

# Realize your vision



`${solution_edition}`  
`${last_modified}`

# 차례

차례	ii
서문	v
<b>1. 개요</b>	<b>1</b>
1.1 IPA 솔루션 소개	1
1.2 지원 기능	1
1.3 유즈케이스	2
<b>2. 시작하기</b>	<b>3</b>
2.1 시스템 요구사항	3
2.2 라이선스 준비	4
2.3 IPA Designer 설치하기	4
2.4 IPA Bot 설치하기	6
<b>3. IPA Designer 사용하기</b>	<b>8</b>
3.1 IPA Designer 화면 살펴보기	8
메뉴	9
디자이너 설정	10
샘플	12
툴바	13
프로젝트 화면	14
프로세스 검색 화면	15
라이브러리 화면	16
프로세스 플로우 전용 라이브러리 화면	17
디자인 화면	17
액티비티 찾기	18
차트 방향 변경하기	18
미니맵 보기	19
화면 크기 변경하기	20
출력 화면	21
공용 리소스 화면	22
변수 화면	22
속성 화면	23
도움말 화면	24

3.2 새 프로젝트 및 프로세스 플로우 시작하기	25
새 프로젝트 또는 프로세스 플로우 만들기	25
새 작업 만들기	26
새 프로세스 만들기	27
새 이벤트 만들기	27
리소스 추가하기	28
새 그룹 만들기	29
프로젝트 편집하기	30
작업 이름 변경하기	30
작업 복사하기	30
작업 이동하기	31
작업 삭제하기	32
3.3 프로젝트 및 프로세스 플로우 저장하기	32
3.4 프로젝트 실행하기	33
3.5 프로세스 플로우 실행하기	34
<b>4. 프로젝트 디자인하기</b>	<b>35</b>
4.1 레코딩 기능으로 디자인하기	35
4.2 라이브러리로 디자인하기	38
액티비티 추가하기	38
링크에 액티비티 추가하기	38
액티비티 그룹에 액티비티 추가하기	39
액티비티 사이에 액티비티 추가하기	39
액티비티 편집하기	39
액티비티 삭제하기	39
액티비티 복사하기 1	39
액티비티 복사하기 2	40
액티비티 이동하기 1	40
액티비티 이동하기 2	40
작업 추가하기	40
여러 액티비티 선택하기	41
선택 반전하기	41
비활성화하기	41
액티비티 그룹 접기	41
라이브러리 알아보기	41
앱, 웹 라이브러리	42
엑셀 라이브러리	82
SAP 라이브러리	120
이미지 인식 라이브러리	129

시스템 라이브러리	145
FlowControl 라이브러리	153
Knox Portal 라이브러리	164
스크립트 전용 라이브러리	170
라이브러리 공통 속성	171
<b>5. 프로세스 플로우 디자인하기</b>	<b>173</b>
5.1 프로세스로 디자인하기	173
프로세스 추가하기	173
대기 동작 추가하기	174
5.2 라이브러리로 디자인하기	176
프로세스 플로우 전용 라이브러리 알아보기	176
<b>6. 서버 연동하기</b>	<b>183</b>
6.1 서버 연동 준비하기	183
6.2 활성화 및 로그인하기	184
활성화하기	184
로그인하기	185
6.3 프로젝트 및 프로세스 플로우 배포하기	186
6.4 프로젝트 및 프로세스 플로우 가져오기	187
6.5 공용 리소스 사용하기	188
<b>7. IPA Bot 사용하기</b>	<b>189</b>
7.1 사용 환경 설정하기	189
7.2 봇 구동하기	191
<b>부록 A. 문제 해결하기</b>	<b>193</b>
A.1 설치 및 등록 관련 문제	193
A.2 자동화 시나리오 작성 및 실행 관련 문제	194
A.3 서버 연계 관련 문제	196
<b>부록 B. FAQ</b>	<b>198</b>
<b>용어 사전</b>	<b>200</b>

# 서문

## 법적 고지 사항

이 매뉴얼을 사용하기 전에 다음 사항을 읽어주십시오.

퍼낸 곳	삼성에스디에스 주식회사
주소	서울특별시 송파구 올림픽로 35길 125
대표 전화	+82 2 1661 3388
전자 메일	<a href="mailto:global.cs@samsung.com">global.cs@samsung.com</a>
홈페이지	<a href="http://www.samsungsds.com">www.samsungsds.com</a>

이 문서에서 다루는 내용은 삼성에스디에스 주식회사가 제공하는 신뢰할 수 있는 정보입니다. 그러나 부정확한 내용이나 오타로 인해 발생하는 문제는 삼성에스디에스 주식회사에서 책임지지 않습니다.

이 문서의 내용과 제품의 사양은 사전 예고 없이 변경될 수 있습니다. 개정에 관한 상세한 정보는 삼성에스디에스 주식회사의 인터넷 홈페이지([www.samsungsds.com](http://www.samsungsds.com))에서 확인할 수 있습니다.

이 문서에 대한 저작권을 포함한 지식재산권은 삼성에스디에스 주식회사에 있습니다. 삼성에스디에스 주식회사의 사전 허가 없이 설명서 내용의 일부 또는 전부를 무단 사용하거나 복제하는 것은 금지되어 있으며, 이는 삼성에스디에스 주식회사의 지식재산권 침해에 해당됩니다.

Copyright © SAMSUNG SDS All rights reserved

## 사용 대상

이 매뉴얼은 Samsung SDS IPA(이하 IPA) 솔루션으로 업무 자동화 프로세스를 디자인하거나 실무에 적용하는 사용자를 대상으로 합니다.

사용자가 IPA 솔루션을 효과적으로 사용하기 위해서는 다음과 같은 지식과 경험이 필요합니다.

- 업무 자동화에 대한 기본적인 지식
- Windows 기반 앱 사용에 대한 기본적인 지식

## 매뉴얼 구성

이 매뉴얼은 다음과 같은 내용으로 구성되어 있습니다.

### 1장. 개요

IPA 솔루션에 대해 소개합니다.

### 2장. 시작하기

IPA 솔루션을 사용하기 위해 필요한 준비 사항과 설치 방법을 설명합니다.

### 3장. IPA Designer 사용하기

IPA Designer의 기본적인 사용 방법을 설명합니다.

### 4장. 프로젝트 디자인하기

IPA Designer를 이용해 업무 자동화 프로젝트를 디자인하는 방법에 대해 설명합니다.

### 5장. 프로세스 플로우 디자인하기

IPA Designer를 이용해 업무 자동화 프로세스 플로우를 디자인하는 방법에 대해 설명합니다.

### 6장. 서버 연동하기

IPA 서버에 연동하여 사용할 수 있는 기능에 대해 설명합니다.

### 7장. IPA Bot 사용하기

IPA Bot의 사용 방법을 설명합니다.

### 부록 A. 문제 해결하기

IPA 솔루션 사용시 빈번히 발생하는 사용자 문제에 대해 가이드합니다.

### 부록 B. FAQ

IPA 솔루션 사용시 자주 발생하는 질문에 대해 정리되어 있습니다.

### 용어 사전

매뉴얼에 사용된 용어에 대해 설명합니다.

## 표기 규약

문서 내용의 이해를 돕기 위해 다음과 같은 표기 규약을 사용합니다.

볼드체 활자	볼드체 활자는 그래픽 유저 인터페이스 요소와 메뉴와 디렉토리 등을 표기할 때 사용합니다. 다만 그래픽 유저 인터페이스 중에서 포털, 창, 팝업창, 마법사, 페이지는 볼드체 활자가 아닌 큰따옴표로 표기합니다.
이탤릭체	이탤릭체는 사용자가 입력해야 하는 변수나 파라미터값을 표기할 때 사용합니다.
고정폭 활자	고정폭 활자는 프로그래밍과 관련된 용어나 코드를 표기할 때 사용합니다. 본문에서 언급할 때는 구분을 위해 명령어, 파라미터명, 레지스트리명, 스크립트명, 프로세스명 등은 고정폭 활자 볼드체로 표기하고, 변수와 파라미터값은 고정폭 활자 이탤릭체로 표기합니다.

## 부연 설명 및 지침

제품을 사용할 때 알아 두면 좋은 팁이나 예외 사항, 제한 사항 등 추가적인 정보를 제공할 때 Note를 사용합니다.



프로젝트를 생성하면 자동으로 빈 프로세스가 생성됩니다.

시스템, 데이터, 단말기 등의 손상 가능성을 알릴 때 Attention을 사용합니다.



속성 화면의 입력값은 실행 시 JavaScript로 변환되므로 JavaScript의 문법에 맞게 입력해야 합니다.

## 개정 이력

솔루션 버전	매뉴얼 버전	매뉴얼 변경 일자	매뉴얼 변경 사항
0.997	1.0	2018.10.31	최초 제정

솔루션 버전	매뉴얼 버전	매뉴얼 변경 일자	매뉴얼 변경 사항

# 1. 개요

사용자는 IPA Designer 및 IPA Bot을 이용해 IPA 솔루션의 자동화 프로세스를 디자인하거나 실무에 적용할 수 있습니다. IPA 솔루션을 효과적으로 사용하기 위해 해당 솔루션의 목적과 기능에 대해 알아보세요.

"개요"에서 다루는 내용은 다음과 같습니다.

- IPA 솔루션 소개
- 지원 기능
- 유즈케이스

## 1.1 IPA 솔루션 소개

IPA(Intelligent Process Automation)는 사람이 수행하던 표준화된 업무를 봇이 대신 처리하도록 자동화하는 솔루션입니다. 봇에게 사용자의 동작을 학습시키면 봇은 학습된 동작을 그대로 모방하여 수행합니다. 봇이 단순 반복적인 작업을 자동으로 처리하는 동안 사용자는 보다 창의적인 업무에 집중하여 비용 절감 및 업무 생산성, 정확성, 효율성을 향상시킬 수 있습니다.

## 1.2 지원 기능

IPA 솔루션은 Windows 앱, 인터넷 익스플로러, 엑셀, SAP 등의 환경을 지원합니다.

규칙 기반의 단순 반복 작업, 시스템 간 정형화된 데이터 입출력, 원본 데이터와 처리 데이터의 비교 등 다양한 업무에 활용할 수 있습니다.

## 1.3 유즈케이스

### Knox Portal 과금 청구 프로세스 자동화

IPA 솔루션을 사용하여 Knox Portal의 사용 요금 청구 프로세스를 자동화합니다.  
매달 3명의 직원이 수행하던 아래와 같은 과정이 자동으로 처리됩니다.

프로세스	대상 앱	내용
사용명세서 메일 전송	웹, 엑셀	<ul style="list-style-type: none"> <li>• Knox Portal 정산 시스템 접속</li> <li>• 사용 내역 다운로드</li> <li>• Excel 양식 정리</li> <li>• 각 회사 담당자에게 메일 전송</li> </ul>
정산 확인 공문 전달	웹, 엑셀	<ul style="list-style-type: none"> <li>• 회신 확인</li> <li>• 과금 내역 확인, 업데이트</li> <li>• 공문 작성</li> <li>• 공문 발송</li> </ul>
계약 생성	SAP, 엑셀	<ul style="list-style-type: none"> <li>• ITO 계약 화면 접속</li> <li>• SAP 계약 정보 확인</li> <li>• 담당자 확인</li> <li>• 계약 생성</li> </ul>

### 일일 업무 보고 프로세스 자동화

시스템에서 필요한 데이터를 추출하여 업무 보고서를 작성하고 메일로 전송하는 과정을 자동으로 처리합니다.  
여기에서는 사업체의 출하 업무를 예로 들어 설명합니다.

프로세스	대상 앱	내용
업무 보고서 작성	웹, 엑셀	<ul style="list-style-type: none"> <li>• 업무 시스템에서 데이터(출고 물량, 전일 출하 물량, 출하 마감 시간 등) 추출</li> <li>• Excel 파일에 자동 입력</li> <li>• 저장</li> </ul>
메일 전송	웹	<ul style="list-style-type: none"> <li>• 수신자 지정</li> <li>• 제목 작성</li> <li>• 보고서 첨부</li> <li>• 메일 전송</li> </ul>

# 2.

## 시작하기

IPA 솔루션을 사용하기 위해서는 라이선스를 준비한 후 IPA Designer 및 IPA Bot을 설치해야 합니다.

"시작하기"에서 다루는 내용은 다음과 같습니다.

- 시스템 요구사항
- 라이선스 준비
- IPA Designer 설치하기
- IPA Bot 설치하기

### 2.1 시스템 요구사항

IPA 솔루션을 설치하고 실행하기 위해 권장되는 하드웨어 및 소프트웨어 사양은 다음과 같습니다.

#### Hardware

---

- CPU Dual Core 이상
- Memory 4 G 이상
- HDD 200 G 이상

#### Software

---

- Window 7 64 bit 이상
- .NET Framework 4.5 이상
- IE11 이상

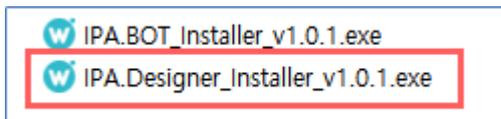
## 2.2 라이선스 준비

라이선스는 관리자에게 문의하세요.

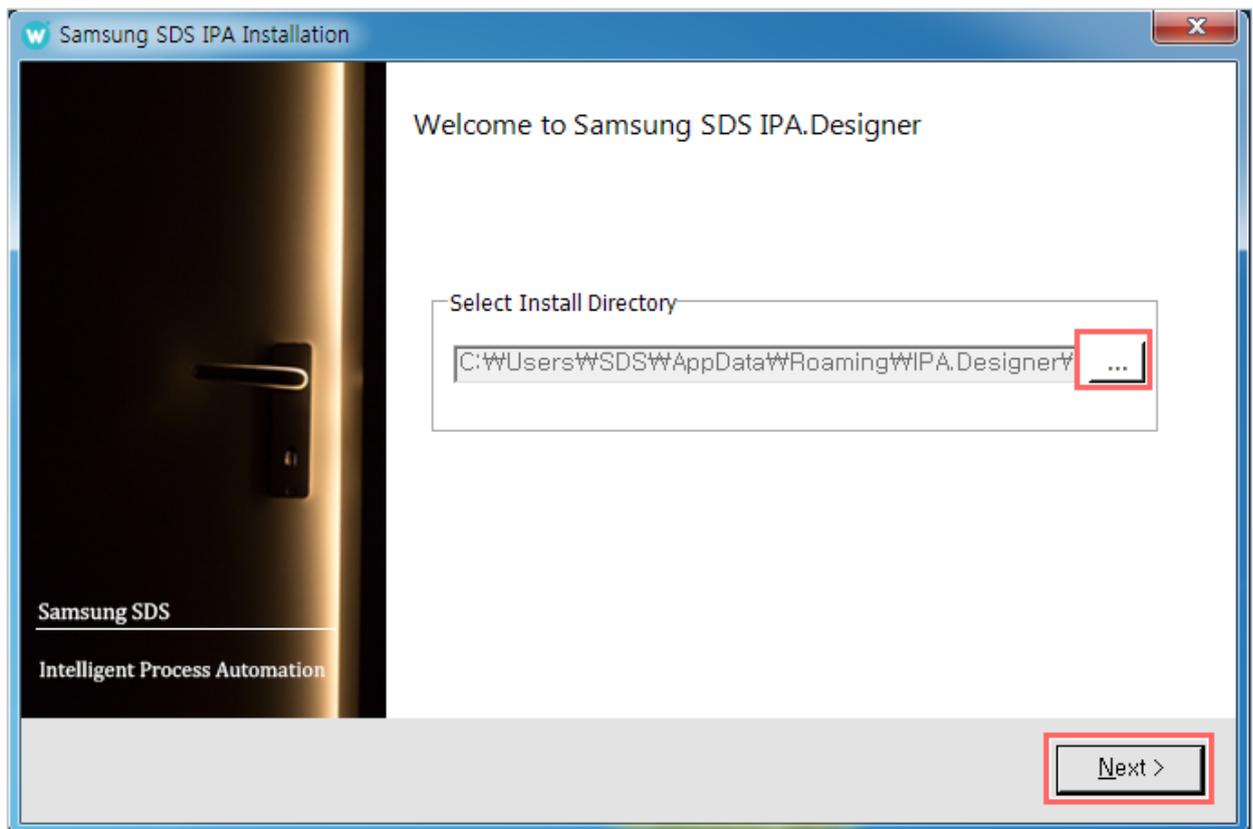
## 2.3 IPA Designer 설치하기

IPA Designer를 설치하려면 아래 절차를 따르세요.

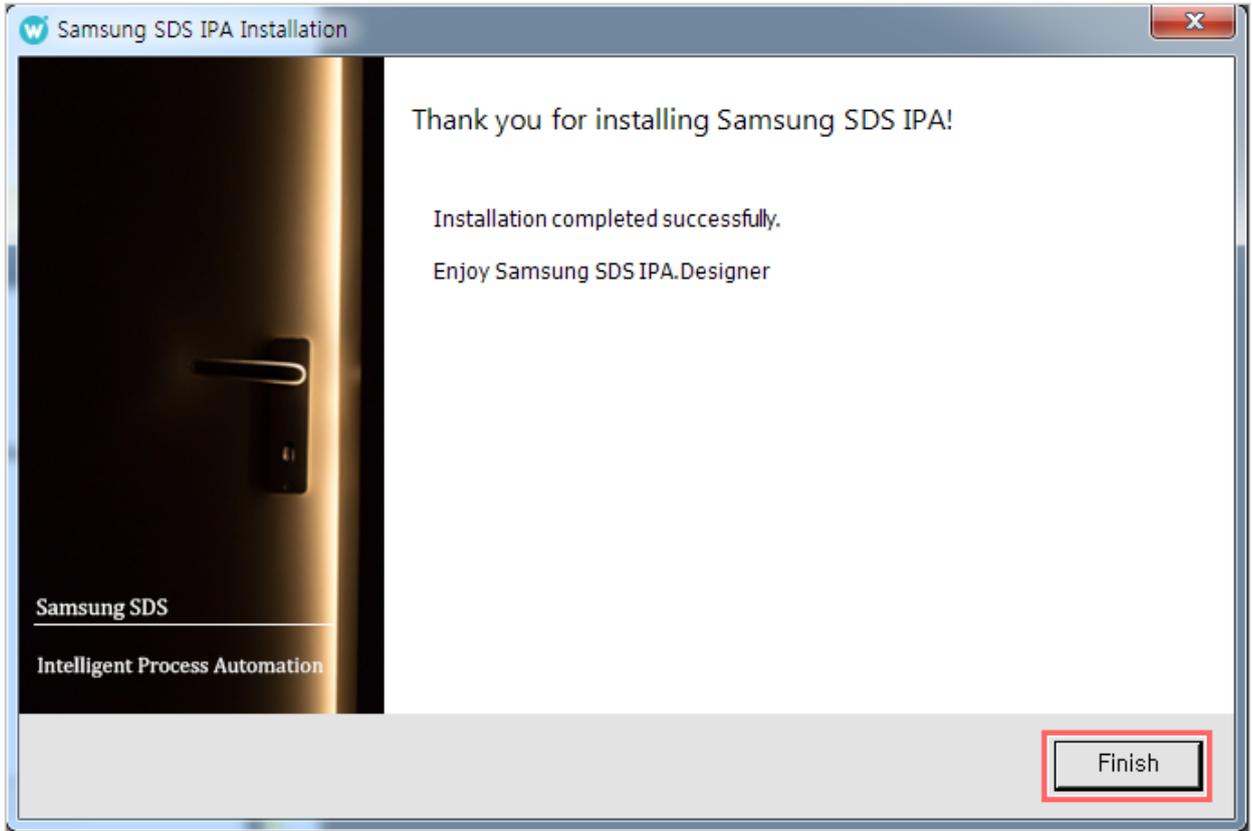
1. IPA Designer 설치 파일을 실행하세요.



2. 설치할 경로를 선택한 후 **Next**를 클릭하세요.



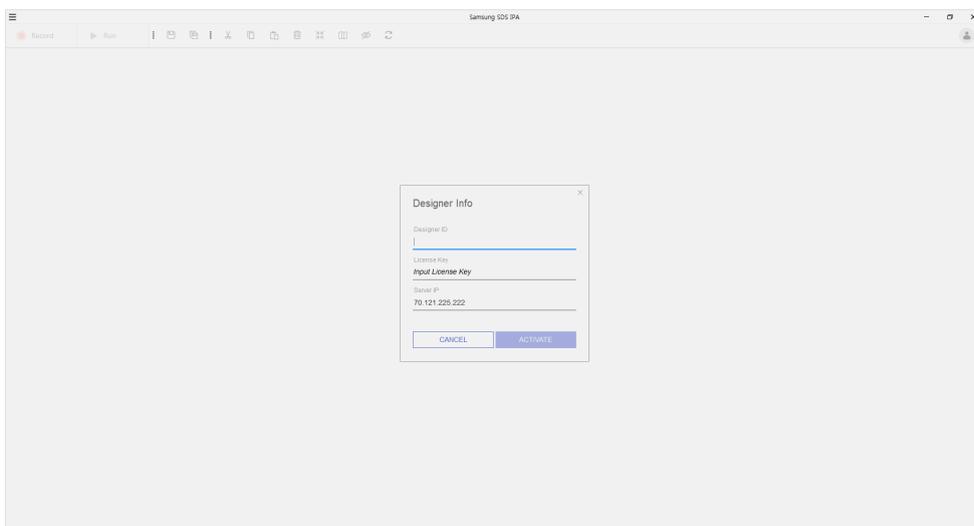
3. **Finish**를 클릭해 설치를 완료하세요.



설치가 완료되면 바탕화면에 바로 가기 아이콘이 추가됩니다.

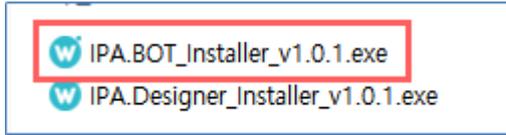


4. 바로 가기 아이콘을 더블클릭해 IPA Designer가 정상적으로 실행되는지 확인하세요.

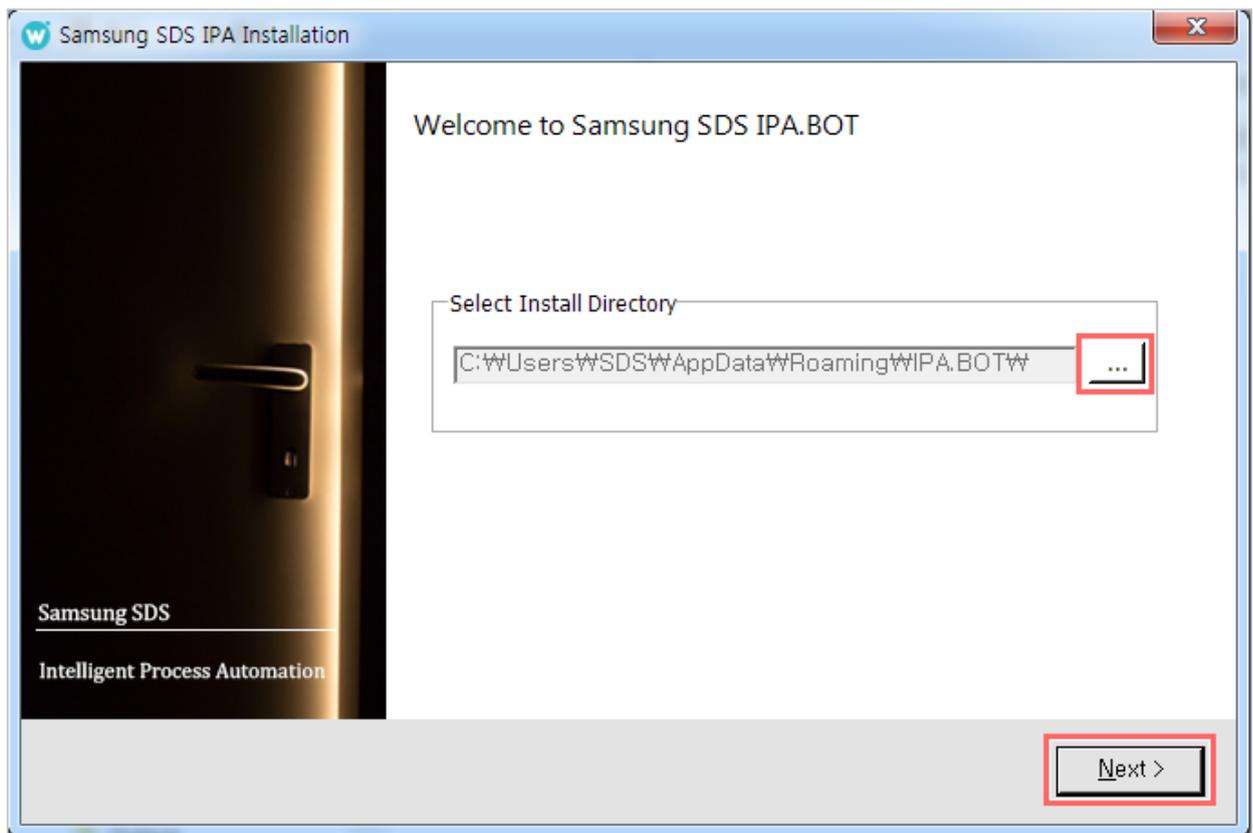


## 2.4 IPA Bot 설치하기

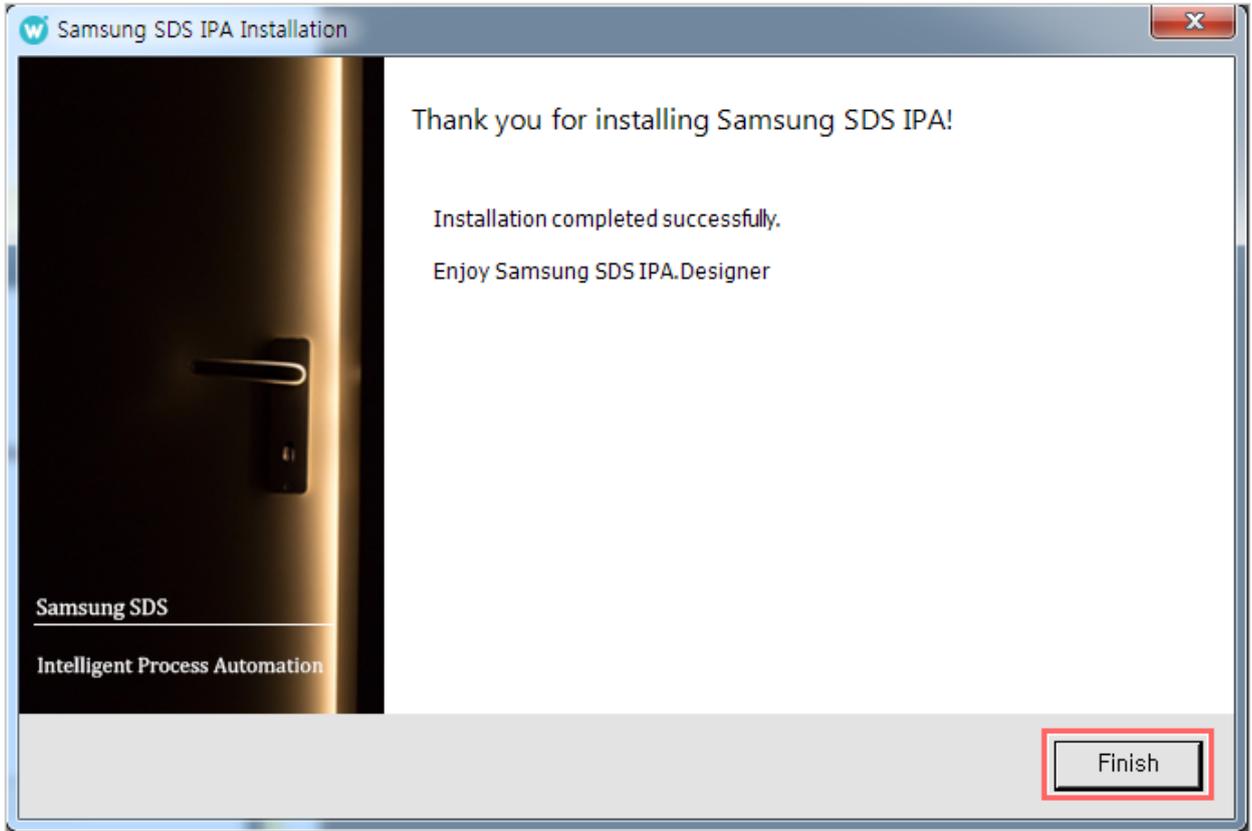
1. IPA Bot 설치 파일을 실행하세요.



2. 설치할 경로를 선택한 후 **Next**를 클릭하세요.



3. **Finish**를 클릭해 설치를 완료하세요.



설치가 완료되면 바탕화면과 우측 하단에 바로 가기 아이콘이 추가됩니다.



4. 바로 가기 아이콘을 더블클릭해 IPA Bot이 정상적으로 실행되는지 확인하세요.

# 3.

## IPA Designer 사용하기

IPA Designer는 사용자가 자동화 프로젝트나 프로세스 플로우를 디자인할 수 있는 프로그램입니다. 디자인하기에 앞서 화면의 구성 요소와 기능을 이해하고 IPA Designer의 기본적인 사용 방법을 이해합니다.

"IPA Designer 사용하기"에서 다루는 내용은 다음과 같습니다.

- IPA Designer 화면 살펴보기
- 새 프로젝트 및 프로세스 플로우 시작하기
- 프로젝트 및 프로세스 플로우 저장하기
- 프로젝트 실행하기
- 프로세스 플로우 실행하기

### 3.1 IPA Designer 화면 살펴보기

IPA Designer를 구성하는 화면 요소를 살펴보고 사용법을 알 수 있습니다. IPA Designer는 아래와 같은 화면 요소로 구성되어 있습니다.

- 메뉴
- 툴바
- 프로젝트 화면
- 프로세스 플로우 검색 화면
- 라이브러리 화면
- 프로세스 플로우 전용 라이브러리 화면
- 디자인 화면
- 출력 화면
- 변수 화면
- 공용 리소스 화면
- 속성 화면
- 도움말 화면

## 메뉴

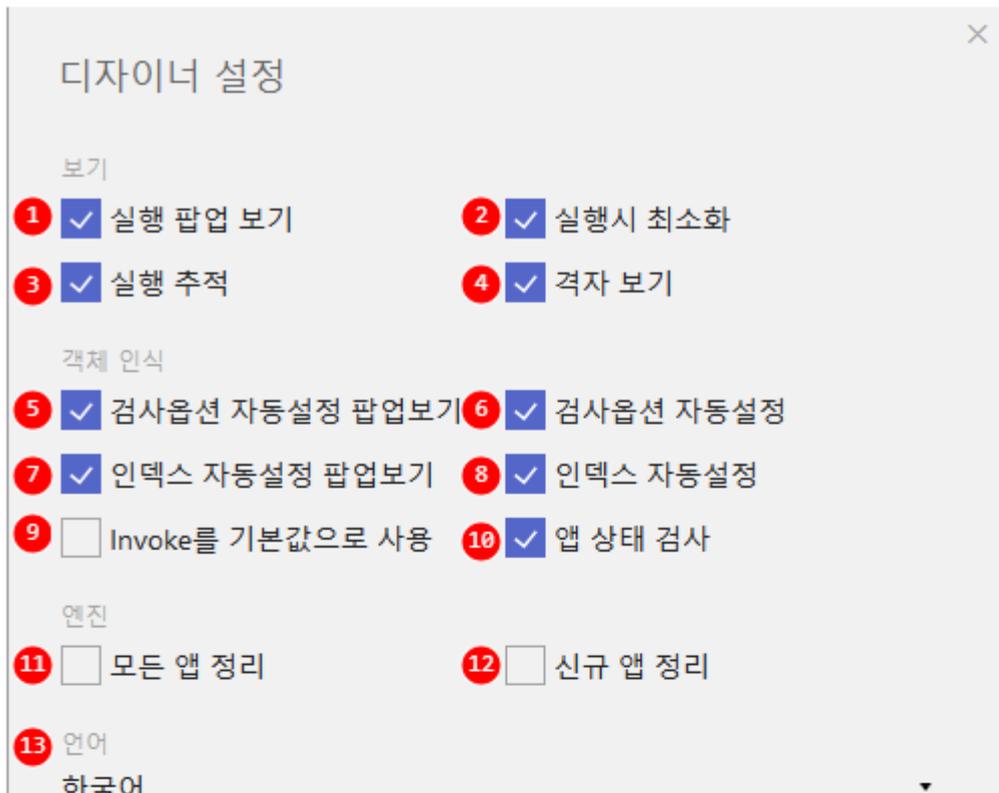
IPA Designer의 기본 메뉴 및 설정 메뉴를 사용할 수 있습니다.  
화면 왼쪽 상단에서  누르세요.



번호	설명
①	새로운 프로젝트나 프로세스 플로우를 만듭니다.
②	저장된 프로젝트나 프로세스 플로우를 엽니다.
③	최근에 작업한 파일을 엽니다.
④	현재 작업 중인 파일을 저장합니다.
⑤	현재 작업 중인 프로젝트를 다른 이름으로(.ipa 확장자) 저장합니다.
⑥	열려 있는 모든 파일을 저장합니다.
⑦	서버에서 프로젝트나 프로세스 플로우를 가져옵니다.
⑧	프로젝트나 프로세스 플로우를 서버로 배포합니다.
⑨	현재 작업 중인 프로젝트나 프로세스 플로우의 복사본을 만듭니다.
⑩	레코딩 기능으로 프로젝트를 디자인합니다.
⑪	디자인한 프로젝트를 실행하거나 중지합니다.
⑫	CPU 및 메모리 사용량을 확인하고 검사할 수 있습니다.
⑬	IPA Designer 및 서버 설정을 변경합니다.
⑭	도움말을 확인합니다.
⑮	프로그램을 종료합니다.

## 디자이너 설정

IPA Designer의 설정을 변경할 수 있습니다.  
메뉴에서 **설정 > 디자이너 설정**을 클릭하세요.



14 실행 모드  
기본

15 스텝 Interval(ms)  
1

16 키입력 Interval(ms)  
10

17 엔진 서비스 포트  
9002

18 앱 정리 제외 목록

19 스텝 이미지 최대 개수  
0

20 수행이력 백업 개수  
10

21 앱 Interval(ms)  
100

22 자동 저장 Interval (min)  
3

취소      확인

번호	설명
1	실행 시 디자이너 최소화 여부를 선택하는 팝업이 나타납니다.
2	실행 시 디자이너 최소화 여부를 설정합니다.
3	현재 어느 스텝이 진행중인지 추적하여 디자이너에서 표시합니다.
4	디자인 화면에 격자를 보여줍니다.
5	검사옵션 설정 팝업을 보여줍니다.
6	검사옵션을 자동으로 설정합니다.
7	인덱스 설정 팝업을 보여줍니다.
8	인덱스를 자동으로 설정합니다.
9	Invoke 를 기본값으로 사용할지 여부를 설정합니다.

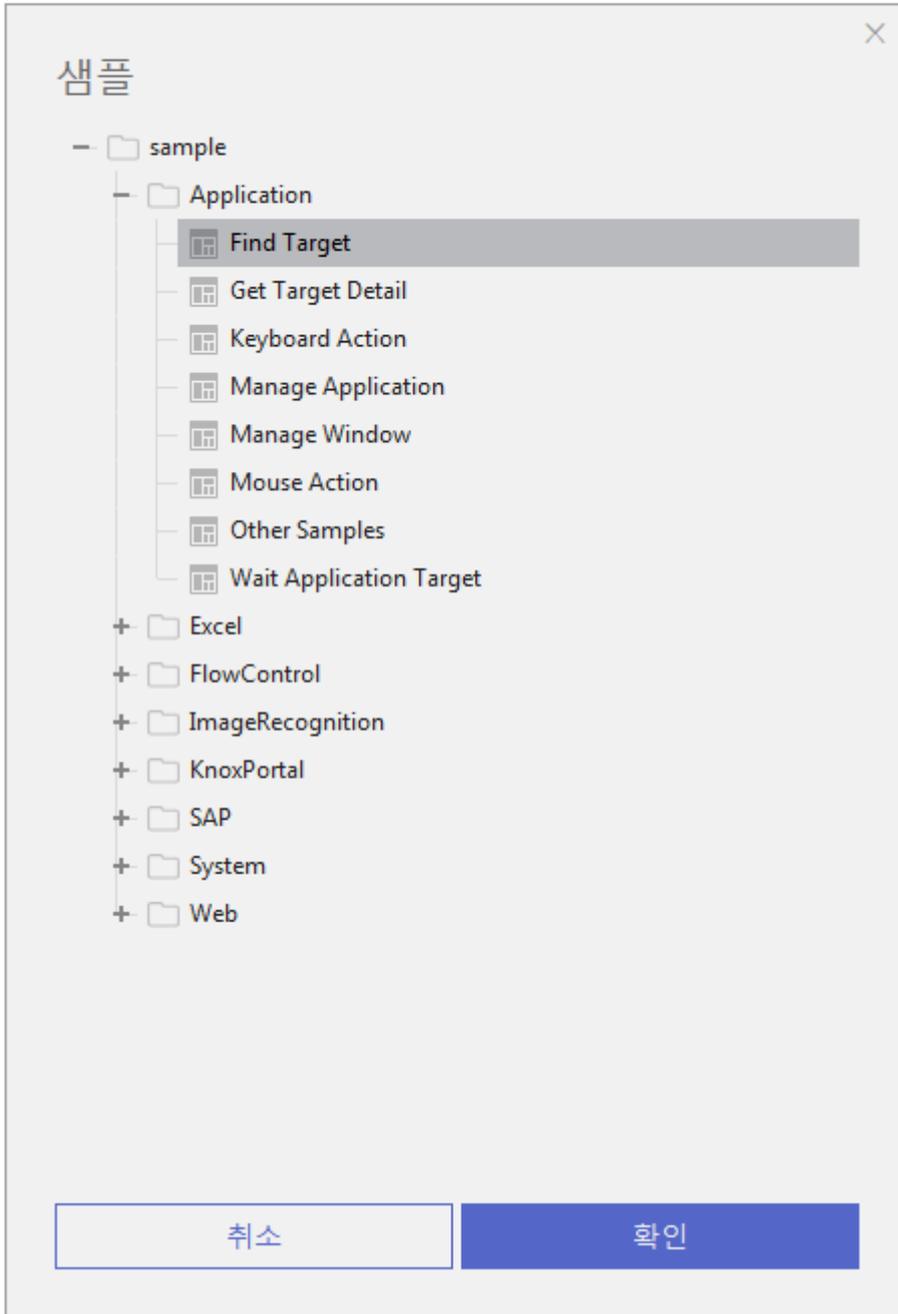
번호	설명
10	대상 앱이 응답없음 상태인 경우 무시하고 지나갑니다.
11	안정적인 자동화 실행을 위해 모든 앱을 정리합니다.
12	안정적인 자동화 실행을 위해 봇에 의해 실행된 모든 앱을 정리합니다.
13	디자이너 언어팩을 설정합니다.
14	실행 모드를 설정합니다. (일반 모드, 안정성 우선 모드, 성능 우선 모드)
15	각 스텝별 수행 간격을 설정합니다.
16	키 입력 간격을 설정합니다.
17	엔진이 실행될 포트를 설정합니다.
18	앱 정리 시 제외될 목록을 설정합니다. 목록은 세미콜론으로 구분합니다. (예 : notepad.exe;mspaint.exe)
19	수행화면을 저장할 이미지 최대 개수를 설정합니다. 예를 들어 스텝이 10개이고 최대 개수를 3으로 하면 마지막 3개인 8, 9, 10번째 스텝 실행 시 화면이 저장됩니다.
20	백업할 수행 이력 개수를 설정합니다.
21	각 스텝별로 앱이 달라졌을 때 수행 간격을 설정합니다. App Title 정보로 구분되므로 동일한 프로그램이라도 제목이 달라지면 다른 앱으로 간주됩니다. (예 : 메모장 - 제목 없음, 메모장 - 테스트)
22	프로세스 혹은 작업 편집 시 자동으로 임시 저장될 간격을 설정합니다.

## 샘플

IPA Designer에서 제공하는 라이브러리에 대한 샘플을 확인할 수 있습니다.

메뉴에서 **도움말 > 샘플**을 클릭하세요.

샘플은 배포에 포함되지 않으며, 파일을 다운받은 후 Designer 실행파일이 있는 경로에 **Samples** 폴더로 복사하면 정상적으로 표시됩니다.



## 툴바

툴바는 사용자가 IPA Designer를 사용하기 위해 필요한 도구를 모아둔 화면입니다. 디자인, 실행, 저장, 편집 순으로 정렬되어 있습니다.



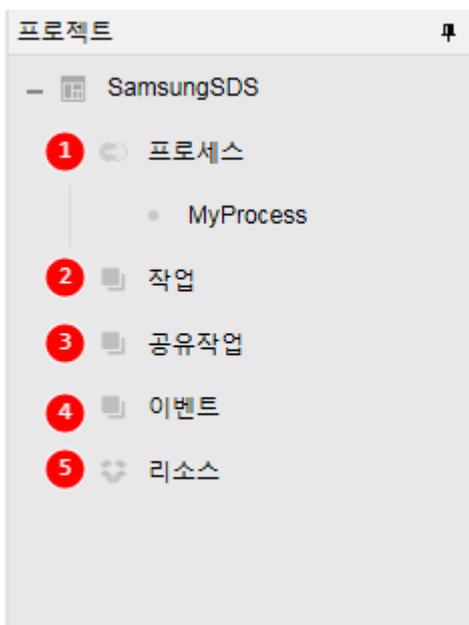
그룹을 마우스로 드래그해 툴바의 순서를 변경할 수 있습니다.



번호	설명
①	레코딩을 시작합니다.
②	현재 열려 있는 프로세스나 작업을 실행합니다.
③	현재 열려 있는 프로세스나 작업을 저장합니다.
④	열려 있는 모든 프로세스나 작업을 저장합니다.
⑤	실행을 취소합니다.
⑥	선택된 노드를 잘라냅니다.
⑦	선택된 노드를 복사합니다.
⑧	복사하거나 잘라낸 노드를 붙여넣기 합니다.
⑨	선택된 노드를 삭제합니다.
⑩	선택된 액티비티 그룹을 접거나 펼칩니다.
⑪	미니맵을 보이거나 숨깁니다.
⑫	선택된 노드를 비활성화합니다.
⑬	선택된 노드를 제외한 모든 노드를 선택합니다.

## 프로젝트 화면

프로젝트의 구조를 보여주는 화면입니다. 프로젝트는 프로세스, 작업, 이벤트, 리소스로 구성되어 있습니다.



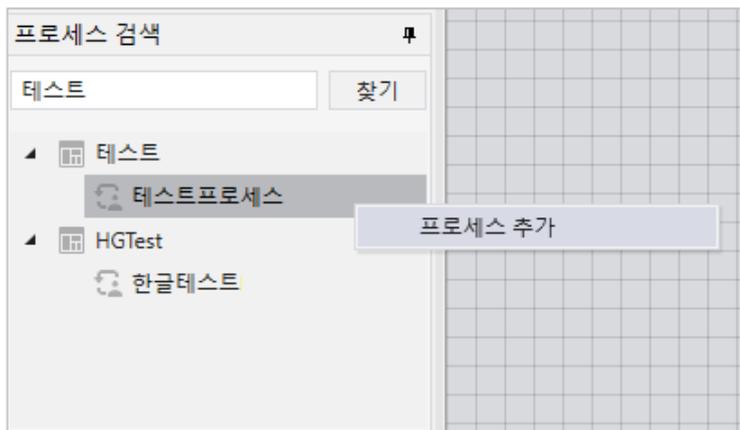
번호	설명
①	프로세스는 붓에서 수행하는 업무 단위로서 다양한 라이브러리와 작업으로 구성됩니다. 하위 그룹을 가질 수 있으며 자기 자신을 제외한 프로세스를 포함할 수 있습니다. 마우스 오른쪽 버튼을 클릭하여 복사, 이름 변경, 삭제, 서버에서 가져오기 등의 편집 기능을 사용할 수 있습니다.
②	작업은 프로세스와 동일하게 다양한 라이브러리와 작업으로 구성되지만 붓에서 수행할 수 없는 업무 단위입니다. 하위 그룹을 가질 수 있으며 자기 자신을 제외한 작업을 포함할 수 있습니다. 마우스 오른쪽 버튼을 클릭하여 복사, 이름 변경, 삭제, 서버에서 가져오기 등의 편집 기능을 사용할 수 있습니다.
③	공유작업은 서버에 있는 공유작업만 가져올 수 있으며 수정이 불가능합니다. 마우스 오른쪽 버튼을 클릭하여 삭제, 새로고침, 서버에서 가져오기 등의 편집 기능을 사용할 수 있습니다.
④	이벤트는 프로세스나 작업의 수행 전후, 에러 발생 시 사용할 수 있습니다. 내용은 작업과 동일하지만 서버에서 가져오기 기능은 제공하지 않습니다.
⑤	리소스는 프로젝트 수행에 필요한 파일 단위입니다. 사용자의 PC에 있는 파일을 가져와서 프로젝트 폴더 내에 별도로 관리합니다.

프로젝트를 구성하는 각 항목들의 관계는 아래 이미지를 참고하세요.



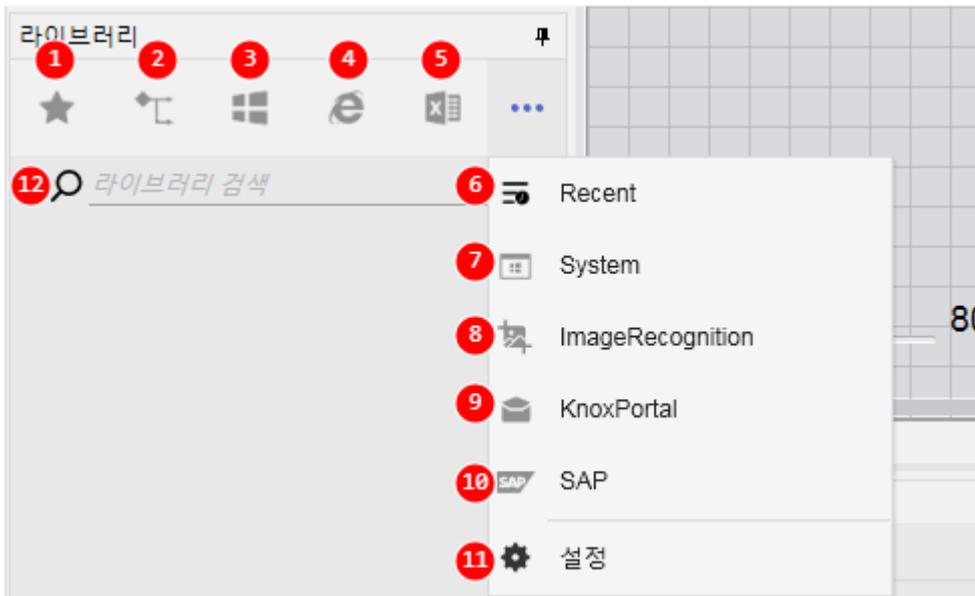
## 프로세스 검색 화면

프로세스 플로우를 디자인하기 위해 프로세스를 검색하고 사용할 수 있는 화면입니다. 자세한 내용은 [4. 프로젝트 디자인하기](#)를 참고하세요.



