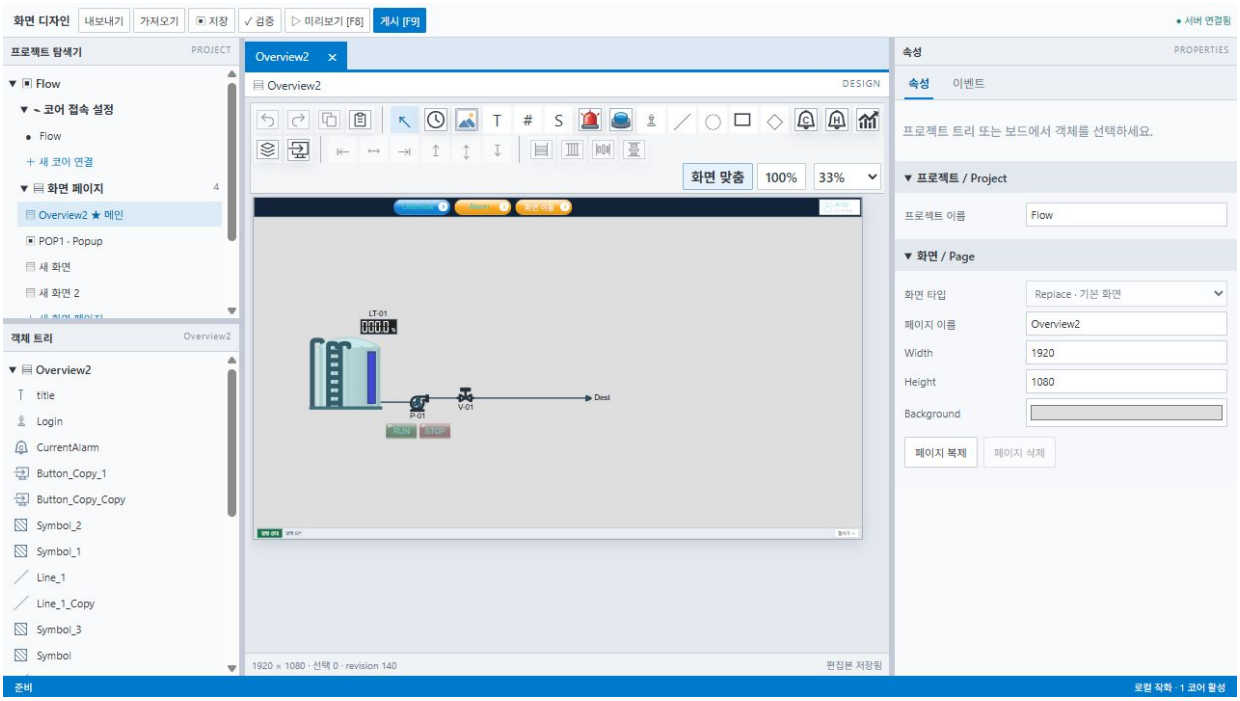
JView

산업용 SCADA / HMI

사용자 매뉴얼

RELEASE 1.0.9 | 2026-10-15



현재 디자이너의 Flow 예제 프로젝트

설치부터 작화, 태그 연결, 스크립트, 게시 및 운영 진단까지

이 문서는 1.0.9 설치 패키지와 소스를 기준으로 작성했습니다. 화면 그림은 현재 디버그 디자이너에서 캡처한 예제입니다. 실제 프로젝트명·태그명·데이터는 사용 환경에 따라 다릅니다.

