16, UTF32, Unicode
3	읽어 온 파일명의 Text를 저장할 변수 명을 지정합니다. (Type: String)

## ■ Write File Change

특정폴더내의 파일이름을 필터 적용하여, 파일의 생성, 삭제, 변경, 이름 변경에 대한 내용을 감지하여 알려주는 이벤트 아이템입니다.

Detection Type은 어떻게 감지할 것인지를 의미합니다. 파일 혹은 폴더가 변경되어지는 방식이나 방법(생성, 삭제, 변경, 이름 재설정)을 의미합니다. Detection Filters는 무엇을 감지할 것인지를 의미합니다. 파일 혹은 폴더의 형식(파일이름, 폴더이름, 파일크기 등)을 의미합니다. Detection Type과 Detection Filters를 합치면 무엇이 어떻게 변경되는지 감지한다가 됩니다. 예를 들어 Detection Filters를 FileName으로 체크하고, Detection Type을 Created로 체크하면 “파일이름의 파일이 생성되는 것을 감지한다.”를 의미합니다.



No.	설명
1	File Name : 대기하여 감시할 파일 이름을 입력하세요. File Name은 String형식의 값입니다.
2	Path : 대기하여 감시할 경로를 입력하세요. Path는 string형식의 값입니다.
3	SubDirectory : 하위 폴더 포함 여부를 입력합니다. SubDirectory는 Bool형식의 값입니다.
4	Time Out : 대기를 종료 할 시간을 입력합니다. Time Out은 UInt형식의 값입니다. Time Out은 분 단위입니다. 0 입력 시 이벤트가 발생할때까지 계속 대기 합니다.
5	Result : 결과값을 받아올 변수명을 입력합니다. Result는 Bool형식의 변수입니다.
6	<p>Detection Type : 대기하여 감시할 형식을 입력합니다.</p> <ul style="list-style-type: none"><li>&gt; Created : 해당 파일 또는 폴더가 생성 되었을 때를 감지 합니다.</li><li>&gt; Deleted : 해당 파일 또는 폴더가 삭제 되었을 때를 감지 합니다.</li><li>&gt; Changed : 해당 파일 또는 폴더가 변경 되었을 때를 감지 합니다.</li><li>&gt; Renamed : 해당 파일 또는 폴더 이름이 변경 되었을 때를 감지 합니다.</li></ul>

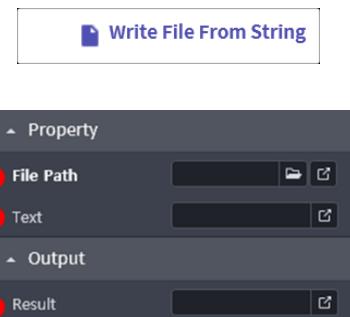
---

	<p>※ Detection Type 별 연계되는 Detection Filters는 아래와 같습니다.</p> <p>Renamed - FlieName, DirectoryName</p> <p>Created - FileName,DirectoryName</p> <p>Deleted - FileName,DirectoryName</p> <p>Changed - Attributes, Size, LastWrite, LastAccess, CreationTime, Security</p>
7	<p>Detection Filters : 대기하여 감시할 필터를 입력합니다.</p> <ul style="list-style-type: none"> <li>&gt; FileName : 파일 이름입니다.</li> <li>&gt; Directoryname : 디렉터리의 이름입니다.</li> <li>&gt; Attributes : 파일 또는 폴더의 특성입니다.</li> <li>&gt; Size : 파일 또는 폴더의 크기입니다.</li> <li>&gt; LastWrite : 파일 또는 폴더에 마지막으로 기록된 날짜입니다.</li> <li>&gt; LastAccess : 파일 또는 폴더를 마지막으로 열었던 날짜입니다.</li> <li>&gt; CreationTime : 파일 또는 폴더를 만든 시간입니다.</li> <li>&gt; Security : 파일 또는 폴더의 보안 설정입니다.</li> </ul>

---

## ■ Write File From String

String을 지정된 경로의 파일로 저장합니다.



No.	설명
1	저장할 파일의 전체 경로를 지정합니다. (Type: String, 기존 파일 존재 시 Overwrite 수행)
2	파일로 저장할 Text를 결과값을 갖고 있는 변수 명을 지정합니다. (Type: String)
3	실행 결과의 성공/실패 여부를 저장할 변수 명을 지정합니다. (Type: Boolean)

## 5.6.6 Windows Credential

System에서 사용하는 Credential(자격증명)를 관리합니다.

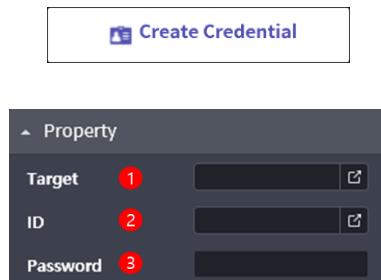
Credential은 WorkCenter의 Asset을 이용할 수 없는 환경에서 보다 안전하게 Application 계정정보를 관리하는 기능을 위하여 사용합니다.

Credential은 다음과 같은 Event Item으로 구성되어 있습니다.

- Create Credential
- Delete Credential
- Get Credential
- Text to SecureString

### ■ Create Credential

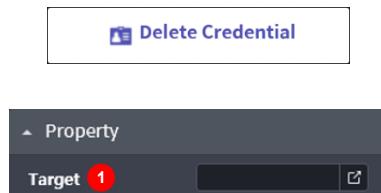
Credential을 생성합니다.



No.	설명
1	등록할 Credential의 이름을 지정합니다. (Type: String)
2	사용자 ID를 입력합니다. (Type: String)
3	사용자 Password를 입력합니다. (Type: String)

### ■ Delete Credential

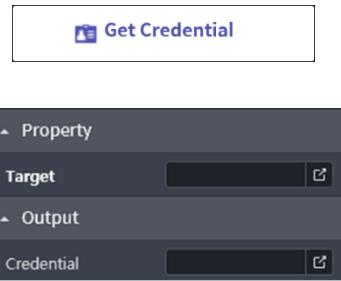
Credential을 삭제합니다.



No.	설명
1	삭제하려는 Credential의 이름을 입력합니다. (Type: String)

## ■ Get Credential

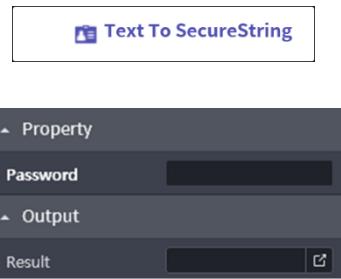
Credential을 가져옵니다.



No.	설명
1	가져오려는 Credential의 이름을 지정합니다. (Type: String)
2	Credential의 정보가 저장될 변수 명을 지정합니다. (Type: Credential)

## ■ Text to SecureString

사용자가 지정한 Text를 암호화를 위하여 SecureString으로 변환합니다.



No.	설명
1	사용자의 Password를 입력합니다. (Type: String)
2	결과를 저장할 변수 명을 지정합니다. (Type: SecureString)

## 5.6.7 Zip

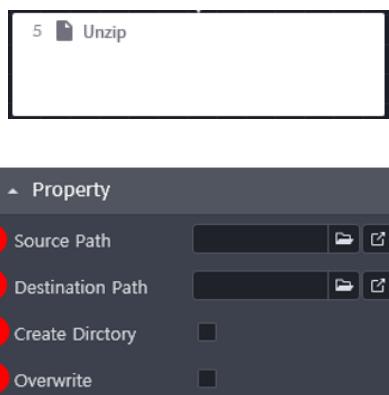
파일 또는 폴더를 압축하거나, 압축해제 할 때 사용합니다.

Zip은 다음과 같은 Event Item으로 구성되어 있습니다.

- Unzip
- Zip

### ■ Unzip

Unzip은 압축 파일을 해제할 때 사용하며 새로운 디렉토리를 생성한 후 압축 해제할 지 또는 이미 파일이 있는 경우 덮어쓸 지 여부를 선택 할 수 있습니다.

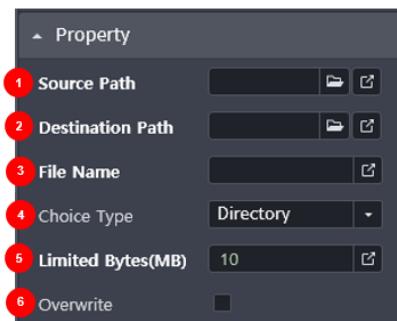


No.	설명
1	Source Path : 압축해제할 파일이 현재 위치한 전체경로를 입력합니다. (Type: String)
2	Destination Path : 압축해제할 목적지 디렉토리의 전체경로를 입력합니다. (Type: String)
3	Create Directory : 압축파일을 해제할 디렉토리를 만들 것인지 여부를 설정합니다. (Type: Boolean)
4	Overwrite : 파일이 존재하는 경우, 덮어쓰기 여부를 설정합니다.

## ■ Zip

Zip은 파일 또는 디렉토리를 압축할 때 사용 합니다.

여러 개의 파일을 압축하는 경우, 파일 다이얼로그에서 여러 개의 파일을 선택하여 압축하는 방법과  
StringArray변수에 개별파일 절대경로를 저장하고 이 변수를 넘겨서 압축하는 방법이 가능합니다.



No.	설명
1	Source Path : 압축할 파일 또는 디렉토리가 현재 위치한 전체경로를 입력합니다. (Type: String or StringArray)
2	Destination Path : 압축된 파일이 저장 될 디렉토리의 전체경로를 입력합니다. (Type: String)
3	File Name : 확장자(.zip)를 포함한 생성할 파일명을 입력합니다. (Type: String)
4	Choice Type : 압축 대상 종류를 선택합니다. >Directory : 디렉토리 하위의 파일/하위디렉토리를 하나의 압축파일로 압축합니다. >File : 선택된 파일을 하나의 압축파일로 압축합니다.
5	Limited Bytes : 생성할 압축파일의 제한 크기입니다. (Type: String) 기본값은 10 Mega bytes입니다.
6	Overwrite : 압축파일이 존재하는 경우, 덮어쓰기 여부를 설정합니다.