## 라이브러리 화면

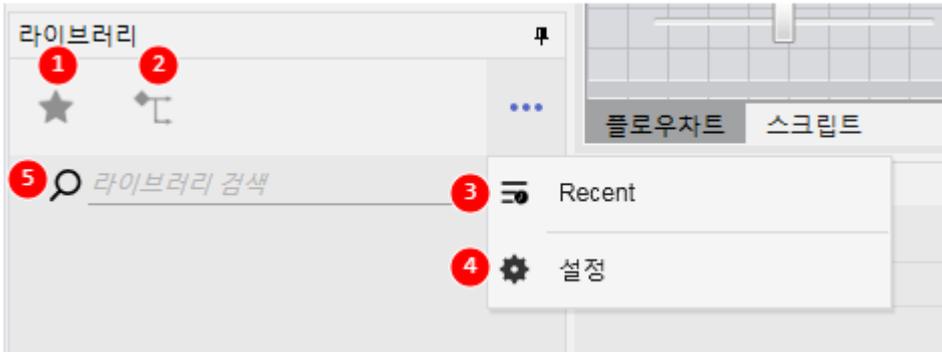
다양한 기능을 모아놓은 라이브러리에서 사용자는 원하는 기능을 찾아 프로세스, 작업 및 이벤트의 플로우차트를 구성할 수 있습니다. 라이브러리는 사용할 수 있는 대상 앱이나 기능에 따라 앱, 웹, 엑셀, SAP, 이미지 인식, 시스템, FlowControl, KnoxPortal 항목으로 구분되어 있습니다. 라이브러리 화면에서 사용할 항목을 선택한 후 원하는 기능을 찾아보세요. 자주 사용하는 라이브러리는 즐겨찾는 라이브러리에 추가할 수 있습니다. 사용할 수 있는 라이브러리 및 자세한 설명은 [4.2 라이브러리로 디자인하기](#) 및 [라이브러리 알아보기](#)를 참고하세요.



번호	설명
1	즐겨찾는 라이브러리
2	FlowControl 라이브러리
3	앱 라이브러리
4	웹 라이브러리
5	엑셀 라이브러리
6	최근 사용한 라이브러리
7	시스템 라이브러리
8	이미지 인식 라이브러리
9	KnoxProtal 라이브러리
10	SAP 라이브러리
11	라이브러리 설정
12	라이브러리 검색

## 프로세스 플로우 전용 라이브러리 화면

프로세스 플로우 디자인 화면에서는 프로세스 플로우 전용 라이브러리를 사용할 수 있습니다. If, For 등 플로우차트를 구성하기 위한 라이브러리는 프로젝트 디자인 화면과 동일합니다. 프로세스 플로우 전용 라이브러리에 대한 자세한 설명은 [프로세스 플로우 전용 라이브러리 알아보기](#)를 참고하세요.



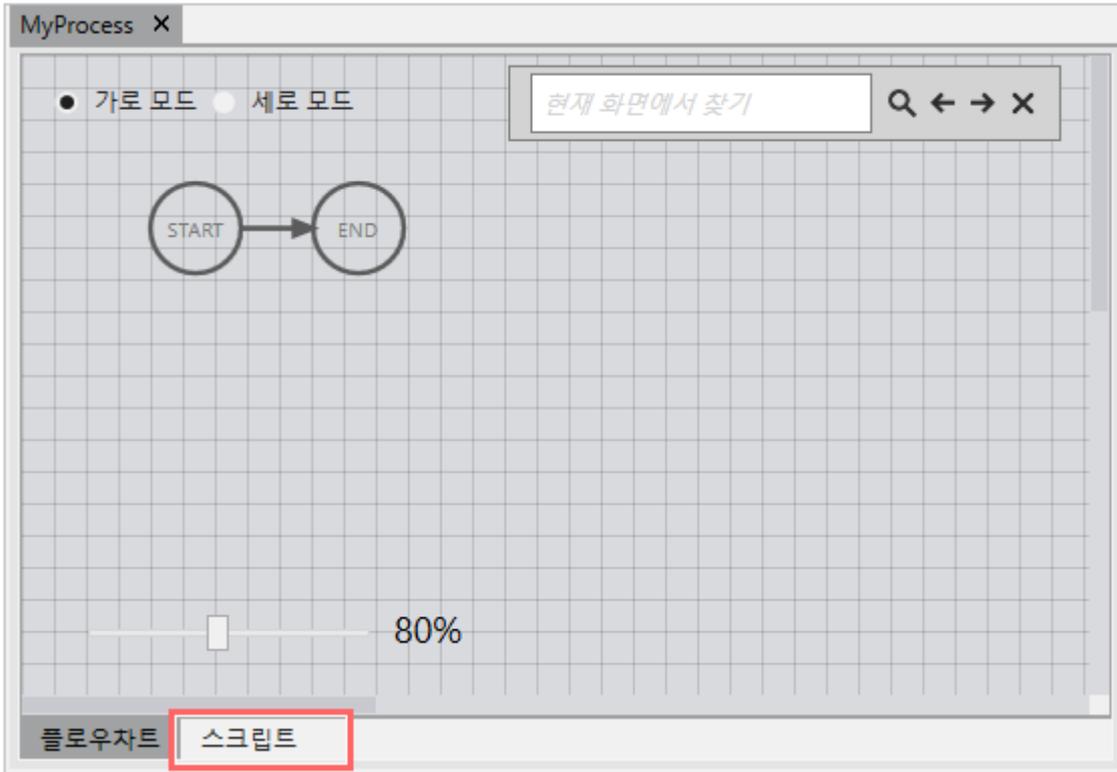
번호	설명
1	즐거찾는 라이브러리
2	프로세스 플로우 라이브러리
3	최근 사용한 라이브러리
4	라이브러리 설정
5	라이브러리 검색

## 디자인 화면

IPA Designer의 메인 화면입니다. 다양한 라이브러리와 작업을 추가해 플로우차트를 구성하고 편집할 수 있습니다. 플로우차트는 모두 스크립트로 변환되며, 변환된 스크립트는 **스크립트** 탭을 클릭해 확인할 수 있습니다.



- 스크립트 탭을 클릭하면 변경된 플로우차트가 자동으로 저장됩니다.
- 스크립트 작성 시 사용할 수 있는 라이브러리는 [Knox Portal 라이브러리](#)를 참고하세요.



## 액티비티 찾기

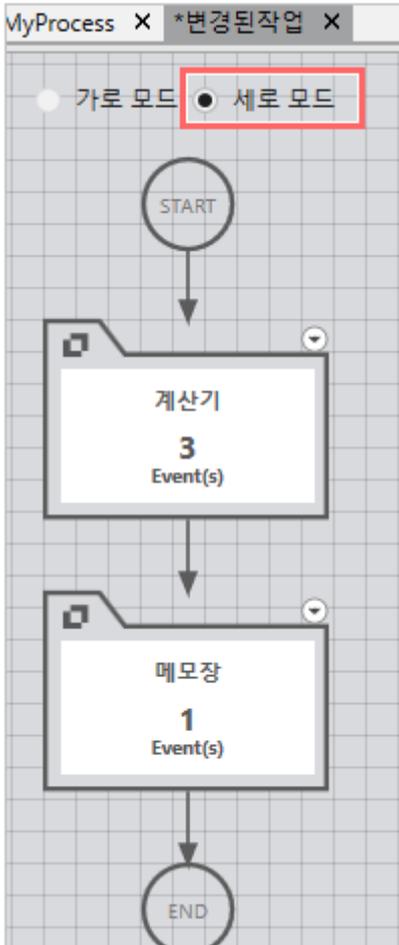
입력한 내용을 포함하는 액티비티를 검색할 수 있습니다.

디자인 화면의 검색창에 검색할 내용을 입력한 후  클릭하거나 키보드의 Enter 키를 누르세요.

입력된 내용이 포함된 노드가 강조되며 첫 번째 노드가 선택됩니다.

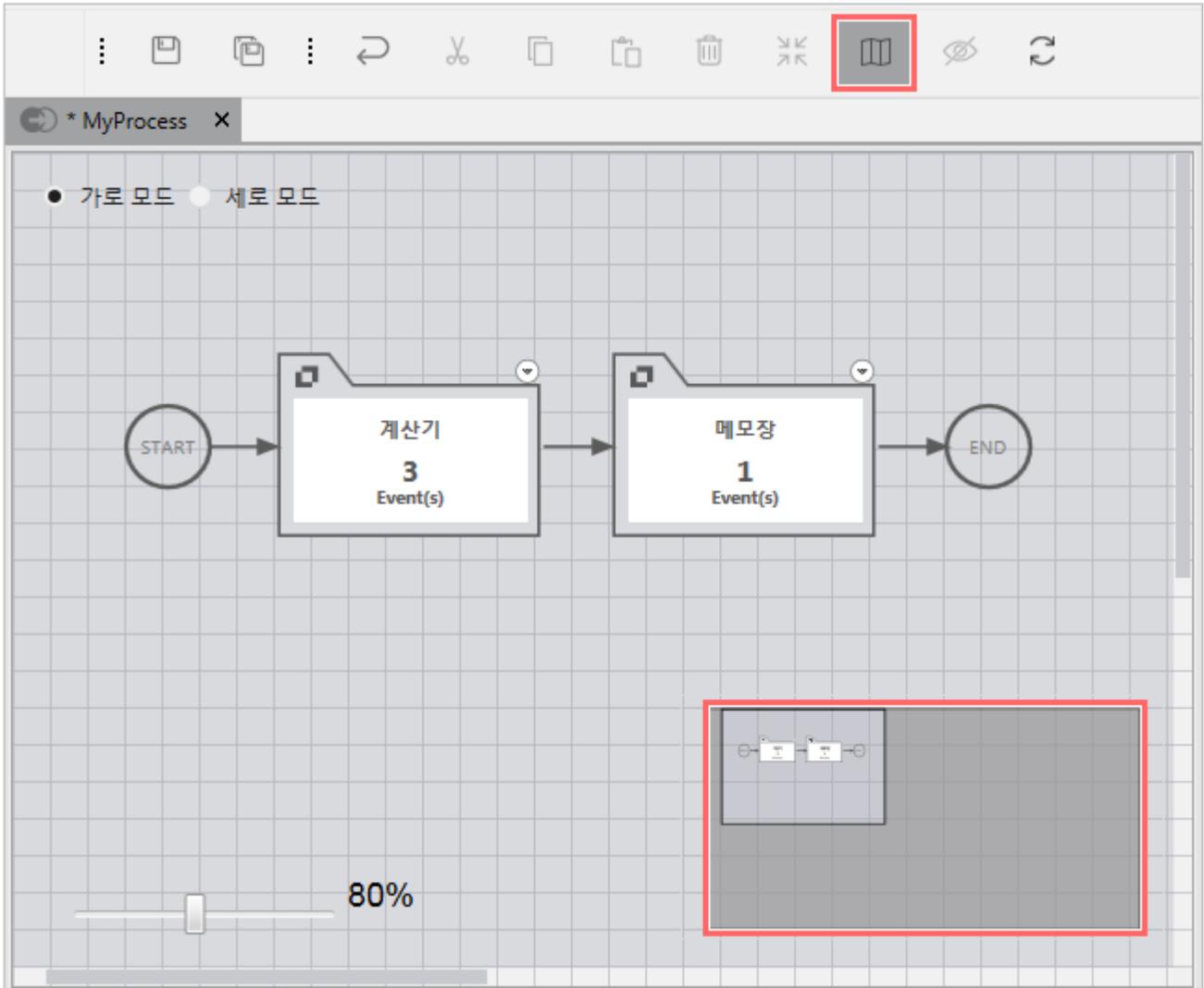
## 차트 방향 변경하기

차트를 세로 방향으로 변경하려면 디자인 화면에서 **세로 모드**를 클릭하세요.



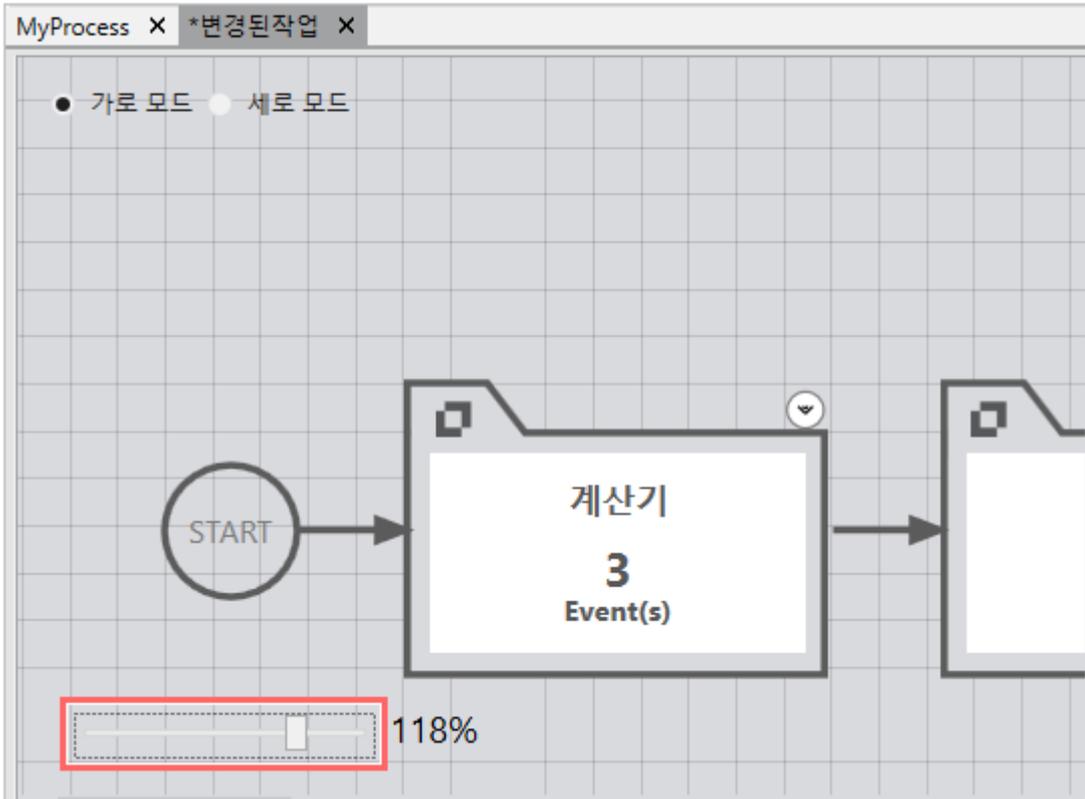
## 미니맵 보기

사용자는  클릭해 차트 화면에 미니맵을 표시할 수 있습니다. 사용자는 마우스를 통해 미니맵을 조작하여 쉽게 원하는 위치로 화면을 이동시킬 수 있습니다.



## 화면 크기 변경하기

플로우차트가 길어져서 한눈에 보기 불편한 경우 사용자는 화면을 확대하거나 축소할 수 있습니다. 화면 좌측 하단의 조절 바를 드래그해 화면 크기를 변경하세요.



## 출력 화면

프로세스나 작업이 수행될 때 로그를 출력하는 화면으로 현재 진행 상황 및 에러 메시지를 확인할 수 있습니다.

```

출력
2019-01-15 15:27:08,803 [1] DEBUG WorkFlowViewModel
Load work flow : Test프로세스플로우
2019-01-15 15:27:08,906 [1] DEBUG DiagramGroupViewer
DiagramViewer : Test프로세스플로우 Subscribe Events
2019-01-15 15:27:09,015 [1] INFO ProcessViewer
Load Process : Test프로세스플로우
2019-01-15 15:27:09,015 [1] DEBUG ProcessViewer
C:\Users\SW\Documents\IPA Designer Projects\Test프로세스플로우\Test프로세스플로우.flow is loaded.
    
```

공용 리소스 변수 출력

## 공용 리소스 화면

서버에 등록되어 있는 공용 리소스를 가져와서 사용하기 위한 화면입니다. 공용 리소스를 사용하려면 먼저 로그인が必要です. 자세한 내용은 [6.5 공용 리소스 사용하기](#)를 참고하세요.



## 변수 화면

변수 화면은 라이브러리의 속성 값으로 사용되는 변수를 추가하고 관리하는 화면입니다. 변수의 초기값을 설정할 수 있으며, 필요 시 설명을 입력해 어떤 변수인지 쉽게 파악할 수 있습니다.

변수는 사용 범위 및 목적에 따라 4가지 타입으로 구분할 수 있습니다.

- **Global:** 현재 프로젝트 전체에서 사용할 수 있는 전역 변수입니다.
- **Normal:** 현재 작업 중인 프로세스나 작업 내에서만 사용할 수 있는 멤버 변수입니다.
- **Input:** 프로세스나 작업 수행 시 파라미터로 전달하기 위해 필요한 멤버 변수입니다.
- **Output:** 프로세스나 작업 수행 완료 시 출력값을 받아오기 위한 멤버 변수입니다.

Global 타입의 변수는 'Global.var1'과 같이 'Global' 키워드를 이용합니다.

Normal, Input, Output 타입의 변수는 'this.var1'과 같이 'this' 키워드를 이용합니다.



## 속성 화면

선택한 노드의 상세 정보를 확인하고 속성값을 설정할 수 있는 화면입니다. 선택한 항목에 따라 다른 정보가 표시됩니다. 자세한 설명은 [라이브러리 알아보기](#)에서 각 라이브러리의 속성 화면을 참고하세요.

속성
☰

---

### Activity

Index

---

Check Name  True  False

Check Value  True  False

Check Size  True  False

### Target

Application

---

Title

---

Type

---

Name

---

Value

---

### More ▲

Timeout

---

Search Bounds

---

Object Path

---

Check Title  True  False

Focus Before  True  False

Draw Bounds  True  False

Ignore Error  True  False

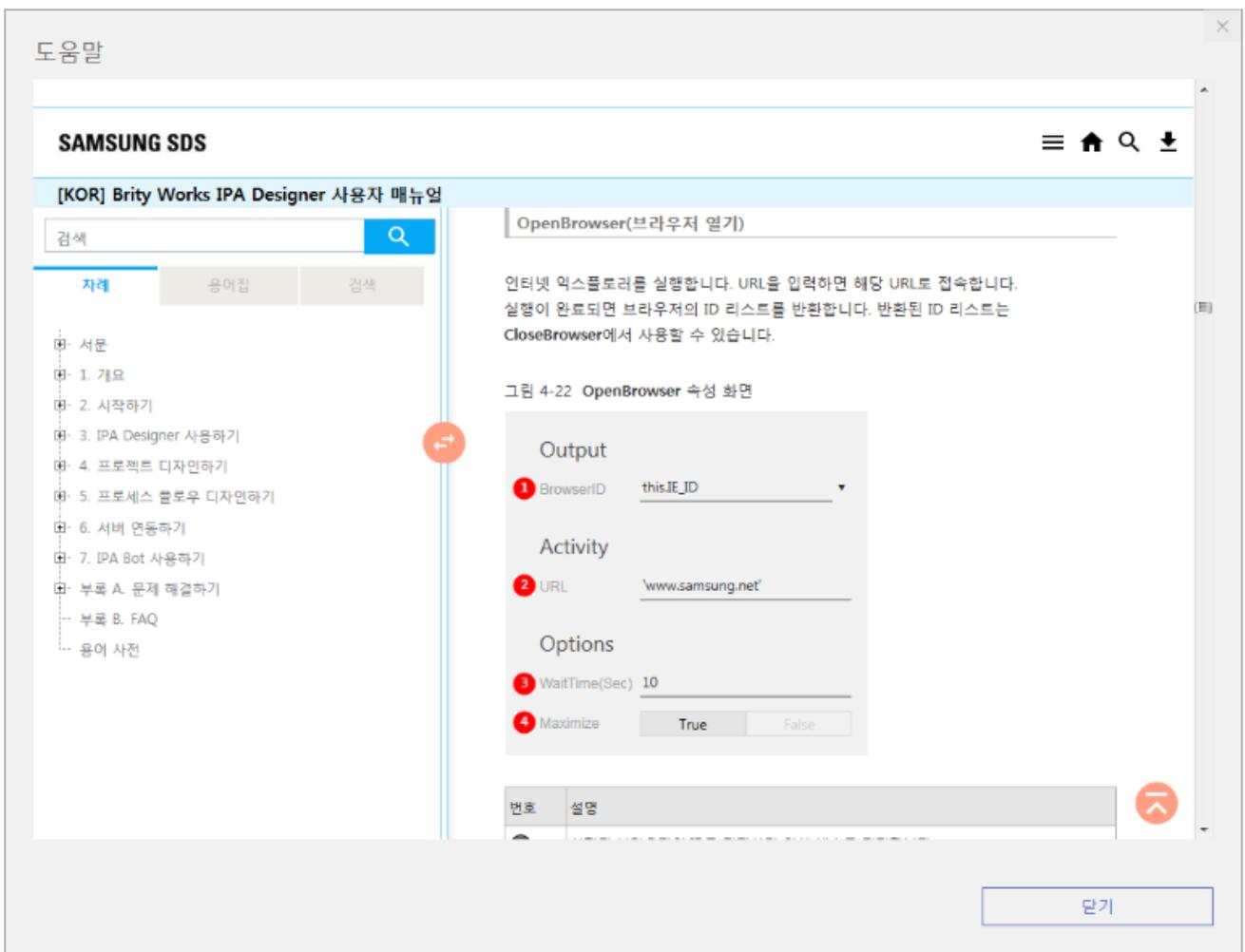
Check Path  True  False



속성 화면의 입력값은 실행 시 JavaScript로 변환되므로 JavaScript의 문법에 맞게 입력해야 합니다. 입력값은 변수 화면에서 Global, this 키워드를 이용해 선언한 변수만 사용할 수 있으며, 스크립트 내에서 선언한 지역 변수는 사용할 수 없습니다.

## 도움말 화면

라이브러리에 대한 상세 정보를 확인할 수 있는 화면입니다. 1개의 액티비티를 클릭 후 F1 키를 누르면 해당 액티비티에 대한 도움말 화면이 표시됩니다.



## 3.2 새 프로젝트 및 프로세스 플로우 시작하기

새 프로젝트 또는 프로세스 플로우를 시작합니다. 프로세스, 작업, 이벤트를 추가하거나 기존의 리소스를 추가하여 프로젝트를 디자인할 수 있습니다. 유사한 작업은 그룹으로 묶어 편리하게 관리할 수 있습니다.

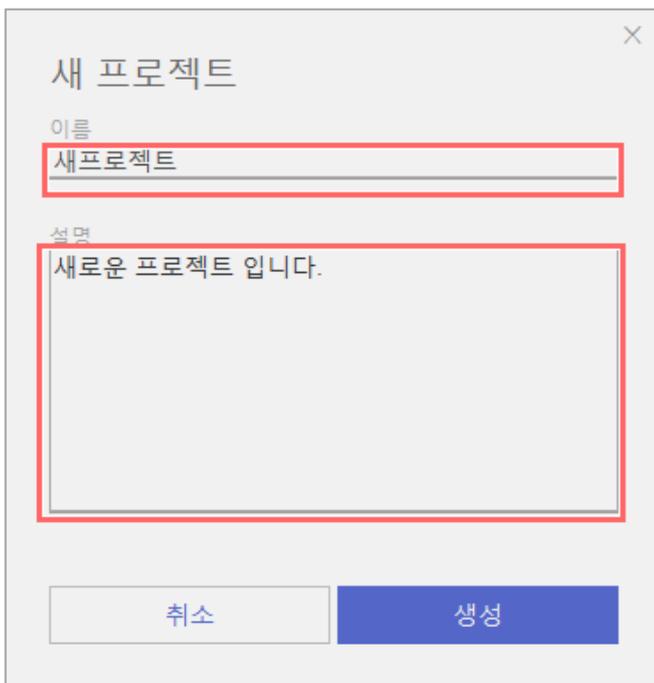
"새 프로젝트 및 프로세스 플로우 시작하기"에서는 아래 내용에 대해 설명합니다.

- 새 프로젝트 또는 프로세스 플로우 만들기
- 새 작업 만들기
- 새 프로세스 만들기
- 새 이벤트 만들기
- 리소스 추가하기
- 새 그룹 만들기
- 프로젝트 편집하기

### 새 프로젝트 또는 프로세스 플로우 만들기

새 프로젝트 또는 프로세스 플로우를 만들려면 아래 절차를 따르세요.

1.  > 새로 만들기를 클릭하세요.
2. 새 프로젝트 또는 새 프로세스 플로우를 클릭하세요.
3. "새 프로젝트" 팝업창에서 새로 만들 프로젝트 또는 프로세스 플로우의 이름과 설명을 입력하세요.



새 프로젝트

이름  
새 프로젝트

설명  
새로운 프로젝트 입니다.

취소    생성

4. **생성**을 클릭하세요.

새 프로젝트 또는 프로세스 플로우가 생성됩니다.

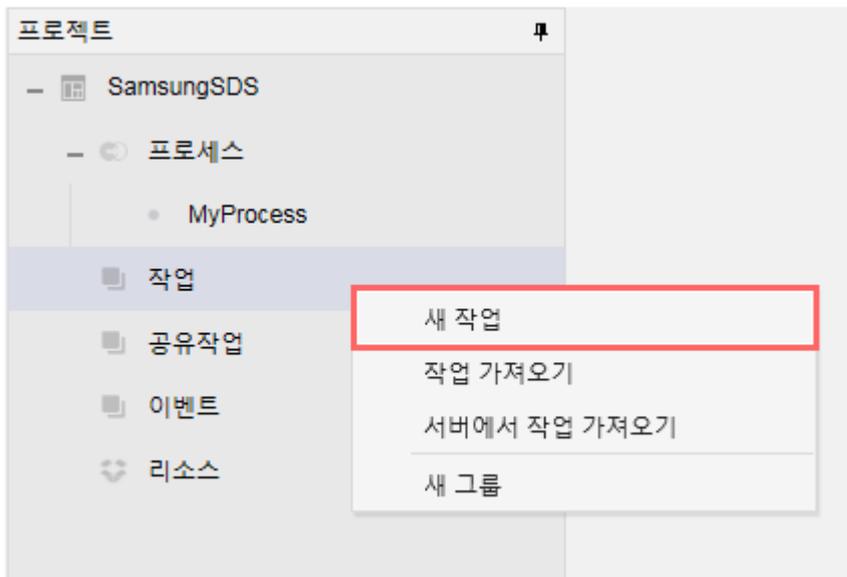


프로젝트를 생성하게 되면 사용자의 편의를 위하여 "MyProcess" 라는 이름의 빈 프로세스가 만들어집니다.

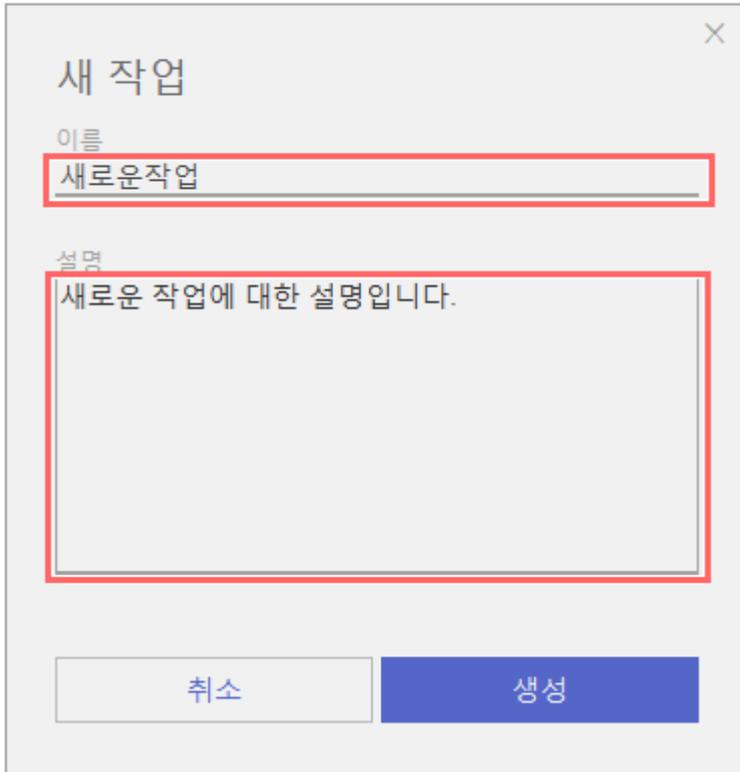
## 새 작업 만들기

새 작업을 만들려면 아래 절차를 따르세요.

1. 프로젝트 화면에서 **작업**을 마우스 오른쪽 버튼으로 클릭한 후 **새 작업**을 클릭하세요.



2. "새 작업" 팝업창에서 새로 만들 작업의 이름을 입력한 후 **생성**을 클릭하세요.



새 작업

이름  
새로운작업

설명  
새로운 작업에 대한 설명입니다.

취소      생성

새 작업이 생성되고 디자인 화면이 열립니다.

## 새 프로세스 만들기

새 프로세스를 만들려면 아래 절차를 따르세요.

1. 프로젝트 화면에서 **프로세스**를 마우스 오른쪽 버튼으로 클릭한 후 **새 프로세스**를 클릭하세요.
2. "새 프로세스" 팝업창에서 새로 만들 프로세스의 이름을 입력한 후 **생성**을 클릭하세요.

새 프로세스가 생성되고 디자인 화면이 열립니다.

## 새 이벤트 만들기

새 이벤트를 만들려면 아래 절차를 따르세요.

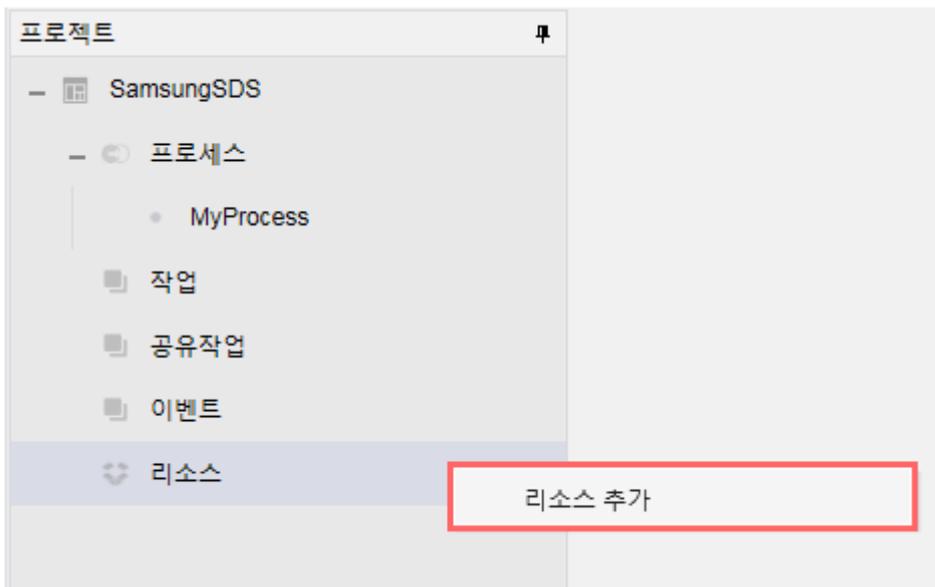
1. 프로젝트 화면에서 **이벤트**를 마우스 오른쪽 버튼으로 클릭한 후 **새 이벤트**를 클릭하세요.
2. "새 이벤트" 팝업창에서 새로 만들 이벤트의 이름을 입력한 후 **생성**을 클릭하세요.

새 이벤트가 생성되고 디자인 화면이 열립니다.

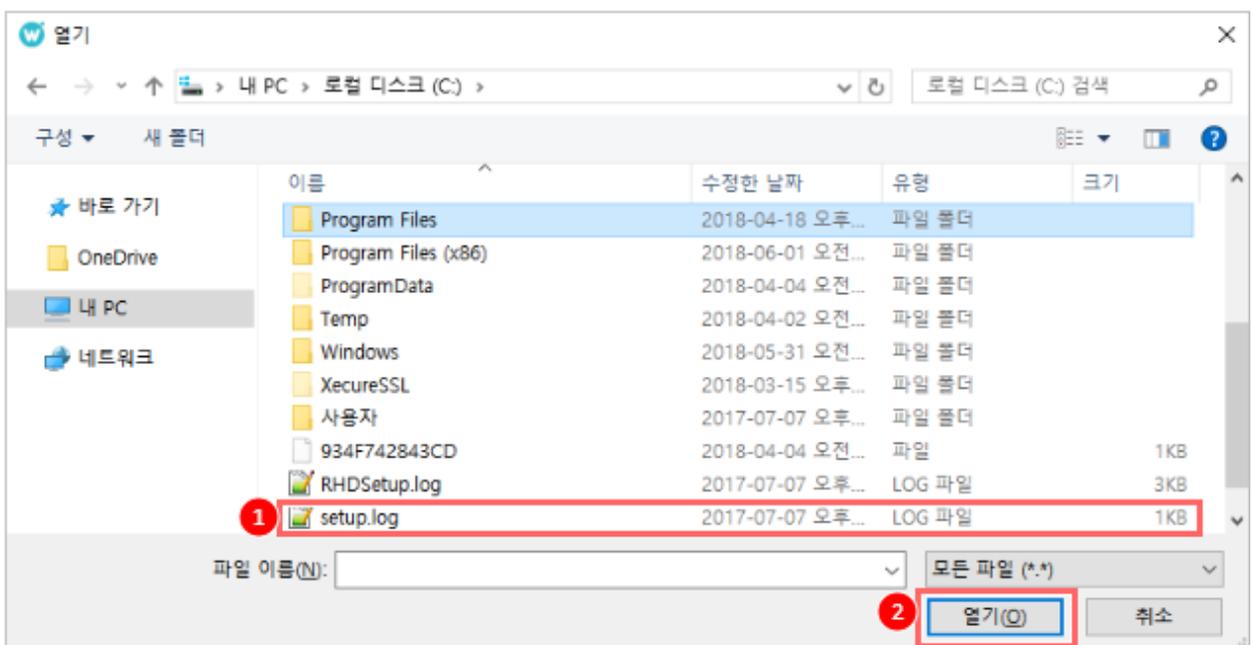
## 리소스 추가하기

프로젝트에서 사용하기 위한 파일을 추가할 수 있습니다. 이 때 파일은 프로젝트 내의 별도 폴더로 관리됩니다.

1. 프로젝트 화면에서 리소스를 마우스 오른쪽 버튼으로 클릭한 후 리소스 추가를 클릭하세요.



2. 추가하려는 파일을 선택하세요.
3. 열기를 클릭하세요.



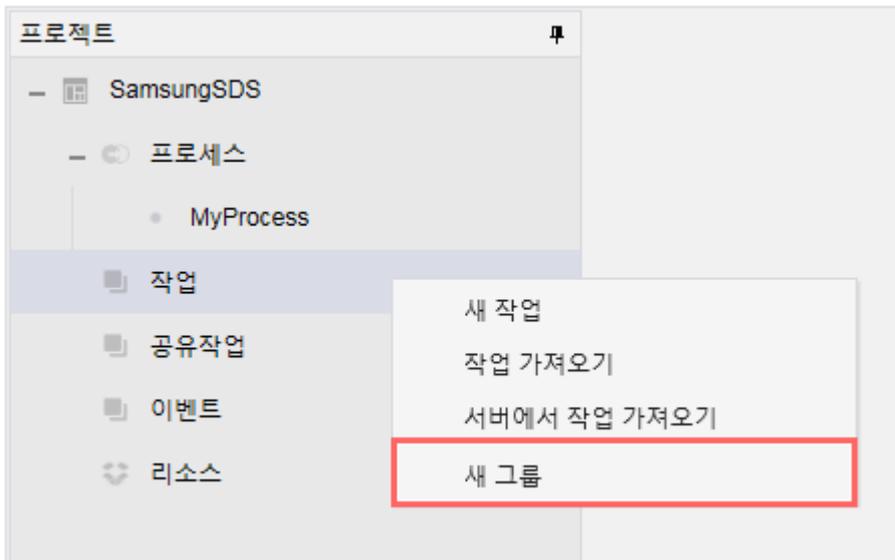
리소스가 추가됩니다.

## 새 그룹 만들기

프로세스나 작업이 많아지는 경우 그룹을 만들어 쉽게 관리할 수 있습니다. 하나의 그룹에는 다른 여러 하위 그룹이 포함될 수 있습니다.

프로세스, 작업의 그룹을 동일한 방법으로 만들 수 있습니다. 여기에서는 작업을 예를 들어 설명합니다.

1. 프로젝트 화면에서 작업을 마우스 오른쪽 버튼으로 클릭한 후 새 그룹을 클릭하세요.



2. "새 그룹" 팝업창에서 새로 만들 그룹의 이름을 입력한 후 생성을 클릭하세요.



새 그룹이 생성됩니다.

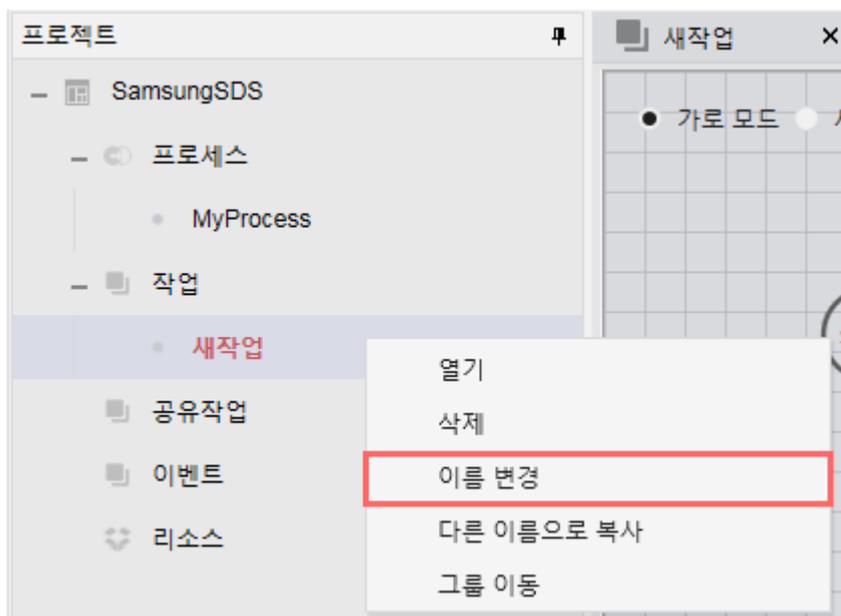
## 프로젝트 편집하기

프로세스, 작업, 이벤트를, 리소스를 동일한 방법으로 편집할 수 있습니다. "프로젝트 편집하기"에서는 작업을 예로 들어 설명합니다.

### 작업 이름 변경하기

작업의 이름을 변경할 수 있습니다.

1. 프로젝트 화면에서 이름을 변경할 작업을 마우스 오른쪽 버튼으로 클릭한 후 **이름 변경**을 클릭하세요.



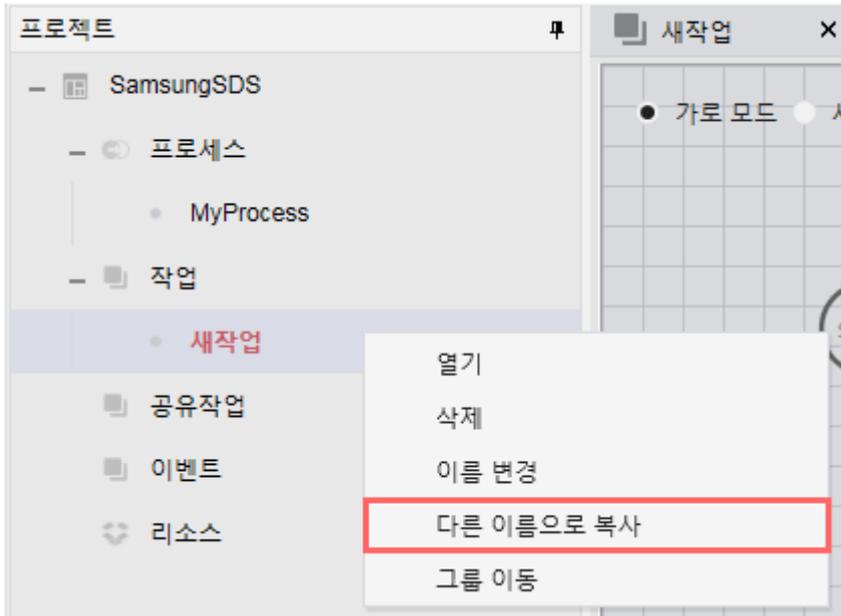
2. "이름 변경" 팝업창에서 변경할 이름을 입력한 후 **확인**을 클릭하세요.

작업의 이름이 변경됩니다.

### 작업 복사하기

비슷한 동작의 작업을 새로 만들 필요 없이 복사하여 사용할 수 있습니다.

1. 프로젝트 화면에서 복사할 작업을 마우스 오른쪽 버튼으로 클릭한 후 **다른 이름으로 복사**를 클릭하세요.



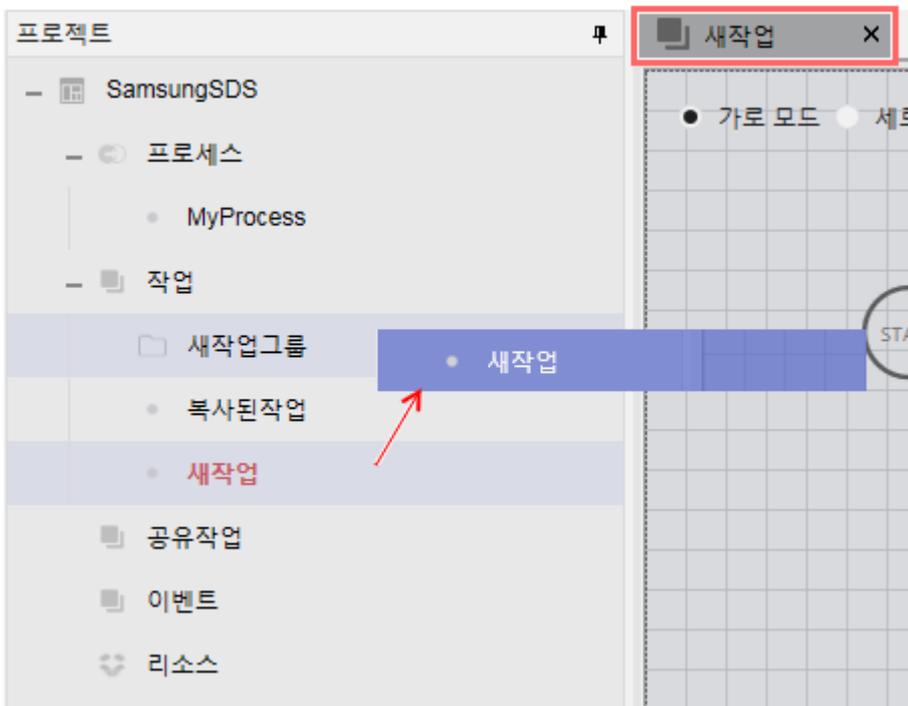
2. "복사" 팝업창에서 복사할 작업의 이름을 입력한 후 **확인**을 클릭하세요.

복사된 작업이 열립니다.

## 작업 이동하기

작업을 그룹 내 또는 그룹 밖으로 이동할 수 있습니다.

위치를 변경할 작업을 선택한 후 마우스로 드래그해 원하는 위치로 이동하세요.

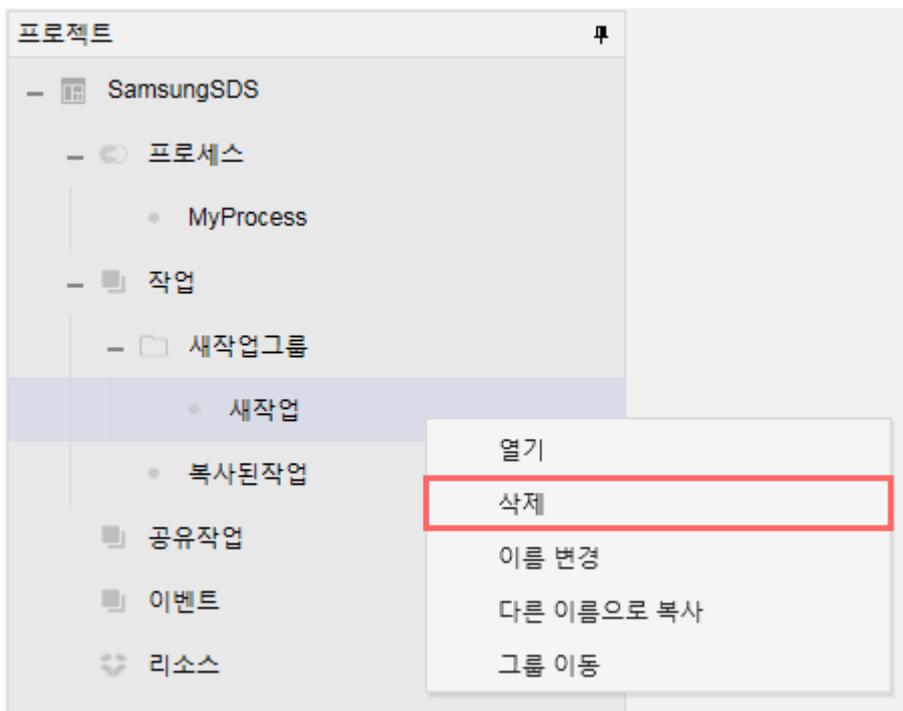


 작업이 열려 있는 경우에는 작업을 닫은 후 이동하세요.

## 작업 삭제하기

더 이상 사용하지 않거나 필요 없는 작업을 삭제할 수 있습니다.

1. 프로젝트 화면에서 삭제할 작업을 마우스 오른쪽 버튼으로 클릭한 후 **삭제**를 클릭하세요.



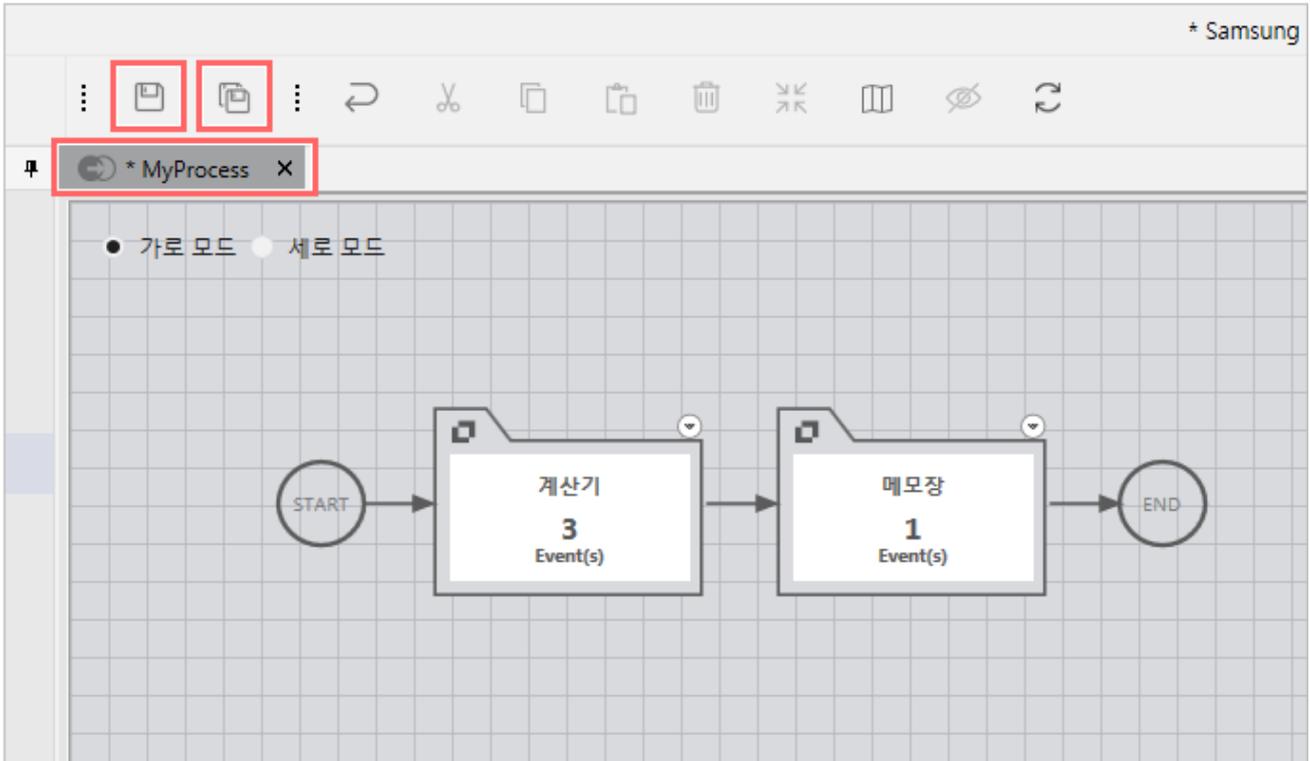
2. "확인" 팝업창에서 **확인**을 클릭하세요.