목차

01. 제품 구성과 작업 흐름 3

02. 설치와 첫 접속 4

03. 디자이너 화면 구성 5

04. 프로젝트와 화면 관리 6

05. JCore 연결 설정 7

06. 게시 태그 관리와 CSV 8

07. 객체 배치와 정렬 9

08. 공통 속성과 기능별 바인딩 10

09. 숫자·문자열·램프 표시 11

10. 버튼·페이지 이동·로그인 12

11. 심볼·이미지·시계 13

12. 팝업 화면 14

13. 현재 알람과 알람 이력 15

14. 실시간·이력 트렌드 16

15. 이벤트와 C# 스크립트 17

16. 스크립트 API와 예제 18

17. 미리보기·게시·운영 20

18. 백업·이전·업데이트 21

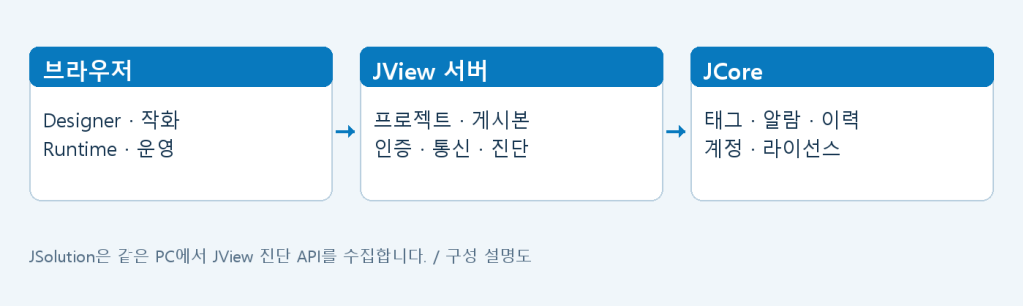
19. JSolution 진단과 문제 해결 22

20. 단축키와 운영 점검표 24

읽는 순서: 처음 설치하는 사용자는 01~06장, 작화 담당자는 07~16장, 운영 담당자는 17~20장을 참고하세요.

01. 제품 구성과 작업 흐름

JView는 브라우저에서 공정 화면을 제작하고 JCore의 태그·알람·이력 데이터를 표시하는 SCADA/HMI 제품입니다. JView 서버가 프로젝트와 게시본을 보관하고, 사용자는 디자이너에서 작화한 뒤 런타임으로 운영합니다.



시스템 구성과 데이터 흐름

주요 구성

구성	역할
Designer	페이지·객체·태그 연결·스크립트를 편집하는 작화 화면
JView.Server	프로젝트 저장, 게시, 사용자 세션 및 JCore 통신 처리
Runtime	게시한 화면을 표시하고 허용된 태그 조작 수행
JCore	실제 태그, 라이선스, 계정 권한, 알람 및 이력 제공
JSolution	같은 PC의 JView 진단 API를 통해 상태·이벤트 수집

처음 만드는 순서

1. JView 설치 후 디자이너에 접속합니다.
2. 코어 연결의 Host, TLS 포트, 인증서 지문과 계정을 입력합니다.
3. 설정을 저장하고 태그 읽어오기를 실행합니다.
4. 게시 태그 보기에서 사용할 태그를 선택합니다.
5. 화면을 만들고 객체를 배치한 후 태그를 연결합니다.
6. 프로젝트를 저장하고 검증·미리보기로 동작을 확인합니다.
7. 게시를 실행하고 운영 주소 또는 저장한 HTML로 접속합니다.

저장과 게시의 차이

저장은 편집본을 서버에 보관합니다. 게시는 운영에 사용할 버전을 확정합니다. 편집본을 저장한 것만으로 운영 중인 화면과 스크립트가 바뀌지 않습니다. HTML은 운영 서버로 연결하는 진입 파일이며 화면 전체를 독립 실행하는 오프라인 패키지가 아닙니다.

02. 설치와 첫 접속

설치 환경

Windows x64 환경과 .NET Framework 4.8 이상이 필요합니다. 설치 프로그램은 관리자 권한을 요청합니다. 개발 도구나 Node.js를 운영 PC에 별도로 설치할 필요는 없습니다. 브라우저 접속 PC에는 JView 서버를 설치하지 않아도 됩니다.

1. JView_Setup.exe를 실행합니다.
2. 설치 안내에 따라 진행하고 완료 여부를 확인합니다.
3. 바탕화면의 JView Designer 바로가기를 실행합니다.
4. 디자이너가 열리면 서버 연결 상태를 확인합니다.
5. 다른 PC에서는 localhost 대신 JView 서버의 IP를 사용합니다.

대상	기본 주소 또는 위치
로컬 디자이너	http://localhost:5080/designer
로컬 런타임	http://localhost:5080/runtime
다른 PC 접속	http://서버IP:5080/designer 또는 /runtime
Windows 서비스	JView.Server / 표시 이름 JView Server
프로그램 폴더	%ProgramFiles%₩JAYON STUDIO₩JView
데이터 폴더	%ProgramData%₩JAYON STUDIO₩JView