## 5.7. Visual Recognition

OCR 관련 기능들을 모아 놓은 것입니다.

Visual Recognition은 총 1개의 카테고리로 구성되어 있습니다.

- IMG
- OCR
- PDF

### 5.7.1. IMG

IMG는 이미지 파일을 읽어서 이미지 변수로 저장, 이미지 파일로 저장, 이미지 보여주기를 하는 기능입니다.

IMG은 다음과 같은 Event Item으로 구성되어 있습니다.

- Load Image
- Save Image
- Show Image

위의 Event Item은 실패 발생시 Continue Run이 체크되지 않으면 에러발생으로 스크립트 종료 처리됩니다.

#### ■ Load Image

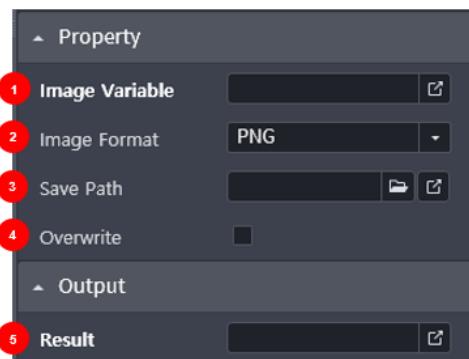
이미지 파일을 읽어서, 이미지 형식의 변수에 저장합니다.



No.	설명
1	Image Path : 로드할 이미지 파일의 경로를 입력합니다. Image Path는 String형식의 값입니다.
2	Result : 결과값을 저장 할 변수명을 입력합니다. Result는 Bitmap형식의 변수입니다.

## ■ Save Image

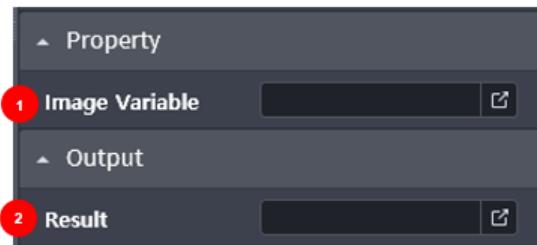
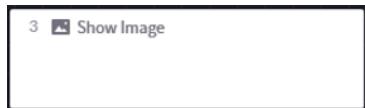
이미지 형식의 변수를 받아서, 저장하고자 하는 위치에 이미지 파일의 형식을 선택하여 이미지 파일로 저장합니다.



No.	설명
1	Image Variable : 저장할 이미지 변수를 입력 합니다. Image Variable은 Bitmap형식의 값입니다.
2	Image Format : 이미지 파일 저장 포맷을 선택 합니다. 이미지 저장 포맷은 PNG, JPG, BMP, TIFF 중 한가지를 선택하세요.
3	Overwrite : 덮어쓰기 여부를 설정합니다.
4	Result : 결과값을 저장 할 변수명을 입력합니다. Result는 Bool형식의 변수입니다.

## ■ Show Image

이미지 형식의 변수를 받아서, 팝업창으로 보여줍니다.



No.	설명
1	Image Variable : 보여줄 이미지 변수를 입력 합니다. Image Variable은 Bitmap형식의 값입니다.
2	Result : 결과값을 저장 할 변수명을 입력합니다. Result는 Bool형식의 변수입니다.

## 5.7.2. OCR

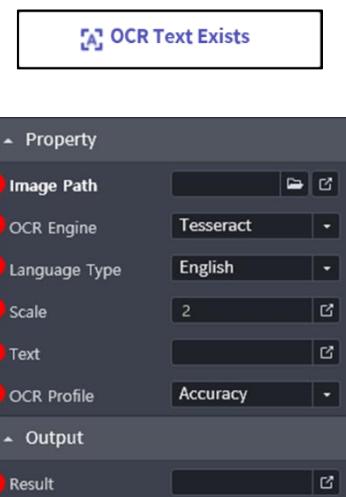
OCR은 수기로 작성했거나 인쇄한 문자를 기계가 읽을 수 있는 문자로 변환하는 기능입니다.

OCR은 다음과 같은 Event Item으로 구성되어 있습니다.

- OCR Text Exists
- Read OCR Document
- Read OCR Image

### ■ OCR Text Exists

OCR을 통해 추출한 Text와 입력된 Text가 동일한지 비교합니다.

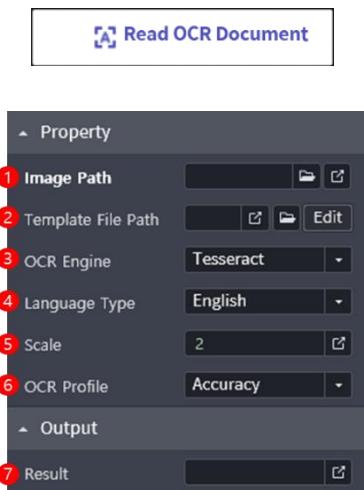


No.	설명
1	OCR을 수행할 이미지의 경로를 지정합니다. (Type: String) ※ 오류발생 기준: > Text에 기술된 문자를 ImagePath에서 정의된 이미지에서 찾지 못한 경우 > ImagePath에서 정의된 이미지가 비 정상인 경우
2	사용할 OCR Engine을 지정합니다. (OCR Engine: Tesseract와 ABBYY를 지원) ※ 오류발생 기준: > OCR Engine으로 ABBYY가 지정되어 있으나 라이선스가 없는 경우
3	사용할 언어 타입을 지정합니다. > 언어Type: English, Korean, Eng_Kor(영한혼용), Number(숫자)
4	OCR 수행 시 사용되는 이미지 스케일을 입력합니다. (Type: Int, default:2)
5	비교할 Text를 입력합니다. (Type: String)
6	OCR처리 시 기능상의 우선 순위를 지정합니다. > Accuracy: 정확도 우선 정책으로 속도가 느림 > Speed: 속도 우선 정책으로 정확도가 떨어짐

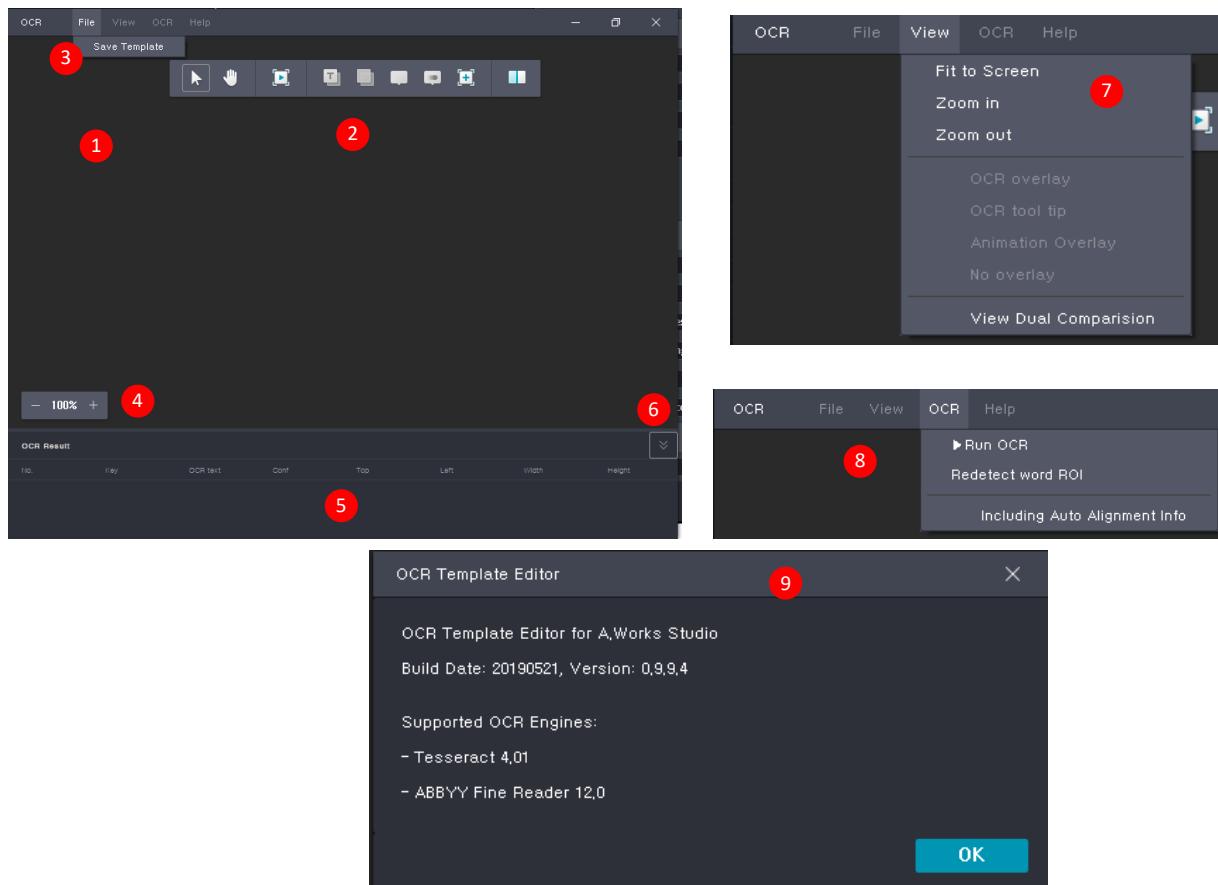
7 처리 결과를 저장할 변수 명을 선택합니다. (Type: Boolean)

## ■ Read OCR Document

템플릿 파일을 활용하여 OCR을 수행합니다.