선택한 작업이 삭제됩니다.

## 3.3 프로젝트 및 프로세스 플로우 저장하기

플로우차트에 변경된 사항이 있는 경우 제목 표시줄의 제목 앞에 \* 표시가 추가됩니다. 현재 활성화된 플로우차트를 저장하려면 툴바에서  클릭하세요. 열려 있는 모든 플로우차트를 저장하려면  클릭하세요.

 > 저장 또는 모두 저장을 클릭해 저장할 수도 있습니다.

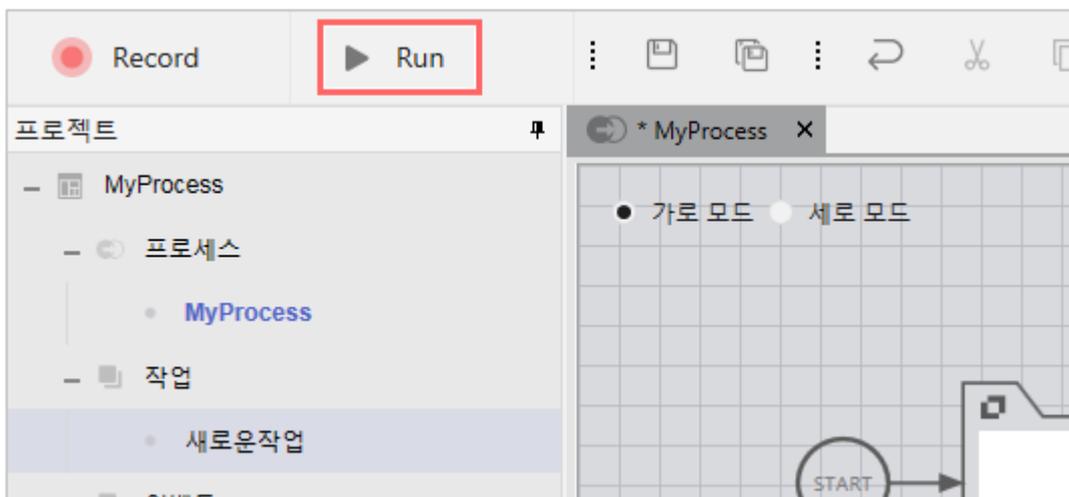


### 3.4 프로젝트 실행하기

활성화되어 있는 프로세스나 작업을 실행하려면 툴바 화면에서 ▶ Run을 클릭하세요.

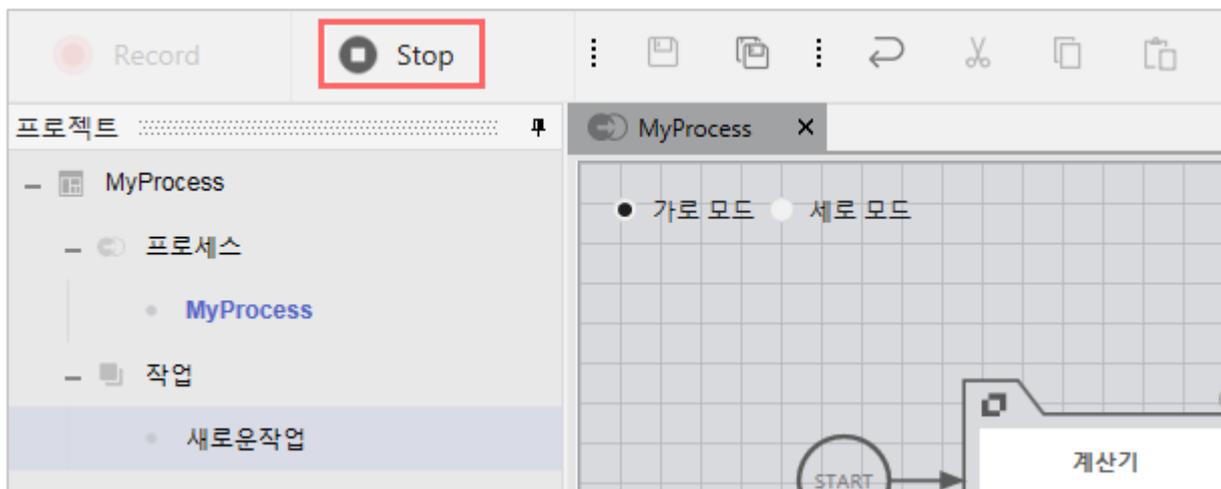
또는 ≡ > 실행하기 > 실행하기를 클릭하세요.

차트가 저장되지 않은 경우 먼저 자동으로 저장됩니다.



**notes** 프로세스나 작업을 처음 수행하는 경우 최소화 여부를 묻는 팝업창이 열립니다. **예**를 클릭해 최소화하면 IPA Designer 프로그램이 숨겨지고, 수행이 완료되면 다시 나타납니다.

수행을 강제로 종료하려면 툴바 화면에서  Stop을 클릭하거나 키보드의 ESC 키를 누르세요.  
또는  > 실행하기 > 중지하기를 클릭하세요.



**notes**  > 설정 > 디자이너 설정에서 실행 추적을 선택한 경우 진행 중인 노드가 자동으로 강조됩니다.

### 3.5 프로세스 플로우 실행하기

프로세스 플로우는 배포한 후 서버를 통해서 실행할 수 있으며, IPA Designer에서 직접 실행할 수 없습니다. 서버 연동에 대한 자세한 내용은 6. 서버 연동하기를 참고하세요.

# 4.

## 프로젝트 디자인하기

사용자는 레코딩 기능을 이용해 동작을 기록하거나, 라이브러리 화면에서 사용하려는 라이브러리를 찾아 액티비티에 추가하여 프로젝트를 디자인할 수 있습니다.

"프로젝트 디자인하기"에서는 아래 내용에 대해 설명합니다.

- 레코딩 기능으로 디자인하기
- 라이브러리로 디자인하기

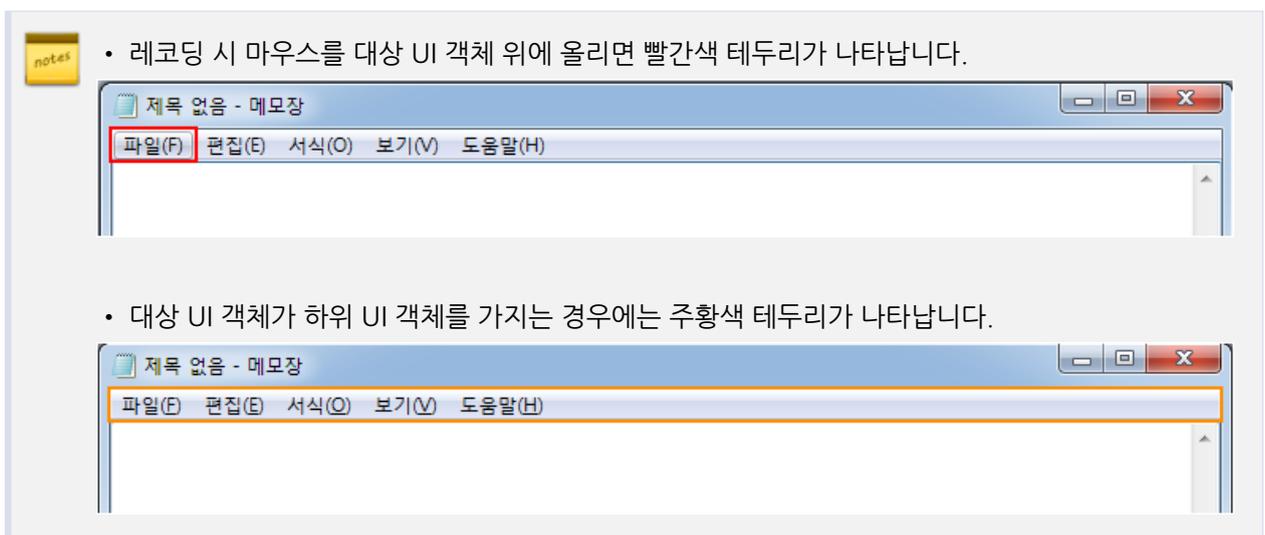
### 4.1 레코딩 기능으로 디자인하기

레코딩 버튼을 눌러 사용자의 동작을 시나리오로 작성합니다. 마우스 클릭, 우클릭, 더블클릭과 키보드 입력 동작을 기록할 수 있습니다.

1.  Record를 클릭해 레코딩을 시작하세요.

또는  > 녹화하기를 클릭하세요.

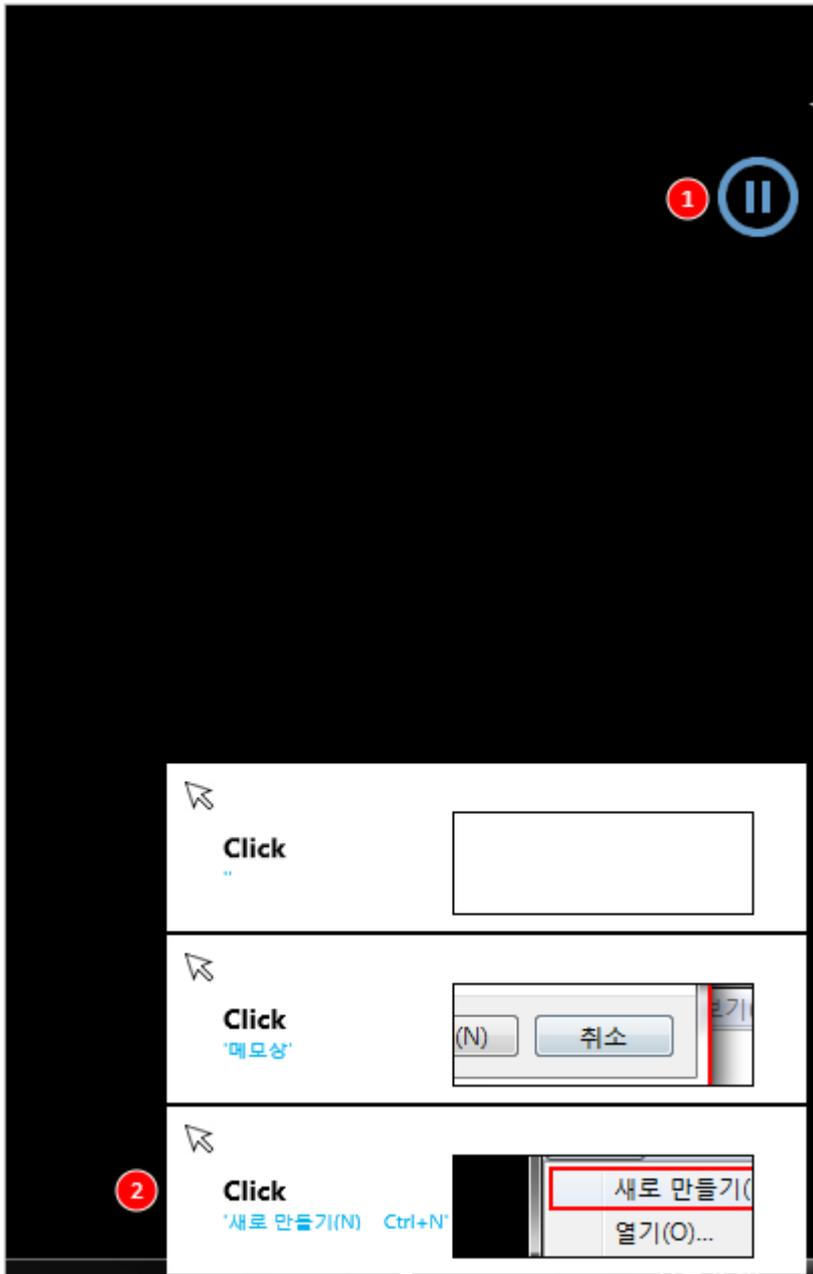
기록이 시작되면 IPA Designer 창은 최소화됩니다.





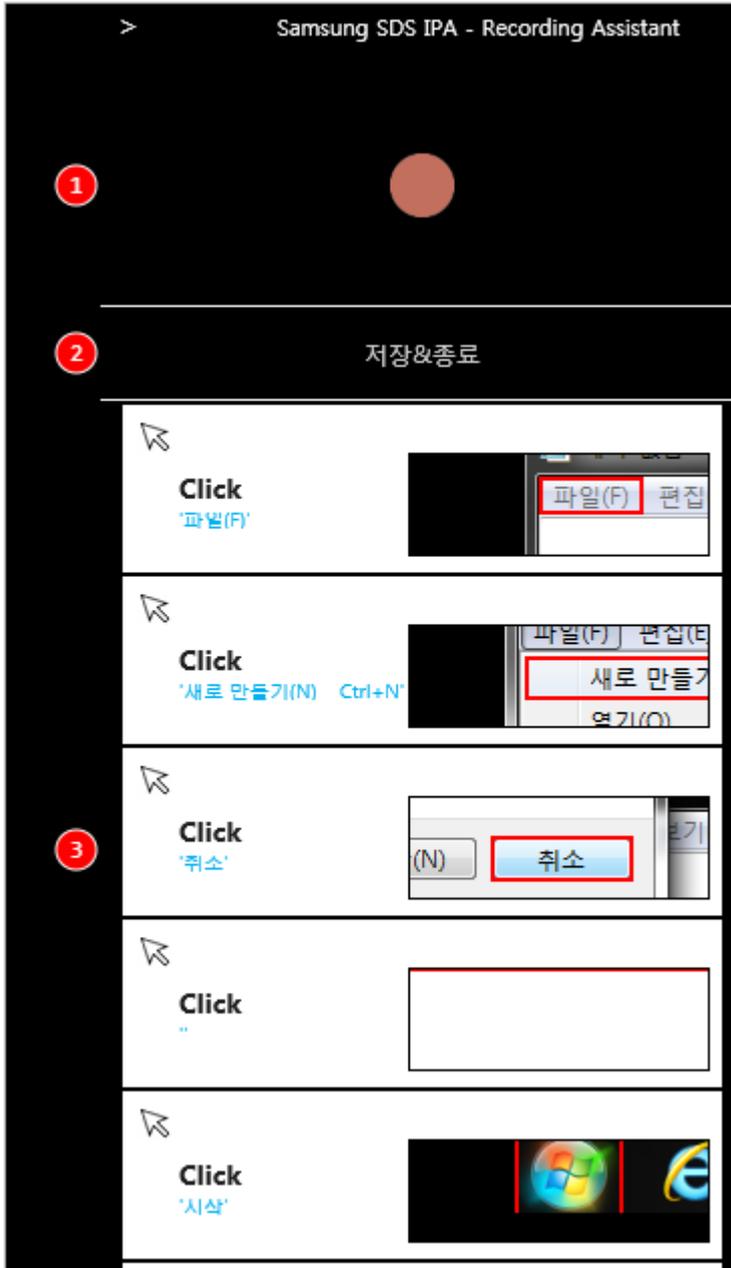
되도록이면 빨간색 테두리가 나타나는 UI 객체를 대상으로 레코딩하세요. 주황색 테두리가 나타나는 UI 객체의 경우 하위 UI 객체를 가지므로 실행 시 검색이 되지 않을 수 있습니다.

2. 프로그램 열기, 파일 열기 등 레코딩할 동작을 처음부터 시작하세요.  
레코딩이 시작되면 아래와 같은 레코딩 화면이 화면 우측에 표시됩니다.



번호	설명
①	레코딩을 일시 중지합니다. 레코딩이 중지되면 레코딩 화면이 열리고 기록된 이벤트 목록이 나타납니다.
②	사용자의 마우스, 키보드 행위가 발생하면 해당 이벤트를 기록하여 화면 우측 하단에 알림창을 표시합니다.

3. 동작 수행이 완료되거나 레코딩을 일시 중지하려면  클릭하세요.  
기록된 이벤트 목록이 나타납니다.



번호	설명
①	레코딩을 다시 시작합니다.
②	레코딩을 종료합니다. 레코딩이 종료되면 IPA Designer 화면이 다시 나타납니다.
③	사용자가 기록한 이벤트 목록을 보여줍니다.

4. 레코딩을 완료하려면 이벤트 목록 화면에서 **저장&종료**를 클릭하세요.

레코딩한 동작이 플로우차트에 추가됩니다.



64 비트 버전의 소프트웨어를 사용할 경우, 반드시 64 비트 버전의 드라이버를 설치하세요. 32 비트 버전의 드라이버는 해당 소프트웨어에 호환되지 않습니다.

## 4.2 라이브러리로 디자인하기

라이브러리를 이용해 프로젝트를 디자인할 수 있습니다.

### 액티비티 추가하기

액티비티는 플로우차트를 구성하는 동작을 의미합니다.

사용자는 원하는 라이브러리를 디자인 화면의 플로우차트로 드래그해 액티비티를 추가할 수 있습니다.

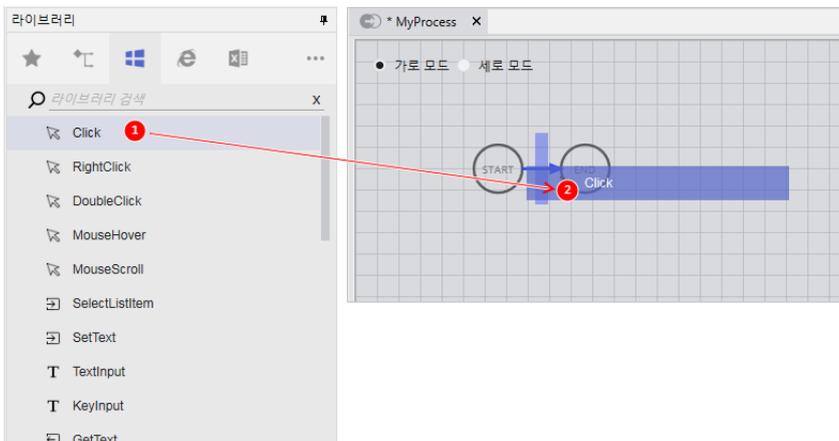


If, For 등 FlowControl 항목의 일부 라이브러리는 링크에만 추가 가능하며, 액티비티 그룹 또는 액티비티 사이에는 추가할 수 없습니다.

### 링크에 액티비티 추가하기

링크는 액티비티와 액티비티를 연결하는 부분입니다. 디자인 화면에서는 각 액티비티 사이의 화살표로 표시됩니다.

라이브러리 화면에서 추가할 라이브러리를 선택한 후 원하는 링크로 드래그하세요. 파란색 사각형이 나타났을 때 마우스를 드롭하면 액티비티가 추가됩니다.

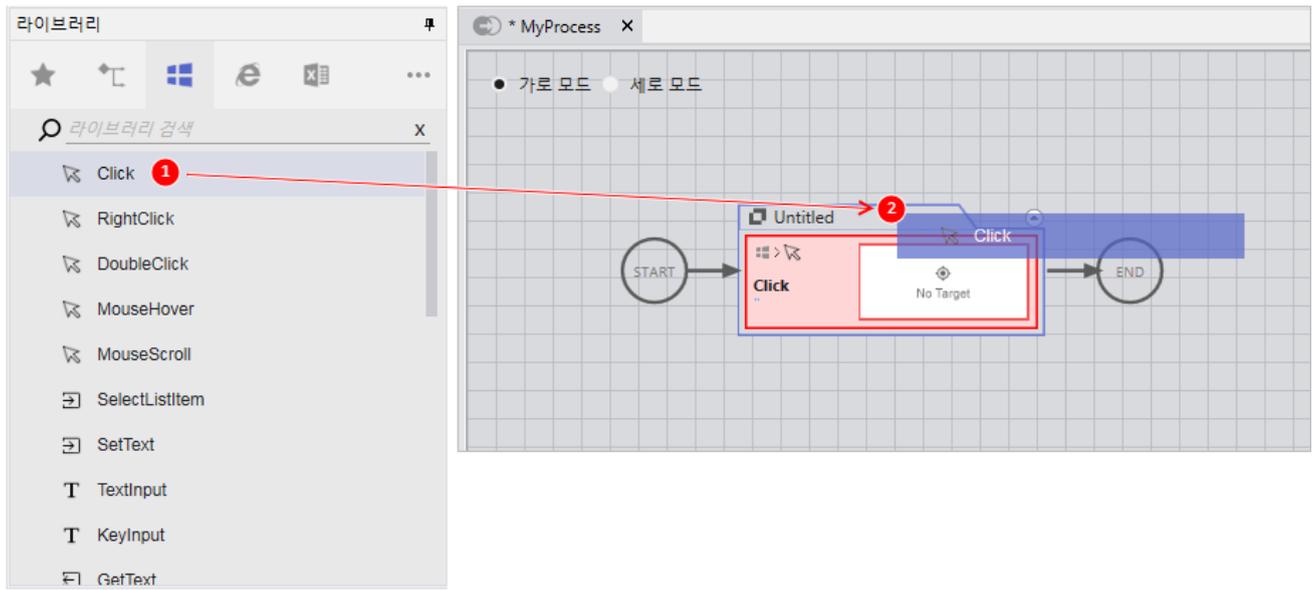


## 액티비티 그룹에 액티비티 추가하기

액티비티 그룹은 여러 개의 액티비티를 하나의 동작으로 묶는 단위입니다.

라이브러리 화면에서 추가할 라이브러리를 선택한 후 원하는 액티비티 그룹의 제목 부분으로 드래그하세요.

액티비티 그룹의 색상이 파란색으로 변경되었을 때 마우스를 드롭하면 액티비티 그룹의 마지막 위치에 액티비티가 추가됩니다.



## 액티비티 사이에 액티비티 추가하기

액티비티 그룹 내의 액티비티 사이에 액티비티를 추가할 수 있습니다.

라이브러리 화면에서 추가할 라이브러리를 선택한 후 원하는 액티비티 그룹의 액티비티 사이로 드래그하세요.

파란색 사각형이 나타났을 때 마우스를 드롭하면 액티비티 사이에 새로운 액티비티가 추가됩니다.

## 액티비티 편집하기

### 액티비티 삭제하기

삭제할 액티비티를 선택한 후 툴바에서  클릭하세요. 또는 키보드의 Delete 키를 누르세요.

### 액티비티 복사하기 1

툴바나 키보드의 단축 키를 이용해 액티비티를 복사할 수 있습니다.

1. 복사할 액티비티를 선택하세요.

2. 툴바에서  클릭하거나 키보드의 Ctrl+C 키를 누르세요.
3. 붙여넣을 위치를 선택하세요.
4. 툴바에서  클릭하거나 키보드의 Ctrl+V 키를 누르세요.

## 액티비티 복사하기 2

마우스 드래그 동작으로 액티비티를 복사할 수 있습니다.

1. 복사할 액티비티를 선택하세요.
2. 키보드의 Ctrl 키를 누른 채 붙여넣을 위치로 드래그하세요.

## 액티비티 이동하기 1

툴바나 키보드의 단축 키를 이용해 액티비티를 이동할 수 있습니다.

1. 이동할 액티비티를 선택하세요.
2. 툴바에서  클릭하거나 키보드의 Ctrl+X 키를 누르세요.
3. 붙여넣을 위치를 선택하세요.
4. 툴바에서  클릭하거나 키보드의 Ctrl+V 키를 누르세요.

## 액티비티 이동하기 2

마우스 드래그 동작으로 액티비티를 이동할 수 있습니다.

이동할 액티비티를 선택한 후 붙여넣을 위치로 드래그하세요.

## 작업 추가하기

작업을 액티비티처럼 다른 작업에 추가할 수 있습니다.

추가할 작업을 선택한 후 원하는 링크로 드래그하세요. 파란색 사각형이 나타났을 때 마우스를 드롭하면 작업이 추가됩니다.



- 작업은 링크에만 추가할 수 있으며, 액티비티 그룹에 포함할 수 없습니다.
- 자기 자신을 작업으로 추가할 수 없습니다.
- 작업 삭제, 복사, 이동은 액티비티와 동일한 방식으로 할 수 있습니다.

## 여러 액티비티 선택하기

여러 액티비티를 함께 선택해 한 번에 복사, 이동, 삭제할 수 있습니다.

마우스로 선택할 노드를 드래그하세요. 또는 Ctrl 키를 누른 상태로 여러 개의 노드를 선택하세요.



- 키보드의 Shift 키를 누른 상태로 노드를 클릭하면 여러 액티비티를 연속적으로 선택할 수 있습니다. 단, 동일한 액티비티 그룹 내에서만 선택할 수 있습니다.
- 키보드의 Ctrl+A 키를 누르면 현재 선택된 액티비티가 소속된 액티비티 그룹을 선택하거나 플로우차트 내의 모든 노드를 한 번에 선택할 수 있습니다.

## 선택 반전하기

선택된 액티비티를 제외한 모든 액티비티를 한 번에 선택할 수 있습니다.

툴바에서 클릭하세요.

## 비활성화하기

자동화 수행 시 사용자는 특정한 액티비티를 제외할 수 있습니다.

수행하지 않을 액티비티를 하나 이상 선택한 후 툴바에서 클릭하세요.

## 액티비티 그룹 접기

액티비티가 많이 추가되는 경우 액티비티 그룹을 접어 간략하게 표시할 수 있습니다. 액티비티 그룹을 접으면 제목과 포함된 액티비티의 개수가 표시됩니다.

액티비티 그룹 우측 상단의 클릭하세요.

여러 개의 액티비티 그룹을 접으려면 1개 이상의 액티비티 그룹을 선택한 후 툴바에서 클릭하세요.

## 라이브러리 알아보기

라이브러리는 사용할 수 있는 대상 앱이나 기능에 따라 앱, 웹, 엑셀, SAP, 이미지 인식, 시스템, FlowControl, KnoxPortal 항목으로 구분되어 있습니다. 라이브러리 화면에서 사용할 항목을 선택한 후 원하는 라이브러리의 이름을 입력해 검색하세요.

## 앱, 웹 라이브러리

**App, Web** 라이브러리에서 제공하는 기능을 이용해 앱 또는 웹 화면을 제어할 수 있습니다. Windows가 제공하는 UI 자동화 기술로 화면에 있는 UI 객체를 식별한 후, 사용자가 지정한 대상 UI 객체를 찾아 지정한 동작을 수행시킵니다.

앱, 웹 라이브러리에서 사용할 수 있는 기능은 다음과 같습니다.

- UI 객체 인식 관련 기능(Find, FindElement, RunAction, WaitAppear, WaitDisappear, IsExist, IsVisible 등)
- 마우스 동작 기능(MouseEvent, Click, RightClick, DoubleClick, MouseHover, SelectListItem 등)
- 키보드 동작 기능(KeyboardEvent, SetText, TextInput, KeyInput, GetText 등)
- 실행 및 종료 기능(OpenApp, ActivateApp, CloseApp, CloseAll, OpenWindow, CloseWindow, OpenBrowser, CloseBrowser 등)
- 메일 기능(POP3,SMTP)(SendMail, ReceiveMail, DownloadMailAttachment 등)
- 기타 기능(GetHierarchyInfo, SendRequest, GetGridItems, GetListItemIndex, GetGridItemIndex 등)
- WebDriver 이용 기능(Selenium, Headless) - beta



대상 앱이 일반적인 개발 방식을 따르지 않는 경우, **App** 라이브러리가 의도대로 동작하지 않을 수 있습니다.

### Find(UI 객체 찾기)

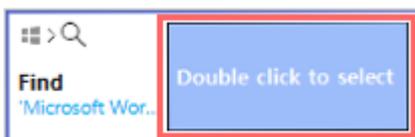
사용자가 지정한 대상 UI 객체를 찾습니다.

UI 객체 관련 앱 라이브러리의 기본이 되는 라이브러리입니다. 예를 들어, **Click** 라이브러리는 **Find** 라이브러리로 검색된 객체를 대상으로 **MouseEvent** 라이브러리의 마우스 이벤트를 발생시킵니다.



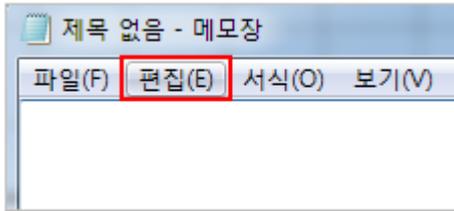
대상 UI 객체를 검색할 때에는 먼저 대상 UI 객체 지정 시 기록된 마우스의 위치에 있는 UI 객체가 대상 UI 객체의 정보와 일치하는지 확인합니다. 일치하지 않을 경우 대상 UI 객체의 상위 경로를 따라 내려가는 방식으로 찾습니다. 이 방식으로도 대상 UI 객체를 찾지 못할 경우, 대상 앱 내의 모든 UI 객체 정보를 수집한 후 대상 UI 객체의 정보와 비교하여 대상 UI 객체를 찾습니다.

1. 노드에서 **No Target > Double click to select**를 더블클릭하세요.



2. UI 객체에 마우스 포인터를 올리세요.

빨간색 테두리가 나타납니다.



3. 대상 UI 객체로 지정할 UI 객체에 빨간색 테두리가 나타나면 클릭하세요.

해당 UI 객체가 대상 UI 객체로 지정됩니다.

 • 하위 UI 객체를 가지는 경우 주황색 테두리가 나타납니다. 단, 대상을 안정적으로 검색하려면 하위 UI 객체를 가지지 않는 UI 객체를 지정하세요. 하위 UI 객체를 가질 경우 레코딩 시점과 자동화 수행 시점의 차이 발생 시 대상을 검색하기 어려울 수 있습니다. 불가피하게 하위 UI 객체를 지정해야 하는 경우에는 자동화 수행 시 화면 구성이 변경될 경우를 고려하여 신중하게 지정하세요.

• 다른 UI 객체를 누른 후 대상 UI 객체를 선택해야 하는 경우, 또는 화면을 이동한 후 대상 UI 객체를 지정해야 하는 경우에는 키보드의 Ctrl 키를 눌러 지정을 일시 정지하거나 다시 시작할 수 있습니다.

• 대상 UI 객체 지정 시 마우스를 누른 시점에 테두리가 표시된 객체가 검색 대상이 되며, 마우스를 놓은 시점의 위치가 대상 UI 객체의 좌측 상단을 기준 상대좌표로 기록됩니다. 이를 활용하면 대상 UI 객체의 정보가 불분명한 경우 주변의 특정 가능한 객체를 지정한 후, 상대좌표의 위치로 마우스를 클릭할 수 있습니다.

• UI 객체 테두리로 인해 팝업 화면이 사라지거나 대상 앱이 오동작하는 경우 키보드의 Alt 키를 한 번 누르면 객체 테두리가 표시되지 않습니다. 한 번 더 누르면 테두리를 다시 표시해 줍니다.

• 마우스를 위치시킨 UI 객체가 아닌 다른 위치에 테두리가 그려지는 경우, 키보드의 Shift 키를 한 번 눌러 **Search Around** 옵션을 활성화한 후 재시도하세요. 자세한 내용은 [그림 4-1 Find 속성 화면의 Search Around 항목 설명](#)을 참고하세요.

대상 UI 객체를 확인하거나 변경, 추가하려면 노드에서 지정한 대상 UI 객체의 캡처 화면을 더블클릭한 후 "Find" 팝업창에서 **실행** 또는 **재지정**, **추가** 를 클릭하세요.

- **실행**: 지정된 대상 UI 객체를 확인할 수 있습니다. 대상 UI 객체에 빨간색 테두리가 3초간 깜박거린 후 사라집니다.

 속성 창의 객체 정보에 변수를 입력한 경우에는 검사를 수행할 수 없습니다. 이 경우, 노드를 마우스 오른쪽 버튼으로 클릭한 후 **Run step**을 클릭하면 해당 노드만 실행됩니다.

- **재지정**: 대상 UI 객체를 재지정할 수 있습니다.
- **추가**: 대상 UI 객체를 추가할 수 있습니다. 추가하게 되면 **속성 화면**에서 추가된 대상 UI 객체들을 선택할 수 있습니다.

그림 4-1 Find 속성 화면

### Activity

1 Index \_\_\_\_\_

2 All or First  True  False

3 **Check Name**  True  False

Check Value  True  False

Check Size  True  False

### Target

4 Application Q-Dir \_\_\_\_\_

5 Title 'Q-Dir 6.46' \_\_\_\_\_

6 Type Edit \_\_\_\_\_

7 Name 'TestProject2' \_\_\_\_\_

8 Value "" \_\_\_\_\_

### More ▲

9 Timeout 1 \_\_\_\_\_

10 Search Bounds \_\_\_\_\_

11 Object Path 'Window|ATL:0000000140168BFC' \_\_\_\_\_

12 Image 

tProject
<b>tProject2</b>
이 프로젝트

13 Bounds 

X	18	Y	144
W	355	H	19

14 Check Title  True  False

15 Focus Before  True  False

16 Draw Bounds  True  False

17 Ignore Error  True  False

18 Check Path  True  False



번호	설명
1	동일한 정보를 가진 UI 객체 목록에서 대상 UI 객체를 식별하기 위한 Index를 입력합니다. 대상 UI 객체 주변에 보이지 않는 투명한 객체가 있거나, 주변 UI 객체와 영역이 겹치는 형태인 경우에는 대상 UI 객체가 아니라 주변 객체가 인식되는 경우가 있습니다. 이 경우 Search Around 옵션을 사용하여 주변 객체의 영역을 추가로 검사하도록 설정할 수 있습니다.
2	True 인 경우 조건에 맞는 모든 타겟 정보를 배열로 반환하며 False인 경우 처음 발견된 1개 타겟만 반환합니다.
3	UI 객체 비교 시 Name, Value, Size 속성에 대한 검사 여부를 설정합니다.
4	대상 UI 객체 앱의 이름이 나타납니다.
5	대상 UI 객체 앱의 타이틀이 나타납니다. 타이틀은 앱의 상태에 따라 변경될 수 있으므로 UI 객체 비교 시 필수로 일치해야 하는 사항은 아니지만, 동일한 앱이 여러 개 발견되는 경우에는 타이틀이 일치하는 UI 객체가 선택됩니다.
6	대상 UI 객체의 클래스 타입이 나타납니다.
7	대상 UI 객체의 Name 정보가 나타납니다. 대부분 화면에 표시되는 문자열이 나타나지만, 앱의 구현 방식에 따라 값이 있어도 나타나지 않을 수 있습니다. Name, Value, Title의 경우 'ww:' 키워드를 이용하여 정규식을 사용하거나 this.SearchText와 같이 변수를 입력할 수 있습니다. 단, 변수가 입력된 경우 노드의 검사 기능은 사용할 수 없습니다.
8	대상 UI 객체의 Value 정보가 나타납니다. 앱의 구현 방식에 따라 화면에 표시되는 문자열이 Name 항목 대신 해당 항목에 나타나거나, 값이 있어도 나타나지 않을 수 있습니다.
9	대상 UI 객체를 찾는 시간을 초 단위로 입력합니다.
10	전체 영역이 아닌 주어진 영역 내에서 검색합니다.
11	사용자가 지정한 대상 UI 객체 경로 정보를 작은따옴표 안에 입력합니다. '@//Button' 과 같이 @를 입력하게 되면 XPath 구문으로 간주하며, 지정한 대상App 의 전체 UI 구조를 XML 형태로 생성하여 XPath 구문으로 대상 UI 객체 정보를 검색합니다. XML 구조는 GetHierarchyInfo 를 참고하세요.
12	사용자가 지정한 대상 UI 객체의 이미지가 나타납니다.
13	사용자가 지정한 대상 UI 객체의 위치 및 크기가 앱의 좌상단 기준 상대 좌표로 나타납니다.
14	UI 객체 비교 시 Title 속성에 대한 검사 여부를 설정합니다.
15	이벤트를 발생시키기 전에 대상 UI 객체에 포커스를 설정할 수 있습니다. 대상 UI 객체에 포커스를 설정할 수 없는 경우에는 대상 앱의 메인 화면에 포커스를 설정합니다. 다른 앱에 의해 대상 UI 객체가 가려진 경우 대상 앱을 앞으로 가져옵니다. 대상 앱이 여러 개 실행되어 있는 경우, 가장 앞에 위치한 앱에서 먼저 검사하고 대상 UI 객체를 찾지 못하면 뒤에 위치한 앱을 앞으로 가져온 후 다시 검사합니다.
16	실행 시 UI 객체를 찾았을 때 박스가 나타나도록 설정합니다.
17	오류가 발생해도 프로세스나 작업을 계속 진행하도록 설정합니다. IsExist 라이브러리를 이용해 존재 여부에 따라 클릭하거나 무시하는 형태의 스크립트를 작성하는 대신 Ignore

번호	설명
	Error 옵션을 활용할 수 있습니다.
18	대상 UI 객체를 검색할 때 화면 내 UI 객체를 비교하는 대신 상위 UI 객체를 검사하도록 설정합니다. 이 경우 검색 속도는 향상되지만 Name, Value, Size, Title 등의 정보가 다른 UI 객체와 차별되지 않을 경우에는 대상 UI 객체를 정확하게 찾지 못할 수 있습니다.
19	정확한 UI 객체 식별을 위해 주변 UI 객체를 추가로 확인할지 설정합니다.
20	Check Size가 True로 설정되어 있을 때 허용하는 오차의 범위를 입력합니다.
21	사용자가 추가한 대상 UI 객체들 중 원하는 UI 객체를 선택, 삭제합니다.

### MouseScroll(마우스 스크롤하기)

그림 4-2 MouseScroll 속성 화면

The screenshot shows the 'Activity' inspector for 'MouseScroll'. The properties are as follows:

- Index: (empty text field)
- Wheel Value: 1 (indicated by a red circle 1)
- Wheel Interval: 500 (indicated by a red circle 2)
- Check Name: True (selected)
- Check Value: False (selected)
- Check Size: False (selected)

번호	설명
1	스크롤을 내리거나 올릴 횟수입니다. (양수는 아래 방향으로, 음수는 위 방향으로 스크롤 처리)
2	스크롤 수행 간격입니다.

### FindElement(하위 UI 객체 찾기)

Input 항목에 입력한 객체의 하위에서 대상 UI 객체를 검색합니다.

### RunAction(이벤트 발생시키기)

Input 항목에 입력한 대상 UI 객체에 클릭 등의 이벤트를 발생시키거나 주변 객체를 검색할 수 있습니다.

## WaitAppear(나타날 때까지 기다리기)

대상 UI 객체가 화면에 나타날 때까지 기다립니다.  
화면 전환 시 유용하게 사용할 수 있습니다.



오랜 시간동안 기다려야 할 경우 **WaitAppear** 라이브러리 이전에 최소 대기 시간 만큼의 Delay를 추가하는 것을 권장합니다. **WaitAppear** 라이브러리를 사용할 경우, 반복적으로 대상 앱의 UI 객체 정보를 검색하므로 대상 앱에 따라 동작이 느려지거나 오동작할 수 있습니다.

그림 4-3 WaitAppear 속성 화면

**Output**

1 Result RESULT

**Activity**

Timeout 10

Index \_\_\_\_\_

Check Name  True  False

Check Value  True  False

Check Size  True  False

**Target**

Application \_\_\_\_\_

Title "

Type \_\_\_\_\_

Name "

Value "

**More** ▲

Search Bounds \_\_\_\_\_

Object Path "

Check Title  True  False

Focus Before  True  False

Draw Bounds  True  False

2 Ignore Error  True  False

Check Path  True  False

Search Around  True  False

번호	설명
1	대상 UI 객체의 발견 여부를 True 또는 False로 반환합니다.
2	True로 설정할 경우 대상 UI 객체가 발견되지 않으면 <b>Result</b> 에 False를 반환합니다. False로 설정할 경우 대상 UI 객체가 발견되지 않으면 Exception을 발생시킵니다.

## WaitDisappear(사라질 때까지 기다리기)

대상 UI 객체가 화면에서 사라질 때까지 기다립니다.  
화면 전환 시 유용하게 사용할 수 있습니다.

그림 4-4 WaitDisappear 속성 화면

The screenshot shows the configuration interface for the WaitDisappear action. It includes the following fields and settings:

- Output:** 1 Result (RESULT)
- Activity:** Timeout (10)
- Check Name:** True
- Check Value:** False
- Check Size:** False
- Target:** Application, Title, Type, Name, Value (all empty)
- More:**
  - Search Bounds (empty)
  - Object Path (empty)
  - Check Title: False
  - Focus Before: True
  - Draw Bounds: True
  - Ignore Error: 2 True
  - Check Path: False
  - Search Around: False

번호	설명
①	대상 UI 객체가 사라지면 True를, 사라지지 않으면 False를 반환합니다.
②	True로 설정할 경우 대상 UI 객체가 사라지지 않으면 <b>Result</b> 에 False를 반환합니다. False로 설정할 경우 대상 UI 객체가 사라지지 않으면 Exception을 발생시킵니다.

## IsExist(존재 여부 검사하기)

대상 UI 객체가 화면에 있는지 검사합니다.

그림 4-5 IsExist 속성 화면

The screenshot shows the configuration interface for the IsExist property. It is divided into several sections: Output, Activity, Target, and More. The 'Output' section has a 'Result' dropdown set to 'RESULT'. The 'Activity' section includes 'Index', 'Check Name', 'Check Value', and 'Check Size', each with 'True' and 'False' radio buttons. The 'Target' section includes 'Application', 'Title', 'Type', 'Name', and 'Value', each with a text input field. The 'More' section includes 'Timeout' (with a red box around the value '1'), 'Search Bounds', 'Object Path', 'Check Title', 'Focus Before', 'Draw Bounds', 'Check Path', and 'Search Around', each with 'True' and 'False' radio buttons.

**Timeout** 항목에 입력한 시간 내에 화면에서 대상 UI 객체를 찾을 경우 True를, 찾지 못할 경우 False를 반환합니다.



스크롤 아래에 있거나 숨겨진 객체 등 화면에 보이지 않더라도 객체 정보가 존재하면 True로 반환합니다. 이 경우, 객체의 Bounds 정보는 0으로 표시될 수 있습니다.

## IsVisible(화면 표시 여부 검사하기)

대상 UI 객체가 화면에 보이는지 검사합니다.

그림 4-6 IsVisible 속성 화면

The screenshot shows the configuration interface for the **IsVisible** property. It is divided into several sections:

- Output:** A dropdown menu labeled 'Result' is set to 'RESULT'.
- Activity:** Contains fields for 'Index', 'Check Name', 'Check Value', and 'Check Size', each with a 'True' and 'False' radio button. 'Check Name', 'Check Value', and 'Check Size' are currently set to 'True'.
- Target:** Contains fields for 'Application', 'Title', 'Type', 'Name', and 'Value', all of which are currently empty (represented by double quotes).
- More:** Contains several additional fields:
  - Timeout:** A text input field containing the value '1', which is highlighted with a red rectangular box.
  - Search Bounds:** An empty text input field.
  - Object Path:** An empty text input field.
  - Check Title:** Radio buttons for 'True' and 'False'.
  - Focus Before:** Radio buttons for 'True' and 'False'.
  - Draw Bounds:** Radio buttons for 'True' and 'False'.
  - Check Path:** Radio buttons for 'True' and 'False'.
  - Search Around:** Radio buttons for 'True' and 'False'.

**IsExist** 라이브러리에 추가로 Bounds 정보를 확인합니다. Bounds 정보가 존재하는 경우 True를 반환하며, 다른 화면에 의해 가려진 경우에도 Bounds 정보가 존재하면 동일하게 True를 반환합니다.

### MouseEvent(마우스 이벤트 발생시키기)

지정한 위치에 클릭 등의 마우스 이벤트를 발생시킵니다.

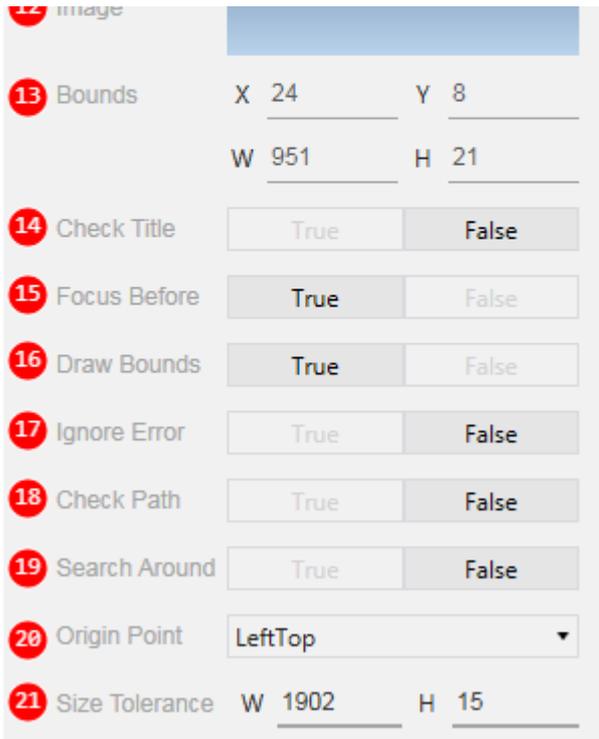
### Click(마우스 클릭하기)

대상 UI 객체를 마우스 왼쪽 버튼으로 클릭합니다.

그림 4-7 Click 속성 화면

The screenshot shows the configuration interface for a Click activity. It is organized into three main sections:

- Activity:**
  - 1 Index: [Empty text field]
  - 2 Click Point: X 303, Y 5
  - 3 Check Name:  True,  False
  - Check Value:  True,  False
  - Check Size:  True,  False
- Target:**
  - 4 Application: notepad
  - 5 Title: '제목 없음 - 메모장'
  - 6 Type: TitleBar
  - 7 Name: ''
  - 8 Value: '제목 없음 - 메모장'
- More:**
  - 9 Timeout: 10
  - 10 Search Bounds: [Empty text field]
  - 11 Object Path: 'Window|Notepad|Win32/TitleBa'
  - 12 Image: [Image preview]



번호	설명
1	동일한 정보를 가진 UI 객체 목록에서 대상 UI 객체를 식별하기 위한 Index를 입력합니다. 대상 UI 객체 주변에 보이지 않는 투명한 객체가 있거나, 주변 UI 객체와 영역이 겹치는 형태인 경우에는 대상 UI 객체가 아니라 주변 객체가 인식되는 경우가 있습니다. 이 경우 Search Around 옵션을 사용하여 주변 객체의 영역을 추가로 검사하도록 설정할 수 있습니다.
2	UI 객체를 찾은 후 클릭할 위치를 UI 객체의 왼쪽 상단 기준 좌표 값으로 입력합니다. 마우스로 직접 지정한 경우에는 자동으로 입력됩니다.
3	UI 객체 비교 시 Name, Value, Size 속성에 대한 검사 여부를 설정합니다.
4	대상 UI 객체 앱의 이름이 나타납니다.
5	대상 UI 객체 앱의 타이틀이 나타납니다. 타이틀은 앱의 상태에 따라 변경될 수 있으므로 UI 객체 비교 시 필수로 일치해야 하는 사항은 아니지만, 동일한 앱이 여러 개 발견되는 경우에는 타이틀이 일치하는 UI 객체가 선택됩니다.
6	대상 UI 객체의 클래스 타입이 나타납니다.
7	대상 UI 객체의 Name 정보가 나타납니다. 대부분 화면에 표시되는 문자열이 나타나지만, 앱의 구현 방식에 따라 값이 있어도 나타나지 않을 수 있습니다. Name, Value, Title의 경우 'www:' 키워드를 이용하여 정규식을 사용하거나 변수를 입력할 수 있습니다. 단, 변수가 입력된 경우 노드의 검사 기능은 사용할 수 없습니다.
8	대상 UI 객체의 Value 정보가 나타납니다. 앱의 구현 방식에 따라 화면에 표시되는 문자열이 Name 항목 대신 해당 항목에 나타나거나, 값이 있어도 나타나지 않을 수 있습니다.
9	대상 UI 객체를 찾는 시간을 초 단위로 입력합니다.
10	전체 영역이 아닌 주어진 영역 내에서 검색합니다.
11	사용자가 지정한 대상 UI 객체 경로 정보를 작은따옴표 안에 입력합니다.
12	사용자가 지정한 대상 UI 객체의 이미지가 나타납니다.