네트워크 동작

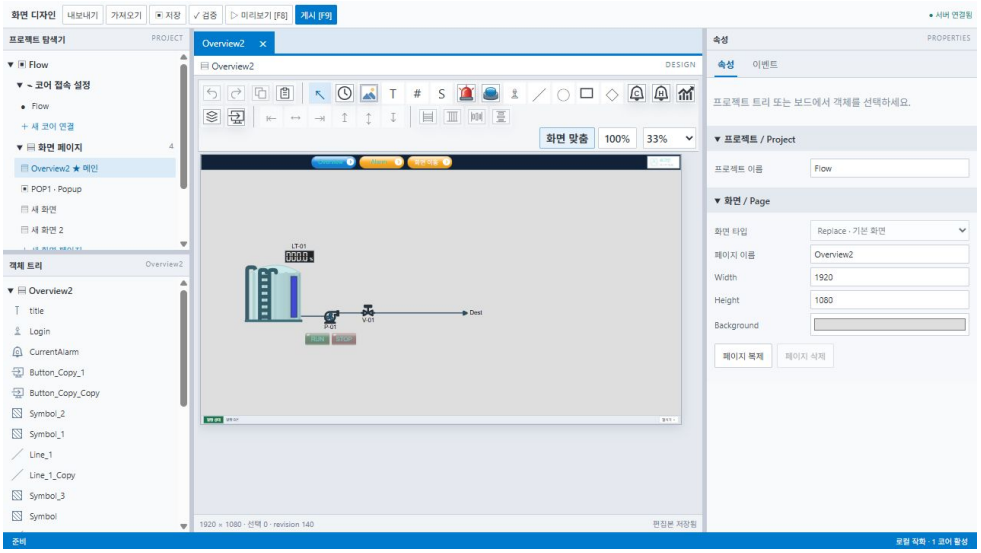
기본 설치는 HTTP 5080을 모든 서버 인터페이스에서 수신합니다. 서버에 IP가 여러 개라면 접근 가능한 각 IP로 디자이너와 런타임에 접속할 수 있습니다. 실제 접근 가능 범위는 Windows 방화벽과 네트워크 구성에 따릅니다. 설치된 디자이너에는 별도 작화 로그인 없이 접근할 수 있으므로 운영자가 접근 범위를 관리해야 합니다.

JView의 웹 포트 5080, JCore 실시간 TLS 포트, JCore 관리 웹 포트 8080은 서로 다른 용도입니다. 사용자 지정 HTTPS·포트를 사용하는 환경은 URL 예약, 인증서 바인딩과 방화벽을 해당 주소에 맞게 구성해야 합니다.

첫 접속이 실패하면

서비스 실행 상태를 확인한 뒤 서버 PC에서 localhost로 먼저 접속합니다. 로컬 접속은 되지만 다른 PC에서만 실패하면 서버 IP·포트와 방화벽을 확인합니다. 서비스 시작 실패 또는 503은 19장의 로그 확인 절차를 사용합니다.

03. 디자이너 화면 구성



현재 디자이너의 주요 작업 영역

영역	사용 방법
상단 프로젝트 메뉴	가져오기·내보내기·저장·검증·미리보기·게시
왼쪽 프로젝트 탐색기	코어 접속 설정, 화면 페이지, 글로벌 스크립트, 심볼 및 게시 이력
왼쪽 객체 트리	현재 페이지의 객체·그룹 선택
중앙 페이지 탭	열린 페이지 사이 전환. 탭 닫기는 페이지 삭제와 다름
중앙 도구 모음	객체 생성, 복사·붙여넣기, 정렬, 확대·축소
오른쪽 속성 / 이벤트	선택한 객체·페이지의 속성 또는 이벤트 편집

패널 크기 변경

좌우 패널과 중앙 화면 사이의 구분 바를 드래그하여 너비를 조절합니다. 왼쪽 프로젝트 탐색기와 객체 트리 사이의 구분 바는 높이를 조절합니다. 구분 바를 더블클릭하면 초기 크기로 돌아갑니다.

화면 확대

화면 맞춤은 전체 페이지가 작업 영역에 들어오도록 배율을 계산합니다. 100%는 원래 배율로 표시합니다. 확대 콤보박스 또는 Ctrl+마우스 휠로 배율을 바꿀 수 있습니다. 확대는 편집 화면의 보기 배율이며 객체의 실제 Width·Height 값을 변경하지 않습니다.

속성과 이벤트

일반 설정은 속성 탭, 객체 동작 코드는 이벤트 탭에서 편집합니다. 객체 더블클릭으로 제공되는 편집 창에서도 기능별 바인딩과 스크립트를 나누어 확인할 수 있습니다. 객체 종류에 해당하지 않는 속성은 표시하지 않습니다.

04. 프로젝트와 화면 관리

현재 서버는 하나의 활성 프로젝트를 운영합니다. 하나의 프로젝트에 여러 기본 화면과 팝업 화면을 구성합니다. 프로젝트 탐색기에서 페이지를 선택하면 해당 페이지의 객체가 트리에 나타납니다.

새 화면 만들기

1. 왼쪽의 새 화면 페이지를 선택합니다.

2. 페이지 이름과 화면 용도를 정합니다.

3. 기본 화면은 Replace, 별도 창처럼 띄울 화면은 Popup으로 구성합니다.

4. Width·Height 및 Background를 지정합니다.

5. 객체를 배치하고 저장합니다.

새 화면의 기본 크기는 1920 × 1080입니다. 설치할 모니터와 표시 목적에 맞춰 조정할 수 있습니다. 메인 표시는 프로젝트의 시작 페이지를 뜻합니다.