No.	설명
1	OCR을 수행할 이미지의 경로를 지정합니다. (Type: String) ※ 오류발생 기준: > Text에 기술된 문자를 ImagePath에서 정의된 이미지에서 찾지 못한 경우 > ImagePath에서 정의된 이미지가 비 정상인 경우
2	사용할 OCR Template File의 경로를 지정합니다. 사용자 지정의 경우 [Edit] 버튼을 클릭하여 지정합니다. (※아래의 OCR Template Editor 참조)
3	사용할 OCR Engine을 지정합니다. (OCR Engine: Tesseract와 ABBYY를 지원) ※ 오류발생 기준: > OCR Engine으로 ABBYY가 지정되어 있으나 라이선스가 없는 경우
4	사용할 언어 타입을 지정합니다. > 언어Type: English, Korean, Eng_Kor(영한혼용), Number(숫자)
5	OCR 수행 시 사용되는 이미지 스케일을 입력합니다. (Type: Int, default:2) ※ 이미지 스케일은 OCR 수행 전 이미지 배율입니다. Ex) 2 -> 2 배
6	OCR처리 시 기능상의 우선 순위를 지정합니다. > Accuracy: 정확도 우선 정책으로 속도가 느림 > Speed: 속도 우선 정책으로 정확도가 떨어짐
7	처리 결과를 저장할 변수 명을 선택합니다. (Type: DataTable)



No.	설명
1	OCR Template Editor 화면입니다.
2	OCR Template Editor 화면에서 사용 가능한 기능 버튼들입니다.
3	File Menu에서 Template을 저장합니다.
4	이미지를 확대 또는 축소할 수 있습니다.
5	OCR 처리한 결과를 출력하는 영역입니다.
6	OCR 처리한 결과를 접기/펼쳐보기 버튼입니다.
7	OCR Template Editor 화면 설정 Menu입니다.
8	OCR 실행 Menu입니다.
9	OCR Template Editor의 Help를 클릭 시 화면에 표시하는 기본 정보입니다.

#### @@ Info @@

※ OCR Template Editor의 사용방법에 “5.9 OCR Template Editor의 사용방법”을 참조하십시오.

## ■ Read OCR Image

OCR을 수행하기 위해 원하는 영역을 이미지로 저장합니다. (Indicator 제공)



No.	설명
1	사용할 OCR Engine을 지정합니다. (OCR Engine: Tesseract와 ABBYY를 지원) ※ 오류발생 기준: > OCR Engine으로 ABBYY가 지정되어 있으나 라이선스가 없는 경우
2	사용할 언어 타입을 지정합니다. > 언어Type: English, Korean, Eng_Kor(영한혼용), Number(숫자)
3	OCR 수행 시 사용되는 이미지 스케일을 입력합니다. (Type: Int, default:2)
4	OCR처리 시 기능상의 우선 순위를 지정합니다. > Accuracy: 정확도 우선 정책으로 속도가 느림 > Speed: 속도 우선 정책으로 정확도가 떨어짐
5	Indicator로 지정한 이미지의 Top position(X좌표)을 가져옵니다. (Type: Int)
6	Indicator로 지정한 이미지의 Left position(Y좌표)을 가져옵니다. (Type: Int)
7	Indicator로 지정한 이미지의 너비를 가져옵니다. (Type: Int)
8	Indicator로 지정한 이미지의 높이를 가져옵니다. (Type: Int)
9	처리 결과를 저장할 변수 명을 입력합니다. (Type: DataTable, String)

### 5.7.3. PDF

PDF는 PDF문서의 텍스트를 추출/이미지 추출 또는 PDF 문서 합치기/PDF 문서 페이지 수 카운트를 하는 기능입니다.

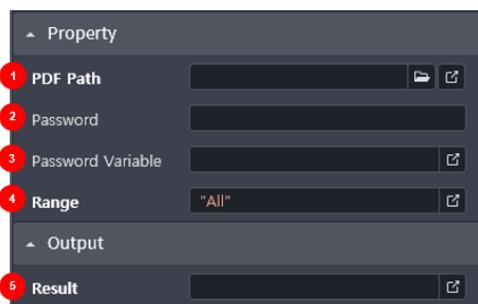
PDF는 다음과 같은 Event Item으로 구성되어 있습니다.

- Extract Images From PDF
- Extract Text From PDF
- Merge PDF Files
- Pages Count From PDF

#### ■ Extract Images From PDF

지정한 PDF 문서에서 이미지를 추출해서 반환합니다.

PDF 파일을 읽어서, 영역의 이미지를 추출하여 Array<Bitmap>형식의 변수에 저장합니다.

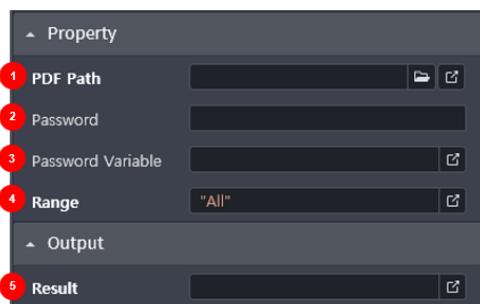


No.	설명
1	PDF Path : 읽을 PDF 파일의 경로를 입력합니다. PDF Path는 String형식의 값입니다.
2	Password : 사용자 패스워드를 입력합니다. Password는 String형식의 값입니다.
3	Password Variable : 사용자 패스워드 정보를 가지고 있는 변수명을 입력합니다. Password Variable은 String형식의 변수입니다. Password와 Password Variable중 하나만 입력하세요.
4	Range : PDF 파일의 페이지 범위를 입력 합니다. Range는 String형식의 변수입니다. 페이지 번호, 페이지 범위 및 All 중에서 선택 명시하여 사용합니다. 예시)"1" 또는 "1-2" 또는 "All"
5	Result : 결과값을 저장 할 변수명을 입력합니다. Result는 Array<Bitmap>형식의 변수입니다.

## ■ Extract Text From PDF

지정한 PDF 문서에서 문자열을 추출해서 반환합니다.

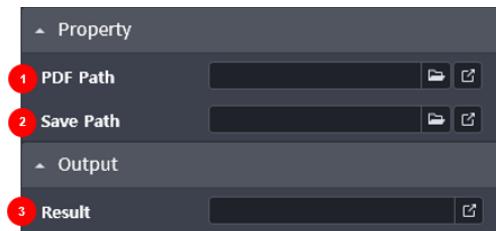
PDF 파일을 읽어서, 영역의 텍스트를 추출하여 String형식의 변수에 저장합니다.



No.	설명
1	PDF Path : 읽을 PDF 파일의 경로를 입력합니다. PDF Path는 String형식의 값입니다.
2	Password : 사용자 패스워드를 입력합니다. Password는 String형식의 값입니다.
3	Password Variable : 사용자 패스워드 정보를 가지고 있는 변수명을 입력합니다. Password Variable은 String형식의 변수입니다. Password와 Password Variable 중 하나만 입력하세요.
4	Range : PDF 파일의 페이지 범위를 입력 합니다. Range는 String형식의 변수입니다. 페이지 번호, 페이지 범위 및 All 중에서 선택 명시하여 사용합니다. 예시) "1" 또는 "1-2" 또는 "All"
5	Result : 결과값을 저장 할 변수명을 입력합니다. Result는 String형식의 변수입니다.

## ■ Merge PDF Files

OCR을 통해 추출한 Text와 입력된 Text가 동일한지 비교합니다.

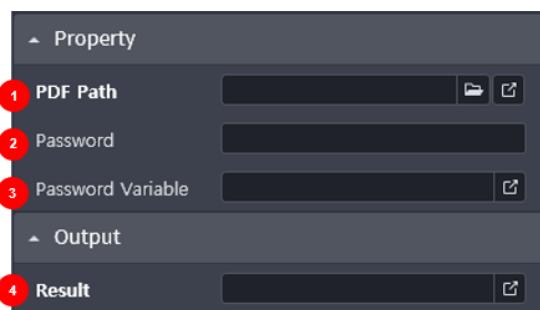
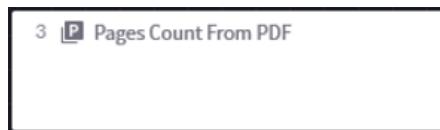


No.	설명
1	PDF Path : 읽을 PDF 파일의 경로를 입력합니다. PDF Path는 String형식의 값입니다.

2	Save Path : 저장할 PDF 파일의 경로를 입력합니다. PDF Path는 String형식의 값입니다.
3	Result : 결과값을 받아올 변수명을 입력합니다. Result는 Bool형식의 변수입니다.

## ■ Pages Count From PDF

지정한 PDF 문서의 전체 페이지 수를 반환합니다.



No.	설명
1	PDF Path : 읽을 PDF 파일의 경로를 입력합니다. PDF Path는 String형식의 값입니다.
2	Password : 사용자 패스워드를 입력합니다. Password는 String형식의 값입니다.
3	Password Variable : 사용자 패스워드 정보를 가지고 있는 변수명을 입력합니다. Password Variable은 String형식의 변수입니다. Password와 Password Variable중 하나만 입력하세요.
4	Result : 결과값을 저장 할 변수명을 입력합니다. Result는 Int형식의 변수입니다.

## 5.8 WorkCenter

WorkCenter와 연동되는 기능입니다.

### @@ 주의 @@

본 기능은 WorkCenter와 연동되어 있을 때 동작합니다.

WorkCenter는 총 3개의 카테고리로 구성되어 있습니다.

- Asset
- Queue
- Work

### 5.8.1. Asset

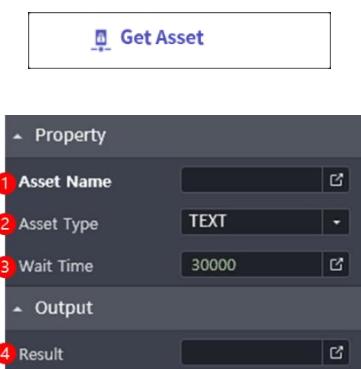
WorkCenter의 Asset을 가져오거나 설정하는 기능입니다.

Asset은 다음과 같은 Event Item으로 구성되어 있습니다.

- Get Asset
- Get Asset Credential
- Set Asset

#### ■ Get Asset

WorkCenter에 저장된 문자열 정보를 얻습니다.