번호	설명
13	사용자가 지정한 대상 UI 객체의 위치 및 크기가 앱의 좌상단 기준 상대 좌표로 나타납니다.
14	UI 객체 비교 시 Title 속성에 대한 검사 여부를 설정합니다.
15	이벤트를 발생시키기 전에 대상 UI 객체에 포커스를 설정할 수 있습니다. 대상 UI 객체에 포커스를 설정할 수 없는 경우에는 대상 앱의 메인 화면에 포커스를 설정합니다. 다른 앱에 의해 대상 UI 객체가 가려진 경우 대상 앱을 앞으로 가져옵니다. 대상 앱이 여러 개가 실행되어 있는 경우, 가장 앞에 위치한 앱에서 먼저 검사하고 대상 UI 객체를 찾지 못하면 뒤에 위치한 앱을 앞으로 가져온 후 다시 검사합니다.
16	실행 시 UI 객체를 찾았을 때 박스가 나타나도록 설정합니다.
17	오류가 발생해도 프로세스나 작업을 계속 진행하도록 설정합니다. IsExist 라이브러리를 이용해 존재 여부에 따라 클릭하거나 무시하는 형태의 스크립트를 작성하는 대신 Ignore Error 옵션을 활용할 수 있습니다.
18	대상 UI 객체를 검색할 때 화면 내 UI 객체를 비교하는 대신 상위 UI 객체를 검사하도록 설정합니다. 이 경우 속도가 향상되지만 Name, Value, Size, Title 등의 정보가 다른 UI 객체와 차별되지 않을 경우에는 대상 UI 객체를 정확하게 찾지 못할 수 있습니다.
19	정확한 UI 객체 식별을 위해 주변 UI 객체를 추가로 확인할지 설정합니다.
20	상대 좌표 계산 시 사용되는 원점(객체의 4개 모서리)을 설정합니다.
21	Check Size가 True로 설정되어 있을 때 허용하는 오차의 범위를 입력합니다.

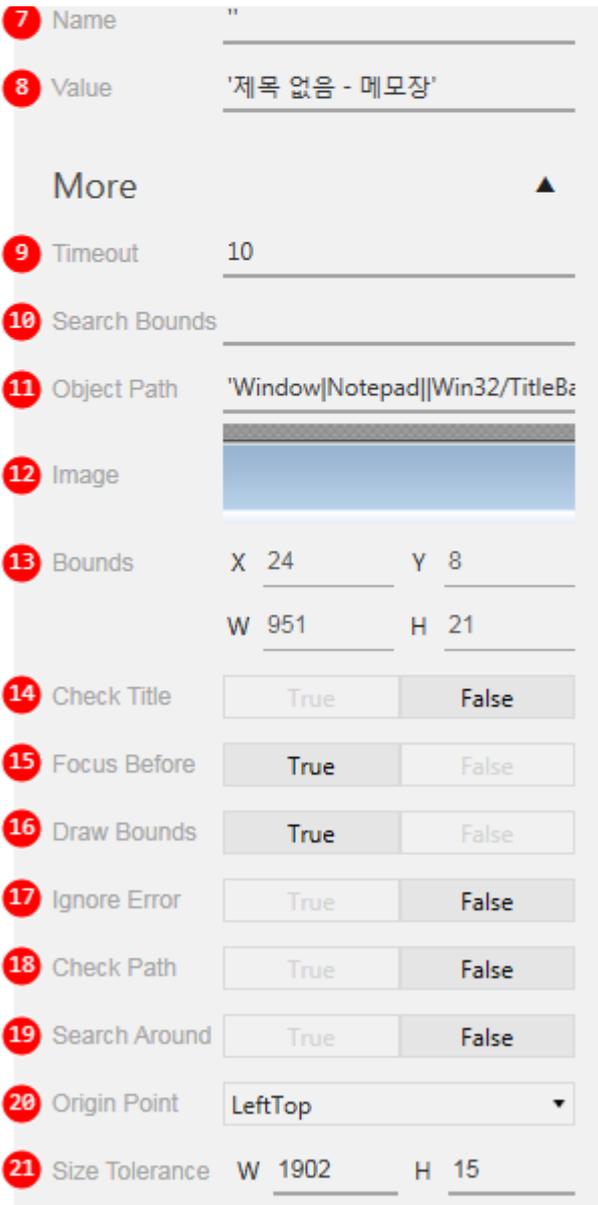
### RightClick(마우스 오른쪽 버튼으로 클릭하기)

대상 UI 객체를 마우스 오른쪽 버튼으로 클릭합니다.

그림 4-8 RightClick 속성 화면

The image shows a configuration window for the 'RightClick' activity. It is divided into two main sections: 'Activity' and 'Target'.

- Activity Section:**
  - Index:** A text input field.
  - Click Point:** Two input fields for X (364) and Y (11).
  - Check Name:** A radio button group with 'True' selected.
  - Check Value:** A radio button group with 'False' selected.
  - Check Size:** A radio button group with 'False' selected.
- Target Section:**
  - Application:** 'notepad'
  - Title:** ''제목 없음 - 메모장''
  - Type:** 'TitleBar'



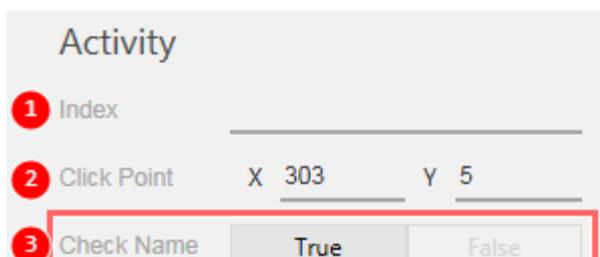
번호	설명
1	동일한 정보를 가진 UI 객체 목록에서 대상 UI 객체를 식별하기 위한 Index를 입력합니다. 대상 UI 객체 주변에 보이지 않는 투명한 객체가 있거나, 주변 UI 객체와 영역이 겹치는 형태인 경우에는 대상 UI 객체가 아니라 주변 객체가 인식되는 경우가 있습니다. 이 경우 Search Around 옵션을 사용하여 주변 객체의 영역을 추가로 검사하도록 설정할 수 있습니다.
2	UI 객체를 찾은 후 클릭할 위치를 UI 객체의 왼쪽 상단 기준 좌표 값으로 입력합니다. 마우스로 직접 지정한 경우에는 자동으로 입력됩니다.
3	UI 객체 비교 시 Name, Value, Size 속성에 대한 검사 여부를 설정합니다.
4	대상 UI 객체 앱의 이름이 나타납니다.
5	대상 UI 객체 앱의 타이틀이 나타납니다. 타이틀은 앱의 상태에 따라 변경될 수 있으므로 UI 객체 비교 시 필수로 일치해야 하는 사항은 아니지만, 동일한 앱이 여러 개 발견되는 경우에는 타이틀이 일치하는 UI 객체가 선택됩니다.
6	대상 UI 객체의 클래스 타입이 나타납니다.

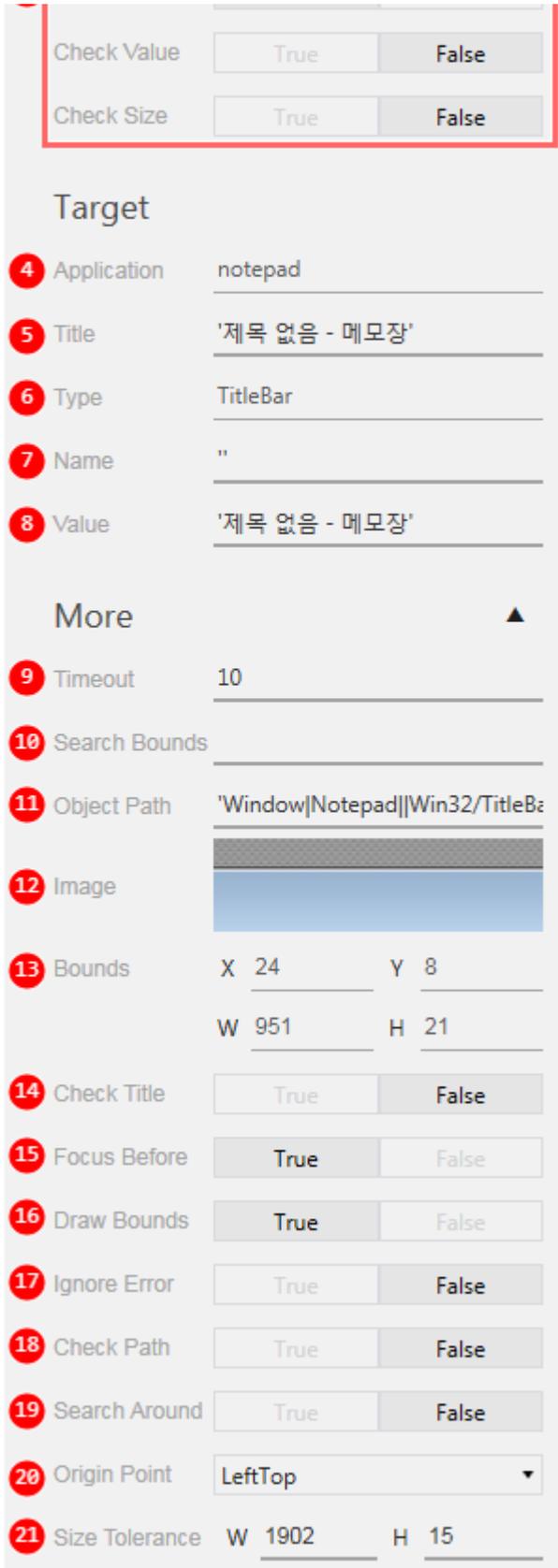
번호	설명
7	대상 UI 객체의 Name 정보가 나타납니다. 대부분 화면에 표시되는 문자열이 나타나지만, 앱의 구현 방식에 따라 값이 있어도 나타나지 않을 수 있습니다. Name, Value, Title의 경우 'www:' 키워드를 이용하여 정규식을 사용하거나 변수를 입력할 수 있습니다. 단, 변수가 입력된 경우 노드의 검사 기능은 사용할 수 없습니다.
8	대상 UI 객체의 Value 정보가 나타납니다. 앱의 구현 방식에 따라 화면에 표시되는 문자열이 Name 항목 대신 해당 항목에 나타나거나, 값이 있어도 나타나지 않을 수 있습니다.
9	대상 UI 객체를 찾는 시간을 초 단위로 입력합니다.
10	전체 영역이 아닌 주어진 영역 내에서 검색합니다.
11	사용자가 지정한 대상 UI 객체 경로 정보를 작은따옴표 안에 입력합니다.
12	사용자가 지정한 대상 UI 객체의 이미지가 나타납니다.
13	사용자가 지정한 대상 UI 객체의 위치 및 크기가 앱의 좌상단 기준 상대 좌표로 나타납니다.
14	UI 객체 비교 시 Title 속성에 대한 검사 여부를 설정합니다.
15	이벤트를 발생시키기 전에 대상 UI 객체에 포커스를 설정할 수 있습니다. 대상 UI 객체에 포커스를 설정할 수 없는 경우에는 대상 앱의 메인 화면에 포커스를 설정합니다. 다른 앱에 의해 대상 UI 객체가 가려진 경우 대상 앱을 앞으로 가져옵니다. 대상 앱이 여러 개가 실행되어 있는 경우, 가장 앞에 위치한 앱에서 먼저 검사하고 대상 UI 객체를 찾지 못하면 뒤에 위치한 앱을 앞으로 가져온 후 다시 검사합니다.
16	실행 시 UI 객체를 찾았을 때 박스가 나타나도록 설정합니다.
17	오류가 발생해도 프로세스나 작업을 계속 진행하도록 설정합니다. IsExist 라이브러리를 이용해 존재 여부에 따라 클릭하거나 무시하는 형태의 스크립트를 작성하는 대신 Ignore Error 옵션을 활용할 수 있습니다.
18	대상 UI 객체를 검색할 때 화면 내 UI 객체를 비교하는 대신 상위 UI 객체를 검사하도록 설정합니다. 이 경우 속도가 향상되지만 Name, Value, Size, Title 등의 정보가 다른 UI 객체와 차별되지 않을 경우에는 대상 UI 객체를 정확하게 찾지 못할 수 있습니다.
19	정확한 UI 객체 식별을 위해 주변 UI 객체를 추가로 확인할지 설정합니다.
20	상대 좌표 계산 시 사용되는 원점(객체의 4개 모서리)을 설정합니다.
21	Check Size가 True로 설정되어 있을 때 허용하는 오차의 범위를 입력합니다.

### DoubleClick(더블클릭하기)

대상 UI 객체를 더블클릭합니다.

그림 4-9 DoubleClick 속성 화면





번호	설명
1	동일한 정보를 가진 UI 객체 목록에서 대상 UI 객체를 식별하기 위한 Index를 입력합니다. 대상 UI 객체 주변에

번호	설명
	보이지 않는 투명한 객체가 있거나, 주변 UI 객체와 영역이 겹치는 형태인 경우에는 대상 UI 객체가 아니라 주변 객체가 인식되는 경우가 있습니다. 이 경우 Search Around 옵션을 사용하여 주변 객체의 영역을 추가로 검사하도록 설정할 수 있습니다.
2	UI 객체를 찾은 후 클릭할 위치를 UI 객체의 왼쪽 상단 기준 좌표 값으로 입력합니다. 마우스로 직접 지정한 경우에는 자동으로 입력됩니다.
3	UI 객체 비교 시 Name, Value, Size 속성에 대한 검사 여부를 설정합니다.
4	대상 UI 객체 앱의 이름이 나타납니다.
5	대상 UI 객체 앱의 타이틀이 나타납니다. 타이틀은 앱의 상태에 따라 변경될 수 있으므로 UI 객체 비교 시 필수로 일치해야 하는 사항은 아니지만, 동일한 앱이 여러 개 발견되는 경우에는 타이틀이 일치하는 UI 객체가 선택됩니다.
6	대상 UI 객체의 클래스 타입이 나타납니다.
7	대상 UI 객체의 Name 정보가 나타납니다. 대부분 화면에 표시되는 문자열이 나타나지만, 앱의 구현 방식에 따라 값이 있어도 나타나지 않을 수 있습니다. Name, Value, Title의 경우 'www:' 키워드를 이용하여 정규식을 사용하거나 변수를 입력할 수 있습니다. 단, 변수가 입력된 경우 노드의 검사 기능은 사용할 수 없습니다.
8	대상 UI 객체의 Value 정보가 나타납니다. 앱의 구현 방식에 따라 화면에 표시되는 문자열이 Name 항목 대신 해당 항목에 나타나거나, 값이 있어도 나타나지 않을 수 있습니다.
9	대상 UI 객체를 찾는 시간을 초 단위로 입력합니다.
10	전체 영역이 아닌 주어진 영역 내에서 검색합니다.
11	사용자가 지정한 대상 UI 객체 경로 정보를 작은따옴표 안에 입력합니다.
12	사용자가 지정한 대상 UI 객체의 이미지가 나타납니다.
13	사용자가 지정한 대상 UI 객체의 위치 및 크기가 앱의 좌상단 기준 상대 좌표로 나타납니다.
14	UI 객체 비교 시 Title 속성에 대한 검사 여부를 설정합니다.
15	이벤트를 발생시키기 전에 대상 UI 객체에 포커스를 설정할 수 있습니다. 대상 UI 객체에 포커스를 설정할 수 없는 경우에는 대상 앱의 메인 화면에 포커스를 설정합니다. 다른 앱에 의해 대상 UI 객체가 가려진 경우 대상 앱을 앞으로 가져옵니다. 대상 앱이 여러 개 실행되어 있는 경우, 가장 앞에 위치한 앱에서 먼저 검사하고 대상 UI 객체를 찾지 못하면 뒤에 위치한 앱을 앞으로 가져온 후 다시 검사합니다.
16	실행 시 UI 객체를 찾았을 때 박스가 나타나도록 설정합니다.
17	오류가 발생해도 프로세스나 작업을 계속 진행하도록 설정합니다. <b>IsExist</b> 라이브러리를 이용해 존재 여부에 따라 클릭하거나 무시하는 형태의 스크립트를 작성하는 대신 <b>Ignore Error</b> 옵션을 활용할 수 있습니다.
18	대상 UI 객체를 검색할 때 화면 내 UI 객체를 비교하는 대신 상위 UI 객체를 검사하도록 설정합니다. 이 경우 속도가 향상되지만 Name, Value, Size, Title 등의 정보가 다른 UI 객체와 차별되지 않을 경우에는 대상 UI 객체를 정확하게 찾지 못할 수 있습니다.
19	정확한 UI 객체 식별을 위해 주변 UI 객체를 추가로 확인할지 설정합니다.
20	상대 좌표 계산 시 사용되는 원점(객체의 4개 모서리)을 설정합니다.
21	<b>Check Size</b> 가 <b>True</b> 로 설정되어 있을 때 허용하는 오차의 범위를 입력합니다.

## MouseHover(마우스 포인터 올리기)

마우스 포인터를 대상 UI 객체 위에 올려놓습니다.

그림 4-10 MouseHover 속성 화면

### Activity

- 1 Index
- 2 Click Point X  Y
- 3 
 Check Name  True  False  
 Check Value  True  False  
 Check Size  True  False

### Target

- 4 Application
- 5 Title
- 6 Type
- 7 Name
- 8 Value

### More ▲

- 9 Timeout
- 10 Search Bounds
- 11 Object Path
- 12 Image 
- 13 Bounds X  Y   
W  H
- 14 Check Title  True  False
- 15 Focus Before  True  False
- 16 Draw Bounds  True  False



번호	설명
1	동일한 정보를 가진 UI 객체 목록에서 대상 UI 객체를 식별하기 위한 Index를 입력합니다. 대상 UI 객체 주변에 보이지 않는 투명한 객체가 있거나, 주변 UI 객체와 영역이 겹치는 형태인 경우에는 대상 UI 객체가 아니라 주변 객체가 인식되는 경우가 있습니다. 이 경우 Search Around 옵션을 사용하여 주변 객체의 영역을 추가로 검사하도록 설정할 수 있습니다.
2	UI 객체를 찾은 후 클릭할 위치를 UI 객체의 왼쪽 상단 기준 좌표 값으로 입력합니다. 마우스로 직접 지정한 경우에는 자동으로 입력됩니다.
3	UI 객체 비교 시 Name, Value, Size 속성에 대한 검사 여부를 설정합니다.
4	대상 UI 객체 앱의 이름이 나타납니다.
5	대상 UI 객체 앱의 타이틀이 나타납니다. 타이틀은 앱의 상태에 따라 변경될 수 있으므로 UI 객체 비교 시 필수로 일치해야 하는 사항은 아니지만, 동일한 앱이 여러 개 발견되는 경우에는 타이틀이 일치하는 UI 객체가 선택됩니다.
6	대상 UI 객체의 클래스 타입이 나타납니다.
7	대상 UI 객체의 Name 정보가 나타납니다. 대부분 화면에 표시되는 문자열이 나타나지만, 앱의 구현 방식에 따라 값이 있어도 나타나지 않을 수 있습니다. Name, Value, Title의 경우 'www:' 키워드를 이용하여 정규식을 사용하거나 변수를 입력할 수 있습니다. 단, 변수가 입력된 경우 노드의 검사 기능은 사용할 수 없습니다.
8	대상 UI 객체의 Value 정보가 나타납니다. 앱의 구현 방식에 따라 화면에 표시되는 문자열이 Name 항목 대신 해당 항목에 나타나거나, 값이 있어도 나타나지 않을 수 있습니다.
9	대상 UI 객체를 찾는 시간을 초 단위로 입력합니다.
10	전체 영역이 아닌 주어진 영역 내에서 검색합니다.
11	사용자가 지정한 대상 UI 객체 경로 정보를 작은따옴표 안에 입력합니다.
12	사용자가 지정한 대상 UI 객체의 이미지가 나타납니다.
13	사용자가 지정한 대상 UI 객체의 위치 및 크기가 앱의 좌상단 기준 상대 좌표로 나타납니다.
14	UI 객체 비교 시 Title 속성에 대한 검사 여부를 설정합니다.
15	이벤트를 발생시키기 전에 대상 UI 객체에 포커스를 설정할 수 있습니다. 대상 UI 객체에 포커스를 설정할 수 없는 경우에는 대상 앱의 메인 화면에 포커스를 설정합니다. 다른 앱에 의해 대상 UI 객체가 가려진 경우 대상 앱을 앞으로 가져옵니다. 대상 앱이 여러 개 실행되어 있는 경우, 가장 앞에 위치한 앱에서 먼저 검사하고 대상 UI 객체를 찾지 못하면 뒤에 위치한 앱을 앞으로 가져온 후 다시 검사합니다.
16	실행 시 UI 객체를 찾았을 때 박스가 나타나도록 설정합니다.
17	오류가 발생해도 프로세스나 작업을 계속 진행하도록 설정합니다. IsExist 라이브러리를 이용해 존재 여부에 따라 클릭하거나 무시하는 형태의 스크립트를 작성하는 대신 Ignore

번호	설명
	<b>Error</b> 옵션을 활용할 수 있습니다.
18	대상 UI 객체를 검색할 때 화면 내 UI 객체를 비교하는 대신 상위 UI 객체를 검사하도록 설정합니다. 이 경우 속도가 향상되지만 Name, Value, Size, Title 등의 정보가 다른 UI 객체와 차별되지 않을 경우에는 대상 UI 객체를 정확하게 찾지 못할 수 있습니다.
19	정확한 UI 객체 식별을 위해 주변 UI 객체를 추가로 확인할지 설정합니다.
20	상대 좌표 계산 시 사용되는 원점(객체의 4개 모서리)을 설정합니다.
21	<b>Check Size</b> 가 True로 설정되어 있을 때 허용하는 오차의 범위를 입력합니다.

### SelectListItem(리스트에서 특정 항목 선택하기)

콤보 박스 또는 리스트에서 대상 UI 객체를 찾아 선택합니다.  
대상 UI 객체가 ListItem 타입인 경우에만 사용할 수 있습니다.



대상 UI 객체가 ListItem 타입인 경우에도 대상 앱의 구현 방식에 따라 동작하지 않거나 정보가 변경되지 않는 경우가 있으므로 주의가 필요합니다.

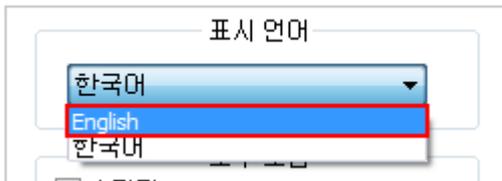


그림 4-11 SelectListItem 속성 화면

### Activity

Index \_\_\_\_\_

Check Name  True  False

Check Value  True  False

Check Size  True  False

### Target

Application explorer \_\_\_\_\_

Title '작업 전환기' \_\_\_\_\_

**2** Type ListItem \_\_\_\_\_

**3** Name 'SDS - Knox Portal - Internet Expl' \_\_\_\_\_

**4** Value "" \_\_\_\_\_

### More ▲

Timeout 10 \_\_\_\_\_

Search Bounds \_\_\_\_\_

Object Path 'List|TaskListThumbnailWnd|Win: \_\_\_\_\_

Image 

Bounds X 12 Y 9  
W 266 H 34

Check Title  True  False

**1** Focus Before  True  False

Draw Bounds  True  False

Ignore Error  True  False

Check Path  True  False

Search Around  True  False

Size Tolerance W 532 H 15

번호	설명
①	이벤트를 발생시키기 전에 대상 UI 객체에 포커스를 설정할 수 있습니다. 포커스를 설정하면 열려있던 콤보 박스가 닫힐 수 있으므로 기본값은 <b>False</b> 로 설정되어 있습니다.
②	대상 UI 객체의 타입이 나타납니다. <b>ListItem</b> 타입인 경우에만 <b>SelectListItem</b> 라이브러리를 사용할 수 있습니다.
③	선택할 항목의 텍스트를 작은따옴표 안에 입력합니다.
④	대상 UI 객체에 따라 텍스트가 <b>Name</b> 이 아닌 <b>Value</b> 항목에 표시될 수 있습니다. 이 경우 <b>Value</b> 항목에 텍스트를 입력하세요.

### GetListItems(리스트에 포함된 모든 항목 가져오기)

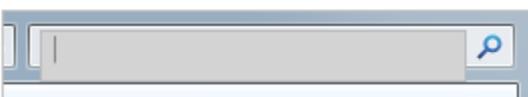
리스트, 트리 또는 메뉴 등에서 대상 UI 객체와 동일한 경로에 있는 항목들을 반환합니다.

### KeyboardEvent(키보드 이벤트 발생시키기)

대상 UI 객체와 관계 없이 키보드 이벤트를 발생시킵니다.

### SetText(텍스트 입력하기)

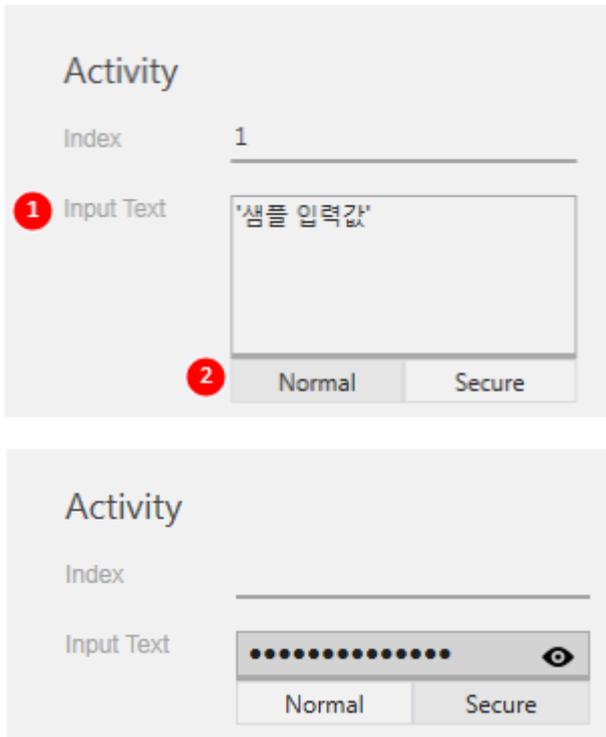
대상 UI 객체에 텍스트를 입력합니다. 텍스트 박스와 같이 대상 UI 객체가 ValuePattern을 지원하고 읽기 전용이 아닌 경우에만 사용할 수 있습니다.



대상 UI 객체를 선택해 텍스트 박스가 나타나면 텍스트를 입력하세요. 키보드의 Shift+Enter 키를 눌러 여러 줄을 입력할 수 있습니다.

입력이 완료되면 Enter 키를 눌러 종료하세요.

그림 4-12 SetText 속성 화면



번호	설명
1	입력할 텍스트를 작은따옴표 안에 입력합니다. 줄바꿈이나 따옴표 등의 특수 문자는 JavaScript의 코딩 규칙에 따라 \를 이용하여 표기합니다.
2	입력한 텍스트를 암호화할지 설정합니다. Secure를 선택한 경우에도 Input Text 항목에서  을 통해 원문을 확인 가능하며, 파일에 저장될 때에만 암호화됩니다.

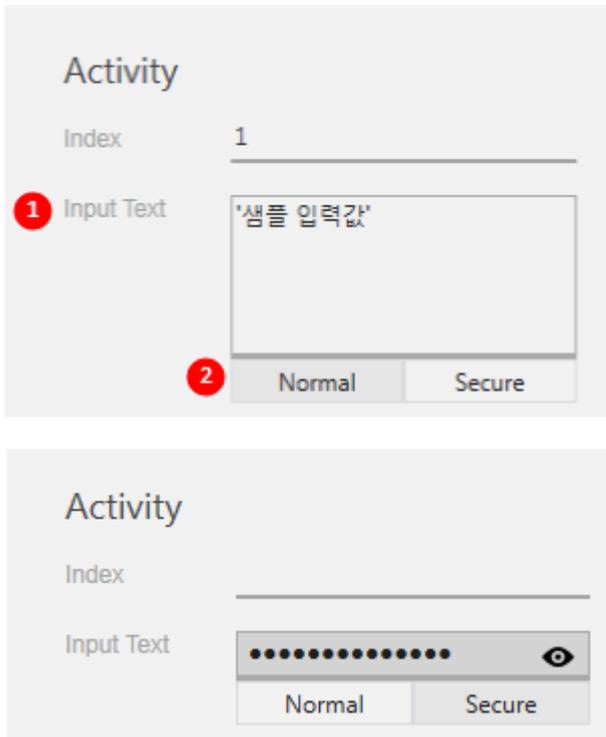
 암호화된 문자열이 포함된 프로젝트를 다른 PC로 복사하여 사용할 경우, 다른 PC에서는 원문을 확인할 수 없습니다.

### TextInput(텍스트 입력하기)

대상 UI 객체를 찾은 후 텍스트를 입력합니다.

Windows의 문자 입력 API를 이용하여 텍스트를 하나씩 대상 UI 객체로 전달합니다. 옵션 설정에 따라 키보드 이벤트 API를 사용하여 텍스트를 입력할 수도 있습니다.

그림 4-13 TextInput 속성 화면



번호	설명
1	입력할 텍스트를 작은따옴표 안에 입력합니다. 줄바꿈이나 따옴표 등의 특수 문자는 JavaScript의 코딩 규칙에 따라 \를 이용하여 표기합니다.
2	입력한 텍스트를 암호화할지 설정합니다. Secure를 선택한 경우에도 Input Text 항목에서  을 통해 원문을 확인 가능하며, 파일에 저장될 때에만 암호화됩니다.

 • 유니코드에서 지원하지 않는 문자의 경우 입력되지 않을 수 있습니다.

• `{{KeyCode}}` 형태의 키워드를 이용하여 문자열 사이에 키보드 이벤트를 발생시킬 수 있습니다. 예를 들어, `{{Enter}}`는 키보드의 Enter 키를 누르는 동작이고, `{{+Enter}}`와 `{{-Enter}}`와 같이 키의 다운과 업을 분리하여 표시할 수 있습니다. `{{}}` 사이에 입력할 키워드는 아래 링크를 참고하세요.

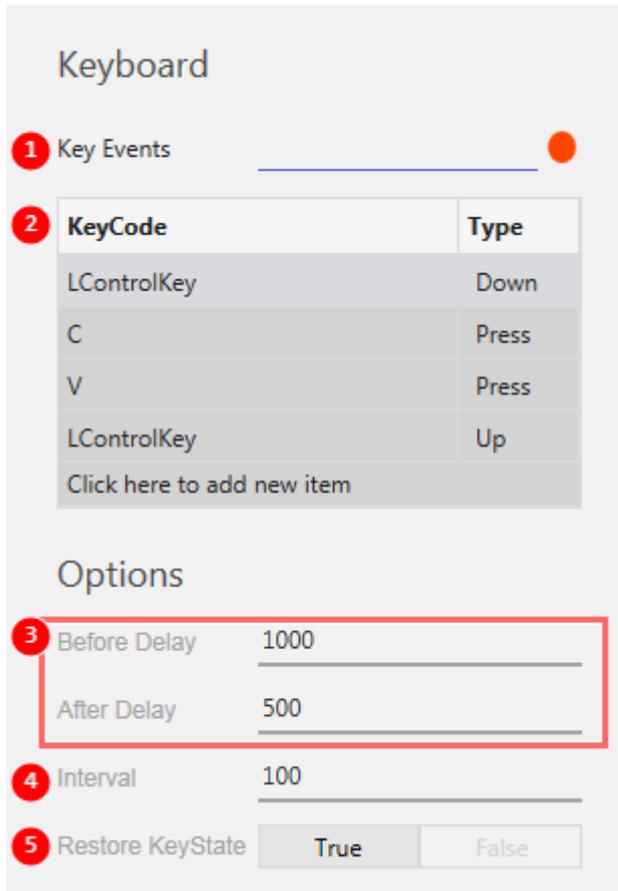
<https://docs.microsoft.com/ko-kr/dotnet/api/system.windows.forms.keys?redirectedfrom=MSDN&view=netframework-4.7.2>

### KeyInput(키보드 버튼 누르기)

키보드 버튼을 누르고 떼는 등의 키보드 이벤트를 발생시킵니다. Ctrl+Shift+C와 같이 여러 개의 키를 동시에 누르는 경우에도 사용할 수 있습니다.

대상 UI 객체를 지정하지 않고, 현재 포커스가 설정된 UI 객체에 키보드 이벤트가 발생합니다.

그림 4-14 KeyInput 속성 화면

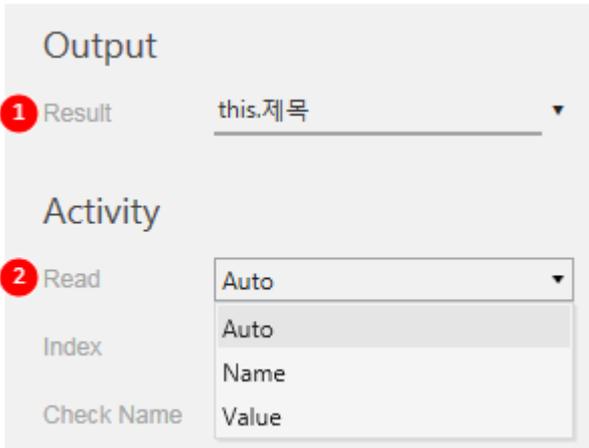


번호	설명
1	 눌러 키보드 이벤트를 레코딩할 수 있습니다.
2	레코딩된 키보드 이벤트를 확인할 수 있습니다. 이벤트 항목을 선택한 후 레코딩하면 선택된 항목 위에 새로운 이벤트가 추가됩니다. 이벤트 항목을 선택한 후 키보드의 Delete 키를 누르면 해당 이벤트가 삭제됩니다. Click here to add new item을 클릭한 후 KeyCode와 Type 항목에 입력해 설정해 키보드 이벤트를 수동으로 추가할 수 있습니다. Type 항목에는 Press, Down, Up, Text, Variable, Delay를 입력할 수 있습니다.
3	키보드 이벤트를 발생시키기 전과 후의 대기시간을 설정합니다.
4	버튼을 누르는 간격을 설정합니다. 단축키의 경우 대상 앱에 따라 간격이 짧으면 동작하지 않을 수 있습니다.
5	KeyInput 수행 후 Ctrl, Alt, Shift 키 상태 초기화할지 선택합니다.

GetText(텍스트 읽어오기)

대상 UI 객체의 텍스트를 읽어옵니다.  
Name과 Value 중 읽고자 하는 항목을 선택할 수 있습니다.

그림 4-15 GetText 속성 화면



번호	설명
①	대상 UI 객체에서 읽어온 텍스트를 할당할 변수를 지정합니다.
②	대상 UI 객체에서 읽으려는 텍스트의 타입을 선택합니다. <b>Auto</b> 를 선택하면 대상 UI 객체가 Edit 또는 Document 타입일 경우 <b>Value</b> 값을, 그 외의 경우에는 <b>Name</b> 값을 우선적으로 읽습니다.

### GetBounds(바운드 정보 가져오기)

대상 UI 객체의 바운드 정보를 가져옵니다. 결과는 x, y, width, height 형태의 문자열로 반환됩니다. 반환된 문자열은 Assign 혹은 ExecuteScript 에서 eval 함수를 이용하여 숫자 배열로 변환하여 사용할 수 있습니다.

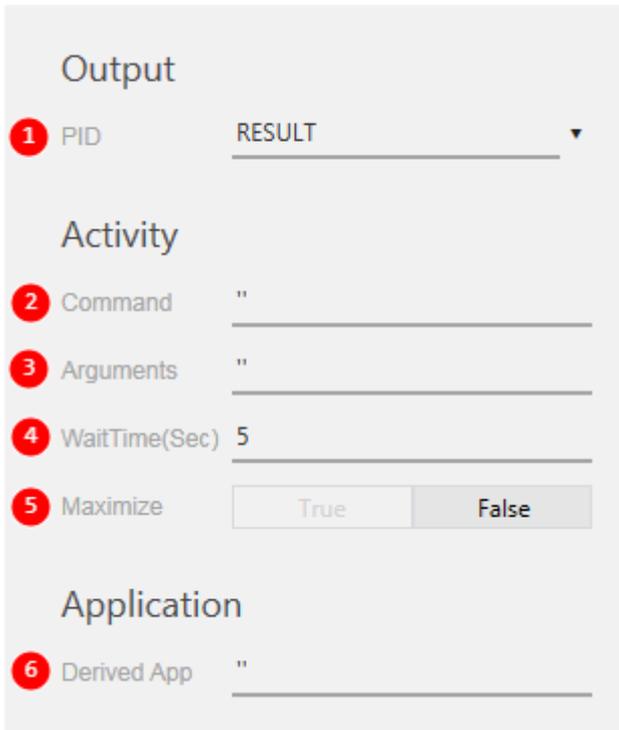
### OpenApp(앱 실행하기)

지정한 앱을 실행합니다. 앱이 실행되면 실행된 앱의 PID(Process ID)를 반환합니다.



OpenApp이 반환하는 PID는 **ActivateApp**, **CloseApp**에서 활용할 수 있습니다.

그림 4-16 OpenApp 속성 화면



번호	설명
1	실행한 앱의 PID를 저장하기 위한 변수를 지정합니다.
2	실행할 앱의 실행 명령어를 입력합니다. 앱에 따라 실행 파일에 대한 전체 경로를 입력해야 할 수도 있습니다.
3	앱 실행 시 필요한 실행 인자를 전달할 수 있습니다. 실행 인자가 없는 경우에는 빈 문자열("")을 입력합니다.
4	앱을 실행한 후 정상적으로 실행되었는지 확인하는 시간을 설정합니다.
5	앱이 실행된 후 최대화 상태로 변경할지 설정할 수 있습니다.
6	Command 항목에 입력한 앱이 아닌 다른 앱이 실행되는 경우 실제 실행될 앱의 이름을 지정할 수 있습니다. null을 입력하거나 false를 입력하면 대상 앱의 실행 여부를 검사하지 않습니다.



런처 앱을 통해 실행되거나, 중복으로 실행되지 않는 경우 등 대상 앱의 특성에 따라 앱을 실행한 후에 반환되는 PID와 화면에 나타난 앱의 PID는 다를 수 있습니다.

예를 들어, Windows 10에서 calc.exe를 실행할 경우 calc.exe는 종료되고 calculator.exe가 실행되어 화면에 나타납니다. 이 경우, OpenApp의 PID 항목에는 화면에 나타난 calculator.exe 대신 OpenApp에서 지정한 calc.exe의 PID가 반환됩니다.

### ActivateApp(앱 활성화하기)

선택한 PID에 해당하는 앱을 맨 앞으로 이동시켜 활성화합니다.

그림 4-17 **ActivateApp** 속성 화면

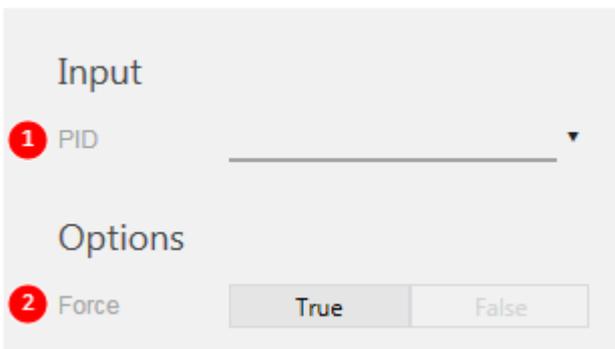


번호	설명
①	활성화할 앱의 PID를 선택합니다.
②	활성화할 앱을 최대화할지 선택합니다.

### CloseApp(앱 종료하기)

선택한 PID에 해당하는 앱을 종료합니다.

그림 4-18 **CloseApp** 속성 화면

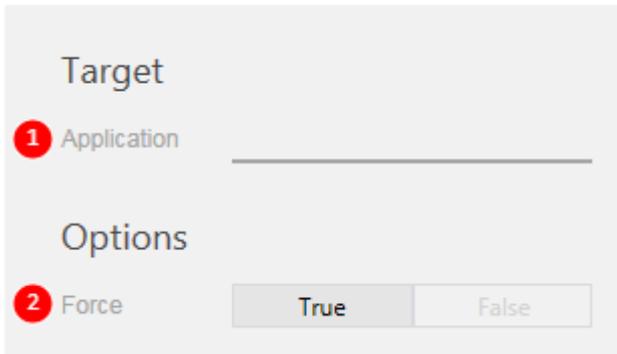


번호	설명
①	종료할 앱의 PID를 선택합니다.
②	<b>True</b> 로 설정하면 앱에 닫기 명령을 전달한 후에도 종료되지 않을 경우, 앱의 상태와 상관없이 강제 종료합니다. <b>False</b> 로 설정하면 앱에 닫기 명령을 한 번만 전달합니다. 앱의 상태에 따라 앱이 종료되지 않을 수 있습니다. 예를 들어, 편집 중인 메모장을 닫을 경우 저장 여부를 묻는 창이 열려 앱이 종료되지 않습니다.

### CloseAll(앱 종료하기)

입력한 앱 이름에 해당하는 앱 또는 모든 앱을 종료합니다.

그림 4-19 CloseAll 속성 화면



번호	설명
1	종료할 앱의 이름을 작은따옴표 안에 입력합니다. 모든 앱을 종료하려면 빈 문자열("")을 입력합니다.
2	True로 설정하면 앱에 닫기 명령을 전달한 후에도 종료되지 않을 경우, 앱의 상태와 상관없이 강제 종료합니다. False로 설정하면 앱에 닫기 명령을 한 번만 전달합니다. 앱의 상태에 따라 앱이 종료되지 않을 수 있습니다. 예를 들어, 편집 중인 메모장을 닫을 경우 저장 여부를 묻는 창이 열려 앱이 종료되지 않습니다.

### OpenWindow(윈도우 열기)

앱을 실행한 후 대상 UI 객체가 나타나는지 확인하고 앱을 최대화합니다.

OpenApp 라이브러리로 실행할 수 없는 런처 형태의 앱 실행 등에 활용할 수 있습니다.

### CloseWindow(윈도우 닫기)

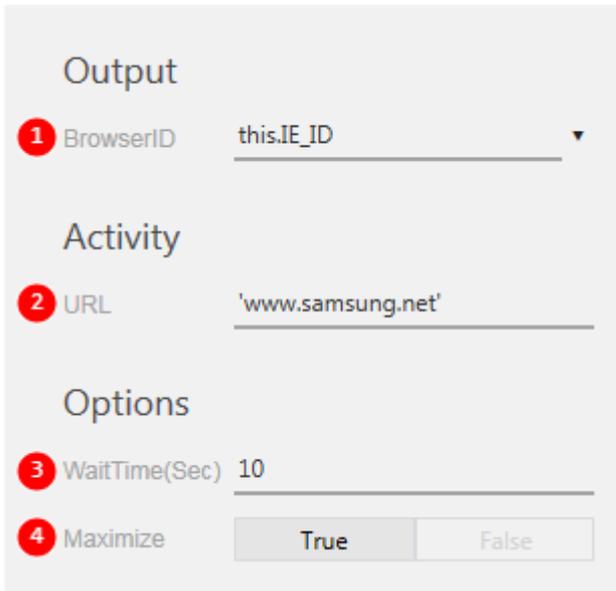
대상 UI 객체가 소속된 앱의 윈도우를 닫습니다.

### OpenBrowser(브라우저 열기)

인터넷 익스플로러를 실행합니다. URL을 입력하면 해당 URL로 접속합니다.

실행이 완료되면 브라우저의 ID 리스트를 반환합니다. 반환된 ID 리스트는 CloseBrowser에서 사용할 수 있습니다.

그림 4-20 OpenBrowser 속성 화면



번호	설명
①	실행된 브라우저의 ID를 저장하기 위한 변수를 지정합니다.
②	브라우저를 실행한 후 접속할 URL을 작은따옴표 안에 입력합니다. 빈 브라우저를 열려면 빈 문자열('')을 입력합니다.
③	브라우저가 실행되어 화면에 나타날 때까지의 대기시간을 설정합니다.
④	브라우저 창을 최대화할지 설정합니다.



- 브라우저는 대부분 하나의 ID를 가지지만, 팝업창이 열리는 등의 경우에는 여러 개의 ID를 가질 수 있습니다.
- 브라우저의 ID는 **OpenApp**의 PID와는 달리 앱 화면의 ID이므로 **CloseApp**에서는 사용할 수 없습니다.

### CloseBrowser(브라우저 종료하기)

선택한 ID에 해당하는 브라우저를 종료합니다. **OpenBrowser**가 반환하는 ID 리스트를 저장해야 사용할 수 있습니다.

### SendMail(메일 보내기)

메일을 보냅니다.

그림 4-21 SendMail 속성 화면

### Connection

1 Host Name

2 Port

3 ID

4 Password

Normal    Secure

5 Use SSL  True  False

### Mail

6 Sender

7 Receivers

8 CCs

9 BCCs

10 Subject

11 Body

12 Attachements

13 File Size (MB)

14 IsHtml  True  False

번호	설명
1	메일 서버(SMTP) 호스트명을 입력합니다.
2	메일 서버 포트를 입력합니다.
3	발신자 ID 를 입력합니다.
4	발신자 비밀번호를 입력합니다.
5	SSL 사용 여부를 지정합니다.
6	발신자 메일 주소를 입력합니다.
7	수신자 메일 주소를 입력합니다. (세미콜론으로 구분하여 복수개 지정이 가능합니다. 참조자와 비밀참조자 입력란에도 동일하게 적용됩니다.)
8	참조자 메일 주소를 입력합니다.
9	비밀참조자 메일 주소를 입력합니다.

번호	설명
10	메일 제목을 입력합니다.
11	메일 본문을 입력합니다.
12	첨부 파일 경로를 입력합니다.
13	첨부 파일 제한 용량을 설정합니다. (수행 시 첨부 파일의 용량이 제한 용량보다 큰 경우 에러로 처리됩니다.)
14	Html 포맷을 사용할지 여부를 설정합니다.

## ReceiveMail(메일 받기)

메일을 받아 변수에 할당합니다.