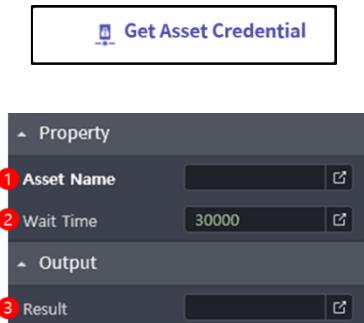


No.	설명
1	사전에 WorkCenter에 정의된 Text Type의 Asset Name을 지정합니다. (Type: String) ※ 오류발생 기준: 사전에 정의되지 않은 Asset Name을 입력할 경우
2	WorkCenter에 있는 해당 Asset의 유형을 지정합니다. > Asset Type: TEXT, BOOLEAN, INTEGER
3	WorkCenter에 Connection을 요청 후 Wait Time을 입력합니다. (Type: Int, 단위: ms)

- 4 가져올 Asset 결과를 저장할 변수 명을 지정합니다. (Type: String)

## ■ Get Asset Credential

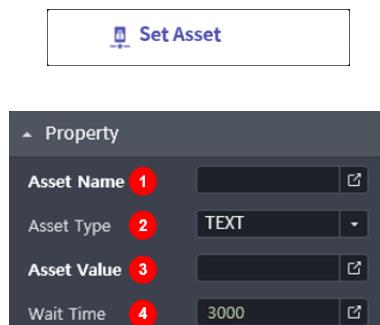
WorkCenter에 저장된 Asset의 Credential(자격증명) 정보를 가져옵니다.



No.	설명
1	WorkCenter에 정의된 Credential Type의 Asset Name을 지정합니다. (Type: String) ※ 오류발생 기준: 사전에 정의되지 않은 Asset Name을 입력할 경우
2	WorkCenter에 Connection을 요청 후 Wait Time을 입력합니다. (Type: Int, 단위: ms)
3	가져올 Asset Credential 결과를 저장할 변수 명을 지정합니다. (Type: Credential) ※ Asset 중 ID/PW와 같은 Credential 정보를 가져옴

## ■ Set Asset

WorkCenter에 저장된 Asset에 데이터를 입력합니다.



No.	설명
1	WorkCenter에 정의된 Asset Name을 지정합니다. (Type: String) ※ 오류발생 기준: 사전에 정의되지 않은 Asset Name을 입력할 경우
2	WorkCenter에 있는 해당 Asset 의 유형을 지정합니다. > Asset Type: TEXT, BOOLEAN, INTEGER

<b>3</b>	WorkCenter에 정의된 Asset Name에 설정할 변수 명 또는 값을 지정합니다. (Type: String)
<b>4</b>	WorkCenter에 Connection을 요청 후 Wait Time을 입력합니다. (Type: Int, 단위: ms)

## 5.8.2. Queue

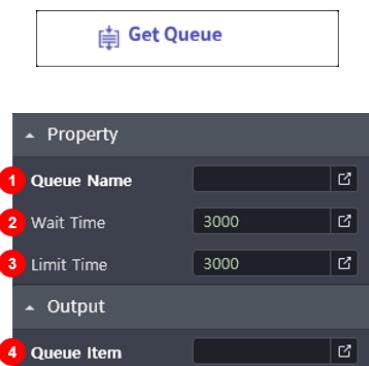
Queue은 WorkCenter와 연동되는 기능입니다.

Queue는 다음과 같은 Event Item으로 구성되어 있습니다.

- Get Queue
- Get Queue List
- Put Queue
- ReQueue
- Transact Queue

### ■ Get Queue

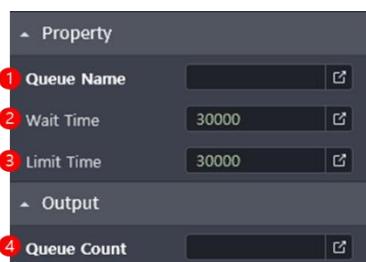
WorkCenter에 생성된 Queue Item 중 가장 오래된 1개의 Queue Item의 정보를 가져옵니다.



No.	설명
<b>1</b>	WorkCenter에 정의된 Queue Name을 지정합니다. (Type: String) ※ 오류발생 기준: 사전에 정의되지 않은 Queue Name을 입력할 경우
<b>2</b>	WorkCenter에 접속 Wait Time을 입력합니다. (Type: Int, 단위: ms)
<b>3</b>	WorkCenter에서 Queue Data를 찾는데 걸리는 최대시간 설정합니다. (Type: Int, 단위: ms)
<b>4</b>	결과로 가져올 Queue Item의 변수 명을 입력합니다. (Type: QueueItem)

## ■ Get Queue Count

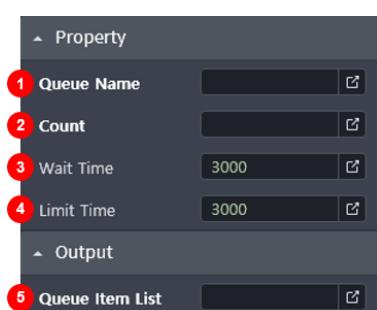
Queue에 남아있는 메시지의 개수를 가져옵니다.



No.	설명
1	WorkCenter에 정의된 Queue Name을 지정합니다. (Type: String) ※ 오류발생 기준: 사전에 정의되지 않은 Queue Name을 입력할 경우
2	WorkCenter에 접속 Wait Time을 입력합니다. (Type: Int, 단위: ms)
3	WorkCenter에서 Queue Data를 찾는데 걸리는 최대시간 설정합니다. (Type: Int, 단위: ms)
4	Queue에 남아있는 메시지의 개수를 저장할 변수 명을 입력합니다. (Type: Int)

## ■ Get Queue List

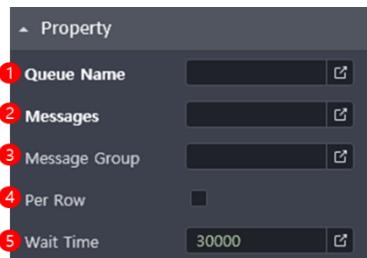
WorkCenter에 생성된 Queue Item list에 대한 정보를 가져옵니다.



No.	설명
1	WorkCenter에 정의된 Queue Name을 지정합니다. (Type: String) ※ 오류발생 기준: 사전에 정의되지 않은 Queue Name을 입력할 경우
2	Queue Item List중 오래된 것을 기준으로 가져올 Item 개수를 지정합니다. (Type: Int)
3	WorkCenter에 접속 Wait Time을 입력합니다. (Type: Int, 단위: ms)
4	WorkCenter에서 Queue Data를 찾는데 걸리는 최대시간 설정합니다. (Type: Int, 단위: ms)
5	결과로 가져올 Queue Item List의 변수 명을 입력합니다. (Type: QueueItemList)

## ■ Put Queue

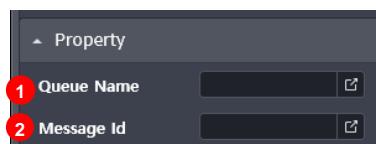
사용자가 신규 Queue Item을 지정하여 WorkCenter Queue에 추가합니다.



No.	설명
1	WorkCenter에 정의할 Queue Name을 지정합니다. (Type: String)
2	Queue에 넣을 DataTable형식의 Message를 지정합니다. (Type: DataTable)
3	Queue Item의 Message Group을 지정합니다. (Type: String)
4	Message Row별로 여러 차례 보낼지 여부를 설정합니다.
5	WorkCenter에 접속 Wait Time을 입력합니다. (Type: Int, 단위: ms)

## ■ ReQueue

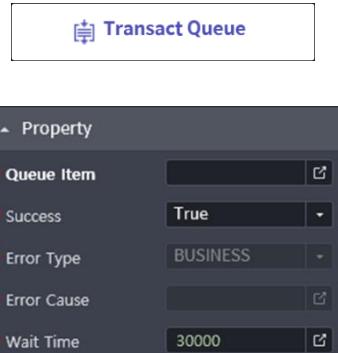
WorkCenter의 Queue의 특정 Message의 상태를 InProgress 또는 Failure에서 New로 변경할 때 사용합니다.



No.	설명
1	Transact를 처리 할 Queue Name을 지정합니다. (Type: String)
2	Transact를 처리 할 Message Id를 지정합니다. (Type: String)

## ■ Transact Queue

WorkCenter의 Queue Item과 Transaction을 합니다.



No.	설명
1	Transaction을 처리할 Queue Item을 지정합니다. (Type: String)
2	성공이 True인지 False인지 지정합니다.
3	Error 발생 시 예외유형으로 Error Type이 BUSINESS인지 SYSTEM인지 지정합니다. > Business: 메시지 데이터 자체 오류가 발생 > System: 메시지 처리 시 시스템의 오류가 발생
4	Error 발생 시 오류 원인을 선택합니다.
5	WorkCenter에 접속 Wait Time을 입력합니다. (Type: Int, 단위: ms)

### 5.8.3. Work

Work는 프로세스 탭에서 사용하는 WorkCenter와 연동되는 기능입니다.

@@ 주의 @@

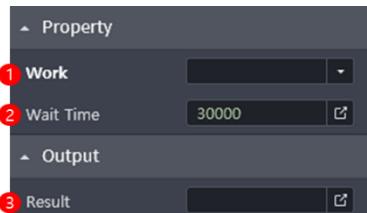
태스크 탭에서는 사용이 불가합니다.

Work는 다음과 같은 Event Item으로 구성되어 있습니다.

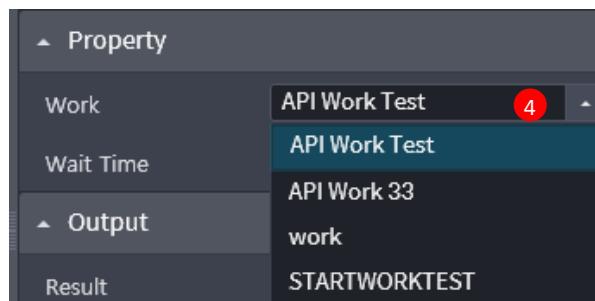
- Start Work

#### ■ Start Work

프로세스 탭에서 작성한 프로세스를 WorkCenter에서 등록한 Work와 연동하여 실행시키기 위한 기능입니다..



No.	설명
1	WorkCenter에서 실행을 위하여 등록한 Work를 지정합니다. ※ 기능이 정상 작동하기 위해서는 WorkCenter에서 Work 등록 시 Work 유형을 API로 등록해야 합니다. Work 유형이 API로 등록 된 Work만 아래의 ④와 같은 DropDownList 리스트로 표시되어 Work를 선택할 수 있습니다.
2	WorkCenter의 Work 실행 Wait Time을 입력합니다. (Type: Int, 단위: ms)
3	처리 결과를 저장할 변수 명을 입력합니다. (Type: Boolean)



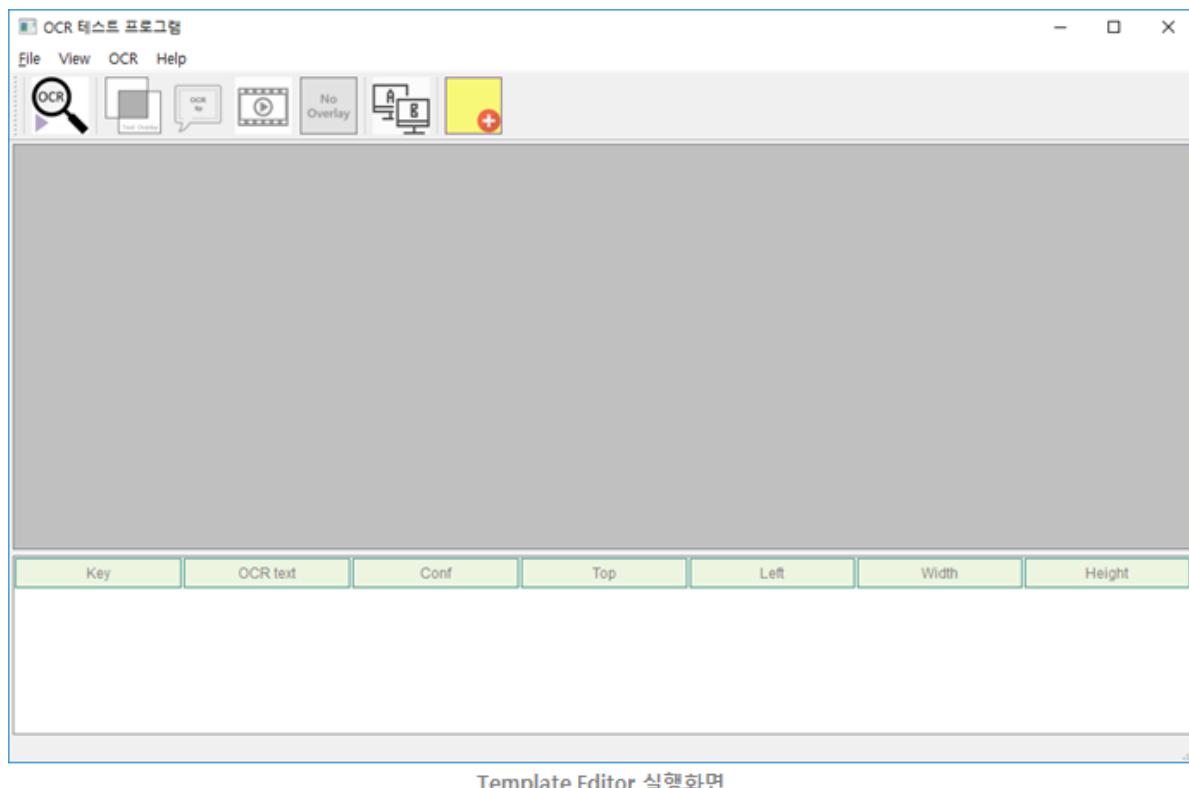
## 5.9 OCR Template Editor 사용설명

### 5.9.1. 프로그램 구성

- Template Editor 프로그램  
*OCRTes.t.exe*
- OCR Test용 PDF 파일  
*OCR\_EVC\_template\_1.pdf*  
*OCR\_EVC\_template\_2.pdf*  
*OCR\_EVC\_template\_3.pdf*
- OCR Test용 Template 파일  
*OCR\_EVC\_template\_1.pdf.tem*

### 5.9.2. 프로그램 실행

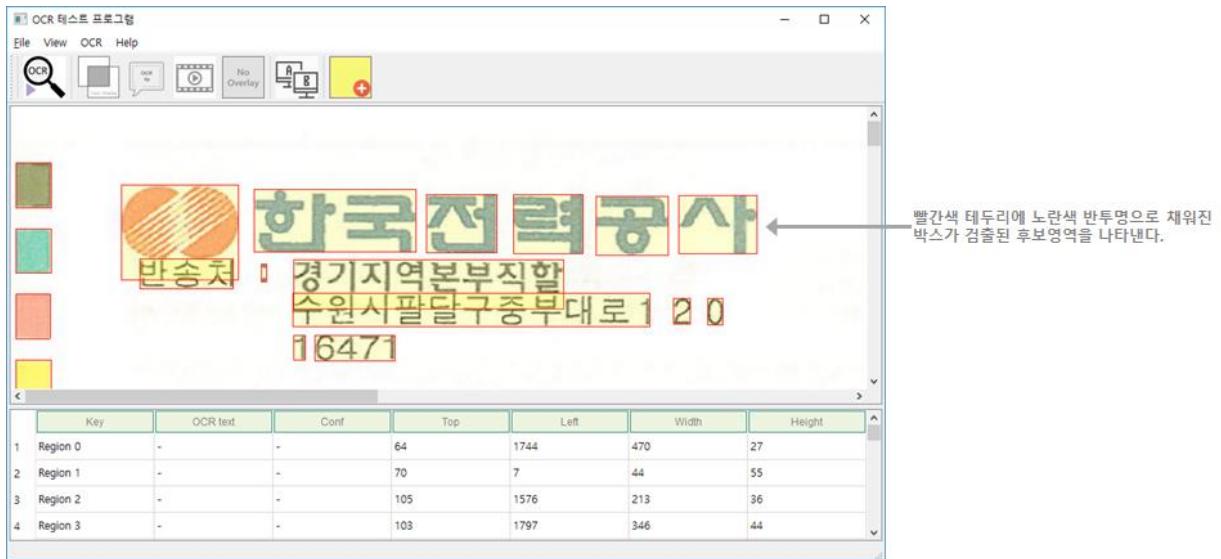
OCRTes.t.exe를 실행하여 Template Editor 프로그램을 실행합니다.



### 5.9.3. PDF 파일 열기

File -> Open 을 클릭 하고 OCR Test용 PDF 파일을 Open합니다..

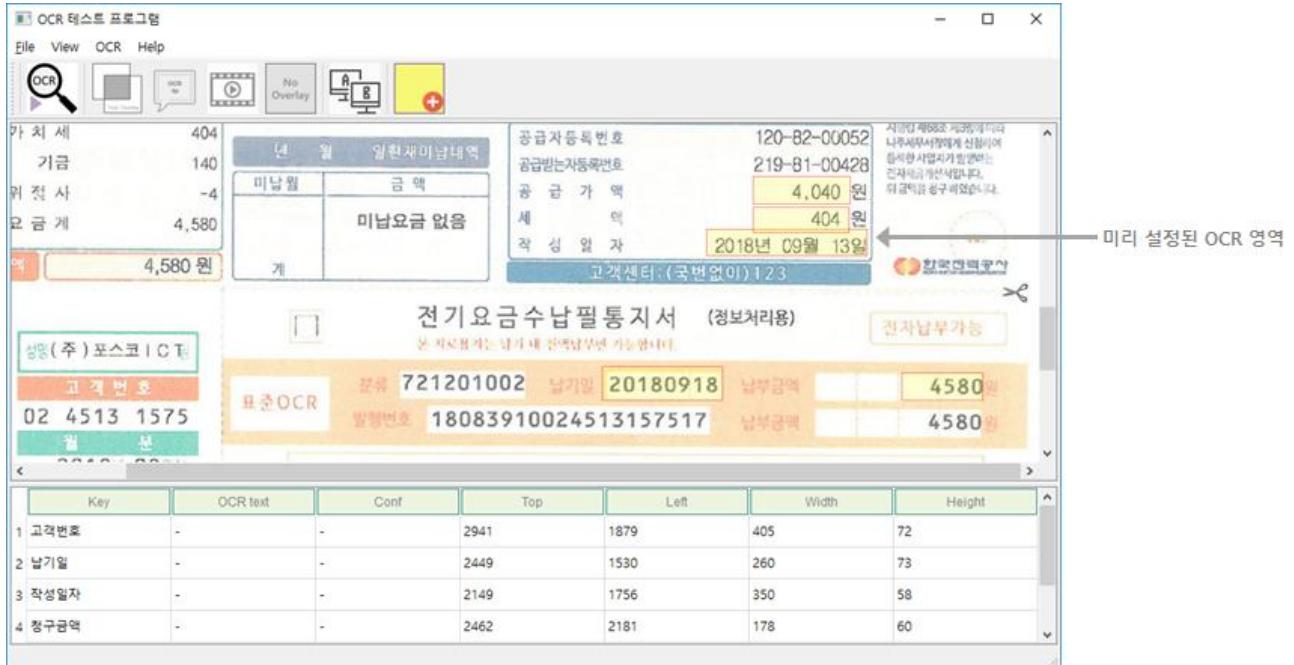
PDF파일을 열면 자동으로 문자 후보 영역이 검출 됩니다.



#### 5.9.4. Template 파일 열기

File -> Load Template 을 클릭 하고 OCR Test용 Template(.tem) 파일을 Open합니다.