그림 4-22 ReceiveMail 속성 화면

The screenshot shows the configuration interface for ReceiveMail, divided into three sections: Output, Connection, and Mail. Red numbered callouts (1-18) point to specific fields and controls:

- Output:** 1. Mails dropdown menu (set to RESULT).
- Connection:** 2. Host Name text field; 3. Port text field; 4. ID text field (containing ""); 5. Password field with Normal/Secure radio buttons; 6. Use SSL field with True/False radio buttons.
- Mail:** 7. Begin Time text field ('2018-01-01 00:00:00'); 8. End Time text field ('2018-01-02 00:00:00'); 9. Sender text field ('SearchSender@samsung.com'); 10. Subject text field ('Search Title'); 11. Body text field ('Search Body Text'); 12. Attachements text field ('Search Attachment Name'); 13. Max Count text field (1); 14. Start No. text field; 15. With Contents field with True/False radio buttons; 16. Ignore HTML... field with True/False radio buttons; 17. Ignore Body... field with True/False radio buttons; 18. Recent First field with True/False radio buttons.

번호	설명
1	조회된 메일정보를 저장하기 위한 변수를 지정합니다.
2	메일 서버(SMTP) 호스트명을 입력합니다.
3	메일 서버 포트를 입력합니다.
4	수신자 ID 를 입력합니다.
5	수신자 비밀번호를 입력합니다.
6	SSL 사용 여부를 지정합니다.
7	입력된 시간 이후에 발송된 메일을 수신합니다.
8	입력된 시간 이전에 발송된 메일 수신합니다. (비어 있는 경우 현재 시간까지 조회합니다.)
9	발신자 메일 주소를 입력합니다.
10	조회할 메일 제목을 입력합니다.
11	조회할 메일 본문을 입력합니다. 값 설정 시 속도가 느려질 수 있습니다.
12	조회할 메일의 첨부 파일명을 입력합니다. 값 설정 시 속도가 느려질 수 있습니다.
13	조회할 메일의 최대 갯수를 입력합니다. (전체 조회인 경우 0으로 입력합니다.)
14	조회할 메일의 시작 번호를 입력합니다.
15	본문, 첨부파일 목록을 포함하여 반환할 것인지 결정합니다.
16	본문 내용 중에서 html 태그 부분은 무시할지 여부를 결정합니다.
17	반환되는 정보 중에서 본문에 있는 첨부는 무시할지 여부를 결정합니다. (본문 첨부를 포함하여 메일을 수신할 경우 <b>False</b> 를 선택합니다.)
18	최근 메일을 먼저 조회할 지 여부를 결정합니다.

### DownloadMailAttachment(메일 첨부파일 저장하기)

메일의 첨부파일을 저장합니다.

### GetHierarchyInfo(Xml 형태로 변환하기)

지정한 객체 또는 앱을 기준으로 하위 객체의 정보를 Xml 형태로 반환합니다. 화면 내의 정보를 한 번에 처리하는 용도 등으로 사용할 수 있습니다.



반환되는 객체 정보는 XmlDocument 객체로 표현됩니다. XmlDocument에 대한 자세한 내용은 아래 링크를 참고하세요.

<https://docs.microsoft.com/ko-kr/dotnet/api/system.xml.xmldocument?view=netframework-4.7.2>

## SendRequest(요청 보내기)

REST API를 호출합니다.

그림 4-23 SendRequest 속성 화면

The screenshot displays the configuration interface for a REST API request. It features a list of properties on the left, each with a red circular icon and a corresponding input field on the right. The properties are: Response (set to RESULT), Method (set to GET), ContentType (set to Text), URL (empty), Authorization (empty), Headers (set to a JSON object), Body (empty), and FormData (set to a JSON object).

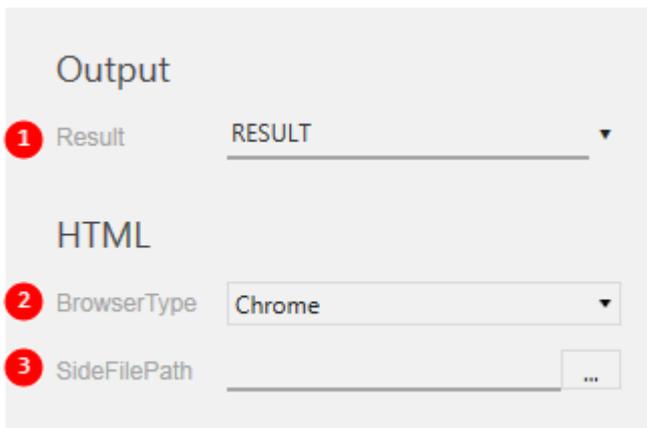
번호	설명
①	호출한 결과를 저장하기 위한 변수를 지정합니다. 지정한 변수의 <b>Headers</b> , <b>Contents</b> 프로퍼티를 사용할 수 있습니다.
②	호출할 방식을 선택합니다.
③	호출할 내용 유형을 선택합니다.
④	호출할 주소를 작은따옴표 안에 입력합니다.
⑤	API Key 를 작은따옴표 안에 입력합니다.
⑥	Header 정보를 작은따옴표 안에 입력합니다.

번호	설명
7	Body 정보를 작은따옴표 안에 입력합니다.
8	호출할 내용 유형이 <b>multipart/formdata</b> 인 경우에만 입력합니다.

### SeleniumSideRunner(Selenium 실행하기)

Selenium IDE Script 를 실행합니다.

그림 4-24 SeleniumSideRunner 속성 화면



번호	설명
1	Script(.side) 실행 결과를 저장하기 위한 변수를 지정합니다.
2	Script(.side)를 실행할 브라우저를 선택합니다.
3	Selenium IDE를 이용하여 레코딩한 Script 파일의 경로를 지정합니다.

### OpenWebDriver(웹드라이버 생성하기)

Selenium 웹드라이버를 생성합니다.

그림 4-25 OpenWebDriver 속성 화면

번호	설명
①	생성된 웹드라이버 객체를 저장하기 위한 변수를 지정합니다.
②	웹드라이버로 접속할 URL 을 작은따옴표 안에 입력합니다.
③	사용할 브라우저를 선택합니다.
④	Selenium 웹드라이버와 통신할 포트를 작은따옴표 안에 입력합니다. default 가 기본값이며 기본포트로 연결합니다.

### CloseWebDriver(웹드라이버 종료하기)

Selenium 웹드라이버를 종료합니다.

그림 4-26 CloseWebDriver 속성 화면

번호	설명
①	종료할 웹드라이버 변수를 입력합니다.

### FindElementsWebDriver(객체 찾기)

웹페이지의 객체를 검색합니다.

그림 4-27 FindElementsWebDriver 속성 화면

번호	설명
①	객체를 검색할 웹드라이버 변수를 입력합니다.
②	검색된 객체를 저장할 변수를 입력합니다.
③	입력한 시간 내에 객체를 찾지 못할 경우 에러가 발생합니다.
④	검색할 객체의 XPath 를 작은 따옴표 안에 입력합니다.
⑤	동일한 XPath의 객체가 여러개일 경우 모두 저장할지 여부를 결정합니다.

### WaitElementWebDriver(객체 생성 대기하기)

지정한 객체가 화면에 생성될때까지 기다립니다.

그림 4-28 WaitElementWebDriver 속성 화면

번호	설명
①	웹드라이버 변수를 입력합니다.
②	지정한 객체가 생성될때까지 대기할 시간을 입력합니다.
③	객체의 XPath 를 입력합니다.

### NavigateWebDriver(페이지 이동하기)

지정한 url 로 페이지를 이동합니다.

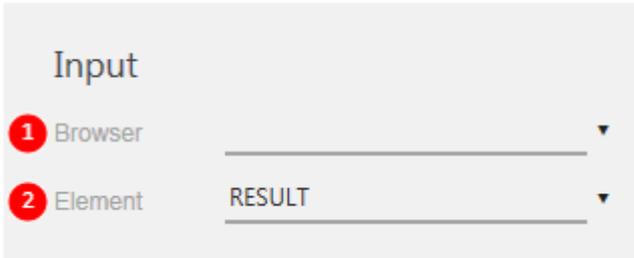
그림 4-29 NavigateWebDriver 속성 화면

번호	설명
①	웹드라이버 변수를 입력합니다.
②	이동할 URL 을 작은따옴표 안에 입력합니다.

### ClickWebDriver(객체 클릭하기)

객체를 클릭합니다.

그림 4-30 ClickWebDriver 속성 화면

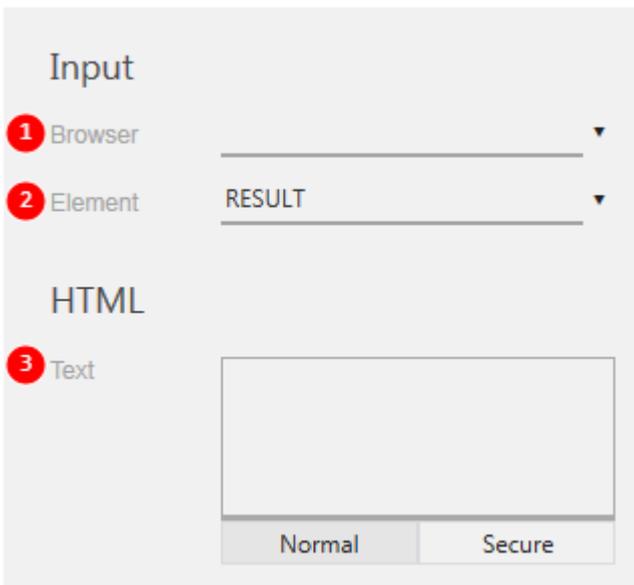


번호	설명
1	웹드라이버 변수를 입력합니다.
2	클릭할 객체를 입력합니다.

### SendKeysWebDriver(문자열 보내기)

객체에 문자열을 보냅니다.

그림 4-31 SendKeysWebDriver 속성 화면



번호	설명
1	웹드라이버 변수를 입력합니다.
2	문자열을 보낼 객체 변수를 입력합니다.
3	보낼 문자열을 입력합니다.

### GetDataWebDriver(데이터 가져오기)

객체에서 지정한 데이터값을 가져옵니다.

그림 4-32 GetDataWebDriver 속성 화면

번호	설명
①	웹드라이버 변수를 입력합니다.
②	가져온 데이터값을 저장할 변수를 입력합니다.
③	가져올 데이터 타입을 선택합니다.
④	가져올 데이터 타입의 이름(키값)을 입력합니다.

### ScreenshotWebDriver(캡처하기)

웹드라이버에 표시되는 화면을 캡처합니다.

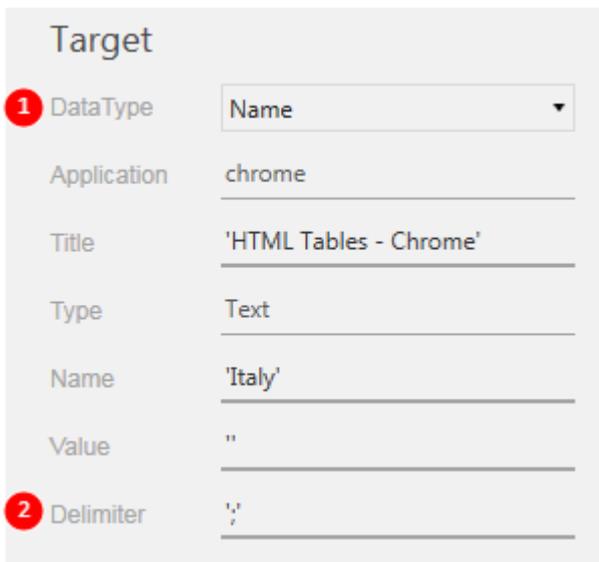
그림 4-33 ScreenshotWebDriver 속성 화면

번호	설명
1	웹드라이버 변수를 입력합니다.
2	캡처된 결과가 저장될 경로를 입력합니다.

### GetGridItems(Grid 항목 가져오기)

Grid 로 표시된 항목들을 읽어옵니다.

그림 4-34 GetGridItems 속성 화면



번호	설명
1	읽어올 항목을 형식을 지정합니다.
2	그리드 내부에 읽어올 항목이 2개 이상인 경우 항목들을 구분할 수 있는 표시를 입력합니다.

### GetListItemIndex(리스트 항목 순서 가져오기)

리스트에서 선택한 항목이 몇 번째 항목인지 가져옵니다.

### GetGridItemIndex(Grid 항목 순서 가져오기)

Grid 에서 선택한 항목이 몇 번째 항목인지 가져옵니다.

## 엑셀 라이브러리

Excel 라이브러리에서 제공하는 기능을 이용해 엑셀 문서를 제어할 수 있습니다.

사용자는 새 문서를 만들거나, 기존 문서를 열어 편집하고 저장할 수 있습니다.

엑셀 라이브러리에서 사용할 수 있는 기능은 다음과 같습니다.

- 새 엑셀 문서 및 워크시트 만들기(NewExcel, CreateExcel, CreateWorkSheet)
- 엑셀 문서 열기, 활성화하기, 저장하기, 닫기, 분리하기, 가져오기(OpenExcel, ActivateExcel, SaveExcel, CloseExcel, SplitExcel, GetActiveExcel, GetExcel, GetSheetName, GetAllSheetNames)
- 워크시트 및 영역 선택하기(SelectWorkSheet, SelectRange, SelectRow, SelectCol)
- 영역 가져오기, 읽기, 쓰기, 생성하기(GetActiveRange, GetDataRange, GetRowCount, GetColCount, SplitRange, MergeRange, ReadRange, WriteRange, MakeRange, ClearRange)
- 필터 적용하기, 해제하기, 정렬하기(SetFilter, SetColorFilter, ReleaseFilter, SortData)
- 문자열 검색하기, 입력하기, 바꾸기(SearchString, WriteString, ReplaceString)
- 복사하기, 붙여넣기, 삭제하기, 이동하기(CopyPaste, CopyWorkSheet, CopyRange, PasteRange, CopyWorkSheet, DeleteWorkSheet, MoveWorkSheet)
- 매크로 실행하기(ExecuteMacro)
- 행 및 열 추가, 선택, 삭제하기(AddRow, AddCol, DeleteRow, DeleteCol)
- 수식 가져오기, 입력하기(ReadCellFormula, WriteCellFormula)
- 스타일, 테두리 적용하기(SetStyleRange, SetBorderRange)
- 숨기기 적용하기, 해제하기(SetHiddenRow, SetHiddenCol)
- 함수 (VLookUp)
- 단축 키 (SendShortCut)

### NewExcel(빈 엑셀 문서 열기)

빈 엑셀 문서를 엽니다.

그림 4-35 NewExcel 속성 화면



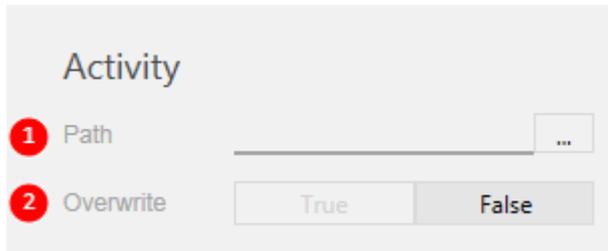
번호	설명
①	엑셀 문서를 할당할 변수를 지정합니다.
②	엑셀 문서를 비활성화할지 설정합니다.

번호	설명
3	엑셀 창을 최대화할지 설정합니다.

### CreateExcel(새 엑셀 문서 만들기)

새 문서를 생성합니다.

그림 4-36 CreateExcel 속성 화면



번호	설명
1	새 엑셀 문서가 생성될 경로 및 파일명을 입력합니다.
2	파일명이 같을 경우 덮어쓰기 여부를 설정합니다.

### CreateWorkSheet(새 워크시트 추가하기)

엑셀 문서에 새 워크시트를 추가합니다.

그림 4-37 CreateWorkSheet 속성 화면



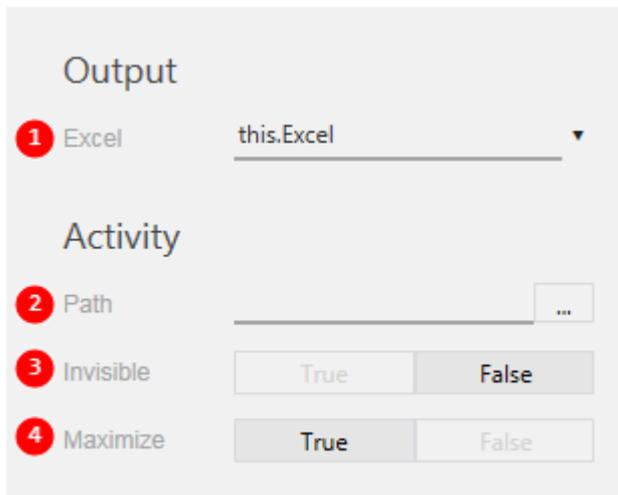
번호	설명
1	열려있는 엑셀 문서가 할당된 변수를 선택합니다.
2	새로 추가할 워크시트의 이름을 작은따옴표 안에 입력합니다.

번호	설명
3	새 워크시트를 생성할 위치를 입력합니다.

### OpenExcel(엑셀 문서 열기)

엑셀 문서를 엽니다.

그림 4-38 OpenExcel 속성 화면



번호	설명
1	열려있는 엑셀 문서를 할당할 변수를 지정합니다. CreateWorkSheet, SelectWorkSheet 등에서 사용할 수 있습니다.
2	열어올 엑셀 파일을 선택하거나 문서의 경로를 입력합니다.
3	엑셀 문서를 비활성화할지 설정합니다.
4	엑셀 창을 최대화할지 설정합니다.

### ActivateExcel(엑셀 창 활성화하기)

비활성화된 엑셀 문서를 활성화시킵니다.

그림 4-39 ActivateExcel 속성 화면

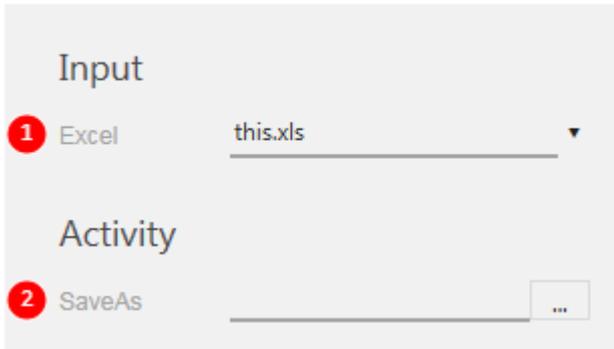


Excel 항목에서 활성화할 엑셀 문서가 할당된 변수를 선택하세요.

### SaveExcel(문서 저장하기)

엑셀 문서를 저장할 수 있습니다.

그림 4-40 SaveExcel 속성 화면

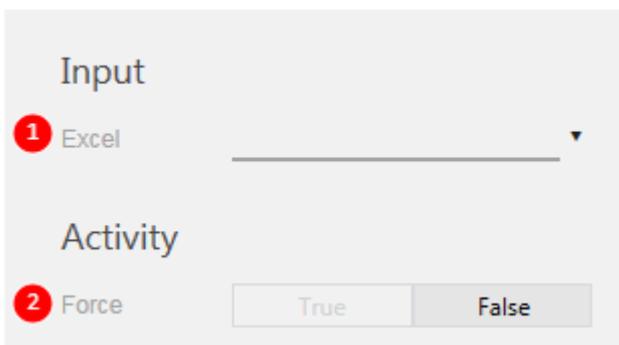


번호	설명
1	저장할 엑셀 문서가 할당된 변수를 선택합니다.
2	다른 이름으로 저장하려면 저장할 경로와 파일명을 선택합니다.

### CloseExcel(엑셀 문서 닫기)

엑셀 문서를 닫습니다.

그림 4-41 CloseExcel 속성 화면



번호	설명
1	종료할 엑셀 문서가 할당된 변수를 선택합니다.
2	True로 설정하면 엑셀에 닫기 명령을 전달한 후에도 종료되지 않을 경우, 엑셀의 프로세스를 강제 종료합니다. False로 설정하면 엑셀에 닫기 명령을 한 번만 전달합니다.

### SplitExcel(엑셀 문서 분리하기)

워크시트가 여러 개인 엑셀 문서의 워크시트 일부 또는 전체를 복사해 새 엑셀 파일을 만듭니다.

그림 4-42 SplitExcel 속성 화면

번호	설명
①	워크시트를 복사할 엑셀 문서가 할당된 변수를 선택합니다.
②	새 엑셀 파일을 저장할 경로를 선택합니다.
③	전체 워크시트를 복사할지 선택합니다.
④	복사할 워크시트의 이름을 작은따옴표 안에 입력합니다.

### GetActiveExcel(활성화된 엑셀 문서 가져오기)

활성화된 엑셀 문서를 가져와 변수에 할당합니다.

그림 4-43 GetActiveExcel 속성 화면

Excel 항목에서 활성화된 문서를 할당할 변수를 선택하세요.

### GetExcel(엑셀 문서 가져오기)

실행된 엑셀 문서 중 지정한 이름의 엑셀 문서를 가져와 변수에 할당합니다.

그림 4-44 GetExcel 속성 화면

The screenshot shows the configuration for the GetExcel activity. Under the 'Output' section, the 'Excel' dropdown is set to 'RESULT'. Under the 'Activity' section, the 'FileName' text box is empty and highlighted with a red border. Below it, there are two sets of buttons: 'Maximize' with 'True' and 'False' buttons, and 'IgnoreError' with 'True' and 'False' buttons.

FileName 항목에 가져올 엑셀 파일의 이름을 작은따옴표 안에 입력하세요.

### GetSheetName(워크시트 이름 가져오기)

입력한 위치에 있는 워크시트의 이름을 가져옵니다.

그림 4-45 GetSheetName 속성 화면

The screenshot shows the configuration for the GetSheetName activity. Under the 'Input' section, the 'Excel' dropdown is empty. Under the 'Output' section, the 'SheetName' dropdown is set to 'RESULT'. Under the 'Sheet' section, the 'Position' text box contains the number '1' and is highlighted with a red border.

**Position** 항목에 이름을 가져올 워크시트의 위치 번호를 입력하세요.

### GetAllSheetNames(엑셀 문서의 워크시트 이름 모두 가져오기)

엑셀 문서에 포함된 워크시트의 이름을 모두 가져옵니다.

그림 4-46 GetAllSheetNames 속성 화면

**SheetNames** 항목에서 워크시트의 이름을 문자열로 반환할 변수를 선택하세요.

### SelectWorkSheet(특정 워크시트 선택하기)

엑셀 문서에서 특정 워크시트를 선택합니다.

그림 4-47 SelectWorkSheet 속성 화면

번호	설명
1	열려있는 엑셀 문서가 할당된 변수를 선택합니다.
2	선택할 워크시트의 이름을 작은따옴표 안에 입력합니다.

### SelectRange(영역 지정하기)

엑셀 문서에서 영역을 선택합니다.

그림 4-48 SelectRange 속성 화면

번호	설명
①	영역을 선택할 엑셀 문서가 할당된 변수를 선택합니다.
②	영역을 선택할 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	선택할 영역의 범위를 작은따옴표 안에 입력합니다. 한 칸만 선택할 경우에는 'A1'과 같이 입력할 수 있습니다.

### SelectRow(행 선택하기)

특정한 행을 선택할 수 있습니다.

그림 4-49 SelectRow 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	행을 선택할 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	선택할 행의 번호를 작은따옴표 안에 입력합니다.

번호	설명
	행 번호 사이에 콜론(:)을 입력할 경우, 첫 번째 숫자부터 두 번째 숫자까지의 행이 모두 선택됩니다. 행 번호 사이에 콤마(,)를 입력할 경우, 해당되는 숫자의 행이 모두 선택됩니다.

### SelectCol(열 선택하기)

특정한 열을 선택할 수 있습니다.

그림 4-50 SelectCol 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	열을 선택할 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	선택할 열의 문자를 작은따옴표 안에 입력합니다. 열 문자 사이에 콜론(:)을 입력할 경우, 첫 번째 문자부터 두 번째 문자까지의 열이 모두 선택됩니다. 열 문자 사이에 콤마(,)를 입력할 경우, 해당되는 문자의 열이 모두 선택됩니다.

### GetActiveRange(선택된 영역 가져오기)

엑셀 문서에서 선택된 영역을 가져옵니다.

그림 4-51 GetActiveRange 속성 화면

선택된 영역을 가져올 엑셀 문서의 변수를 지정합니다.

**notes** dataArray 참조 방법(읽어 들인 Range 기준)  
 dataArray[0] → 1번째 Row  
 dataArray[0].Length → 1번째 Row의 개수  
 dataArray[0][1] → 1번째 Row, 2번째 Column 값

### GetDataRange(데이터 영역 가져오기)

엑셀 문서에서 데이터가 입력된 영역을 가져옵니다.

예를 들어, 아래 그림에서는 "A1:C3" 영역이 **Range** 항목의 변수에 할당됩니다.

	A	B	C	D
1	1			
2		2		
3			3	
4				

그림 4-52 GetDataRange 속성 화면

**Input**  
 1 Excel

**Output**  
 2 Range RESULT

**Sheet**  
 3 SheetName ACTIVE\_SHEET

번호	설명
1	데이터 영역을 가져올 엑셀 문서의 변수를 선택합니다.
2	가져온 데이터 영역이 할당될 변수를 지정합니다.
3	가져올 데이터 영역이 있는 워크시트의 이름을 작은따옴표 안에 입력합니다.

### GetRowCount(행 개수 가져오기)

행의 개수를 가져올 수 있습니다.

그림 4-53 GetRowCount 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	행의 개수를 가져올 변수를 지정합니다.
③	행의 개수를 가져올 워크시트의 이름을 작은따옴표 안에 입력합니다.

### GetColCount(열 개수 가져오기)

열의 개수를 가져올 수 있습니다.

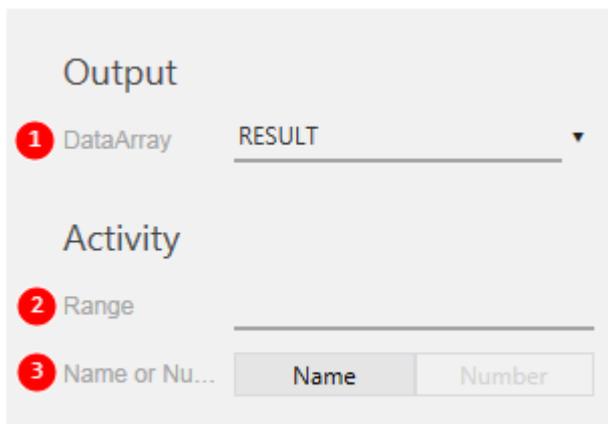
그림 4-54 GetColCount 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	열의 개수를 가져올 변수를 지정합니다.
③	열의 개수를 가져올 워크시트의 이름을 작은따옴표 안에 입력합니다.

### SplitRange(행 또는 열 분리해 할당하기)

주어진 영역에서 행과 열을 분리하여 변수에 할당할 수 있습니다.

그림 4-55 SplitRange 속성 화면



번호	설명
①	분리된 영역의 정보가 할당될 변수를 지정합니다.
②	분리할 영역을 입력합니다.
③	컬럼 이름을 문자 그대로 받을지 숫자('A' = 1)로 변환하여 받을지 여부를 결정합니다.

 "A6:BB2" 입력으로 "A,6,BB,2" 가 this.range 변수에 배열로 할당됩니다.

### MergeRange(셀 영역 합치기)

선택된 영역의 셀을 병합합니다.

그림 4-56 MergeRange 속성 화면

The screenshot shows the configuration interface for the MergeRange function. It is divided into three main sections: Input, Activity, and Sheet. In the Input section, 'Excel' is selected. In the Activity section, the 'IncludeAllData' property is highlighted with a red rectangular box, and the 'True' radio button is selected. In the Sheet section, 'SheetName' is set to 'ACTIVE\_SHEET' and 'Range' is set to ''A1:B2'.

**IncludeAllData** 항목에서 True를 선택하면 모든 셀의 데이터를 병합한 셀에 모두 입력합니다. False를 선택하면 첫 번째 셀의 데이터만 병합된 셀에 입력합니다.

### ReadRange(선택된 영역 읽어오기)

엑셀 문서에서 선택된 영역을 읽어옵니다.

그림 4-57 ReadRange 속성 화면

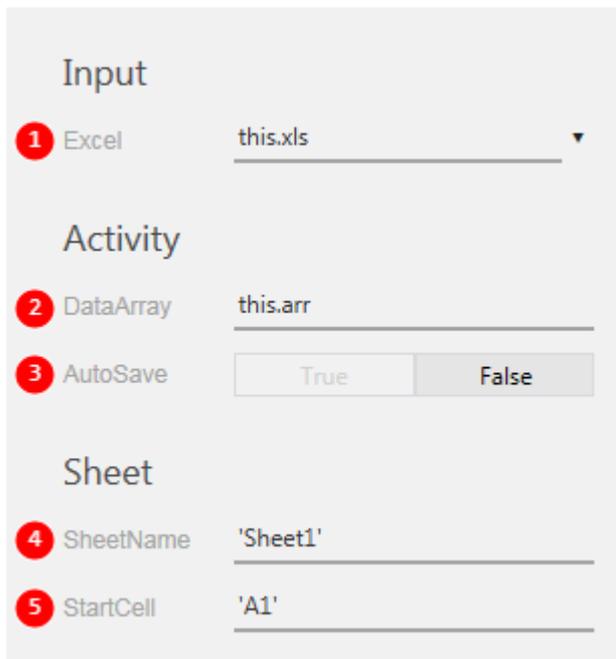
The screenshot shows the configuration interface for the ReadRange function. It is divided into three main sections: Input, Output, and Sheet. In the Input section, 'Excel' is selected and highlighted with a red circle containing the number 1. In the Output section, 'DataArray' is selected and highlighted with a red circle containing the number 2. In the Sheet section, 'ReadCellType' is set to 'Data' (highlighted with a red circle containing the number 3), 'Range' is set to ''A1:B2' (highlighted with a red circle containing the number 4), and 'SheetName' is set to 'ACTIVE\_SHEET' (highlighted with a red circle containing the number 5).

번호	설명
①	읽어올 영역이 있는 엑셀 문서가 할당된 변수를 선택합니다.
②	읽어온 영역을 할당할 변수를 지정합니다.
③	읽어올 영역의 타입을 지정합니다.
④	선택할 영역의 범위를 작은따옴표 안에 입력합니다. 한 칸만 선택할 경우에는 'A1'과 같이 입력할 수 있습니다.
⑤	읽어올 영역이 있는 워크시트의 이름을 작은따옴표 안에 입력합니다.

### WriteRange(데이터 영역 복사하기)

엑셀 문서에서 데이터가 입력된 영역을 복사해 지정한 위치에 입력할 수 있습니다.

그림 4-58 WriteRange 속성 화면



번호	설명
①	데이터가 입력될 엑셀 문서가 할당된 변수를 선택합니다.
②	입력할 데이터 영역이 할당된 엑셀 문서의 변수를 선택합니다.
③	데이터를 입력한 후 자동으로 저장할지 설정합니다.
④	데이터가 입력될 워크시트의 이름을 작은따옴표 안에 입력합니다.
⑤	데이터가 입력될 좌측 상단의 셀 이름을 작은따옴표 안에 입력합니다.

### MakeRange(영역 만들기)

영역을 만들어 변수에 할당할 수 있습니다.  
열이 증가되어 셀 주소를 입력하기 어려울 때 사용합니다.

그림 4-59 MakeRange 속성 화면

### Output

1 Range

### Activity

2 Range

3 InputValue

4 IsRow  True  False

5 IsIncrease  True  False

6 FirstAddress  True  False

번호	설명
1	만들어진 영역이 할당될 변수를 지정합니다.
2	영역이 만들어질 시작점의 위치를 작은따옴표 안에 입력합니다.
3	Range 항목의 위치를 기준으로 행 또는 열을 증가 또는 감소시킬 숫자를 입력합니다.
4	행을 변경하려면 True, 열을 변경하려면 False를 선택합니다.
5	증가시키려면 True, 감소시키려면 False를 입력합니다.
6	Range 항목이 한 셀이 아닌 영역으로 지정된 경우 MakeRange 대상을 시작 주소로 할 지 여부를 결정합니다.



위의 입력으로 나오는 결과는 "A6"이 this.range 변수에 할당됩니다.

### ClearRange(영역 지우기)

지정된 셀 영역의 데이터를 삭제합니다.

그림 4-60 ClearRange 속성 화면

번호	설명
①	데이터를 삭제할 엑셀 문서의 변수를 선택합니다.
②	데이터를 삭제한 후 자동으로 저장할지 설정합니다.
③	데이터를 삭제할 워크시트의 이름을 작은따옴표 안에 입력합니다.
④	데이터를 삭제할 셀 영역을 작은따옴표 안에 입력합니다.

### SetFilter(필터 지정하기)

필터가 있는 엑셀 문서에서 필터를 설정할 수 있습니다. 아래 그림과 같이 드롭다운 버튼이 있을 때 사용할 수 있습니다.

A	B
index ▼	data ▼
11	a
12	b
13	c
24	d
25	e
26	f
37	g
38	h
39	i

그림 4-61 SetFilter 속성 화면

The screenshot shows the configuration for the SetFilter activity. It is organized into four main sections:

- Input:** Item 1, Excel, set to 'this.xls'.
- Output:** Item 2, Range, set to 'this.range'.
- Activity:**
  - Item 3, FilterQuery, set to '=5&=6'
  - Item 4, FilterQuery2, set to '=6&=7'
  - Item 5, IncludeHeader, set to True (radio button selected).
  - Item 6, FilterOperAND, set to True (radio button selected).
- Sheet:**
  - Item 7, SheetName, set to 'Sheet1'
  - Item 8, FilterCell, set to 'A1'

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	결과를 저장할 변수를 지정합니다.
③	필터에 적용할 첫 번째 쿼리 값을 작은따옴표 안에 입력합니다.
④	필터에 적용할 두 번째 쿼리 값을 작은따옴표 안에 입력합니다.
⑤	데이터를 가져올 때 필터가 있는 헤더의 포함 여부를 설정합니다.
⑥	필터 쿼리 조건을 설정합니다. <b>True</b> 로 설정하면 두 가지 쿼리를 모두 만족하는 조건이며, <b>False</b> 로 설정하면 둘 중 하나라도 만족하는 조건입니다.
⑦	데이터를 가져올 워크시트의 이름을 작은따옴표 안에 입력합니다.
⑧	필터가 있는 셀의 위치를 작은따옴표 안에 입력합니다.



필터 쿼리 값 알아보기

- '=test' → test 문자열만 필터링함
- '<>test' → test 문자열만 제외하고 필터링함
- '=test\*' → test로 시작하는 문자열을 필터링함
- '=\*test' → test로 끝나는 문자열을 필터링함

'=\*test\*' → test가 포함된 모든 문자열을 필터링함  
 '<>\*test\*' → test가 포함되지 않은 모든 문자열을 필터링함  
 '=test??' → test로 시작하여 끝에 두 문자가 더 있는 문자열을 필터링함  
 '=?test' → test로 끝나는 앞에 두 문자가 더 있는 문자열을 필터링함  
 '=?test??' → test 양쪽으로 두 문자 이상 있는 모든 문자열을 필터링함  
 '<>test??' → test양쪽으로 두 문자 이상 포함되지 않은 모든 문자열을 필터링함  
 '>10' → 10보다 큰 숫자만 필터링함  
 '>=10' → 10보다 크거나 같은 숫자만 필터링함  
 '<10' → 10보다 작은 숫자만 필터링함

### SetColorFilter(색 필터 지정하기)

폰트나 셀의 색상으로 필터를 지정할 수 있습니다.

그림 4-62 SetColorFilter 속성 화면

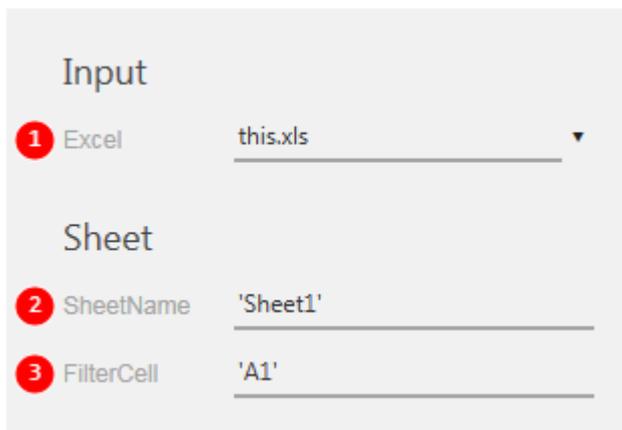
번호	설명
1	엑셀 문서가 할당된 변수를 선택합니다.
2	결과를 저장할 변수를 지정합니다.

번호	설명
3	필터를 지정할 색상 값을 작은따옴표 안에 입력합니다.
4	폰트 색상으로 필터를 지정할지 설정합니다.
5	데이터를 가져올 때 필터가 있는 헤더의 포함 여부를 설정합니다.
6	데이터를 가져올 워크시트의 이름을 작은따옴표 안에 입력합니다.
7	필터가 있는 셀의 위치를 작은따옴표 안에 입력합니다.

### ReleaseFilter(필터 해제하기)

엑셀 문서에 설정된 필터를 해제할 수 있습니다.

그림 4-63 ReleaseFilter 속성 화면



번호	설명
1	엑셀 문서가 할당된 변수를 선택합니다.
2	해제할 필터가 있는 워크시트의 이름을 작은따옴표 안에 입력합니다.
3	필터가 있는 셀의 위치를 작은따옴표 안에 입력합니다.

### SortData(정렬하기)

지정한 열이나 행을 오름차순 또는 내림차순으로 정렬할 수 있습니다.

그림 4-64 SortData 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	오름차순으로 정렬할지 내림차순으로 정렬할지 설정합니다.
③	정렬할 데이터가 행인지 열인지 선택합니다.
④	정렬할 데이터가 있는 워크시트의 이름을 작은따옴표 안에 입력합니다.
⑤	정렬할 데이터가 있는 셀의 위치를 작은따옴표 안에 입력합니다.

### SearchString(문자열 검색하기)

엑셀 문서에서 문자열을 검색할 수 있습니다.  
문자열을 찾은 경우 문자열이 있는 위치가 반환됩니다.

그림 4-65 SearchString 속성 화면

The screenshot shows the 'SearchString' configuration window. It is organized into four main sections:

- Input:** A dropdown menu labeled 'Excel' with a red circle '1' next to it.
- Output:** A dropdown menu labeled 'Range' with the value 'RESULT' and a red circle '2' next to it.
- Activity:**
  - A text input field labeled 'SearchString' containing the value ''string'' with a red circle '3' next to it.
  - Two radio buttons for 'PartMatch': 'True' (selected) and 'False', with a red circle '4' next to the 'True' button.
  - Two radio buttons for 'CaseMatch': 'True' and 'False' (selected), with a red circle '5' next to the 'False' button.
  - Two radio buttons for 'Value or For...': 'Value' (selected) and 'Formula', with a red circle '6' next to the 'Value' button.
- Sheet:**
  - A text input field labeled 'SheetName' containing the value 'ACTIVE\_SHEET' with a red circle '7' next to it.
  - A text input field labeled 'Range' containing the value ''1:1048576'' with a red circle '8' next to it.

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	검색된 문자열의 주소가 반환될 변수를 지정합니다.
③	검색할 문자열을 작은따옴표 안에 입력합니다.
④	부분 검색 허용 여부를 설정합니다.
⑤	대소문자 구분 여부를 설정합니다.
⑥	값을 찾을지 수식을 찾을지 여부를 설정합니다.
⑦	문자열을 검색할 엑셀 문서의 워크시트 이름을 작은따옴표 안에 입력합니다.
⑧	문자열을 검색할 엑셀 문서의 범위를 설정합니다. "1:1048576"이 기본값이며, 1번 줄부터 1048576번째 줄까지 검색한다는 의미입니다.

### WriteString(문자열 입력하기)

엑셀 문서에서 설정한 셀에 문자열을 입력할 수 있습니다.

그림 4-66 WriteString 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	입력할 문자열을 작은따옴표 안에 입력합니다.
③	자동 저장 여부를 설정합니다.
④	문자열이 입력될 워크시트 이름을 작은따옴표 안에 입력합니다.
⑤	문자열이 입력될 셀의 주소를 작은따옴표 안에 입력합니다.

### ReplaceString(문자열 변경하기)

특정 문자열을 검색하여 다른 문자열로 변경할 수 있습니다.

그림 4-67 ReplaceString 속성 화면

The screenshot shows the configuration interface for the ReplaceString activity. It is divided into three main sections: Input, Activity, and Sheet. 
   
1. **Input**: A dropdown menu is set to 'Excel'.
   
2. **Activity**: Contains three input fields: 'InputString', 'ReplaceString', and a red-bordered box containing two toggle buttons: 'PartMatch' (set to 'True') and 'CaseMatch' (set to 'False').
   
3. **Sheet**: Contains two text input fields: 'SheetName' with the value 'ACTIVE\_SHEET' and 'Range' with the value ''1:1048576''.

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	검색할 문자열을 입력합니다.
③	변경할 문자열을 입력합니다.
④	매칭 방식을 선택합니다.
⑤	문자열을 변경할 워크시트 이름을 작은따옴표 안에 입력합니다.
⑥	문자열이 변경될 셀의 주소를 작은따옴표 안에 입력합니다.

### CopyPaste(복사하고 붙여넣기)

엑셀 문서의 특정 영역을 복사하고 붙여넣을 수 있습니다.

그림 4-68 CopyPaste 속성 화면

The screenshot shows a configuration window for CopyPaste with three main sections: Input, Activity, and Sheet. Each section contains several properties with numbered callouts (1-12) pointing to specific controls.

- Input:**
  - 1 SourceExcel: Dropdown menu
  - 2 DestExcel: Dropdown menu
- Activity:**
  - 3 PasteType: Dropdown menu (value: All)
  - 4 PasteOperation: Dropdown menu (value: None)
  - 5 DestAutoSave: Toggle (True/False)
  - 6 PasteSkipBla...: Toggle (True/False)
  - 7 PasteTransp...: Toggle (True/False)
  - 8 PasteUseLink: Toggle (True/False)
- Sheet:**
  - 9 SourceSheet: Text input (value: ACTIVE\_SHEET)
  - 10 SourceRange: Text input (value: 'A1:B2')
  - 11 DestSheet: Text input (value: ACTIVE\_SHEET)
  - 12 TargetRange: Text input (value: 'A1:B2')

번호	설명
1	복사할 영역이 있는 엑셀 문서의 변수를 선택합니다.
2	붙여넣기할 엑셀 문서의 변수를 선택합니다.
3	붙여넣기할 값의 유형을 선택합니다.
4	붙여넣기할 연산을 선택합니다.
5	붙여넣기한 엑셀 문서를 자동으로 저장할지 선택합니다.
6	내용이 있는 셀만 붙여넣도록 설정합니다.
7	복사할 영역의 행/열을 바꿔 붙여넣을지 선택합니다.
8	복사할 영역의 링크를 붙여넣기할지 선택합니다.
9	복사할 영역이 있는 워크시트의 이름을 작은따옴표 안에 입력합니다.
10	엑셀 문서의 복사할 영역을 작은따옴표 안에 입력합니다.
11	붙여넣기할 워크시트의 이름을 작은따옴표 안에 입력합니다.
12	붙여넣기할 데이터의 위치를 작은따옴표 안에 입력합니다.

## CopyWorkSheet(특정 워크시트 복사하기)

엑셀 문서에서 특정 워크시트를 복사합니다.

그림 4-69 CopyWorkSheet 속성 화면

번호	설명
①	복사할 워크시트의 이름을 작은따옴표 안에 입력합니다.
②	복사한 워크시트를 붙여넣을 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	복사될 워크시트 위치의 바로 앞 워크시트 이름을 작은따옴표 안에 입력합니다.
④	복사될 워크시트 위치의 바로 뒤 워크시트 이름을 작은따옴표 안에 입력합니다.

## CopyRange(영역 복사하기)

지정한 영역의 데이터를 복사합니다.

## PasteRange(영역 붙여넣기)

복사한 영역 데이터를 붙여 넣습니다.

그림 4-70 PasteRange 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	붙여넣기할 데이터의 유형을 선택합니다.
③	붙여넣기할 연산을 선택합니다.
④	내용이 있는 데이터만 붙여넣도록 설정합니다.
⑤	복사한 데이터의 행/열을 바꿔 붙여넣을지 선택합니다.
⑥	복사한 데이터의 링크를 붙여넣기할지 선택합니다.
⑦	붙여넣기할 워크시트의 이름을 작은따옴표 안에 입력합니다.
⑧	붙여넣기할 데이터의 위치를 작은따옴표 안에 입력합니다.

### DeleteWorkSheet(특정 워크시트 삭제하기)

엑셀 문서에서 특정 워크시트를 삭제합니다.

그림 4-71 DeleteWorkSheet 속성 화면

번호	설명
①	열려있는 엑셀 문서가 할당된 변수를 선택합니다.
②	삭제할 워크시트의 이름을 작은따옴표 안에 입력합니다.

### ExecuteMacro(매크로 실행하기)

엑셀 문서에 저장된 매크로를 실행할 수 있습니다.  
매크로가 만들어져 있어야 사용할 수 있습니다.

그림 4-72 ExecuteMacro 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	실행할 매크로의 이름을 작은따옴표 안에 입력합니다.
③	매크로가 실행될 워크시트의 이름을 작은따옴표 안에 입력합니다.

## AddRow(행 추가하기)

엑셀 문서에 행을 추가할 수 있습니다.

그림 4-73 AddRow 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	행을 추가할 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	추가할 위치의 행 번호를 작은따옴표 안에 입력합니다. 행 번호 사이에 콜론(:)을 입력할 경우, 첫 번째 숫자부터 두 번째 숫자까지의 행이 모두 추가됩니다. 행 번호 사이에 콤마(,)를 입력할 경우, 해당되는 숫자의 행이 모두 추가됩니다.

## AddCol(열 추가하기)

엑셀 문서에 열을 추가할 수 있습니다.

그림 4-74 AddCol 속성 화면

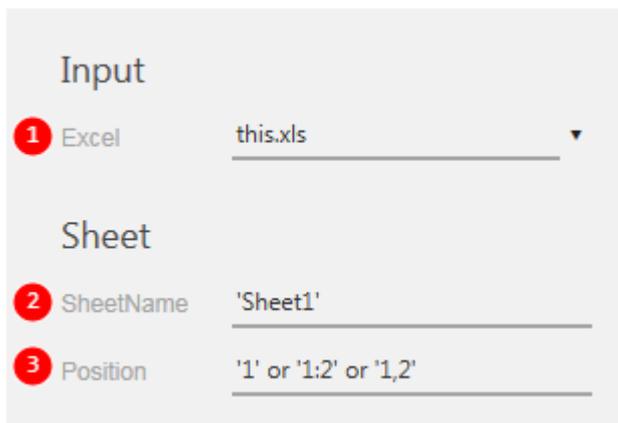
번호	설명
----	----

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	열을 추가할 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	추가할 위치의 열 문자를 작은따옴표 안에 입력합니다. 열 문자 사이에 콜론(:)을 입력할 경우, 첫 번째 문자부터 두 번째 문자까지의 열이 모두 추가됩니다. 열 문자 사이에 콤마(,)를 입력할 경우, 해당되는 문자의 열이 모두 추가됩니다.

### DeleteRow(행 삭제하기)

특정한 행을 삭제할 수 있습니다.

그림 4-75 DeleteRow 속성 화면



번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	행을 삭제할 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	삭제할 행의 번호를 작은따옴표 안에 입력합니다. 행 번호 사이에 콜론(:)을 입력할 경우, 첫 번째 숫자부터 두 번째 숫자까지의 행이 모두 삭제됩니다. 행 번호 사이에 콤마(,)를 입력할 경우, 해당되는 숫자의 행이 모두 삭제됩니다.

### DeleteCol(열 삭제하기)

특정한 열을 삭제할 수 있습니다.

그림 4-76 DeleteCol 속성 화면

The screenshot shows a configuration interface for the DeleteCol function. It is divided into two sections: 'Input' and 'Sheet'. Under 'Input', there is a dropdown menu labeled 'Excel' with the value 'this.xls' and a red circle with the number '1'. Under 'Sheet', there are two dropdown menus: 'SheetName' with the value ''Sheet1'' and a red circle with the number '2', and 'Position' with the value ''A' or 'A:B' or 'A,B'' and a red circle with the number '3'.

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	열을 삭제할 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	삭제할 열의 문자를 작은따옴표 안에 입력합니다. 열 문자 사이에 콜론(:)을 입력할 경우, 첫 번째 문자부터 두 번째 문자까지의 행이 모두 삭제됩니다. 열 문자 사이에 콤마(,)를 입력할 경우, 해당되는 문자의 열이 모두 삭제됩니다.

### ReadCellFormula(특정 셀의 입력값 가져오기)

특정 셀에 입력된 수식을 가져올 수 있습니다.

그림 4-77 ReadCellFormula 속성 화면

The screenshot shows a configuration interface for the ReadCellFormula function. It is divided into three sections: 'Input', 'Output', and 'Sheet'. Under 'Input', there is a dropdown menu labeled 'Excel' with the value 'this.xls' and a red circle with the number '1'. Under 'Output', there is a dropdown menu labeled 'Formula' with the value 'this.formula' and a red circle with the number '2'. Under 'Sheet', there are two dropdown menus: 'SheetName' with the value ''Sheet1'' and a red circle with the number '3', and 'Cell' with the value ''A1'' and a red circle with the number '4'.

번호	설명
----	----

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	읽어온 수식을 할당할 변수를 지정합니다.
③	수식을 읽어올 워크시트의 이름을 작은따옴표 안에 입력합니다.
④	수식을 읽어올 셀 주소를 작은따옴표 안에 입력합니다.

### WriteCellFormula(특정 셀에 수식 입력하기)

특정 셀에 수식을 입력할 수 있습니다.

그림 4-78 WriteCellFormula 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	셀에 입력할 수식을 작은따옴표 안에 입력합니다.
③	자동 저장 여부를 설정합니다.
④	수식을 입력할 워크시트의 이름을 작은따옴표 안에 입력합니다.
⑤	수식을 입력할 셀 주소를 작은따옴표 안에 입력합니다.

### SetStyleRange(영역 스타일 지정하기)

지정한 영역의 스타일을 설정합니다.

그림 4-79 SetStyleRange 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	스타일을 지정할 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	선택할 영역의 범위를 작은따옴표 안에 입력합니다. 한 칸만 선택할 경우에는 'A1'과 같이 입력할 수 있습니다.
④	폰트 이름을 선택합니다.
⑤	가로 정렬 방식을 선택합니다.
⑥	세로 정렬 방식을 선택합니다.
⑦	폰트 크기를 입력합니다. "0" 이 기본값이며, 폰트 크기를 변경하지 않는다는 의미입니다.

번호	설명
8	폰트 색상을 입력합니다. 색상은 RGB 숫자를 작은따옴표 안에 콤마(,)로 구분하여 입력합니다. 예를 들어, 파란색의 경우 '0,0,255' 를 입력합니다.
9	배경 색상을 입력합니다. 색상은 RGB 숫자를 작은따옴표 안에 콤마(,)로 구분하여 입력합니다. 예를 들어, 파란색의 경우 '0,0,255' 를 입력합니다.
10	너비를 입력합니다. "0" 이 기본값이며, 너비를 변경하지 않는다는 의미입니다.
11	높이를 입력합니다. "0" 이 기본값이며, 높이를 변경하지 않는다는 의미입니다.
12	텍스트 굵게 표시 여부를 선택합니다. Current 는 현재 엑셀 문서에 설정된 내용을 변경하지 않는다는 의미입니다.
13	텍스트 기울임꼴로 표시 여부를 선택합니다. Current 는 현재 엑셀 문서에 설정된 내용을 변경하지 않는다는 의미입니다.
14	텍스트 밑줄 표시 여부를 선택합니다. Current 는 현재 엑셀 문서에 설정된 내용을 변경하지 않는다는 의미입니다.

### SetBorderRange(영역 테두리 지정하기)

지정한 영역의 테두리를 설정합니다.

그림 4-80 SetBorderRange 속성 화면

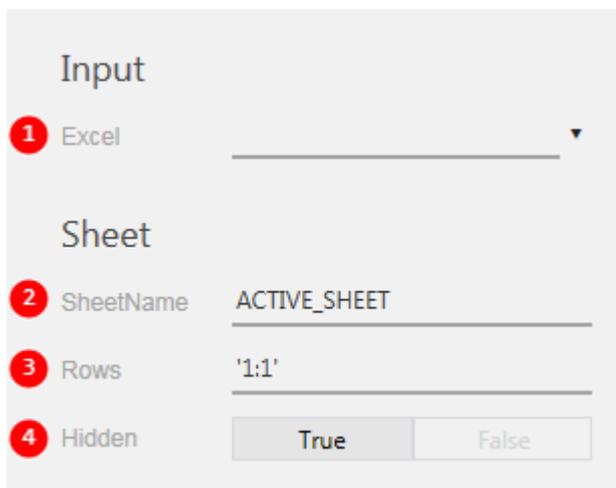
The image shows a property editor for the SetBorderRange function. It is organized into three main sections: Input, Sheet, and Cell. In the Input section, a dropdown menu is set to 'Excel'. In the Sheet section, there are two text input fields: 'SheetName' with the value 'ACTIVE\_SHEET' and 'Range' with the value 'A1:B2'. In the Cell section, there are two controls: a dropdown menu for 'Border' and a radio button group for 'Overlay' where 'True' is selected.

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	테두리를 지정할 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	선택할 영역의 범위를 작은따옴표 안에 입력합니다. 한 칸만 선택할 경우에는 'A1'과 같이 입력할 수 있습니다.
④	테두리 종류를 선택합니다.
⑤	True 로 설정할 경우 엑셀 문서에 설정된 테두리에 선택한 테두리를 추가합니다. False 로 설정할 경우 엑셀 문서에 설정된 테두리를 삭제하고 선택한 테두리를 추가합니다.

### SetHiddenRow(행 숨기기)

지정한 행을 숨기기/숨기기 취소 합니다.

그림 4-81 SetHiddenRow 속성 화면



번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	테두리를 지정할 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	선택할 행 영역의 범위를 작은따옴표 안에 입력합니다. 한 행만 선택할 경우에는 '1'과 같이 입력할 수 있습니다.
④	True 로 설정할 경우 선택된 행 영역을 숨기기 합니다. False 로 설정할 경우 선택된 행 영역을 숨기기 취소합니다.

### SetHiddenCol(열 숨기기)

지정한 열을 숨기기/숨기기 취소 합니다.

그림 4-82 SetHiddenCol 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	테두리를 지정할 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	선택할 열 영역의 범위를 작은따옴표 안에 입력합니다. 한 열만 선택할 경우에는 '1'과 같이 입력할 수 있습니다.
④	True 로 설정할 경우 선택된 열 영역을 숨기기 합니다. False 로 설정할 경우 선택된 열 영역을 숨기기 취소합니다.

### MoveWorkSheet(Sheet 이동하기)

지정한 워크시트의 위치를 옮깁니다.

그림 4-83 SetHiddenCol 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	이동할 워크시트의 이름을 작은따옴표 안에 입력합니다.
③	복사될 워크시트 위치의 바로 앞 워크시트 이름을 작은따옴표 안에 입력합니다.
④	복사될 워크시트 위치의 바로 뒤 워크시트 이름을 작은따옴표 안에 입력합니다.

## VLookUp(검색)

엑셀문서에서 원하는 정보를 다중조건을 이용하여 검색합니다.

그림 4-84 VLookUp 속성 화면

번호	설명
①	엑셀 문서가 할당된 변수를 선택합니다.
②	원하는 값을 찾기 위한 키값을 작은 따옴표 안에 입력합니다.
③	값을 찾을 영역을 작은 따옴표 안에 입력합니다.
④	찾을 영역에서 가져올 값이 있는 열의 위치를 입력합니다.
⑤	찾은 값을 표시할 워크시트의 이름을 작은따옴표 안에 입력합니다.
⑥	찾은 값을 표시할 셀을 작은 따옴표 안에 입력합니다.

번호	설명
7	정확하게 일치하는 값을 찾을 것인지, 유사한 값을 찾을 것인지를 결정합니다. (True : 유사한 값, False : 정확한 값)

## SendShortCut(단축 키)

단축 키를 엑셀에서 실행합니다.

그림 4-85 SendShortCut 속성 화면

The screenshot shows the configuration interface for the SendShortCut action. It is divided into three sections: Input, Activity, and Sheet. Red numbered callouts (1-9) point to specific fields:

- 1** Excel: A dropdown menu set to ACTIVE\_EXCEL.
- 2** Keys: A text input field containing an empty string "".
- 3** BeforeDelay: A numeric input field set to 1000.
- 4** AfterDelay: A numeric input field set to 500.
- 5** ShortCutName: A text input field containing an empty string "".
- 6** ALT: A toggle button set to True.
- 7** CTRL: A toggle button set to False.
- 8** SHIFT: A toggle button set to False.
- 9** SheetName: A dropdown menu set to ACTIVE\_SHEET.

번호	설명
1	엑셀 문서가 할당된 변수를 선택합니다.
2	단축 키 문자열을 입력합니다.
3	단축 키 실행 전 지연할 시간을 입력합니다.
4	단축 키 실행 후 지연할 시간을 입력합니다.
5	단축 키 이름을 입력합니다.
6	ALT key pressed 유무를 선택합니다.
7	CTRL key pressed 유무를 선택합니다.

번호	설명
8	SHIFT key pressed 유무를 선택합니다.
9	단축 키가 실행할 워크시트의 이름을 작은따옴표 안에 입력합니다.

 단축 키 문자열 알아보기

- '~' → Enter 키 동작
- '{BS}', '{BACKSPACE}' → 백스페이스 키 동작
- '{DELETE}' → Delete 키 동작
- '{CAPSLOCK}' → Caps Lock 키 동작
- '{ESC}' → ESC 키 동작
- '{F1}' → F1 키(도움말) 동작
- '{LEFT}' → 왼쪽 화살표 키 동작
- '{RIGHT}' → 오른쪽 화살표 키 동작
- '{UP}' → 위쪽 화살표 키 동작
- '{DOWN}' → 아래쪽 화살표 키 동작
- '{PGDN}' → Page Down 키 동작
- '{PGUP}' → Page Up 키 동작
- '{TAB}' → Tab 키 동작
- '+' → Shift 키 동작 ( '+ab' → Ab , '+(ab)' → AB)
- '^' → Ctrl 키 동작
- '%' → Alt 키 동작

## SAP 라이브러리

SAP 라이브러리에서 제공하는 기능을 이용해 SAP GUI를 제어할 수 있습니다.

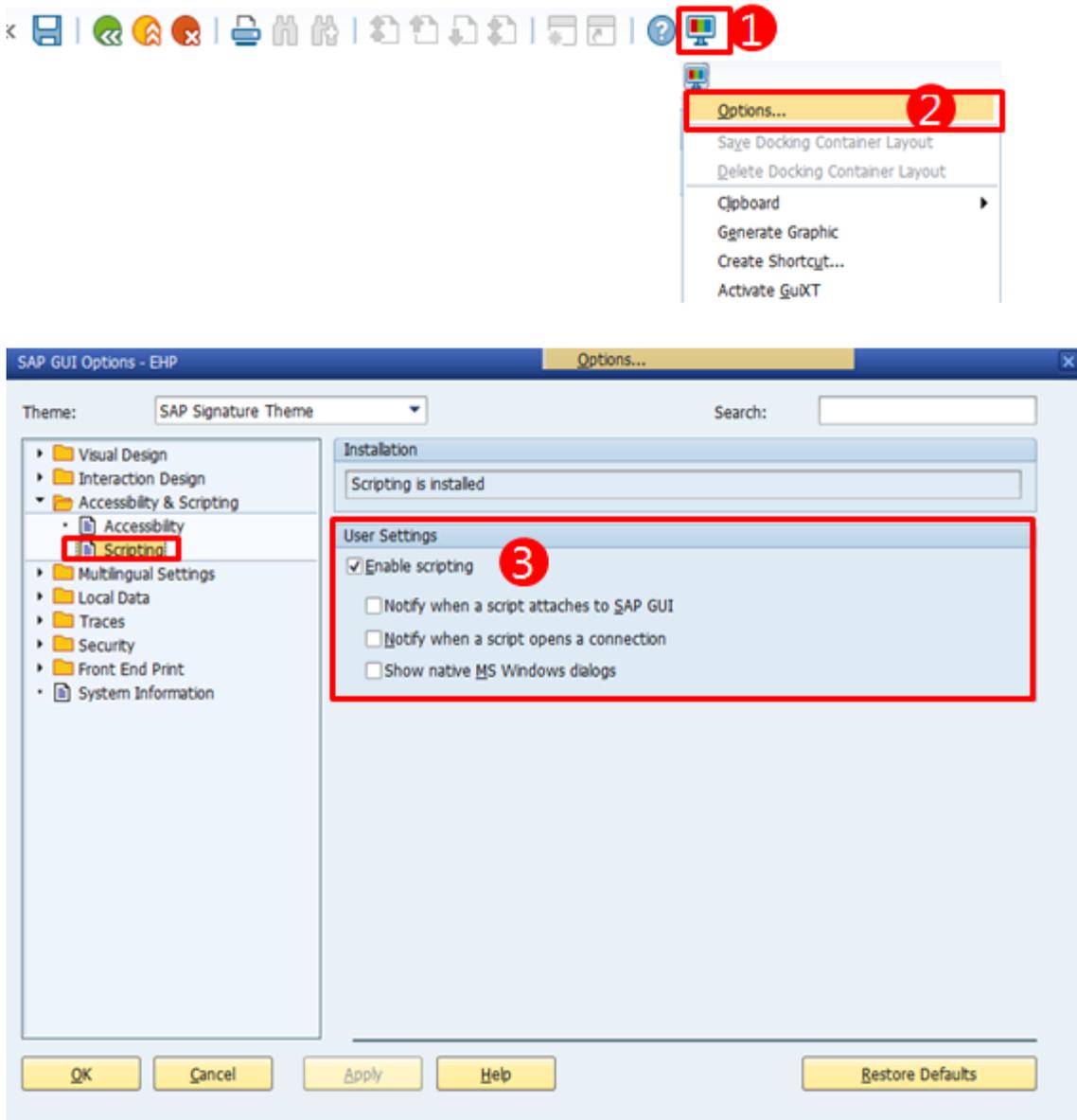
사용자는 SAP GUI에 데이터를 입력하거나, 입력된 데이터를 가져와 가공하고 처리할 수 있습니다.

SAP 라이브러리에서 사용할 수 있는 기능은 다음과 같습니다.

- 마우스 동작 기능(Click, DoubleClick, RightClick, MouseHover)
- 키보드 동작 기능(SetText, GetText)
- 대상 UI 객체 인식 관련 기능(IsExist, WaitSAPAppear, WaitSAPDisappear)
- SAP Grid 관련 기능(GetGridColumnID, GetGridRowCount, GetGridColCount)
- 콤보 박스 관련 기능(SelectComboltem)

### SAP 옵션 설정하기

SAP 라이브러리 사용을 위해서는 아래와 같이 설정이 필요합니다.



번호	설명
①	SAP 툴바에서 옵션 버튼을 클릭합니다.
②	메뉴에서 Options 를 선택합니다.
③	Enable Scripting은 체크하고 나머지는 모두 체크를 해제합니다.

### Click(클릭하기)

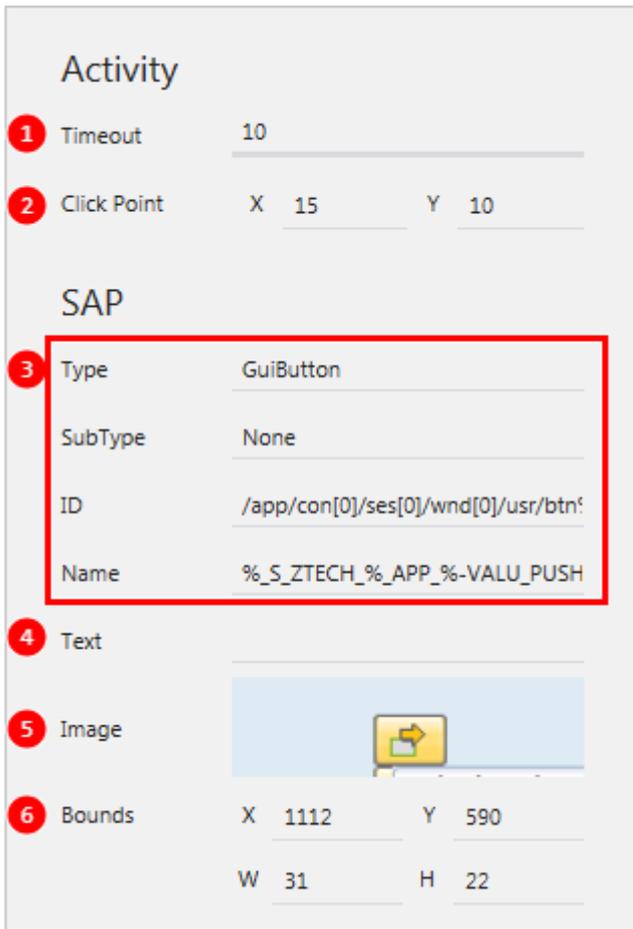
사용자가 지정한 대상 UI 객체를 찾아 클릭합니다.

1. 노트에서 **No Target > Double click to select**를 더블클릭하세요.
2. UI 객체에 마우스 포인터를 올리면 빨간색 테두리가 나타납니다. 클릭 이벤트를 발생시킬 UI 객체에 빨간색

테두리가 나타났을 때 클릭하세요.

해당 UI 객체가 대상 UI 객체로 지정됩니다.

그림 4-86 Click 속성 화면



번호	설명
①	대상 UI 객체를 찾는 시간을 초 단위로 입력합니다.
②	대상 UI 객체를 찾은 후 클릭할 위치를 대상 UI 객체의 왼쪽 상단 기준 좌표 값으로 입력합니다. 마우스로 직접 지정한 경우에는 자동으로 입력됩니다.
③	대상 UI 객체의 타입, 서브타입, ID, Name 정보가 나타납니다.
④	대상 UI 객체의 텍스트 값이 나타납니다.
⑤	사용자가 지정한 대상 UI 객체의 이미지가 나타납니다.
⑥	사용자가 지정한 대상 UI 객체의 위치 및 크기가 화면 기준 좌표로 나타납니다.

### DoubleClick(더블클릭하기)

대상 UI 객체를 더블클릭합니다.

그림 4-87 DoubleClick 속성 화면

번호	설명
①	대상 UI 객체를 찾는 시간을 초 단위로 입력합니다.
②	대상 UI 객체를 찾은 후 더블클릭할 위치를 대상 UI 객체의 왼쪽 상단 기준 좌표 값으로 입력합니다. 마우스로 직접 지정한 경우에는 자동으로 입력됩니다.
③	대상 UI 객체의 타입, 서브타입, ID, Name 정보가 나타납니다.
④	대상 UI 객체의 텍스트 값이 나타납니다.
⑤	사용자가 지정한 대상 UI 객체의 이미지가 나타납니다.
⑥	사용자가 지정한 대상 UI 객체의 위치 및 크기가 화면 기준 좌표로 나타납니다.

#### RightClick(마우스 오른쪽 버튼으로 클릭하기)

대상 UI 객체를 마우스 오른쪽 버튼으로 클릭합니다.

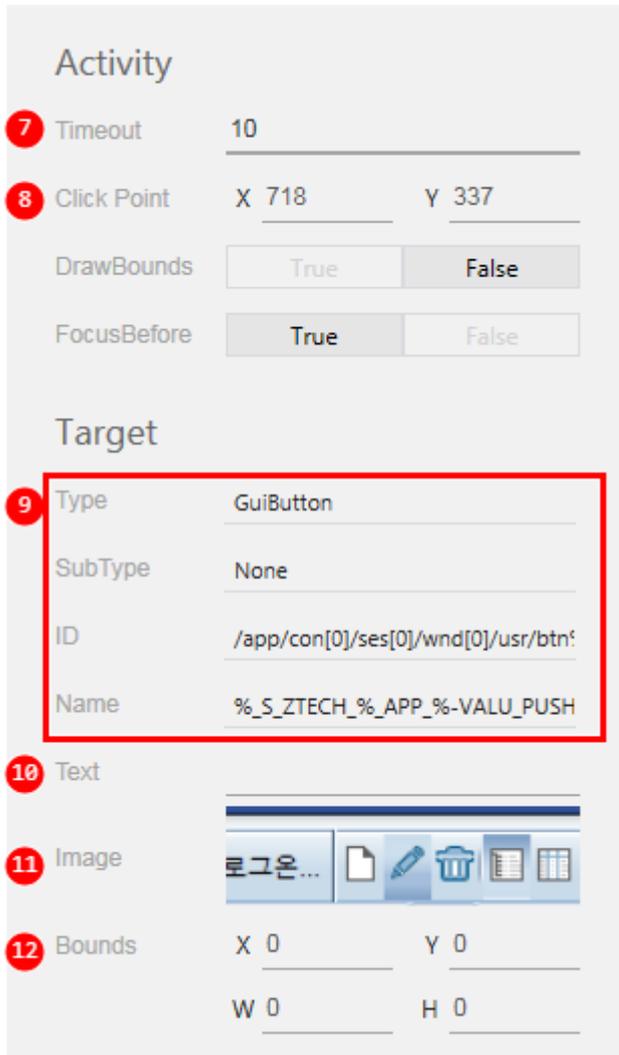
그림 4-88 RightClick 속성 화면

번호	설명
①	대상 UI 객체를 찾는 시간을 초 단위로 입력합니다.
②	대상 UI 객체를 찾은 후 클릭할 위치를 대상 UI 객체의 왼쪽 상단 기준 좌표 값으로 입력합니다. 마우스로 직접 지정한 경우에는 자동으로 입력됩니다.
③	대상 UI 객체의 타입, 서브타입, ID, Name 정보가 나타납니다.
④	대상 UI 객체의 텍스트 값이 나타납니다.
⑤	사용자가 지정한 대상 UI 객체의 이미지가 나타납니다.
⑥	사용자가 지정한 대상 UI 객체의 위치 및 크기가 화면 기준 좌표로 나타납니다.

### MouseHover(마우스 포인터 올리기)

마우스 포인터를 대상 UI 객체 위에 올려놓습니다.

그림 4-89 MouseHover 속성 화면

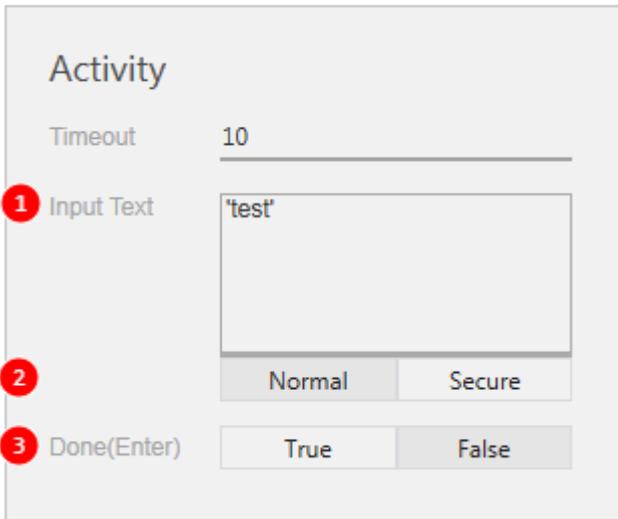


번호	설명
①	대상 UI 객체를 찾는 시간을 초 단위로 입력합니다.
②	대상 UI 객체를 찾은 후 마우스 포인터를 올려놓을 위치를 대상 UI 객체의 왼쪽 상단 기준 좌표 값으로 입력합니다. 마우스로 직접 지정한 경우에는 자동으로 입력됩니다.
③	대상 UI 객체의 타입, 서브타입, ID, Name 정보가 나타납니다.
④	대상 UI 객체의 텍스트 값이 나타납니다.
⑤	사용자가 지정한 대상 UI 객체의 이미지가 나타납니다.
⑥	사용자가 지정한 대상 UI 객체의 위치 및 크기가 화면 기준 좌표로 나타납니다.

### SetText(텍스트 입력하기)

대상 UI 객체에 텍스트를 입력합니다. 텍스트 박스와 같이 대상 UI 객체가 ValuePattern을 지원하고 읽기 전용이 아닌 경우에만 사용할 수 있습니다.

그림 4-90 **SetText** 속성 화면



번호	설명
①	입력할 텍스트를 작은따옴표 안에 입력합니다.
②	입력한 텍스트를 암호화할지 설정합니다.
③	텍스트를 입력한 후 Enter 키를 입력할지 설정합니다.

### GetText(텍스트 읽어오기)

대상 UI 객체의 텍스트를 읽어옵니다.

그림 4-91 **GetText** 속성 화면



번호	설명
①	대상 UI 객체에서 읽어온 텍스트를 할당할 변수를 지정합니다.
②	대상 UI 객체에서 읽어온 텍스트가 나타납니다.

### IsExist(존재 여부 검사하기)

대상 UI 객체가 화면에 있는지 검사합니다.

Timeout 항목에 입력한 시간 내에 화면에서 대상 UI 객체를 찾을 경우 True를, 찾지 못할 경우 False를 반환합니다.

### WaitSAPAppear(나타날 때까지 기다리기)

대상 UI 객체가 화면에 나타날 때까지 기다립니다.

화면 전환 시 유용하게 사용할 수 있습니다.

### WaitSAPDisappear(사라질 때까지 기다리기)

대상 UI 객체가 화면에서 사라질 때까지 기다립니다.

화면 전환 시 유용하게 사용할 수 있습니다.

### GetGridColumnID(GridColumnID 가져오기)

영역의 ColumnID를 가져옵니다.

그림 4-92 GetGridColumnID 속성 화면

번호	설명
①	ColumnID를 가져올 컬럼명을 입력합니다.
②	지정한 대상 UI 객체가 타이틀인지 여부를 선택합니다.

### GetGridRowCount(행 개수 가져오기)

행의 개수를 가져올 수 있습니다.

### GetGridColumnCount(열 개수 가져오기)

열의 개수를 가져올 수 있습니다.

### SelectComboItem(ComboItem 값 선택하기)

콤보 박스 형태의 UI 객체에서 원하는 값을 선택할 수 있습니다.

그림 4-93 SelectComboItem 속성 화면

Activity									
Timeout	10								
SAP									
Type	GuiComboBox								
SubType	None								
ID	/app/con[0]/ses[0]/wnd[0]/usr/cmt								
Name	GV_SCN100_KOART								
Text	Sale								
SelectIndex	0								
Image	<div style="border: 1px solid gray; padding: 2px;"> <span style="border: 1px solid gray; padding: 1px;">ZUIB, Ub, U /</span> </div> <div style="border: 1px solid gray; padding: 2px;"> <span style="border: 1px solid gray; padding: 1px;">Sale</span> </div>								
Bounds	<table border="0"> <tr> <td>X</td><td>2384</td><td>Y</td><td>422</td></tr> <tr> <td>W</td><td>87</td><td>H</td><td>20</td></tr> </table>	X	2384	Y	422	W	87	H	20
X	2384	Y	422						
W	87	H	20						

SelectIndex 항목에 선택할 아이템의 index를 입력하세요. index는 0부터 시작합니다.

### SearchInGrid(Grid 특정 항목 찾기)

SAP Grid 에서 특정 항목을 검색합니다.

### GetGridData(Grid 데이터 가져오기)

SAP Grid의 모든 데이터를 가져옵니다.

### SAPRunMacro(Macro 실행하기)

SAP Macro 를 실행합니다.

## 이미지 인식 라이브러리

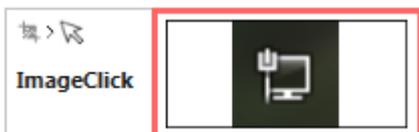
**ImageRecognition** 라이브러리에서 제공하는 기능을 이용해 이미지를 대상 UI로 지정할 수 있습니다. 사용자가 이미지를 캡처하면 해당 이미지와 동일한 UI 요소를 인식하여 라이브러리의 동작을 수행합니다. 이미지의 특정 위치를 클릭하거나, 이미지 안의 문자를 텍스트로 변환하는 등의 동작을 수행할 수 있습니다.

이미지 인식 라이브러리에서 사용할 수 있는 기능은 다음과 같습니다.

- 마우스 동작 기능(ImageClick, ImageHover, ImageRightClick, ImageDoubleClick)
- OCR 관련 기능(GetImageText, GetImageTextInfo, GetImageTable, TextClick, TextDoubleClick, TextRightClick, TextHover)
- 이미지 유무 확인 기능(IsExist, WaitImageAppear, WaitImageDisappear, MatchImages)
- 이미지 캡처 기능(CaptureScreen, CaptureArea, CaptureWindow)
- 이미지 컬러 관련 기능(GetImageColor)

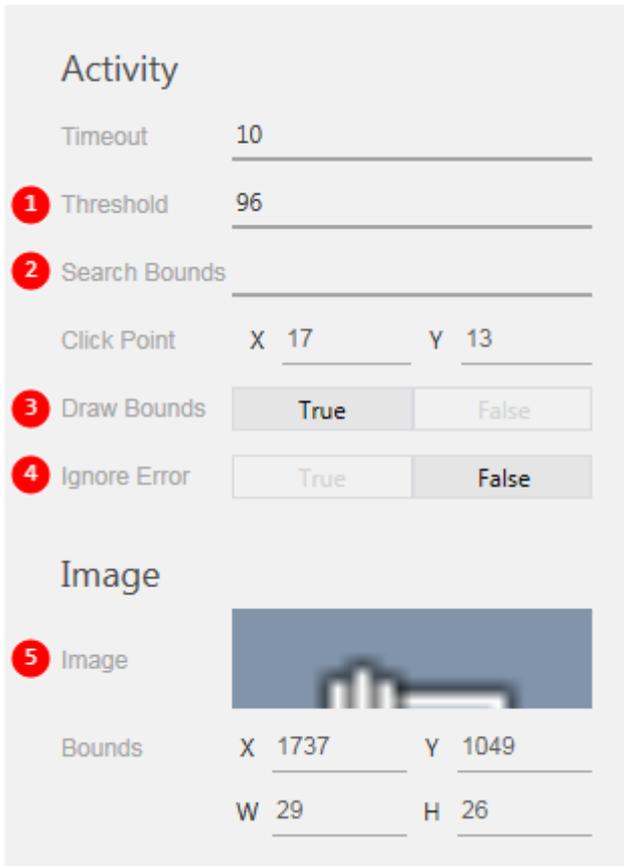
### ImageClick(이미지 클릭하기)

이미지 위의 특정 위치를 클릭할 수 있습니다.



1. 노드에서 **No Target > Double click to select**를 더블클릭하세요.
2. 원하는 영역을 드래그해 캡처하세요.
3. 클릭할 위치를 선택하세요.

그림 4-94 ImageClick 속성 화면

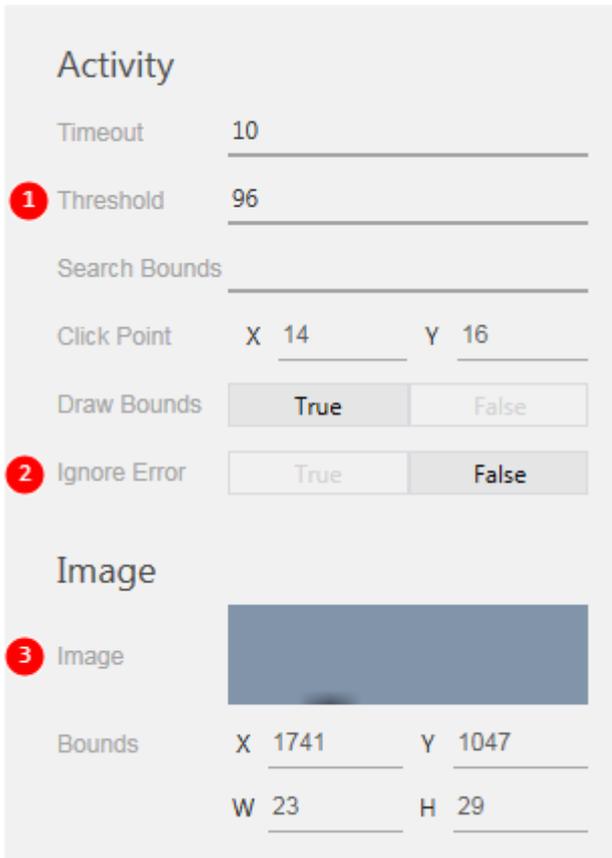


번호	설명
①	0~100 사이의 값을 입력해 이미지의 유사도를 설정할 수 있습니다. 0에 가까울수록 일치하지 않을 확률이 높아집니다.
②	전체 영역이 아닌 주어진 영역 내에서 검색하도록 설정합니다.
③	이미지 영역에 빨간색 테두리를 그릴지 여부를 설정합니다.
④	오류가 발생해도 프로세스나 작업을 계속 진행하도록 설정합니다.
⑤	캡처한 이미지가 나타납니다. 이미지를 더블클릭하면 ImageEditor로 이미지 설정을 변경할 수 있습니다. <ul style="list-style-type: none"> <li>• <b>블러오기:</b> 기존에 캡처한 이미지를 블러웁니다.</li> <li>• <b>잘라내기:</b> 캡처된 이미지를 잘라냅니다.</li> <li>• <b>적용:</b> 클릭할 위치를 다시 선택합니다.</li> <li>• <b>초기화:</b> 변경한 설정을 초기화합니다.</li> </ul>

### ImageHover(이미지 위에 마우스 포인터 올리기)

마우스 포인터를 이미지의 특정 위치에 올려놓습니다.

그림 4-95 ImageHover 속성 화면

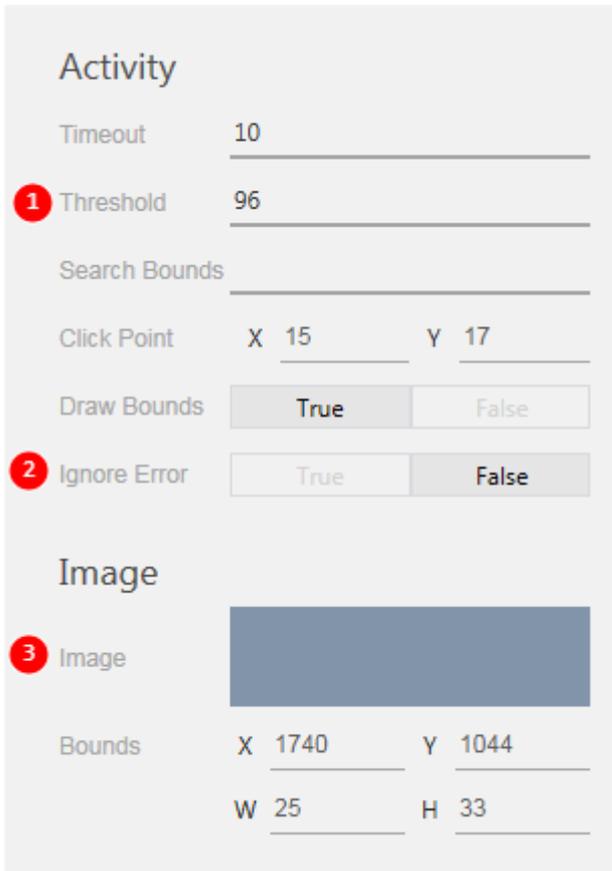


번호	설명
1	0~100 사이의 값을 입력해 이미지의 유사도를 설정할 수 있습니다. 0에 가까울수록 일치하지 않을 확률이 높아집니다.
2	오류가 발생해도 프로세스나 작업을 계속 진행하도록 설정합니다.
3	캡처한 이미지가 나타납니다. 이미지를 더블클릭하면 ImageEditor로 이미지 설정을 변경할 수 있습니다. <ul style="list-style-type: none"> <li>• <b>블러오기:</b> 기존에 캡처한 이미지를 블러웁니다.</li> <li>• <b>잘라내기:</b> 캡처된 이미지를 잘라냅니다.</li> <li>• <b>적용:</b> 클릭할 위치를 다시 선택합니다.</li> <li>• <b>초기화:</b> 변경한 설정을 초기화합니다.</li> </ul>

### ImageRightClick(마우스 오른쪽 버튼으로 이미지 클릭하기)

이미지의 특정 위치를 마우스 오른쪽 버튼으로 클릭합니다.

그림 4-96 ImageRightClick 속성 화면

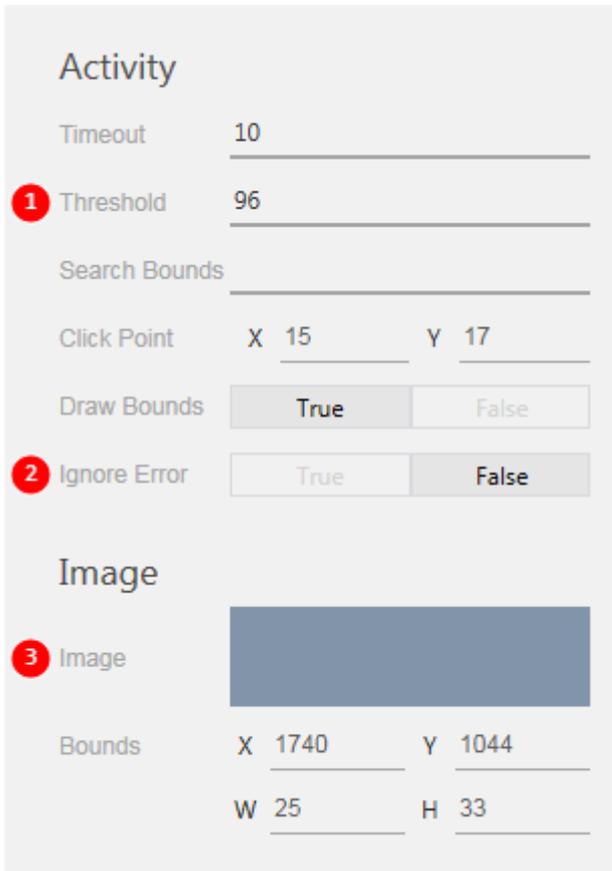


번호	설명
1	0~100 사이의 값을 입력해 이미지의 유사도를 설정할 수 있습니다. 0에 가까울수록 일치하지 않을 확률이 높아집니다.
2	오류가 발생해도 프로세스나 작업을 계속 진행하도록 설정합니다.
3	캡처한 이미지가 나타납니다. 이미지를 더블클릭하면 ImageEditor로 이미지 설정을 변경할 수 있습니다. <ul style="list-style-type: none"> <li>• 불러오기: 기존에 캡처한 이미지를 불러옵니다.</li> <li>• 잘라내기: 캡처된 이미지를 잘라냅니다.</li> <li>• 적용: 클릭할 위치를 다시 선택합니다.</li> <li>• 초기화: 변경한 설정을 초기화합니다.</li> </ul>

### ImageDoubleClick(이미지 더블클릭하기)

이미지의 특정 위치를 더블클릭합니다.

그림 4-97 ImageDoubleClick 속성 화면

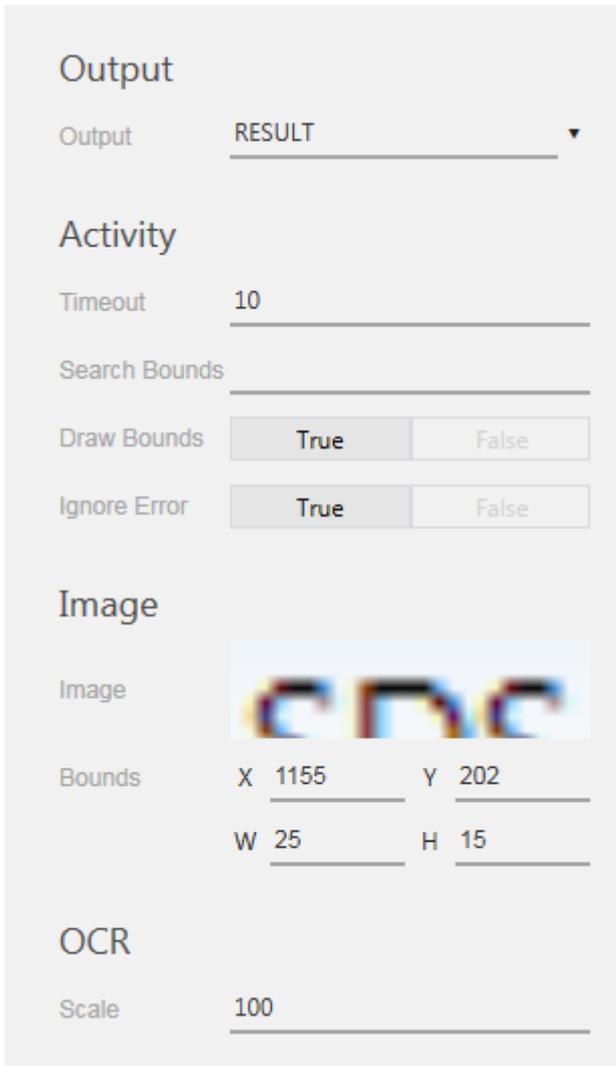


번호	설명
1	0~100 사이의 값을 입력해 이미지의 유사도를 설정할 수 있습니다. 0에 가까울수록 일치하지 않을 확률이 높아집니다.
2	오류가 발생해도 프로세스나 작업을 계속 진행하도록 설정합니다.
3	캡처한 이미지가 나타납니다. 이미지를 더블클릭하면 ImageEditor로 이미지 설정을 변경할 수 있습니다. <ul style="list-style-type: none"> <li>• 불러오기: 기존에 캡처한 이미지를 불러옵니다.</li> <li>• 잘라내기: 캡처된 이미지를 잘라냅니다.</li> <li>• 적용: 클릭할 위치를 다시 선택합니다.</li> <li>• 초기화: 변경한 설정을 초기화합니다.</li> </ul>

### GetImageText(이미지에서 텍스트 읽어오기)

이미지에 나타난 문자를 텍스트로 변환합니다.

그림 4-98 GetImageText 속성 화면



#### GetImageTextInfo(텍스트 정보 읽어오기)

이미지를 인식해 텍스트를 읽어와 이미지 정보(높이, 너비 등)와 같이 반환합니다.

#### GetImageTable(테이블 텍스트 읽어오기)

이미지에서 테이블을 인식해 텍스트를 읽어옵니다.

그림 4-99 GetImageTable 속성 화면

The screenshot shows a configuration window for the GetImageTable activity. It is organized into several sections:

- Output:** A dropdown menu is set to "RESULT".
- Activity:** Includes a "Timeout" field set to "10", a "Search Bounds" field, and two toggle buttons for "Draw Bounds" and "Ignore Error", both of which are currently set to "True".
- Image:** Displays a preview of a black box with the white text "All rights". Below the preview, the "Bounds" are defined with four fields: X (1536), Y (331), W (90), and H (25).
- OCR:** Includes a "Scale" field set to "100".

### TextClick(텍스트 클릭하기)

이미지에서 특정 텍스트를 찾아 클릭할 수 있습니다.

그림 4-100 TextClick 속성 화면

**Output**

Output

**Activity**

Timeout

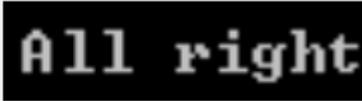
Search Bounds

Index

Draw Bounds

Ignore Error

**Image**

Image 

Bounds X  Y   
W  H

**OCR**

Scale

Target Text

클릭할 텍스트를 TargetText 항목의 작은따옴표 안에 입력하세요.

#### TextDoubleClick(텍스트 더블클릭하기)

이미지에서 특정 텍스트를 찾아 더블클릭합니다.

그림 4-101 TextDoubleClick 속성 화면

The image shows a configuration panel for the TextDoubleClick property. It is organized into several sections:

- Output:** A dropdown menu set to "RESULT".
- Activity:** Includes a "Timeout" field set to "10", and "Search Bounds" and "Index" fields.
- Draw Bounds:** A toggle button set to "True".
- Ignore Error:** A toggle button set to "False".
- Image:** Shows a preview of the text "All rights" on a black background. Below it, "Bounds" are defined with X: 1537, Y: 331, W: 89, and H: 24.
- OCR:** A "Scale" field set to "100".
- Target Text:** An empty text input field, which is highlighted with a red rectangular border.

더블클릭할 텍스트를 TargetText 항목의 작은따옴표 안에 입력하세요.

#### TextRightClick(텍스트 오른쪽 클릭하기)

이미지에서 특정 텍스트를 찾아 오른쪽 버튼으로 클릭합니다.

그림 4-102 TextRightClick 속성 화면

**Output**

Output RESULT ▼

**Activity**

Timeout 10

Search Bounds \_\_\_\_\_

Index \_\_\_\_\_

Draw Bounds  True  False

Ignore Error  True  False

**Image**

Image 

Bounds X 1537 Y 333  
W 87 H 23

**OCR**

Scale 100

Target Text

클릭할 텍스트를 TargetText 항목의 작은따옴표 안에 입력하세요.

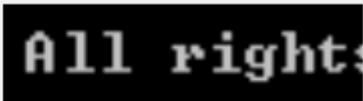
#### TextHover(텍스트 위에 마우스 포인터 올리기)

---

이미지에서 특정 텍스트를 찾아 마우스 포인터를 올려줍니다.

그림 4-103 TextHover 속성 화면

The screenshot shows the TextHover property editor with the following settings:

- Output:** RESULT
- Activity:**
  - Timeout: 10
  - Search Bounds: (empty)
  - Index: (empty)
  - Draw Bounds:  True  False
  - Ignore Error:  True  False
- Image:**
  - Image: 
  - Bounds: X 1537, Y 332, W 88, H 22
- OCR:**
  - Scale: 100
  - Target Text:  (highlighted with a red box)

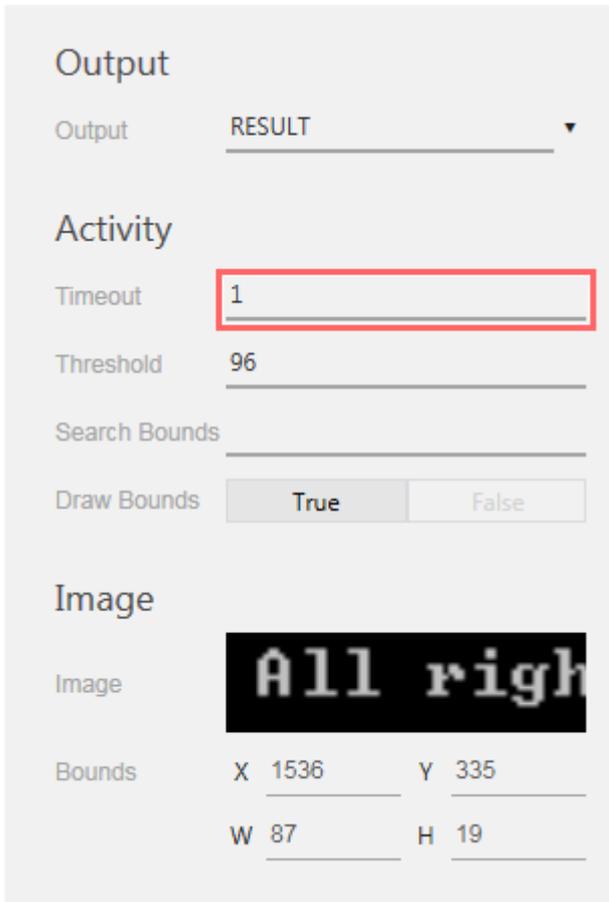
마우스 포인터를 올려놓을 텍스트를 TargetText 항목의 작은따옴표 안에 입력하세요.