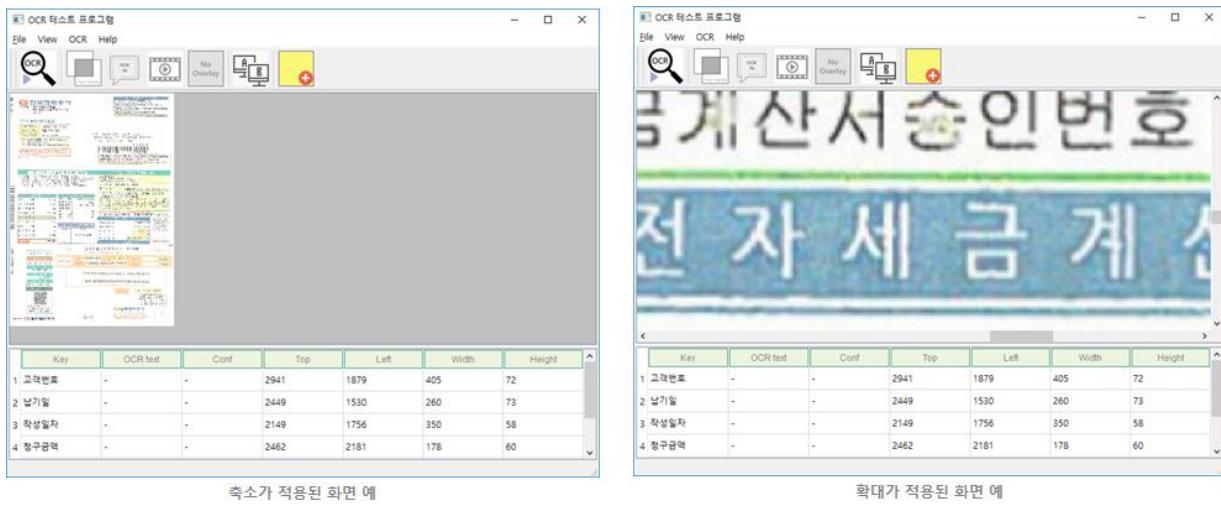
Template 파일을 열면 미리 지정된 OCR 영역의 Template이 자동으로 설정됩니다.



#### 5.9.5. 확대/축소 위치 이동

Ctrl + 마우스 Wheel을 이용하여 확대 축소가 가능합니다.

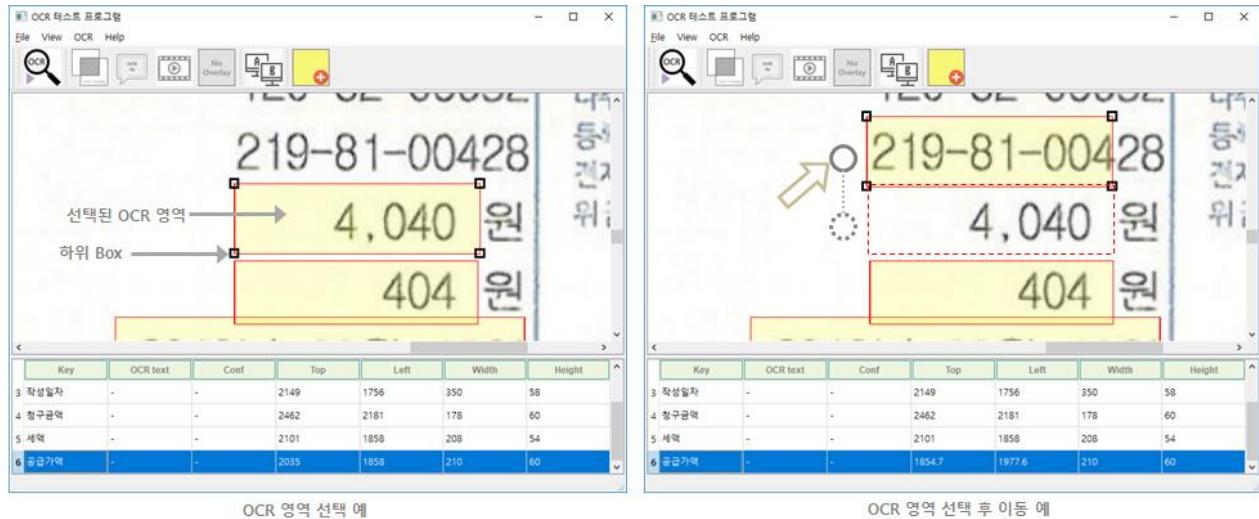
화면이 확대 되면 자동으로 하단 및 우측에 스크롤 바가 생기며 이를 이용하여 화면 이동이 가능합니다.



### 5.9.6. OCR 영역 선택 및 이동

특정 OCR 영역을 나타내는 빨간색 테두리에 노란색 반투명으로 채워진 박스는 마우스 왼쪽 버튼을 클릭하여 선택이 가능합니다.

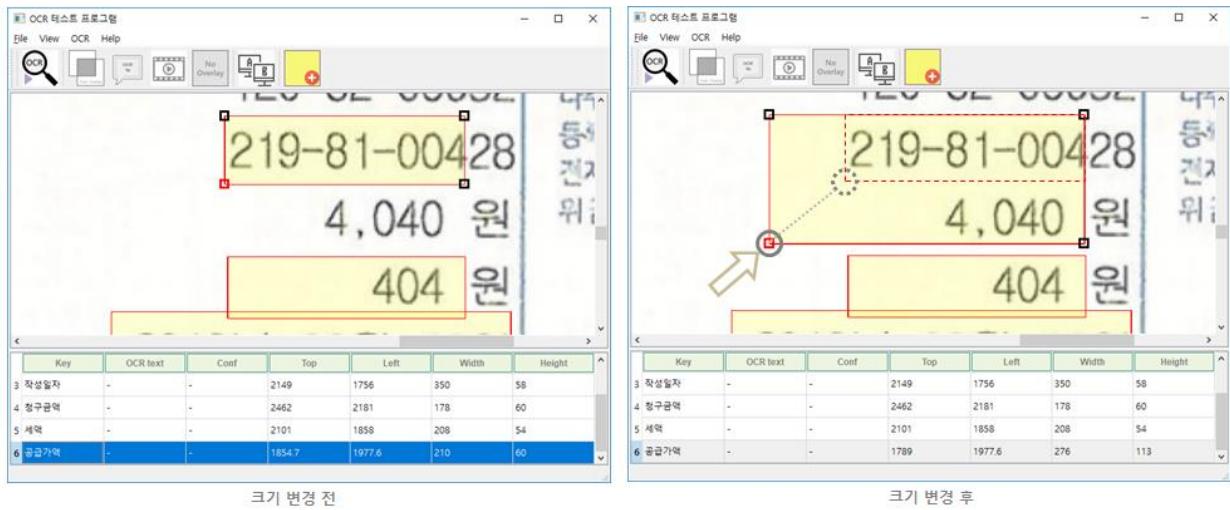
특정 OCR 영역을 선택하면 크기 조절을 위한 하위 박스 4개가 4각형 모서리에 나타나고 이를 통해 선택 유무를 확인 가능합니다. 선택된 OCR 영역은 마우스 드래그를 통해 위치 이동이 가능합니다.



### 5.9.7. OCR 영역 크기 변경

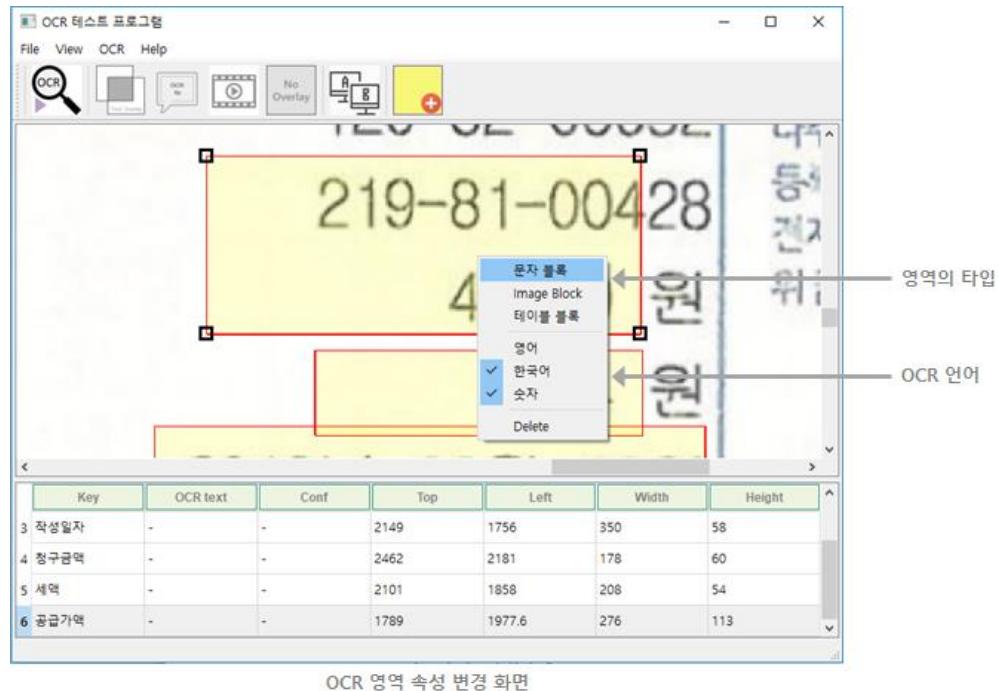
특정 OCR 영역 선택 시 나타내는 하위 박스 4개를 이용하여 좌우 상하 크기 변경이 가능합니다.

하위 박스 영역에 마우스가 위치하면 위치된 하위박스의 색깔이 빨간색으로 변경되고 이때 마우스 드래그를 통해 크기 변경이 가능합니다.