#### IsExist(존재 여부 검사하기)

특정 이미지가 화면에 있는지 검사합니다.

Timeout 항목에 입력한 시간 내에 화면에서 해당 이미지를 찾을 경우 True를, 찾지 못할 경우 False를 반환합니다.

그림 4-104 IsExist 속성 화면



### WaitImageAppear(이미지가 나타날 때까지 기다리기)

특정 이미지가 화면에 나타날 때까지 기다립니다.

**Timeout** 항목에 입력한 시간이 지나면 **False**를 반환합니다.

그림 4-105 WaitImageAppear 속성 화면



**Output**

Output RESULT ▼

**Activity**

Timeout 10

Threshold 96

Search Bounds \_\_\_\_\_

Draw Bounds  True  False

Ignore Error  True  False

**Image**

Image 

Bounds X 1537 Y 331  
W 86 H 25

### WaitImageDisappear(이미지가 사라질 때까지 기다리기)

특정 이미지가 화면에서 사라질 때까지 기다립니다.

**Timeout** 항목에 입력한 시간이 지나면 False를 반환합니다.

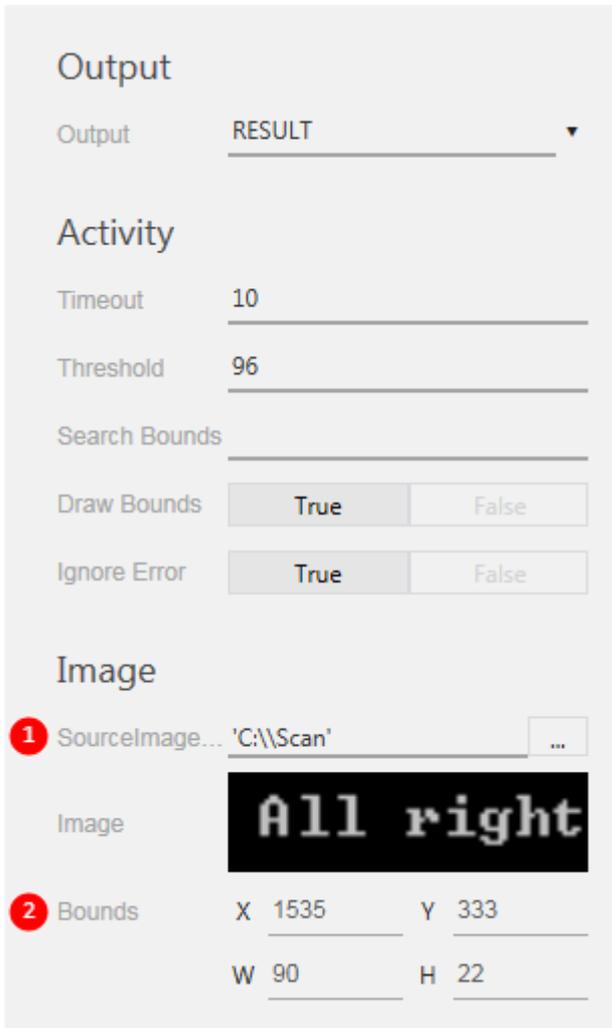
그림 4-106 WaitImageDisappear 속성 화면



### MatchImages(이미지 매칭하기)

지정한 영역 안에서 특정 폴더 내의 이미지와 매칭되는 이미지 파일을 찾아 파일명을 반환합니다. 매칭되는 이미지가 없을 경우 빈 문자열을 반환합니다.

그림 4-107 MatchImages 속성 화면

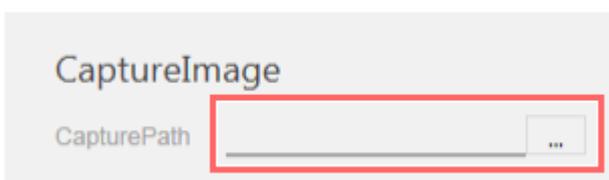


번호	설명
1	비교할 이미지 파일이 있는 경로를 선택합니다.
2	이미지를 비교할 영역을 지정합니다.

### CaptureScreen(전체 화면 캡처하기)

현재 로컬 PC의 전체 화면을 캡처할 수 있습니다.

그림 4-108 CaptureScreen 속성 화면

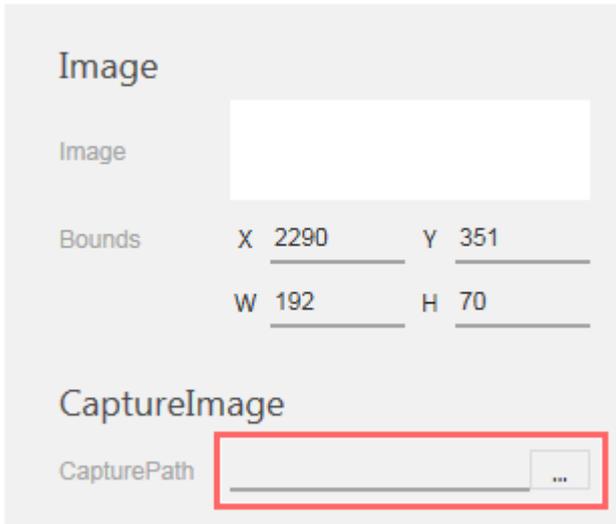


CapturePath 항목에서 경로를 선택하면 해당 경로에 캡처 이미지가 저장됩니다.

### CaptureArea(특정 영역 캡처하기)

PC 화면의 특정 영역을 지정하여 캡처할 수 있습니다.

그림 4-109 CaptureArea 속성 화면

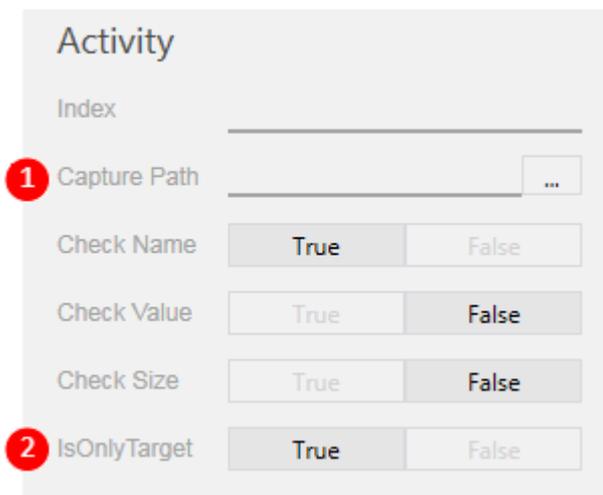


CapturePath 항목에서 경로를 선택하면 해당 경로에 캡처 이미지가 저장됩니다.

### CaptureWindow(특정 윈도우 캡처하기)

PC 화면의 특정 객체 또는 윈도우를 지정하여 캡처할 수 있습니다.

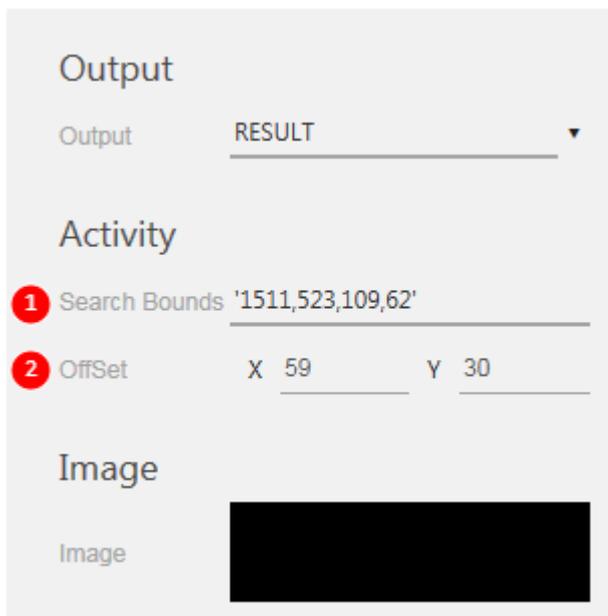
그림 4-110 CaptureWindow 속성 화면



번호	설명
①	이미지 파일을 저장할 경로를 선택합니다.
②	True 인 경우 지정한 객체를 이미지 파일로 저장하며, False 인 경우 지정한 객체가 포함된 윈도우를 이미지 파일로 저장합니다.

### GetImageColor(이미지 색상 가져오기)

이미지에서 지정된 좌표의 색상을 가져올 수 있습니다.



번호	설명
①	색상을 가져올 이미지를 선택합니다.
②	색상을 가져올 좌표를 입력합니다.

## 시스템 라이브러리

System 라이브러리에서 제공하는 기능을 이용해 시스템을 제어할 수 있습니다.

시스템 라이브러리에서 사용할 수 있는 기능은 다음과 같습니다.

- 파일 제어(CreateFile, ExecuteFile, CloseFile, CopyFile, DeleteFile, IsFileExist, GetFileList, MoveFile, RenameFile)
- 디렉토리 제어(CreateDirectory, CopyDirectory, DelectDirectory, IsDirectoryExist, GetDirectoryList, GetSpecialFolder)
- 텍스트 파일 제어(ReadTextFile, WriteTextFile)
- 입력 박스, 메시지 박스 표시(InputBox, MessageBox)

- 편집 관련 기능(GetClipboard, SetClipboard, ClipboardToFile, Copy(Ctrl+C), Cut(Ctrl+X), Paste(Ctrl+V), SelectAll(Ctrl+A), Screenshot(Alt+PrtSc))
- 프로그램 실행(ExecuteCmd)
- 경과 시간 계산(TimeStamp)
- 레지스트리 관련 기능(GetRegistryValue, SetRegistryValue)

### CreateFile(파일 생성하기)

---

지정한 경로의 파일을 생성합니다.

txt, xml, html 등 텍스트 포맷의 파일만 생성이 가능하며 엑셀, 워드 문서 등과 같이 특정 앱에 의해 작성되는 파일 포맷은 생성할 수 없습니다.

### ExecuteFile(파일 실행하기)

---

지정한 파일을 실행합니다.

지정한 파일의 확장자에 따라 시스템에 지정된 앱이 실행되고, 실행된 앱의 PID를 반환합니다.

### CloseFile(파일 종료하기)

---

ExecuteFile 라이브러리로 실행한 후 반환된 PID를 이용해 실행 중인 파일을 종료합니다.

### CopyFile(파일 복사하기)

---

지정한 파일을 복사합니다.

### DeleteFile(파일 삭제하기)

---

지정한 파일을 삭제합니다.

### IsFileExist(존재 여부 검사하기)

---

지정한 파일이 존재하는지 확인합니다.

### GetFileList(파일 목록 가져오기)

---

지정한 폴더의 파일 목록을 가져옵니다.

### CreateDirectory(디렉토리 생성하기)

---

지정한 디렉토리를 생성합니다.

### CopyDirectory(디렉토리 복사하기)

---

지정한 디렉토리를 복사합니다.

### DeleteDirectory(디렉토리 삭제하기)

---

지정한 디렉토리를 삭제합니다.

### IsDirectoryExist(디렉토리 존재 여부 검사하기)

---

지정한 디렉토리가 존재하는지 확인합니다.

### GetDirectoryList(디렉토리 리스트 가져오기)

---

지정한 디렉토리의 하위 디렉토리 리스트를 가져옵니다.

### GetSpecialFolder(사용자 폴더 경로 가져오기)

---

내 문서, 바탕 화면 등 사용자 폴더의 경로를 가져옵니다.

### ReadTextFile(텍스트 파일 내용 가져오기)

---

텍스트 파일을 읽어 문자열로 가져옵니다.

## WriteTextFile(텍스트 파일 쓰기)

문자열을 지정된 경로의 텍스트 문서에 입력하여 저장합니다.

## InputBox(입력 박스 표시하기)

화면에 사용자 입력을 받을 수 있는 입력 박스를 표시합니다.

그림 4-111 InputBox 속성 화면



**Variable** 항목에 입력한 값을 저장할 변수를 지정하세요.  
사용자가 값을 입력하지 않으면 지정된 변수의 값이 문자열로 변환되어 반환됩니다.

**InputBox** 는 IPA Designer에서 수행할 경우에만 동작합니다.  
시나리오 작성 시, 디버깅을 위해 사용할 수 있습니다.

## MessageBox(메시지 박스 표시하기)

화면에 메시지 박스를 표시합니다.

그림 4-112 MessageBox 속성 화면



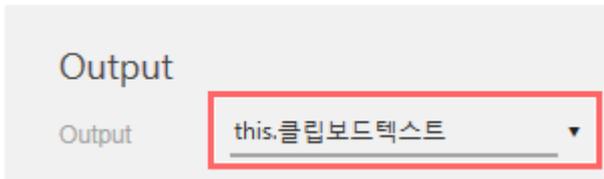
**Arguments** 항목에 메시지 박스에 표시할 값을 작은따옴표 안에 입력하세요.

**MessageBox** 로 출력하는 값은 IPA Designer에서 수행할 경우에만 동작합니다.  
시나리오 작성 시, 디버깅을 위해 사용할 수 있습니다.

## GetClipboard(클립보드 가져오기)

시스템 클립보드에 저장된 값을 가져옵니다.

그림 4-113 GetClipboard 속성 화면



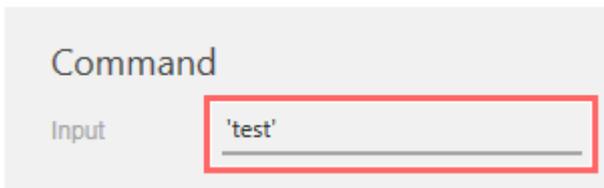
**Output** 항목에 클립보드에서 읽은 값을 저장하기 위한 변수를 지정하세요.

### SetClipboard(클립보드에 복사하기)

입력한 값을 시스템의 클립보드에 저장합니다.

키보드의 Ctrl+C 단축키와 동일한 효과를 가지므로 저장된 값은 붙여넣기 기능에 사용할 수 있습니다.

그림 4-114 SetClipboard 속성 화면

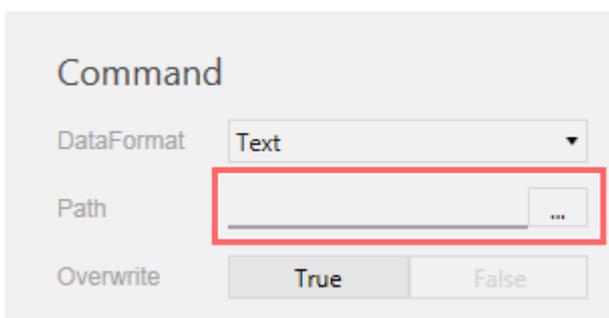


**Input** 항목에 클립보드에 저장할 값을 작은따옴표 안에 입력하세요.

### ClipboardToFile(클립보드 내용 파일로 쓰기)

이미지, 텍스트 등 클립보드에 있는 내용을 파일로 저장합니다.

그림 4-115 ClipboardToFile 속성 화면



**Path** 항목에서 파일을 저장할 경로를 선택하세요.

### Copy(Ctrl+C)(복사하기)

---

현재 포커스를 가진 화면을 대상으로 복사하기 이벤트를 발생시킵니다.  
키보드의 Ctrl+C 단축키와 동일한 효과를 가집니다.

### Cut(Ctrl+X)(잘라내기)

---

현재 포커스를 가진 화면을 대상으로 잘라내기 이벤트를 발생시킵니다.  
키보드의 Ctrl+X 단축키와 동일한 효과를 가집니다.

### Paste(Ctrl+V)(붙여넣기)

---

현재 포커스를 가진 화면을 대상으로 붙여넣기 이벤트를 발생시킵니다.  
키보드의 Ctrl+V 단축키와 동일한 효과를 가집니다.

### SelectAll(Ctrl+A)(전체 선택하기)

---

현재 포커스를 가진 화면을 대상으로 모두 선택 이벤트를 발생시킵니다.  
키보드의 Ctrl+A 단축키와 동일한 효과를 가집니다.

### Screenshot(Alt+PrtSc)(화면 캡처하기)

---

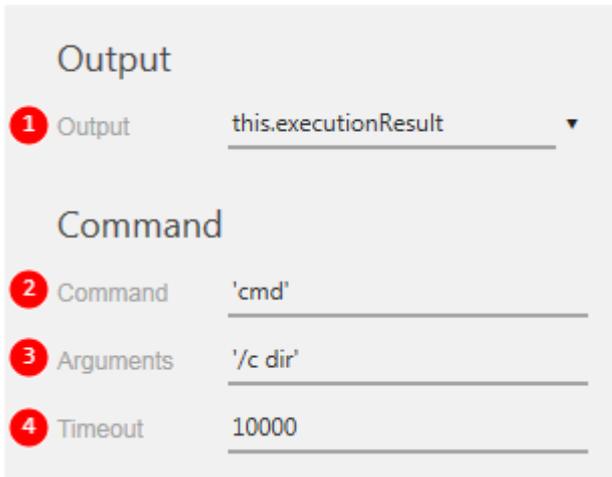
현재 포커스를 가진 화면을 캡처합니다.

### ExecuteCmd(백그라운드에서 프로그램 실행하기)

---

백그라운드에서 프로그램 명령을 실행한 후, 실행 결과를 반환합니다.  
화면에 보이지 않는 상태로 실행됩니다.

그림 4-116 ExecuteCmd 속성 화면



번호	설명
1	대상 프로그램의 실행 결과를 저장할 변수를 지정합니다. 반환값은 명령 수행 시 발생하는 표준 출력(Standard Output) 문자열이 됩니다.
2	실행할 명령어를 작은따옴표 안에 입력합니다.
3	실행 명령어에 필요한 인자를 작은따옴표 안에 입력합니다.
4	명령이 수행 완료될 때까지 대기하는 시간을 입력합니다. (ms 단위)

### TimeStamp(경과 시간 계산하기)

지정한 시간부터 현재 시간까지의 경과 시간을 밀리세컨드(millisecond) 단위로 반환합니다.

그림 4-117 TimeStamp 속성 화면



번호	설명
1	경과 시간을 저장하기 위한 변수를 설정합니다.
2	경과 시간을 계산할 시작 시간을 입력합니다.

### GetRegistryValue(레지스트리 값 가져오기)

---

지정한 레지스트리 경로의 키 값을 가져옵니다.

그림 4-118 GetRegistryValue 속성 화면

The screenshot shows a configuration window for the GetRegistryValue command. It has two main sections: 'Output' and 'Command'. In the 'Output' section, there is a dropdown menu labeled 'Output' with 'RESULT' selected. In the 'Command' section, there are three fields: 'BaseKey' is a dropdown menu with 'ClassesRoot' selected; 'KeyName' is a text input field containing an empty string; and 'ValueName' is a text input field containing an empty string.

### SetRegistryValue(레지스트리 값 입력하기)

---

지정한 레지스트리 경로에 키 값을 입력합니다.

그림 4-119 SetRegistryValue 속성 화면

The screenshot shows a configuration window for the SetRegistryValue command. It has one main section: 'Command'. It contains five fields: 'BaseKey' is a dropdown menu with 'ClassesRoot' selected; 'ValueKind' is a dropdown menu with 'String' selected; 'KeyName' is a text input field containing an empty string; 'ValueName' is a text input field containing an empty string; and 'Value' is a text input field containing an empty string.

### MoveFile(파일 이동하기)

---

지정한 경로의 파일을 이동합니다.

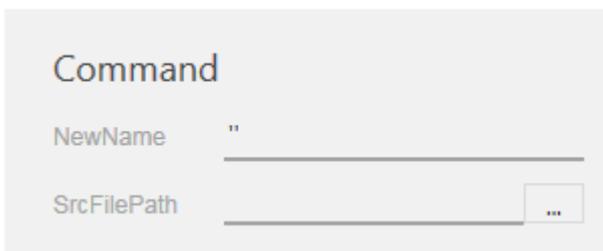
그림 4-120 MoveFile 속성 화면



### RenameFile(파일 이름 변경하기)

지정한 파일의 이름을 변경합니다.

그림 4-121 RenameFile 속성 화면



## FlowControl 라이브러리

FlowControl 라이브러리에서 제공하는 기능을 이용해 프로세스의 흐름을 제어할 수 있습니다.

반복 작업을 수행하거나, 조건에 따라 다른 작업을 설정하거나, 특정 조건으로 프로세스의 흐름을 늦추거나 종료할 수 있습니다.

FlowControl 라이브러리에서 사용할 수 있는 기능은 다음과 같습니다.

- 의사결정(If)
- 반복(For, Foreach, While)
- 변수할당(Assign)
- 사용자 스크립트 작성(ExecuteScript, ExecuteLibrary)
- 프로세스 컨트롤(Delay, TerminateFlow, TerminateAll, Break, Continue)
- 디버깅, 로깅(Debugger, Log, DumpScreen)
- 기타(RetryTask, Reboot, UploadResource, DownloadResource, TryCatch, DataConverter)

### If(의사 결정)

조건에 따라 다른 동작을 수행하는 True/False 분기문을 생성할 수 있습니다. JavaScript의 if/else와 같습니다.

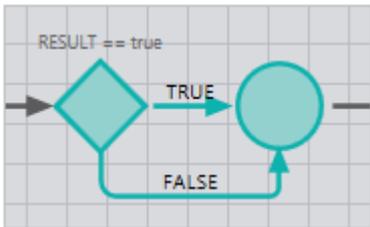
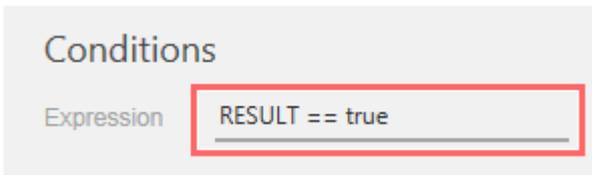


그림 4-122 If 속성 화면



**Expressin** 항목에 조건문을 입력하세요.  
 조건문은 &&나 || 같은 연산자를 이용해 여러 개 입력할 수 있습니다.

### For(반복)

Loop를 생성할 수 있습니다. JavaScript의 for loop와 같습니다.

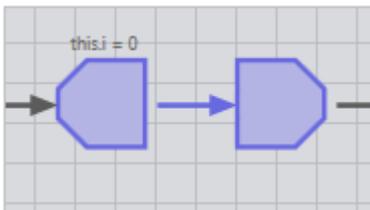
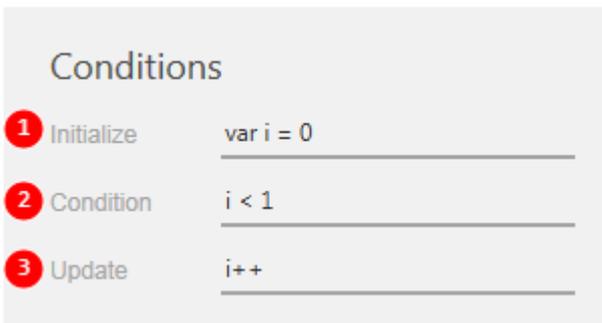


그림 4-123 For 속성 화면



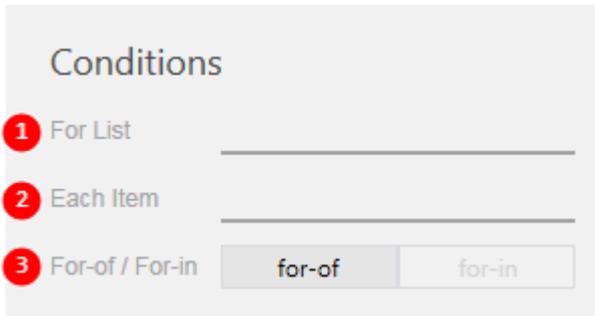
번호	설명
1	index의 초기값을 입력합니다.

번호	설명
②	index의 조건을 입력합니다.
③	index의 업데이트 내용을 입력합니다.

### Foreach(리스트 아이템 차례대로 가져오기)

ListItem을 차례대로 가져오는 Loop를 생성할 수 있습니다. JavaScript의 foreach와 같습니다.

그림 4-124 Foreach 속성 화면



번호	설명
①	데이터가 있는 리스트 타입의 변수를 입력합니다.
②	리스트 타입에서 아이템이 하나씩 할당될 변수를 입력합니다.
③	자바스크립트 구문과 동일한 목적으로 사용합니다.

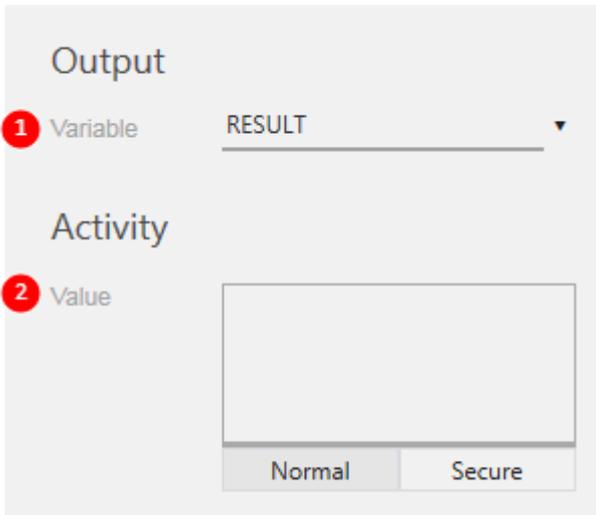
### While(특정 조건까지 수행하기)

특정 조건까지 수행되는 Loop를 생성할 수 있습니다. JavaScript의 while과 같습니다.

### Assign(변수 할당하기)

변수에 값을 할당할 수 있습니다.

그림 4-125 Assign 속성 화면

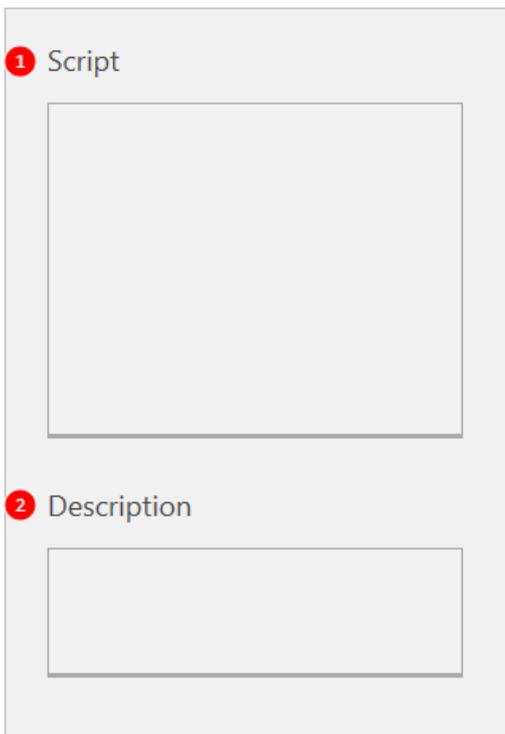


번호	설명
①	값이 할당될 변수를 지정합니다.
②	변수에 할당될 값을 입력합니다.

### ExecuteScript(사용자 정의 스크립트 입력하기)

사용자 정의 스크립트를 입력할 수 있습니다. 스크립트는 JavaScript 문법으로 작성해야 합니다.

그림 4-126 ExecuteScript 속성 화면



번호	설명
1	사용자 정의 스크립트를 입력합니다. 텍스트 상자를 더블클릭하면 ScriptEditor를 사용할 수 있습니다.
2	스크립트에 대한 설명을 작은따옴표 안에 입력합니다.

### ExecuteLibrary(스크립트 함수 실행하기)

사용자가 js 파일로 작성한 스크립트 함수를 실행할 수 있습니다.



• 스크립트는 아래와 같은 형태로 작성하세요.

```
Script.파일명 = new function(){
this.함수명 = function(파라미터 목록){
}
}
```

- 프로그램 설치 시 기본적인 스크립트 함수가 함께 제공됩니다. 자세한 내용은 프로그램 설치 경로의 Libraries 폴더의 .js 파일을 참고하세요.
- 작성한 스크립트 함수 파일은 내 문서의 프로젝트 폴더에 Libraries 폴더를 생성한 후 저장하세요.

### Delay(시간 지연시키기)

사용자가 정의한 시간만큼 지연시킬 수 있습니다.

그림 4-127 Delay 속성 화면

**Time**

1 Second

2 Millisecond

3 Description

번호	설명
----	----

번호	설명
①	초 단위 시간을 입력합니다.
②	밀리초 단위 시간을 입력합니다.
③	지연에 대한 설명을 작은따옴표 안에 입력합니다.

### TerminateFlow(플로우 종료하기)

동작하고 있는 플로우를 종료시킬 수 있습니다. JavaScript의 return과 같습니다.

### TerminateAll(모두 종료하기)

동작하고 있는 프로세스를 종료시킬 수 있습니다. JavaScript의 exit와 같습니다.

### Break(Loop 중지하기)

Loop 안에서 사용자가 정의한 시점에 Loop를 빠져나갈 수 있습니다. JavaScript의 break와 같습니다.

### Continue(Loop 계속 진행하기)

Loop 안에서 사용자가 정의한 시점에 다음 Loop cycle로 넘어 갈 수 있습니다. JavaScript의 continue와 같습니다.

### Debugger(디버그 콘솔 실행하기)

디버그 콘솔을 실행할 수 있습니다. 디버그 콘솔 안에서 사용자는 현재까지 진행된 내용의 변수 상태를 입력하여 확인할 수 있습니다.

### Log(로그 출력하기)

사용자 로그를 지정된 파일에 출력할 수 있습니다. 스크린샷이나 메시지 박스로 출력할 수도 있습니다.

그림 4-128 Log 속성 화면

### DumpScreen(덤프 스크린샷)

프로젝트 폴더 하위 Histories\Logs 폴더 아래 스크린샷과 윈도우 Hierarchy 정보가 저장되며 Fail으로 저장할 경우 서버로 저장된 정보가 전송됩니다.

그림 4-129 DumpScreen 속성 화면

### RetryTask(작업 재실행)

에러가 발생했을 때 OnError 이벤트에서 현재 진행 중이던 작업을 재시작할 수 있습니다.

그림 4-130 RetryTask 속성 화면

### Reboot(재부팅)

컴퓨터를 재시작합니다.

### UploadResource(리소스 업로드)

서버에 리소스(파일)를 업로드 합니다. 서버에 업로드 된 리소스(파일)는 리소스 ID가 반환되며, 이 ID로 다시 리소스(파일)를 업데이트하거나 다운로드 할 수 있습니다. 리소스 ID는 자동으로 생성되거나, 사용자가 입력 할 수 있습니다. 리소스(파일)은 보관기한이 있고, 보관기한이 지나면 서버에서 다운로드 할 수 없습니다.

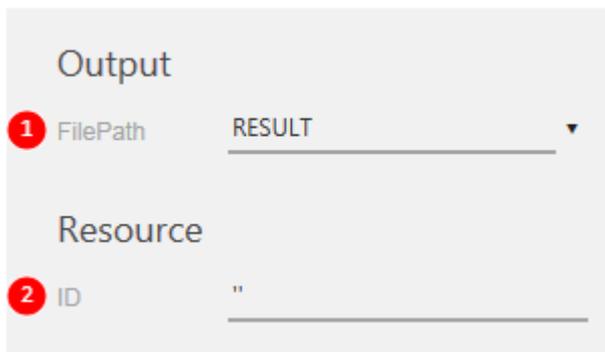
그림 4-131 UploadResource 속성 화면

번호	설명
1	서버에 등록된 ID 를 반환합니다. ID 를 설정하면 동일한 ID 를 받게 되고, 아닐 경우 서버에서 만든 ID 를 반환합니다.
2	지정한 ID 값으로 저장이 됩니다. (덮어쓰기)
3	서버에서 파일을 보관할 기간을 입력합니다.
4	서버에 업로드 할 파일 경로입니다.

### DownloadResource(리소스 다운로드)

서버에서 리소스(파일)를 다운로드 받을 수 있습니다. 다운로드 할 리소스의 리소스 ID가 입력되어야 다운로드 가능합니다. 다운로드 된 리소스(파일)는 프로젝트 폴더내의 SharedResources 폴더에 저장되고, 다운로드에 성공한 경우 파일 경로가 반환됩니다.

그림 4-132 DownloadResource 속성 화면

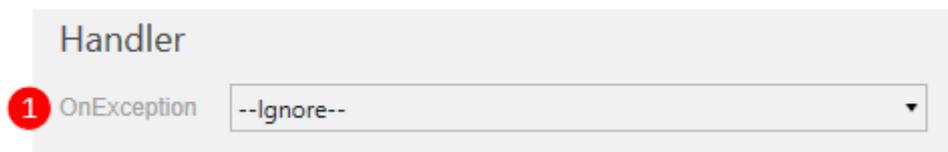


번호	설명
1	다운로드 된 파일의 경로입니다.
2	다운로드 할 파일의 ID 입니다.

### TryCatch(예외 처리)

스크립트 수행 중 예외 처리를 수행합니다. JavaScript의 Try Catch와 같습니다.

그림 4-133 TryCatch 속성 화면



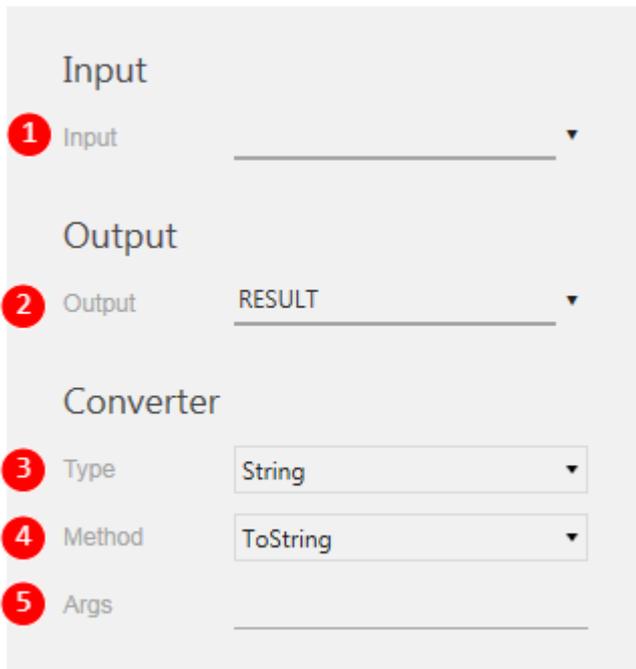
번호	설명
①	Exception 발생시 수행할 이벤트 핸들러입니다.

### DataConverter(데이터 컨버터)

입력된 데이터를 타입 변환 또는 데이터 추출을 통해 변환을 원하는 형태로 바꾸어 반환합니다.



그림 4-134 DataConverter 속성화면



번호	설명
①	DataConverter에 입력될 input 데이터
②	DataConverter에서 반환된 output 데이터
③	컨버터의 타입 목록 (String, Number, Date, Json, Xml)
④	타입별 사용가능한 메소드 목록 (표 아래에 타입별 메소드 설명)
⑤	메소드에 추가로 필요한 전달인자 (optional)

Type	Method	설명	사용예
String	ToString	입력된 데이터를 String타입으로 변환	
	Format	입력된 FormatString에 Arg에 입력한 값을 넣어 반환	Input: 'This is {0}'

Type	Method	설명	사용예
			Args: 'Sample' Output > 'This is Sample'
	Trim	입력된 데이터를 String타입으로 변환하고, 앞 뒤 공백을 제거	
	ReplaceAll	입력된 문자열에서 특정문자열을 찾아 바꿀문자열로 모두 변경	Args: 'aa', 'bb' (aa를 모두 bb로 변경)
	Split	입력된 문자열을 특정 문자로 나누어 배열에 저장	Args: ':' (:로 문자열을 나눔)
	ExtractByRegex	입력된 문자열에서 Args에 입력된 정규식에 일치하는 값을 추출하여 반환	
	IsMatch	입력된 문자열이 Args에 입력된 정규식에 일치하는지 True/False로 반환	
Number	ToNumber	입력된 데이터를 Number타입을 변환	
	Max	입력된 데이터와 Args에 입력된 숫자를 비교하여 큰값을 반환	
	Min	입력된 데이터와 Args에 입력된 숫자를 비교하여 작은값을 반환	
	Round	입력된 값을 소숫점 반올림하여 반환합니다. Arg에 숫자를 입력하면 그 숫자자리수에서 반올림됩니다.	
	Ceil	입력된 값을 소숫점 버림하여 반환합니다.	
	Floor	입력된 값을 소숫점 올림하여 반환합니다.	
	Abs	입력된 값의 절대값을 반환합니다.	
Date	ToDateTime	입력된 dateformat의 문자열을 DateTime으로 변환합니다. Args에 dateformat을 입력할 경우 그 포맷에 맞게 문자열로 반환합니다.	
	ToUTCDateTime	입력된 dateformat의 문자열을 UTCDateTime으로 변환합니다. Args에 dateformat을 입력할 경우 그 포맷에 맞게 문자열로 반환합니다.	
	FromExcelDate	입력된 엑셀에서 가져온 Date문자열을 UTCDateTime으로 변환합니다. Args에 dateformat을 입력할 경우 그 포맷에 맞게 문자열로 반환합니다.	
	AddYears, AddMonths, AddDays, AddHours, AddMinutes, AddSeconds	입력된 datetime에서 Args에 입력된 숫자만큼 시간 또는 날짜를 더하거나 뺄 수 있습니다.	
	DateFormat	입력된 dateformat에 맞게 문자열로 반환합니다.	
Json	ToString	Json객체를 문자열로 반환합니다.	
	ToJsonObject	Json형식의 문자열을 Json객체로 반환합니다.	
	GetValueByKey	Json형식의 문자열에서 Args에 입력한 key의 value를 반환합니다	
Xml	ToString	XmlElement를 문자열로 반환합니다.	

Type	Method	설명	사용예
	GetValueByXPath	Xml형식의 문자열에서 Args에 입력한 xpath의 value를 반환합니다.	

## Knox Portal 라이브러리

Knox Portal 라이브러리에서 제공하는 기능을 이용해 Knox Portal REST API 연계기능을 사용할 수 있습니다.

이 라이브러리를 사용하기 위해서는 디자이너가 로그인 상태여야 합니다.

Knox Portal 라이브러리에서 사용할 수 있는 기능은 다음과 같습니다.

- 메일전송 (SendMail)
- 결재상신, 결재상황조회 (SubmitApproval, GetApprovalStatus)
- 사용자, 부서정보 조회 (GetUserInfo, GetDeptInfo)
- PIMS기능 (CreateSchedule, DeleteSchedule, GetSchedulesInfo)

### SendMail(메일 보내기)

Knox Portal API를 이용해 메일을 전송합니다. 수신인은 최대 200명까지(참조, 비밀참조 포함) 가능하며, 첨부파일은 10개 이하, 총 10MB까지만 지원합니다.

그림 4-135 SendMail 속성 화면

**Mail**

Sender

Receivers

CCs

BCCs

Subject

Body

Attachments

IsHtml  True  False

## SubmitApproval(결재상신)

Knox Portal API를 이용해 결재를 상신합니다. 결재라인을 설정할 때는 아래 값 중에 선택 할 수 있으며, 병렬합의, 병렬결재의 경우 결재자의 순번이 같아야 합니다.

- Approval(결재)
- Consent(합의)
- PostApproval(후결)
- ParallelConsent(병렬합의)
- ParallelApproval(병렬결재)
- Notification(통보)

그림 4-136 SubmitApproval 속성 화면

The screenshot shows a web form for submitting an approval. It includes the following sections:

- Output:** A dropdown menu for 'ApprovalID' with 'RESULT' selected.
- Approval:**
  - 'Title' field with value 'sample title'
  - 'Content' field with value 'sample content'
  - 'Attachments' field with value 'Sample Attachment File Path1;Fi'
  - 'IsHtml' field with radio buttons for 'True' and 'False'.
- Approval Lines:** A table with columns 'Seq', 'Mail', and 'Type'. Below the table is a button that says 'Click here to add new item'.

## GetApprovalStatus(결재 상황 조회)

Knox Portal API를 이용해 결재상황을 조회합니다.

아래 값 중에서 현재 결재상황을 반환합니다.

- FailedEncrypt (암호화 실패)
- Encrypt(암호화중)
- Preliminary(예약상신)
- Postponed(보류)

- InProgress(진행중)
- Approved(완결)
- Rejected(반려)
- Canceled(상신취소)
- PreApproved(전결)
- PostApproved(후완결)

그림 4-137 GetApprovalStatus 속성 화면



### GetUserInfo(사용자 정보 조회)

KnoxPortal ID 또는 메일주소를 이용해 임직원 정보를 가져옵니다.

반환되는 타입은 `UserInfo[]` 이며, `UserInfo` 클래스의 멤버는 아래와 같습니다.

- UserId (KnoxPortal 아이디)
- Name (임직원 이름)
- EngName (임직원 영어이름)
- Department (부서이름)
- EngDepartment (부서 영어이름)
- DepartmentCode (부서코드)
- EmployeeNo (사번)
- EngTitle (영어 직급명)
- Title (직급명)
- Mail (메일주소)
- Mobile (핸드폰 번호)
- Telephone (전화번호)

그림 4-138 GetUserInfo 속성 화면

The screenshot shows a configuration window for the `GetUserInfo` function. It is divided into three sections:

- Output:** A dropdown menu is set to `RESULT`.
- User:** A dropdown menu is set to `ID`, and a search field contains an empty string.
- Knox:** Two radio buttons are present, with `False` selected.

### GetDeptInfo(부서 정보 조회)

KnoxPortal ID 또는 메일주소를 이용해 임직원의 부서정보를 가져옵니다.

반환되는 타입은 `DeptInfo[ ]` 이며, `DeptInfo` 클래스의 멤버는 아래와 같습니다.

- DeptName (부서명)
- EngDeptName (영어 부서명)
- DeptCode (부서코드)
- HighDeptName (상위 부서명)
- EngHighDeptName (영어 상위 부서명)
- HighDeptCode (상위 부서 코드)
- Company (회사명)
- EngCompany (영어 회사명)
- ManagerId (부서장 Knox ID)
- ManagerName (부서장 이름)

그림 4-139 GetDeptInfo 속성 화면

**Output**

DeptInfo RESULT ▼

**User**

Type ID ▼

Search "

**Knox**

IgnoreError  True  False

### CreateSchedule(일정 생성)

KnoxPortal PIMS 일정을 생성 합니다. 일정 생성에 성공하면 스케줄ID가 반환됩니다.

그림 4-140 CreateSchedule 속성 화면

**Output**

ScheduleID RESULT ▼

**PIMS**

Type ID ▼

Search "

Subject

Contents

Place

StartTime '2019-01-01 00:00:00'

EndTime '2019-01-01 01:00:00'

IsAllDay  True  False

## DeleteSchedule(일정 삭제)

KnoxPortal PIMS 일정을 삭제합니다. 삭제할 스케줄ID와 사용자ID가 필요합니다.

그림 4-141 DeleteSchedule 속성 화면

## GetSchedulesInfo(일정 조회)

KnoxPortal PIMS 일정을 조회합니다.

반환되는 타입은 ScheduleInfo[ ]로 ScheduleInfo클래스의 멤버는 아래와 같습니다.

- ScheduleId (스케줄 ID)
- Subject (일정 제목)
- Contents (일정 내용)
- StartTime (시작 시각)
- EndTime (종료 시각)
- AllDayYn (하루종일)
- Place (장소)
- Attendees (참석자)

그림 4-142 GetSchedulesInfo 속성 화면

## 스크립트 전용 라이브러리

스크립트를 작성해 프로젝트를 디자인할 경우 봇 객체 및 기본 스크립트 라이브러리를 활용할 수 있습니다.

### **ToString(obj)**

---

값 형태의 변수, 배열 및 2차원 배열의 정보를 문자열로 반환합니다.

### **GetObjectInfo(obj)**

---

객체 형태 변수의 멤버 데이터 정보를 반환합니다.

### **GetCount(obj)**

---

배열 형태 객체의 데이터 개수를 반환합니다.

### **Print(message), Debug(), Info(), Warn(), Error()**

---

로그 수준에 맞게 로그를 출력할 수 있습니다.

### **CharToInt(string)**

---

문자에 해당하는 코드값을 반환합니다.

### **IntToChar(number)**

---

코드값에 해당하는 문자를 반환합니다.

### **AddHostType(name, typeName)**

---

C#에서 제공하는 클래스를 등록하여 스크립트에서 사용할 수 있습니다.

예)

```
Bot.AddHostType( ' Process ' , ' System.Diagnostics.Process ' );
var proc = new Process();
Proc.Start( ' notepad.exe ' );
var pid =Process.GetProcessById(1234);
```

## CreateObject(typeName)

COM 객체를 생성합니다.

예)

```
var fs = Bot.CreateObject( " Scripting.FileSystemObject " );
var a = fs.CreateTextFile("c:\testfile.txt", true);
a.WriteLine( " This is a test. " );
```

## 라이브러리 공통 속성

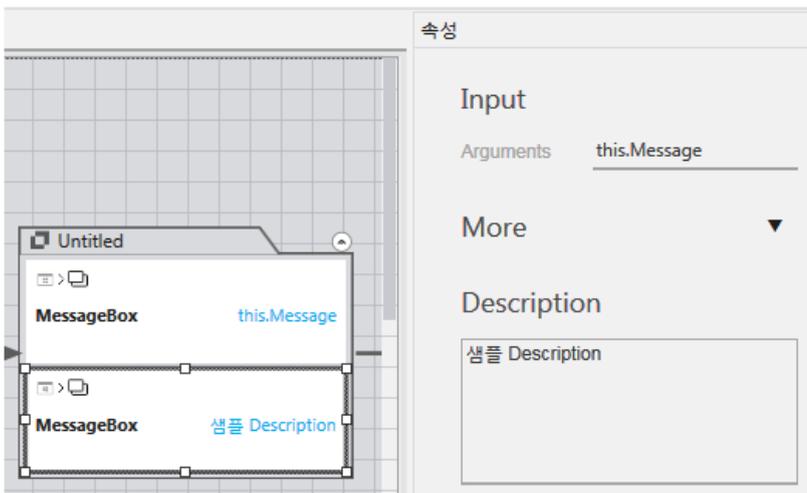
모든 라이브러리에서 공통적으로 제공하는 속성을 설정합니다.

### Description

설명을 추가할 수 있습니다.

Description 이 입력되면 차트의 설명에 표시되고, Description 이 없는 경우에는 각 라이브러리의 대표값이 차트에 표시됩니다.

그림 4-143 Description속성 화면

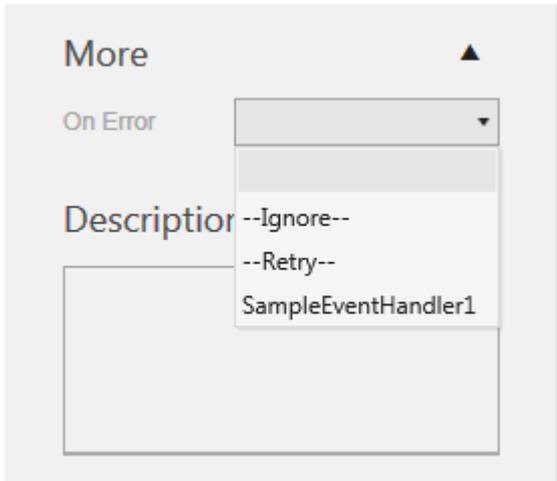


## On Error

각 액티비티 별로 오류가 발생한 경우에 수행할 동작을 지정합니다.

지정하지 않으면 오류를 출력하고 작업이 종료되고, Ignore 를 선택하면 오류 로그를 출력하고 다음 액티비티가 계속 수행됩니다. Retry 를 지정하는 경우에는 1회 재시도하게 되고 재시도시에도 오류가 발생하면 작업이 종료됩니다. 또, 이벤트에 등록된 핸들러를 지정하면 오류발생시 지정한 핸들러가 호출됩니다.

그림 4-144 Description속성 화면



# 5.

## 프로세스 플로우 디자인하기

프로세스 플로우는 여러 프로세스가 처리되는 순서와 흐름을 나타내는 도식입니다. 프로세스 플로우를 디자인하는 방법은 프로젝트와 유사하지만 일부 다른 부분이 있습니다. 이 장에서는 차이점을 위주로 설명합니다.

"프로세스 플로우 디자인하기"에서는 아래 내용에 대해 설명합니다.

- 프로세스로 디자인하기
- 라이브러리로 디자인하기

### 5.1 프로세스로 디자인하기

프로세스 플로우는 서버에 배포된 여러 프로세스를 조합하여 디자인할 수 있습니다.

#### 프로세스 추가하기

서버에 배포된 프로세스를 검색해 프로세스 플로우를 디자인하려면 아래의 절차를 따르세요.

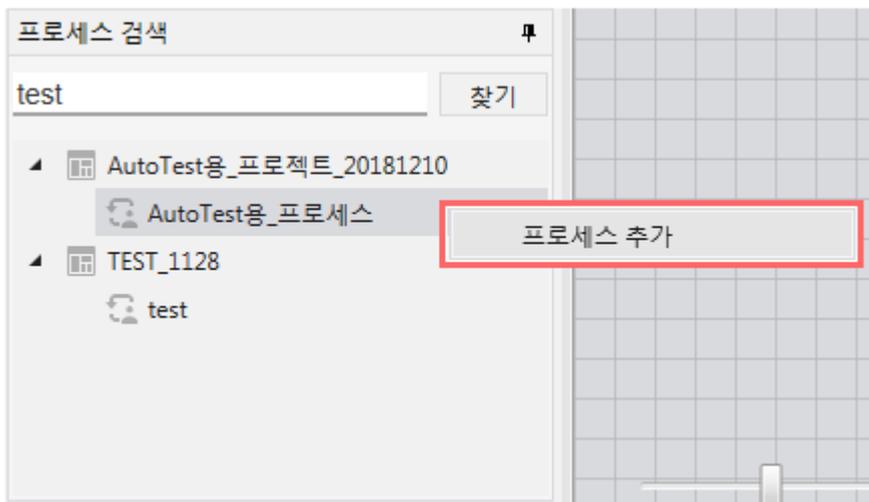


프로세스를 검색하려면 먼저 서버에 로그인해야 합니다.

1. 프로세스 검색 화면의 검색창에 조회할 문구를 입력하고 **찾기**를 클릭하세요.  
입력한 문구를 포함하는 프로세스와 해당 프로세스가 포함된 프로젝트가 나타납니다.



2. 추가할 프로세스를 마우스 오른쪽 버튼으로 클릭한 후 프로세스 추가를 클릭하세요.



3. 프로세스 플로우 전용 라이브러리 화면에서 CallProcess를 검색한 후 플로우차트에 드래그해 추가하세요.
4. 속성 화면에서 호출할 프로세스의 정보를 입력하세요.

프로세스가 추가됩니다.

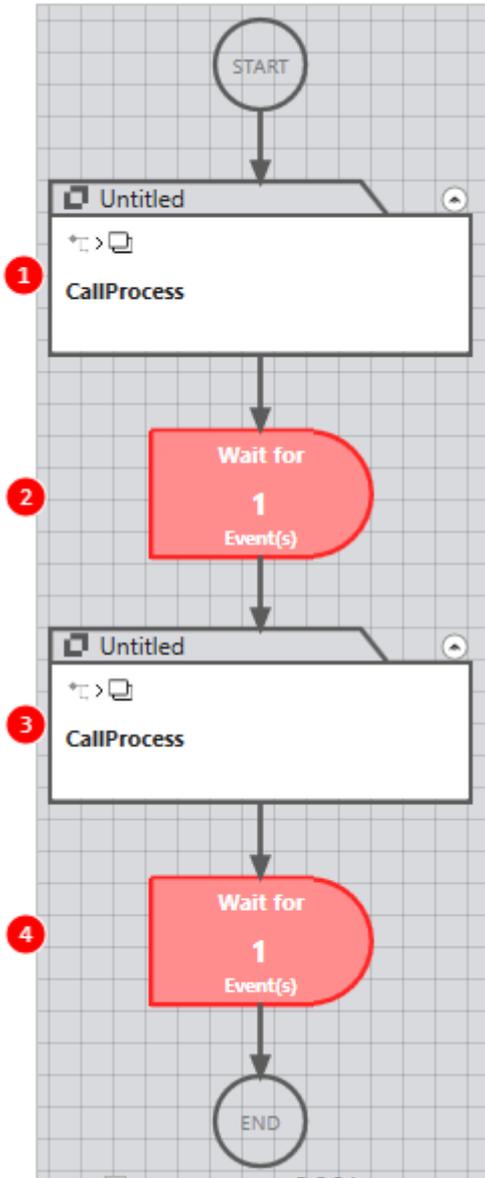


- 프로세스 플로우로 가져온 프로세스의 정보만 조회됩니다.
- 프로세스에 입력값이나 출력값이 있는 경우 속성 화면에 입력하세요.

## 대기 동작 추가하기

프로세스 플로는 순차적으로 진행되는 방식이 아닌 비동기 방식입니다. 프로세스가 순차적으로 수행되어야 하는 경우에는 앞 프로세스가 종료되기까지 기다리는 동작을 적용해야 합니다.

예를 들어, 로그인하기 프로세스를 완료한 후 메일 작성하기 프로세스를 진행해야 할 경우 두 프로세스 사이에 **WaitForMultipleEvents** 라이브러리를 추가합니다.



번호	설명
①	로그인하기
②	'로그인하기' 프로세스 수행 완료 대기
③	메일 작성하기
④	'메일 작성하기' 프로세스 수행 완료 대기

## 5.2 라이브러리로 디자인하기

라이브러리로 프로세스 플로우를 디자인할 수 있습니다.

라이브러리를 이용해 액티비티를 추가하고 편집하는 방법은 [4.2 라이브러리로 디자인하기](#)를 참고하세요.

### 프로세스 플로우 전용 라이브러리 알아보기

#### CallProcess(특정 프로세스 수행하기)

사용자가 지정한 프로세스를 수행하도록 합니다. 프로세스는 서버에 배포되어 있어야 하며 전달해야 할 입력값이나 전달받을 출력값이 있는 경우에는 반드시 함께 명시해야 합니다.

그림 5-1 CallProcess 속성 화면

The screenshot shows the configuration interface for the CallProcess activity. It is divided into three main sections: Input, Output, and Conditions. Each section contains several configuration items, some of which are highlighted with red boxes and numbered callouts (1-5).

- Input Section:**
  - Item 1: ExecutionBot, set to processBot.
  - Item 2: A red box highlights two input variables: var1 (data) and var2 (data).
- Output Section:**
  - Item 3: Result, set to processResult.
  - Item 4: A red box highlights two output variables: var3 and var4.
- Conditions Section:**
  - Item 5: ProcessId, set to MyProcess(836f34ef-22a8-...

번호	설명
1	이전에 수행된 프로세스와 동일한 봇에서 프로세스를 수행시키는 경우에 사용합니다. 이 기능을 사용하려면 이전에 수행된 프로세스의 결과 변수를 선택하고, 사용을 원치 않으면 빈칸으로 둡니다.
2	수행할 프로세스의 입력값으로 전달할 변수를 지정합니다.
3	프로세스의 수행 결과 정보를 저장할 변수를 지정합니다.

번호	설명
4	수행할 프로세스로부터 출력값을 전달 받을 변수를 지정합니다.
5	수행할 프로세스의 아이디를 선택합니다. 검색을 통해 추가된 프로세스만 목록에 나타나며, 선택되는 프로세스에 따라 입력값과 출력값이 달라질 수 있습니다.

### WriteApproval(결재 상신하기)

사용자가 입력한 정보를 기준으로 결재를 상신하는 작업을 수행하도록 합니다.

그림 5-2 WriteApproval 속성 화면

The screenshot shows the configuration interface for the WriteApproval task. It is divided into three main sections:

- Output:** A dropdown menu labeled 'Result' is set to 'approvalResult'.
- Conditions:**
  - Title:** A text input field containing '샘플 제목입니다'.
  - Contents:** A larger text area containing '결재 상신 내용입니다'.
- Approval Line:** A table with columns 'Email' and 'Role'. It contains two entries:
 

Email	Role	
sample_id@samsur	Request	X
sample_id2@samsu	Approve	X

 Below the table is a link that says 'Click here to add new item'.

번호	설명
1	결재 상신 수행 결과 정보를 저장할 변수를 지정합니다.
2	결재 상신 제목을 입력합니다. 작은 따옴표 없이 내용을 바로 입력합니다.
3	결재 상신 내용을 입력합니다. 작은 따옴표 없이 내용을 바로 입력합니다.
4	결재 라인 정보를 입력합니다.

### RequestAsset(자산 요청하기)

서버에 있는 자산을 요청해서 프로세스 플로우에서 사용하도록 합니다. 예를 들어, 공용으로 사용하는 공지사항 제목 템플릿 문자열이 자산으로 등록되어 있는 경우, RequestAsset을 통해서 값을 받아올 수 있습니다.



RequestAsset은 별도로 WaitForMultipleEvents를 추가할 필요 없이 단독으로 사용합니다.

그림 5-3 RequestAsset 속성 화면



번호	설명
1	요청된 자산 객체의 값을 저장할 변수를 선택합니다.
2	요청하려는 자산을 선택합니다. 공용 자산으로 추가한 항목만 목록에 나타납니다.

### WaitForMultipleEvents(이벤트 수행 대기하기)

비동기 방식으로 수행되는 프로세스 플로우에서 하나의 동작이 완료된 이후 다음 동작을 수행하고자 하는 경우 반드시 WaitForMultipleEvents 를 사용해야 합니다. 1개 이상의 여러 동작에 대해 완료되기를 대기할 수 있으며, 하나의 동작이라도 완료되는 경우 및 모든 동작이 전부 완료되는 경우를 옵션으로 지정할 수 있습니다.

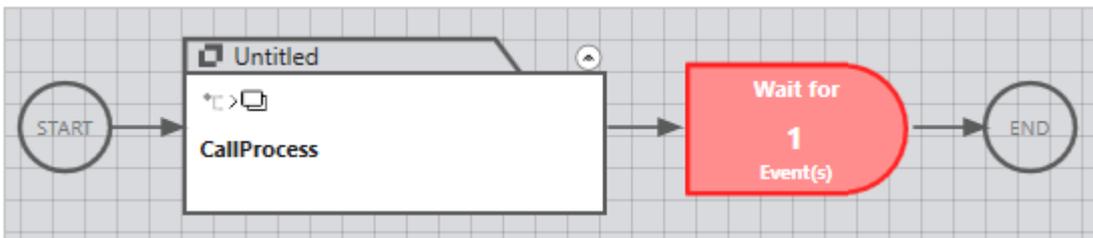
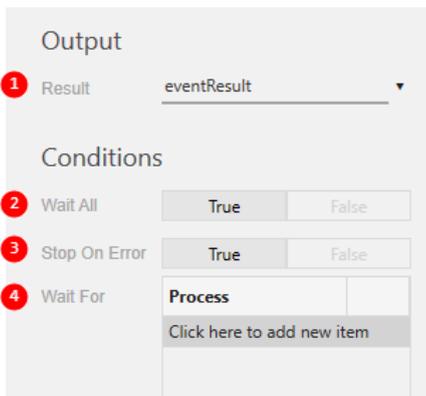


그림 5-4 WaitForMultipleEvents 속성 화면



번호	설명
①	결과 정보가 저장될 변수를 선택합니다.
②	모든 이벤트가 다 발생할 때까지 대기할지, 또는 하나의 이벤트라도 발생할 때까지 대기할지 선택합니다. 예를 들어, 5개의 프로세스 수행 대기 시 False를 설정할 경우에는 5개 중 하나의 프로세스가 완료되면 다음으로 진행합니다.
③	대기하는 이벤트 중 하나라도 실패하는 경우 프로세스 플로우를 중단합니다.
④	대기할 이벤트를 선택합니다. 이벤트의 수행 결과가 저장되는 변수를 선택하면 됩니다. <a href="#">Click here to add new item</a> 을 클릭해 대기할 이벤트를 추가할 수 있습니다. 삭제하려면 <b>X</b> 클릭합니다.

### WaitEmail(메일 수신 대기하기)

메일이 도착하면 특정 동작을 수행할 수 있도록 사용자가 지정한 정보가 담긴 메일을 기다리는 라이브러리입니다. 발신자, 수신자 계정 정보 및 제목에 포함될 문자열 정보를 입력해야 합니다.

그림 5-5 WaitEmail 속성 화면

The screenshot shows the configuration interface for the WaitEmail component, divided into three main sections: Output, Conditions, and Mail. Red numbered callouts (1-6) point to specific fields:

- 1** Output: Points to the 'Result' dropdown menu.
- 2** Conditions: Points to the 'Server URL' text input field containing '112.107.220.150'.
- 3** Conditions: Points to the 'Receiver ID' text input field containing 'sample\_receiver@samsung.com'.
- 4** Conditions: Points to the 'Sender ID' text input field containing 'sample\_sender@samsung.com'.
- 5** Conditions: Points to the 'Receiver PW' password input field.
- 6** Mail: Points to the 'Check Mail' dropdown menu set to 'Contains'.

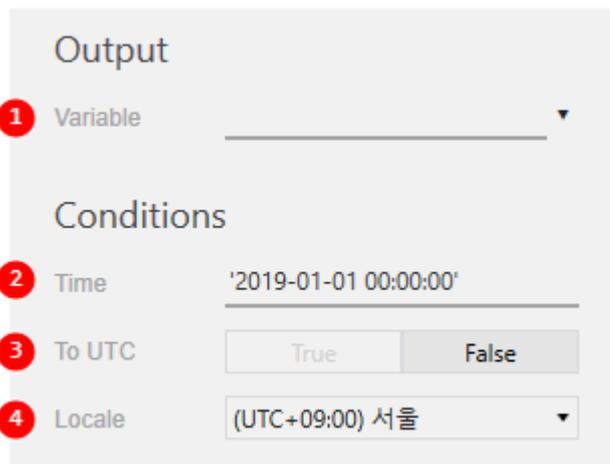
Other visible fields include 'Subject' (sample title), 'Begin Time' (2019-01-01 00:00:00), 'End Time' (2019-01-02 00:00:00), 'To UTC' (radio buttons for True/False), and 'Locale' ((UTC+09:00) 서울).

번호	설명
①	메일 수신 대기 결과를 저장하는 변수를 선택합니다.
②	메일 서버의 주소를 작은따옴표 안에 입력합니다.
③	메일을 받는 계정의 아이디를 작은따옴표 안에 입력합니다.
④	메일을 보내는 계정의 아이디를 작은따옴표 안에 입력합니다.
⑤	메일을 받는 계정의 비밀번호를 입력합니다. Asset 을 선택하는 경우 비밀번호가 저장된 Asset 변수를 입력하고, Input 을 선택하는 경우 계정의 비밀번호를 직접 입력합니다.
⑥	메일의 제목에 포함될 내용을 작은따옴표 안에 입력합니다.

### AssignDateTime(날짜 시간 정보 설정하기)

원하는 날짜와 시간 정보를 변수에 할당합니다.

그림 5-6 AssignDateTime 속성 화면

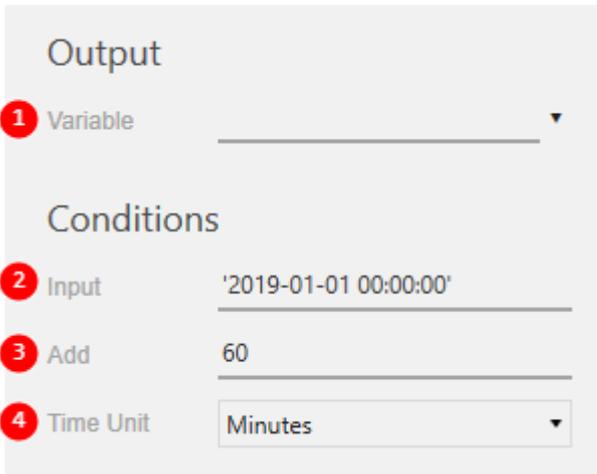


번호	설명
①	시간 정보를 저장하는 변수를 선택합니다.
②	설정할 시간 정보를 입력합니다.
③	시간 정보를 UTC 로 변환할 지 여부를 선택합니다. True 를 선택하는 경우 런타임에 시간 정보를 Locale 기준으로 UTC 로 변환합니다.
④	Locale 정보를 선택합니다. To UTC 옵션이 True 로 선택된 경우 Locale 정보가 활용됩니다.

### AddDateTime(날짜 시간 정보 더하기)

입력받은 값에 원하는 날짜와 시간 정보를 더하여 변수에 할당합니다.

그림 5-7 AssignDateTime 속성 화면

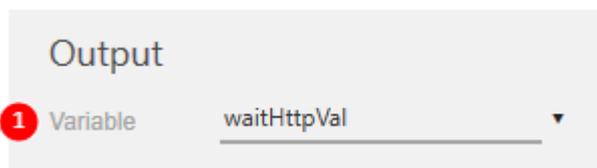


번호	설명
①	시간 정보를 저장하는 변수를 선택합니다.
②	변경 기준이 되는 시간 정보를 입력합니다.
③	더할 시간 값을 숫자로 입력합니다.
④	시간 단위를 선택합니다.

### WaitHttp(URL 접속 대기하기)

임시로 생성된 URL 이 호출되도록 기다리는 라이브러리입니다. 주로 사용자의 계정 인증을 위해 이루어지며 메일, 메신저, SMS 등을 통해 전달되는 URL 을 사용자가 접속할 때까지 대기합니다.

그림 5-8 WaitHttp 속성 화면



번호	설명
①	결과 객체의 값을 저장하는 변수를 선택합니다. 전달 받은 URL 값을 저장하였다가 다른 라이브러리에서 쉽게 사용하기 위해 필요합니다.

### WakeUp(특정 시간까지 대기하기)

사용자가 지정한 특정 시각까지 대기합니다. 예를 들어 새벽 3시에 특정 프로세스가 수행되기를 원하면 사용자는 CallProcess 라이브러리 이전에 WakeUp을 추가하고 시간을 명시합니다. 서버에서 프로세스 플로우 수행 요청을

하면 이 라이브러리 단계에서 대기하고 있다가 새벽 3시가 되면 이후 프로세스가 수행됩니다.

그림 5-9 WakeUp 속성 화면

번호	설명
①	결과 객체를 저장하는 변수를 선택합니다.
②	동작을 수행할 시간을 설정합니다.

# 6.

## 서버 연동하기

IPA Designer를 서버에 연동해 작업한 프로젝트 및 프로젝트 플로우를 서버에 배포할 수 있습니다. 또한 서버에 배포된 프로젝트 및 프로젝트 플로우를 내려받아 편집하거나 사용할 수 있습니다.

"서버 연동하기"에서는 아래 내용에 대해 설명합니다.

- 서버 연동 준비하기
- 활성화 및 로그인하기
- 프로젝트 및 프로젝트 플로우 배포하기
- 프로젝트 및 프로젝트 플로우 가져오기
- 공용 리소스 사용하기

### 6.1 서버 연동 준비하기

서버와 연동하려면 디자이너 ID 및 라이선스 키가 필요합니다.

1. 디자이너 ID 및 라이선스 키를 생성하세요.
2. 계정을 생성하세요.
3.  > 설정 > 서버 설정을 클릭하세요.
4. 서버 IP와 서버 Port를 입력한 후 확인을 클릭하세요.

서버 설정

서버 IP  
182.195.81.243

서버 Port  
8777

User Agent  
IPA Designer 1.0

OCR 서버 URL

네트워크  
 Proxy 사용  OCR Server 접속

보안 프로토콜

취소 확인

서버 주소가 설정됩니다.

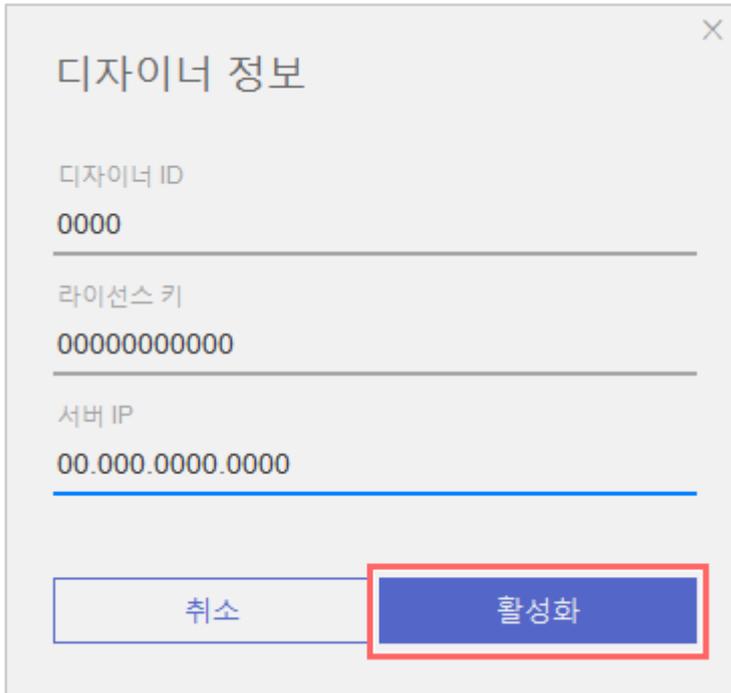
## 6.2 활성화 및 로그인하기

서버에 연동하려면 IPA Designer를 활성화하고 로그인해야 합니다.  
서버에서 발급받은 디자이너 ID와 라이선스 키가 필요합니다.

### 활성화하기

서버에 처음 연동하거나, 디자이너 ID 및 라이선스 키를 변경한 경우 먼저 IPA Designer를 활성화하세요.

1. ☰ > 도움말 > 라이선스를 클릭하세요.
2. "디자이너 정보" 팝업창에서 라이선스 키를 입력한 후 활성화를 클릭하세요. 디자이너 ID 는 활성화 완료 시 자동으로 입력됩니다.



IPA Designer가 활성화됩니다.

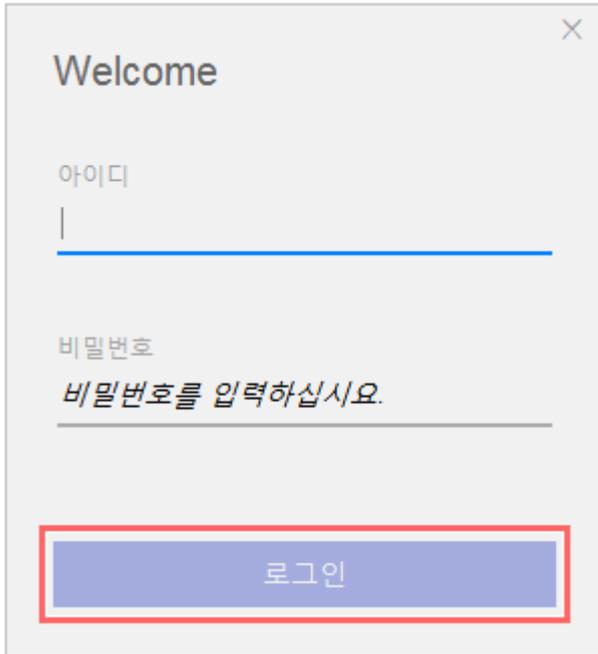
## 로그인하기

서버에서 발급받은 계정을 사용해 로그인하세요.

1. IPA Designer 화면에서  클릭하세요.



2. 서버에서 생성한 ID와 비밀번호를 입력한 후 로그인을 클릭하세요.

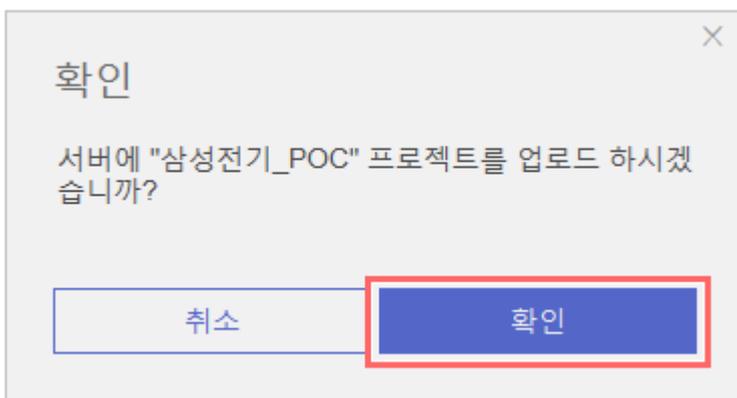


로그인이 완료됩니다.

## 6.3 프로젝트 및 프로세스 플로우 배포하기

IPA Designer에서 작업 중인 프로젝트 및 프로세스 플로우를 서버에 배포할 수 있습니다. 배포한 프로젝트 또는 프로세스 플로우는 서버에서 활성화한 후 봇을 이용해 실행할 수 있습니다.

1. ☰ > 배포하기를 클릭하세요.
2. 배포 확인 팝업창에서 확인을 클릭하세요.



프로젝트 또는 프로세스 플로우가 서버에 배포됩니다.

## 6.4 프로젝트 및 프로세스 플로우 가져오기

서버에 배포된 프로젝트 및 프로세스 플로우를 가져올 수 있습니다. 가져온 프로젝트 또는 프로세스 플로우는 로컬의 IPA Designer에서 편집하거나 실행할 수 있습니다.

1.  > 가져오기를 클릭하세요.
2. 프로젝트 또는 프로세스 플로우를 선택한 후 가져오기를 클릭하세요.
3. 가져올 프로젝트 또는 프로세스 플로우를 선택한 후 가져오기를 클릭하세요.

**프로젝트 리스트** ✕

Full Text Search ✕

네임스페이스 ▼	이름 ▼	버전 ▼	설명 ▼
	PJ_Admin_사용자_20181210	1.80	
	PJ_Admin_메시지_20181210	3.20	
	PJ_Admin_현황모니터링_20181210	1.50	
	PJ_Admin_예약작업_20181210	1.00	
	PJ_Admin_로그인_20181210	1.10	
	AutoTest용_프로젝트_20181210	1.00	
	PJ1206	1.00	
	proj1205_2	1.20	
	PJ1205_2	1.00	
	송재민테스트용	1.00	
	PJ1205_1	1.00	
	PJ1205	1.10	
	proj1127	1.70	
	PJ1128	1.50	
	PJ_Admin_로그인_20181210	1.00	
	CtrlWithClick	1.80	
	VersionTestProject	1.40	
	cc5	1.00	
	adad6	1.00	

선택된 항목이 없습니다.

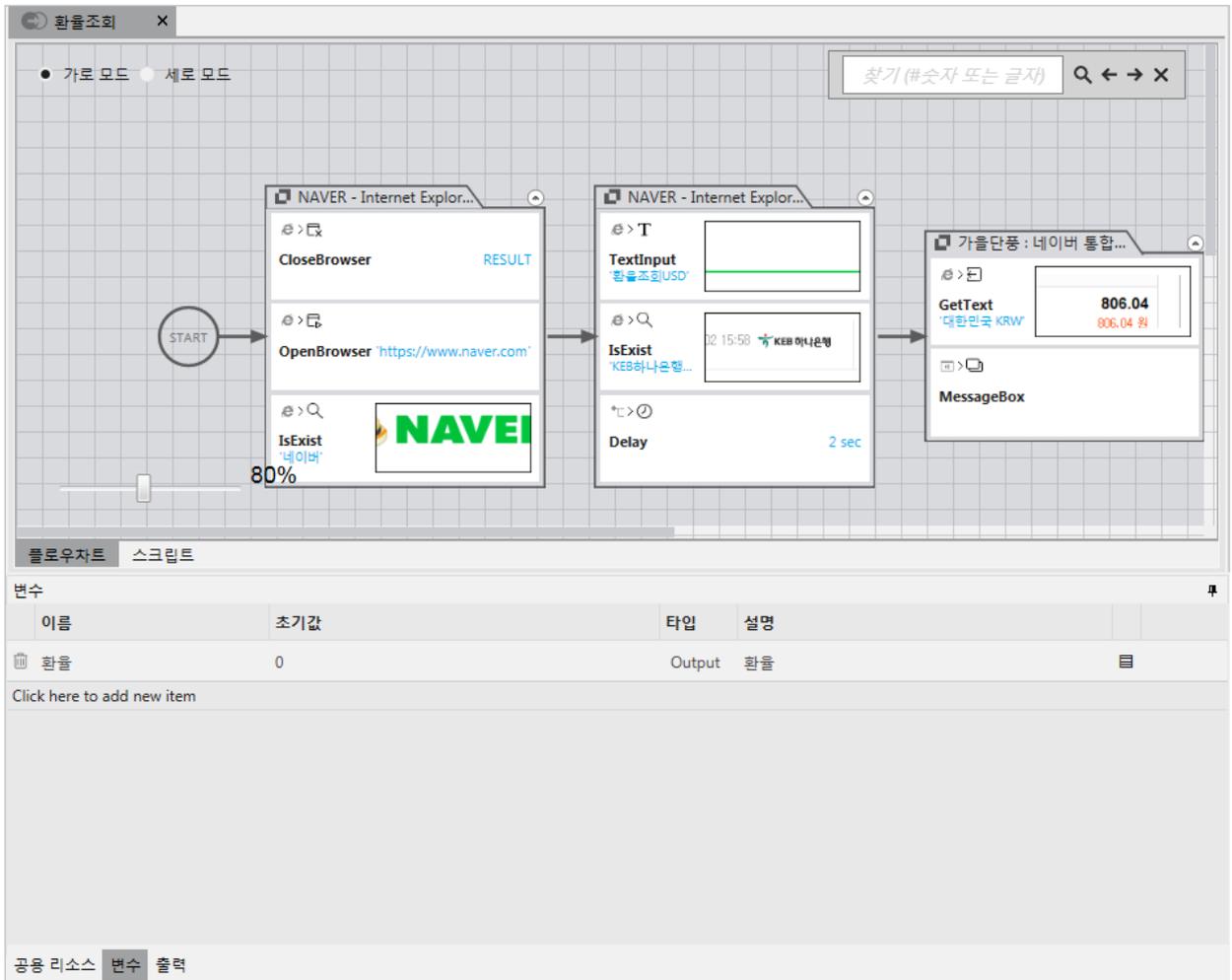
취소
가져오기

가져온 프로젝트 또는 프로세스 플로우가 추가됩니다.

## 6.5 공용 리소스 사용하기

서버에 등록되어 있는 공용 리소스를 조회하고, 필요한 경우 IPA Designer로 가져와 사용할 수 있습니다. 프로젝트에 추가된 리소스는 라이브러리의 속성 화면에서 사용할 수 있습니다.

1. 공용 리소스를 클릭해 공용 리소스 화면으로 이동하세요.



2. Click here to add new item을 클릭하세요.  
서버에 등록된 공용 리소스 목록이 나타납니다.
3. 서버에 등록된 공용 리소스 목록에서 가져올 리소스를 선택한 후 가져오기를 클릭하세요.

리소스가 현재 프로젝트에 추가됩니다.

# 7.

## IPA Bot 사용하기

IPA Bot을 사용해 업무자동화 환경을 구축할 수 있습니다.

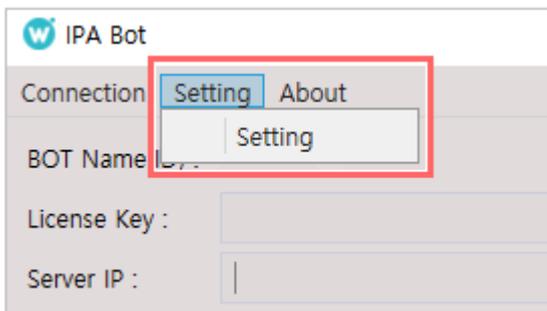
"IPA Bot 사용하기"에서는 아래 내용에 대해 설명합니다.

- 사용 환경 설정하기
- 봇 구동하기

### 7.1 사용 환경 설정하기

IPA Bot을 사용하려면 서버에서 Bot ID와 라이선스 키를 생성해야 합니다.

1. **Setting > Setting**을 클릭하세요.



2. **서버** 탭에서 서버 IP와 서버 Port를 입력한 후 **확인**을 클릭하세요.

**설정**

봇    서버    라이선스

서버 IP: 182.195.81.243

서버 Port: 8777

User Agent: IPA Bot 1.0

OCR 서버 URL: [Empty]

Proxy 사용:

OCR Server 접속시 Pro:

보안 프로토콜: [Empty]

취소    **확인**

3. 라이선스 탭에서 라이선스 키를 입력한 후 **확인**을 클릭하세요.

**설정**

봇    서버    라이선스

클라이언트 ID: [Empty]

클라이언트 이름: ClientName

라이선스 키: 1234-5678-9012

Issuer: AUTH\_CLIENT\_CERTIFICATE

취소    **확인**

4. **Setting > Setting**을 클릭하세요.
5. **라이선스** 탭에서 클라이언트 ID와 라이선스 키를 입력한 후 **확인**을 클릭하세요.

The screenshot shows a settings window titled '설정' (Settings) with a close button (X) in the top right corner. Below the title bar are three tabs: '봇' (Bot), '서버' (Server), and '라이선스' (License), with the '라이선스' tab selected and highlighted in blue. The main area contains four input fields:

- 클라이언트 ID: BOT-000001
- 클라이언트 이름: ClientName
- 라이선스 키: 1234-5678-9012
- Issuer: AUTH\_CLIENT\_CERTIFICATE

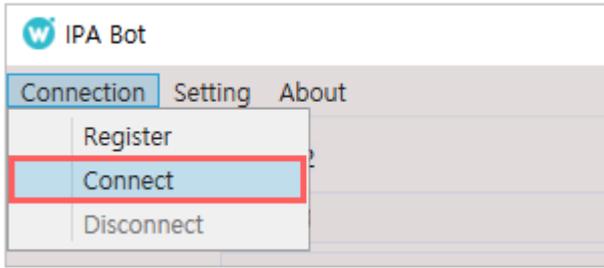
At the bottom of the dialog, there are two buttons: '취소' (Cancel) and '확인' (Confirm). The '확인' button is highlighted with a red border.

사용 환경 설정이 완료됩니다.

## 7.2 봇 구동하기

서버 IP와 클라이언트 ID, 라이선스 키를 설정한 후 봇을 구동할 수 있습니다.

**Connection > Connect**를 클릭하세요.



봇이 구동됩니다.

# 부록 A.

## 문제 해결하기

자주 발생하는 문제 상황과 해결 방법을 설명합니다.  
"문제 해결하기"에서 다루는 내용은 다음과 같습니다.

- 설치 및 등록 관련 문제
- 자동화 시나리오 작성 및 실행 관련 문제
- 서버 연계 관련 문제

### A.1 설치 및 등록 관련 문제

**상황:** IPA Designer 및 IPA Bot이 활성화되지 않습니다.

해결 방법: 인증서가 제대로 설치되지 않은 경우 IPA Designer 및 IPA Bot가 활성화되지 않을 수 있습니다.  
아래와 같이 auth\_server.cer, AUTH\_SERVER\_CERTIFICATE.crt 인증서를 다시 설치하세요.

1. 인증서 화면에서 **인증서 설치**를 클릭하세요.
2. 인증서 가져오기 마법사에서 **다음**을 클릭하세요.
3. **모든 인증서를 다음 저장소에 저장**을 선택한 후 **찾아보기**를 클릭하세요.
4. 인증서를 저장할 경로를 선택한 후 **확인**을 클릭하세요.
5. **다음**을 클릭하세요.
6. 인증서가 저장된 것을 확인한 후 **마침**을 클릭하세요.

상황: 신규 버전 설치가 되지 않습니다.



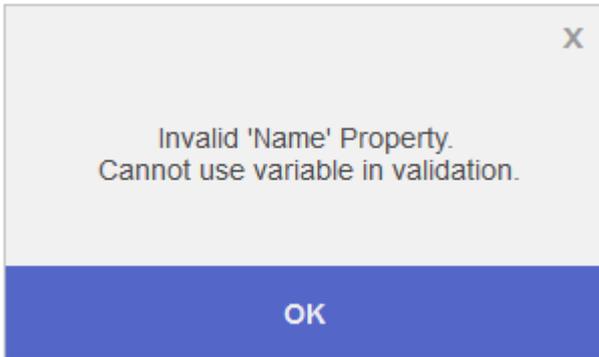
해결 방법: 이전 버전의 IPA Designer 또는 IPA Bot을 종료한 후 신규 버전을 설치하세요.

## A.2 자동화 시나리오 작성 및 실행 관련 문제

상황: 앱의 메뉴들 중 하위 메뉴를 클릭하고 싶습니다. 하위 메뉴를 열기 위해 상위 메뉴를 누르면 대상 UI 객체로 지정되어 원하는 메뉴를 선택할 수 없습니다.

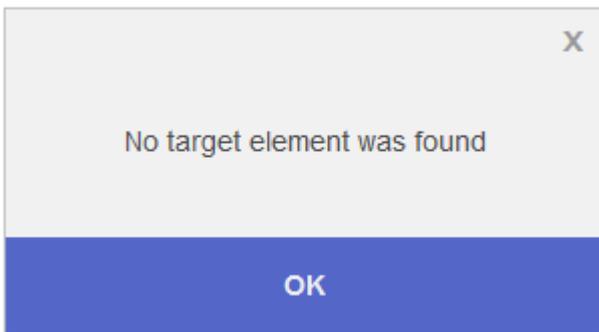
해결 방법: 앱의 메뉴 하위 항목이나 컨텍스트 메뉴와 같이 이전 단계의 UI 객체를 클릭한 후 대상 UI 객체를 선택해야 하는 경우, 이전 단계에 클릭한 UI 객체가 대상 UI 객체로 지정되는 문제가 발생합니다. 또는 지정하려는 UI 객체가 화면에 없어서 마우스를 이용해 해당 화면으로 이동해야 하는 경우에도 동일한 문제가 발생합니다. 이러한 경우에는 키보드의 Ctrl 키를 눌러 대상 지정을 일시 중지할 수 있습니다. 대상 UI 객체를 선택할 수 있는 상태가 되면 다시 Ctrl 키를 눌러 대상 지정을 다시 시작할 수 있습니다.

**상황:** 대상 UI 객체 Test 수행 시 오류가 발생합니다.



해결 방법: **Name, Value** 등 편집 가능한 속성 값은 스크립트 생성 시 스크립트 코드와 동일하게 처리됩니다. 따라서 일반 문자열인 경우 따옴표 안에 입력해야 하며, 변수를 사용할 경우에는 따옴표 없이 입력이 가능합니다. 위 화면의 경우 **Name** 속성에 변수가 지정된 경우에 발생하는 오류입니다. 변수의 값은 실행 시점에 결정되므로, 시나리오 작성 단계의 Test 시점에는 변수의 값을 사용할 수 없습니다. 또는 문자열 입력 시 따옴표가 누락된 경우, 해당 입력 값이 변수 또는 스크립트 구문으로 인식되어 오류가 발생할 수 있습니다.

**상황:** 대상 UI 객체를 찾지 못합니다.



해결 방법: 속성 화면에서 대상 UI 객체의 **Name, Value** 값과 **Check Name, Check Value, Check Size** 항목을 확인하세요. 레코딩 시점과 수행 시점의 대상 UI 객체 정보가 다른 경우 대상 UI 객체를 발견하지 못할 수 있습니다. Chrome 브라우저 등 일부 앱, Flash를 이용한 웹 사이트 등 대상 UI가 객체 정보를 정상적으로 제공하지 않는 경우가 존재합니다. 이 경우 ImageRecognition 라이브러리를 이용하여 이미지 인식 방식으로 대상 UI를 지정할 수 있습니다.

**상황:** 각각의 UI 객체를 단독으로 수행하는 경우에는 잘 동작하는데, 여러 개의 UI 객체를 연달아 사용하면 정상적으로 동작하지 않습니다.

해결 방법: 콤보 박스 항목이나 컨텍스트 메뉴 등 일부 UI 객체는 포커스를 설정하면 화면에서 사라질 수 있습니다. 이러한 경우에는 **Focus Before** 항목을 **False**로 설정한 후 재시도하세요.  
 화면이 변경되는 시점에는 UI 객체 정보를 정상적으로 얻지 못할 수 있습니다. 특히 새로운 윈도우 화면으로 전환되는 경우에는 오류가 발생할 가능성이 높아집니다. 이 경우 각 스텝 사이에 **Delay** 라이브러리를 추가해 수행 속도를 조정하세요.

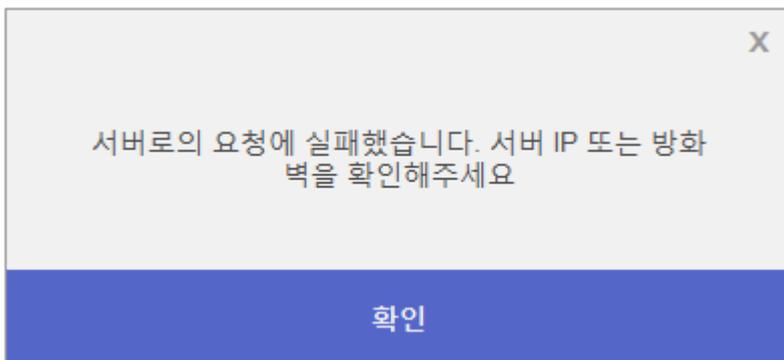
**상황: 다른 작업을 호출하면 오류가 발생합니다.**

해결 방법: 호출하려는 작업의 속성 화면에서 **Input**, **Output** 항목을 확인한 후 오류가 발생한 RunTask의 속성 화면에서 **Input**, **Output** 항목을 확인하세요. 대상 작업의 **Input**, **Output** 변수와 RunTask의 **Input**, **Output** 속성이 일치해야 합니다. 작업의 **Input**, **Output** 변수가 변경되는 경우에는 해당 작업을 호출하는 RunTask에도 변경해야 합니다.

호출하려는 작업이 저장되어있는지 확인하세요. 저장되지 않은 작업은 수정하기 전의 스크립트를 사용하여 실행됩니다.

## A.3 서버 연계 관련 문제

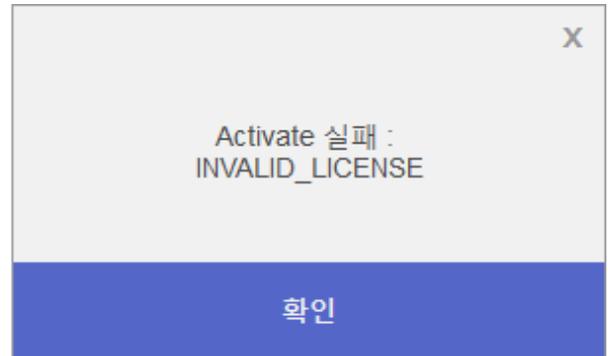
**상황: 서버 연결이 되지 않습니다.**



해결 방법: 서버 IP 또는 방화벽을 확인해주세요.

상황: IPA Designer 활성화가 되지 않습니다.

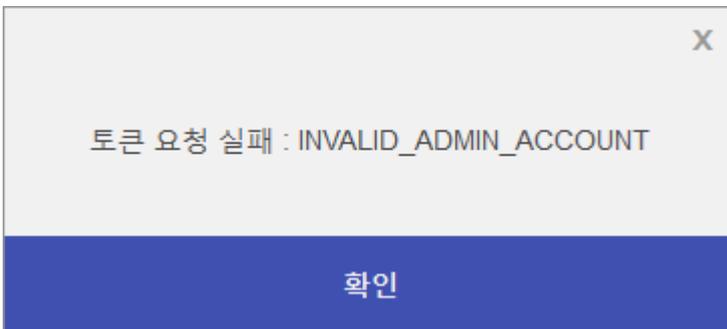
---



해결 방법: Client ID 또는 라이선스 키를 확인하세요.

상황: 로그인이 되지 않습니다.

---



해결 방법: ID 또는 비밀번호를 확인하세요.

# 부록 B.

## FAQ

자주 발생하는 질문과 답변을 확인할 수 있습니다.

### Q. 시나리오를 작성할 때와 해상도가 달라도 실행할 수 있나요?

A. Windows나 SAP, Excel 등에서 제공하는 정보를 이용하여 대상 UI 객체를 검색하는 원리입니다. UI 객체 간의 상하 관계 및 텍스트 정보, 타입 정보 등으로 대상 UI 객체를 식별하므로 해상도가 달라도 기본적인 동작을 수행하는 데에는 문제가 없습니다.

단, 해상도에 따라 화면의 구성이 전반적으로 달라지는 앱의 경우에는 정상적으로 동작하지 않을 수 있습니다.

### Q. Windows 7에서 작성한 시나리오가 Windows 10에서도 동작하나요?

A. 대상 앱이 동일한 경우 Windows의 버전과 상관 없이 동일하게 동작합니다.

단, 대상 앱이 이름과 기능만 동일하고 버전이 다른 경우에는 정상적으로 동작하지 않을 수 있습니다. 예를 들어, Windows 7의 그림판과 Windows 10의 그림판은 동일한 앱이 아니므로 호환되지 않습니다.

### Q. 가상 OS에서도 사용할 수 있나요?

A. 가상 OS에서도 일반 PC처럼 사용할 수 있습니다.

단, 가상 OS의 성능이 낮은 경우가 많으므로 주의해야 합니다. 저사양의 OS 환경에서는 시나리오가 정상적으로 동작하지 않을 수 있습니다. 이러한 경우 Delay 스텝을 추가하거나 설정 화면에서 Speed를 조정하는 등 시간을 조정하여 사용할 수 있습니다.

### Q. Designer 실행 시 V3에서 에러 메시지가 발생하면 어떻게 해야하나요?

A. V3는 EXE 프로그램이 Signing이 되어 있지 않거나,

다수 PC에서의 실행이 확인되지 않으면(클라우드 진단이라고 되어 있습니다)

차단 메시지를 띄웁니다.

V3 환경설정에서 검사예외설정에 Bot.exe, Designer.exe를 포함시켜 보시기 바랍니다.

또는 클라우드 진단을 UnCheck 해보세요.

### Q. 프로그램의 메뉴 또는 우클릭 후 나타나는 컨텍스트 메뉴를 라이브러리의 대상객체로 지정하는 방법이 있나요?

A. 대상객체를 지정 시 Ctrl 키를 입력하면 객체선택 사각형이 녹색으로 표시되며, 녹색인 상태에서는 객체로 입력이 중단되어 메뉴 또는 우클릭 후 다시 Ctrl 키를 입력하여 객체선택 사각형이 적색으로 표시되었을때 객체를 선택할 수 있습니다.

Click 라이브러리를 예로 들면,

1. 대상 지정 메뉴 선택

=> 마우스 이동시 객체 선택 사각형이 적색으로 표시

2. 키보드에서 Ctrl 키 한번 클릭

=> 대상 지정 모드가 Pause 상태로 변경되며, 이후 마우스 이동시 객체 선택 사각형이 녹색으로 표시

3. 지정하고자 하는 메뉴가 나타나도록 마우스 및 키보드 조작

4. 지정대상 객체가 화면에 나타나면 키보드에서 Ctrl 키를 한번 클릭

=> 대상 지정 모드가 Resume 상태로 변경되며, 이후 마우스 이동시 객체 선택 사각형이 적색으로 표시

# 용어 사전