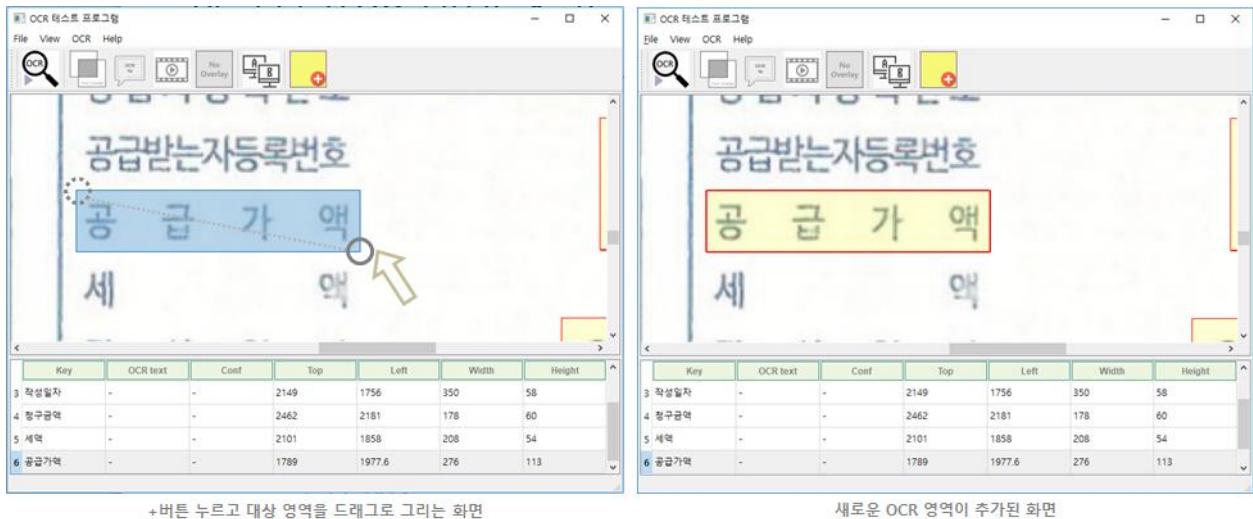
### 5.9.8. OCR 영역 속성 변경

선택된 OCR 영역에서 마우스 오른쪽 버튼을 누르면 Context 메뉴가 나타나 각 OCR 영역의 속성을 변경할 수 있습니다. 속성은 영역의 타입, OCR을 수행할 언어가 있습니다.



### 5.9.9. 새로운 OCR 영역 추가

상단 오른쪽 Toolbar 아이콘을 클릭하고 대상 영역을 드래그로 그리면 새로운 OCR 영역을 추가 할 수 있습니다.

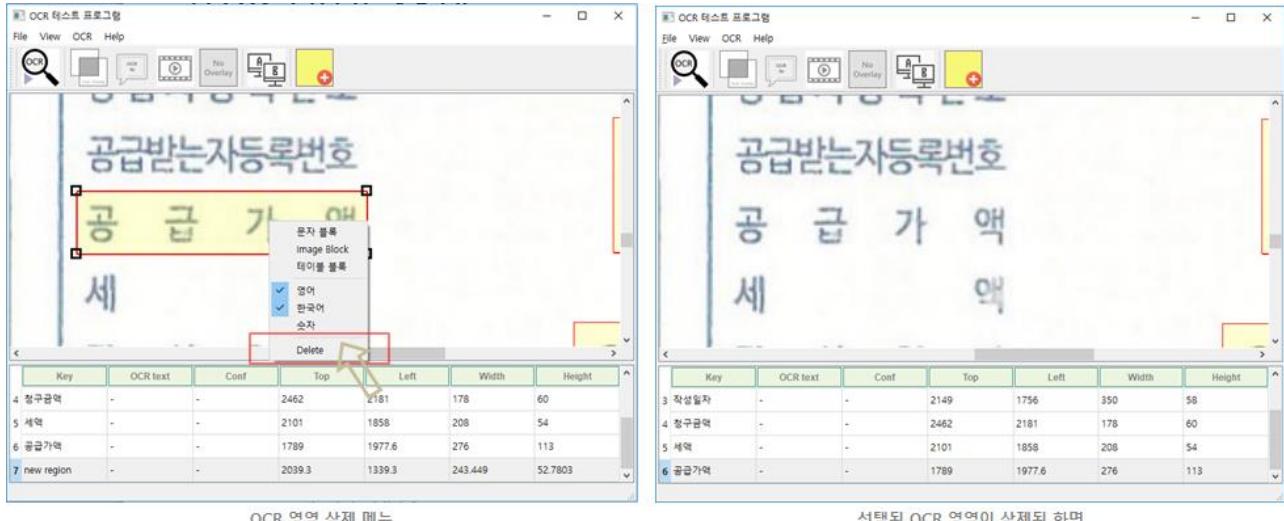


+버튼 누르고 대상 영역을 드래그로 그리는 화면

새로운 OCR 영역이 추가된 화면

### 5.9.10. OCR 영역 삭제

선택된 OCR 영역에서 마우스 오른쪽 버튼을 누르면 Context 메뉴가 나타납니다. Context 메뉴에서 삭제를 클릭하면 선택된 OCR 영역이 삭제됩니다.

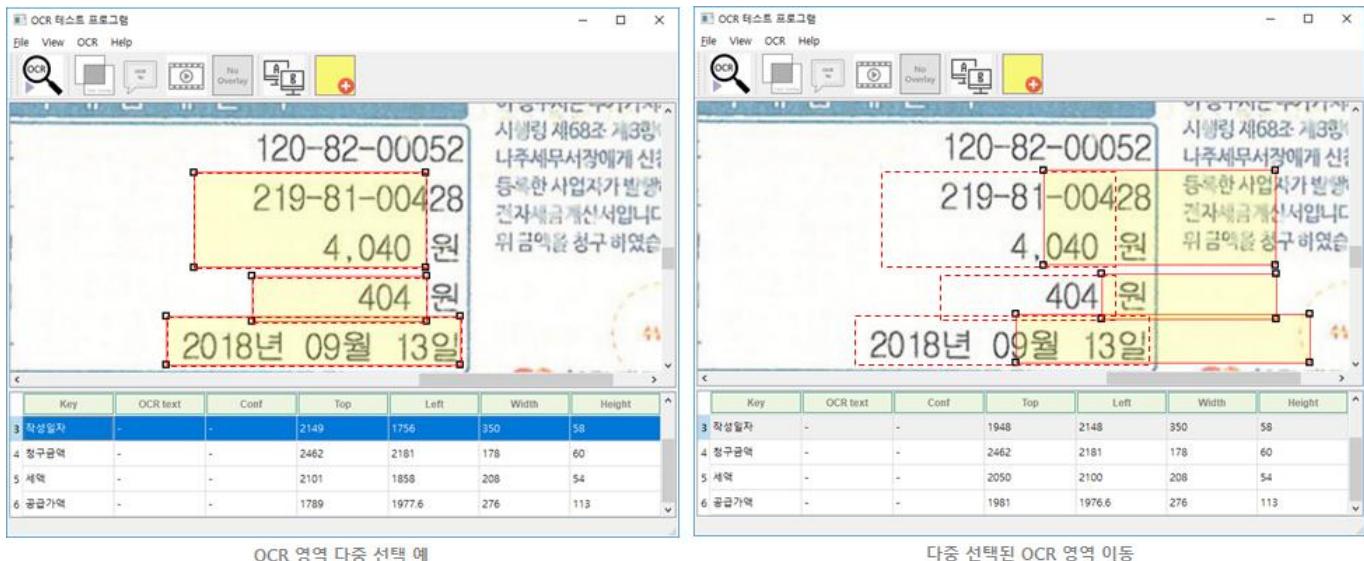


OCR 영역 삭제 메뉴

선택된 OCR 영역이 삭제된 화면

### 5.9.11. OCR 영역 다중 선택 및 동작

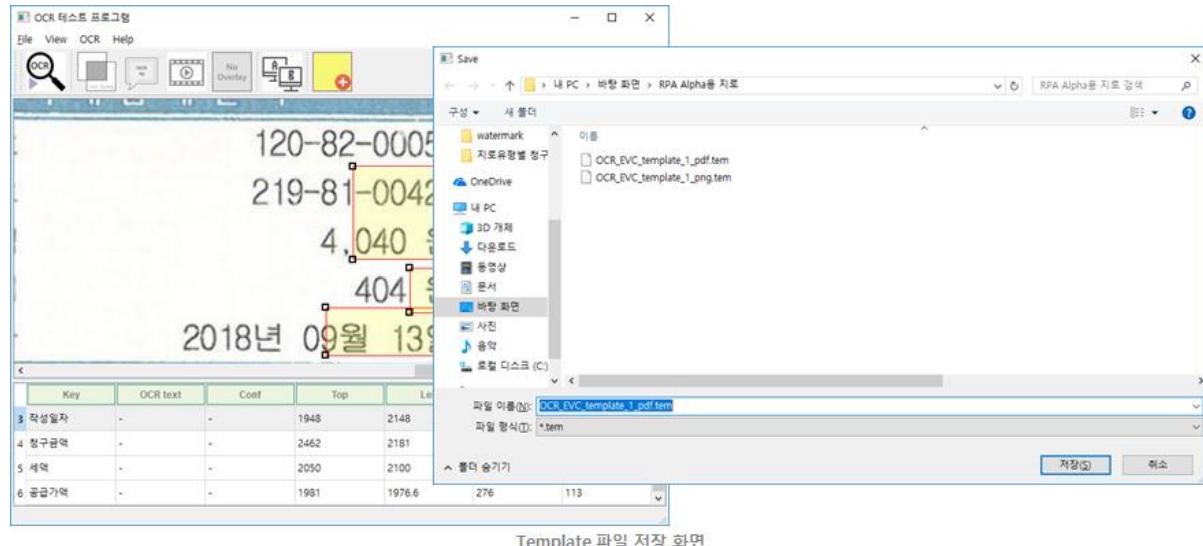
OCR 영역은 드래그 또는 Ctrl 키를 이용하여 다중 선택이 가능합니다. 다중 선택된 OCR 영역은 이동, 속성 변경, 삭제가 한번에 가능합니다.



### 5.9.12. Template 파일 저장

Template Editor를 이용하여 편집된 Template 정보는 Template파일로 저장이 가능합니다.

File -> Save Template를 선택하면 파일저장이 가능합니다.



### 5.9.13. OCR 수행

템플릿 편집이 끝나면 템플릿에 해당하는 OCR 결과를 확인해 볼 수 있습니다.

상단 왼쪽 돋보기 모양의 Toolbar 아이콘을 클릭하면 OCR을 수행하고 결과를 보여 줍니다.

OCR 결과는 하단 테이블 또는 OCR 결과 표시 방법에 따라 화면에 표시됩니다.

Left Window (No Overlay):

Key	OCR text	Conf	Top	Left	Width	Height
3 작성일자	-	-	2149	1756	350	58
4 청구금액	-	-	2462	2181	178	60
5 세액	-	-	2101	1858	208	54
6 곱값가액	-	-	2035	1858	210	60

Right Window (No Overlay):

Key	OCR text	Conf	Top	Left	Width	Height
1 고객번호	02 4513 1575	96	2941	1879	405	72
2 날기일	20180918	95	2449	1530	260	73
3 작성일자	2018년 09월 13일	89	2149	1756	350	58
4 청구금액	4580	39	2462	2181	178	60

OCR 수행 전

OCR 수행 후

## 5.9.14. OCR 결과 비교 Overlay

OCR 수행 후 상단 Toolbar 왼쪽 2번째 아이콘을 클릭하면 OCR 결과를 문서에 Overlay하여 비교 할 수 있는 기능을 제공합니다.

Left Window (Overlay):

Key	OCR text	Conf	Top	Left	Width	Height
1 고객번호	02 4513 1575	96	2941	1879	405	72
2 날기일	20180918	95	2449	1530	260	73
3 작성일자	2018년 09월 13일	89	2149	1756	350	58
4 청구금액	4580	39	2462	2181	178	60

Center: Overlay 적용 화면

Right Window (No Overlay):

Key	OCR text	Conf	Top	Left	Width	Height
1 고객번호	02 4513 1575	96	2941	1879	405	72
2 날기일	20180918	95	2449	1530	260	73
3 작성일자	2018년 09월 13일	89	2149	1756	350	58
4 청구금액	4580	39	2462	2181	178	60

Center: No Overlay 적용 화면

## 5.9.15. OCR 결과 비교 Tooltip

OCR 수행 후 상단 Toolbar 왼쪽 3번째 아이콘을 클릭하면 OCR 결과를 Tooltip 형태로 표시합니다. OCR 결과를 확인하고 싶은 영역으로 마우스를 가져가면 OCR 결과를 Tooltip으로 보여 줍니다.



Tooltip 결과 비교 화면

### 5.9.16. OCR 결과 비교 Animation Text

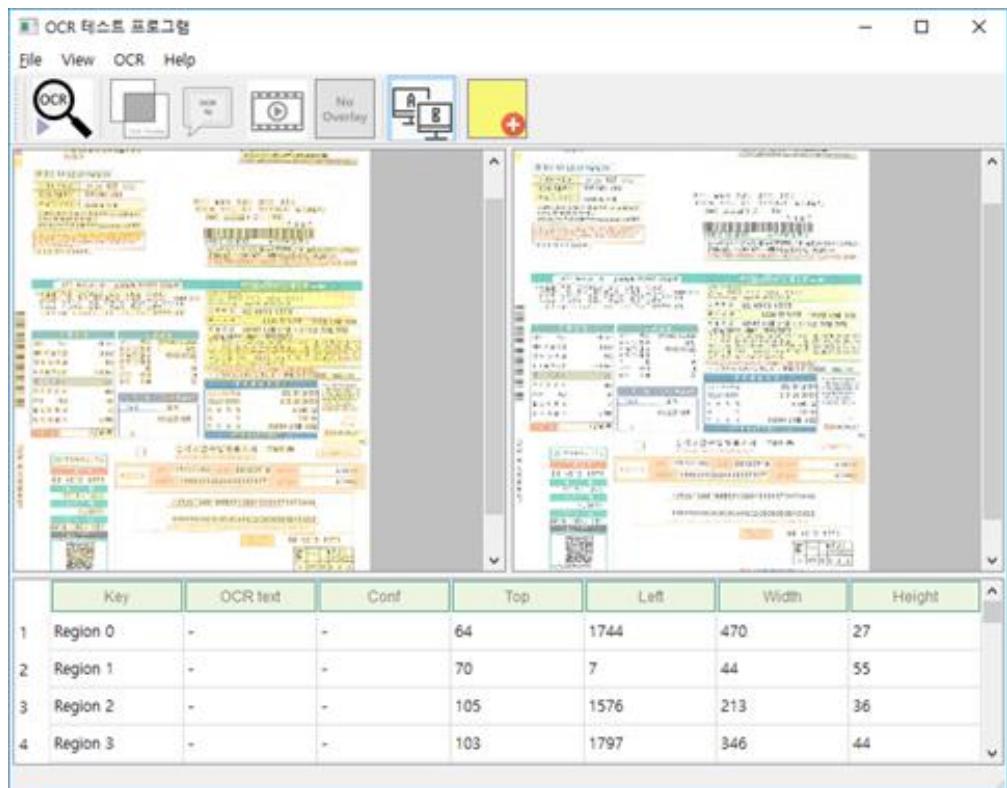
OCR 수행 후 상단 Toolbar 왼쪽 4번째 아이콘을 클릭하면 OCR 결과를 Animation Text 형태로 보여 줍니다. OCR 결과를 확인하고 싶은 영역으로 마우스를 가져가면 OCR 결과가 0.5 정도의 시간 동안 원쪽에서 오른쪽으로 Animation 효과를 보여주며 OCR 결과로 변경되고 영역을 벗어나면 원본 화면을 보여 줍니다.



Animation Text 등장

### 5.9.17. OCR 결과 및 원본 문서 비교

OCR 수행 후 상단 Toolbar 왼쪽 6 번째 아이콘을 클릭하면 OCR 결과 또는 Template Editing 을 수행하는 화면에 추가적으로 원본 문서를 보여주는 화면이 나타납니다. 두 화면은 위치는 일치하지 않지만 한쪽에서 확대 축소를 수행하면 같은 크기로 반대편 화면을 맞추어 줍니다.



축소가 적용된 화면 예

Key	OCR text	Conf	Top	Left	Width	Height
1	Region 0	-	64	1744	470	27
2	Region 1	-	70	7	44	55
3	Region 2	-	105	1576	213	36
4	Region 3	-	103	1797	346	44

## 부록 A. 용어설명

용어	설명
RPA	Robotic Process Automation. 단순 반복업무를 자동화하여 소프트웨어 로봇으로 하여금 수행토록 하는 프로그램으로 화면 및 문자인식기, 인공지능 기술 등 다양한 Event Item들과 결합하여 동작합니다.
Studio	다양한 Flow Control과 Event Item들을 이용하여 자동화 프로세스와 태스크 스크립트를 만들어 A.WORKS Bot의 수행을 검증하는 소프트웨어입니다.
WorkCenter	A.WORKS S.Bot의 실행 및 모니터링, 수행 결과, 일정관리 등 서버에서 수행되는 업무기능을 지원하는 자원관리 시스템입니다.
Mini	A.WORKS Bot의 실행 및 모니터링, 수행 결과, 일정관리 등 로컬PC에서 수행되는 업무기능을 지원합니다.
Bot	A.WORKS Studio를 이용하여 제작된 소프트웨어 로봇으로, 사용자의 로컬PC 환경에서 수행되는 로봇을 A.WORKS A.Bot이라 하고, WorkCenter에 등록되어 스케줄에 따라 수행되는 로봇을 A.WORKS S.Bot이라고 합니다.
Process	A.WORKS Studio로 제작하는 하나의 자동화 업무의 단위입니다
Task	프로세스를 구성하는 하나의 단위 작업항목으로 여러 개의 태스크 조합으로 프로세스를 구성합니다.
Script	일반적으로, 간단한 프로그램들을 연결하기 위해 사용하는 프로그램 컴퓨터 프로세서나 컴파일러가 아닌 다른 프로그램에 의해 번역되고 수행되는 명령어 집합. Studio에서는 프로세스, 태스크, Flow Control, Event Item으로 기술하는 내용 들을 의미합니다.
.awx	A.WORKS Studio를 이용하여 제작한 프로세스 스크립트의 확장자입니다
Resource	작업에 사용되는 문서나 이미지, 동영상 등과 같은 파일들로, 특정의 경로에 저장해두고 프로세스를 수행 시 사용할 수 있는 자원들입니다.
Logic	컴퓨터에 어떤 연산이나 업무를 수행시키기 위한 논리의 구조나 절차 또는 방식을 말합니다.
Library	컴퓨터가 특정 업무 기능을 수행하는데 필요한 기능이나 관련한 함수들을 스크립트에 사용할 수 있도록 파일 형태로 제공하는 것을 말합니다.
Flow Control	프로세스 및 태스크 Logic 구성 시 처리흐름을 제어하기 위하여 특정 조건에 따라 순차처리나 분기, 반복 등의 기능을 수행합니다.
Event Item	컴퓨터에 특정 업무를 수행시키는데 필요한 최소 단위의 기능들의 모음으로 라이브러리 형태로 Studio에서 제공합니다.
Shared Task	WorkCenter에 공유 태스크로 등록된 태스크로, Download하여 자동화 프로세스를 구성합니다. WorkCenter에 Upload하고 등록을 요청하면 서버에서 기능검증 후 Shared Task로 등록하고 공유하여 사용할 수 있습니다.
Property	기능함수, 파일 또는 문자열 등 특정 대상이 가지고 있는 고유의 정보 들로

	기본값 외에 특정 조건에 따라 특정의 값을 설정하여 사용합니다..
Object	Desktop에 표시된 요소로 해당 Object는 고유한 식별자인 Selector를 이용하여 인식합니다. A.WORKS에서는 Object를 인식하는 자동화 기술을 기반으로 커서로 클릭하거나 Keyboard 입력 등 UI 자동화를 구현합니다.
Indicator	Object 기반의 자동화 스크립트를 구현할 때, Object를 지정하는 Event Item의 기능입니다.
Selector	Object를 구분하는 식별자로 문자열로 구성합니다..
OCR	Optical Character Reader. 이미지 형태의 문서를 읽어 Text형태로 변환해 주는 기능입니다.
Image Recognition	Object 선택 시 이미지로 인식하여 사용할 수 있게 해줍니다.
Asset	WorkCenter에서 관리되는 공용Data로, Studio/Mini가 해당 Data를 요청해 사용할 수 있습니다
Browser	월드 와이드 웹(www)에서 모든 정보를 볼 수 있도록 해주는 응용 프로그램입니다. A.WORKS에서는 Microsoft社의 Internet Explorer, Google사의 Chrome을 지원합니다.
SMTP	Simple Mail Transfer Protocol의 약어로 인터넷에서 전자메일을 전송할 때 쓰이는 프로토콜입니다.