이름·복제·삭제

객체 선택을 해제하고 프로젝트 또는 페이지를 선택하면 프로젝트 이름과 페이지 속성을 편집할 수 있습니다. 페이지 복제로 유사한 화면을 만든 후 태그와 이름을 바꾸면 반복 작화 시간을 줄일 수 있습니다. 시작 페이지 등 삭제 제한이 있는 항목은 삭제 버튼 상태를 확인합니다.

페이지 이동 버튼은 대상 이름을 미리 입력할 수 있습니다. 대상 페이지를 아직 만들지 않았거나 삭제했다면 실행 시 대상 없음 오류가 발생할 수 있으므로 게시 전 이동 경로를 점검합니다.

편집본 관리

저장 후 하단의 저장 상태를 확인합니다. 여러 브라우저에서 같은 프로젝트를 동시에 수정하면 저장 충돌이 발생할 수 있습니다. 충돌 시 다른 편집 내용을 확인한 후 최신 상태를 다시 열어 작업합니다.

내보내기는 .jview 파일을 생성합니다. 가져오기는 현재 편집본을 교체하며 운영 게시본을 즉시 변경하지 않습니다. 이전하기 전에 기존 프로젝트를 내보내 보관하세요. 연결과 비밀번호 이전에 관한 차이는 18장을 참고합니다.

05. JCore 연결 설정

왼쪽 코어 접속 설정에서 새 코어 연결을 추가합니다. 한 JCore 서버에 프로젝트가 여러 개 있어도 같은 서버·계정·연결 설정이라면 대상 프로젝트를 여러 개 지정할 수 있습니다. 서버나 인증 구성이 다르면 연결을 분리합니다.

항목	입력 내용
이름	사용자가 구분하기 쉬운 연결 이름
Host	JCore 서버 IP 또는 호스트 이름. http://나 경로 제외
Port	JCore TLS 접속 포트. 신규 연결 기본값 4841
Certificate SHA256	JCore 서버 인증서의 SHA256 지문 64자리
대상 프로젝트	JCore에 등록된 프로젝트 이름. 여러 개는 쉼표로 구분
계정명·비밀번호	태그 관리 및 자동 조회에 사용할 공통 JCore 계정
관리 API 주소	Host에서 http://Host:8080/api/ 형태로 자동 구성

연결 절차

1. JCore가 실행 중이고 TLS 및 관리 웹 API에 접근 가능한지 확인합니다.
2. Host·Port·Certificate SHA256을 입력합니다.
3. 대상 프로젝트를 입력합니다. 비우면 전체 프로젝트를 대상으로 합니다.
4. Admin 또는 User 계정과 비밀번호를 입력하고 설정을 저장합니다.
5. 연결을 활성화하고 프로젝트를 저장합니다.
6. 태그 읽어오기와 게시 태그 보기를 진행합니다.

현재 버전의 TLS 연결에는 유효한 SHA256 지문이 필요합니다. 임의 값이나 빈 값으로 검증을 우회하지 않습니다. 인증서를 교체했다면 실제 새 지문과 연결 설정을 일치시켜야 합니다.

계정과 연결 상태

Admin과 User는 태그 관리 권한이 같고 Guest는 제외합니다. 공통 계정 입력은 하나지만 관리 웹 토큰과 JView 실시간 세션은 내부에서 구분합니다. 자동 조회 계정 저장이 태그 쓰기 권한까지 부여하지는 않으며 조작에는 별도 운영 로그인이 필요합니다.

TLS 설정됨 · Runtime 로그인 필요는 연결 설정과 실제 운영 세션이 다르다는 의미입니다. 연결 버튼만 눌렀다고 운영 로그인과 태그 등록이 모두 끝난 것은 아닙니다.

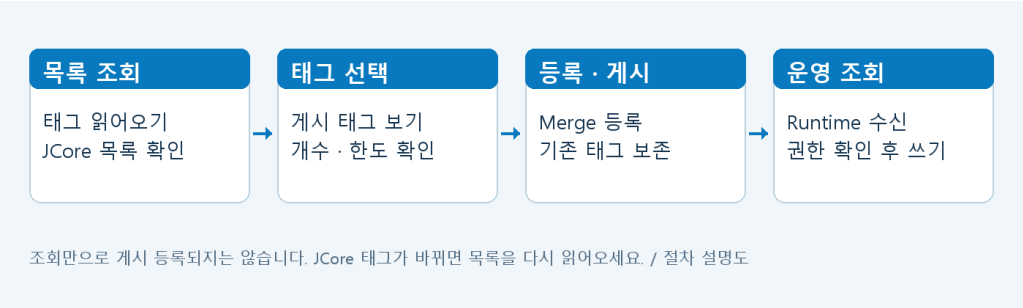
태그 관리에는 JCore master 2d9f3dd의 변경이 반영된 빌드가 필요합니다. 기존 JCore 1.0.33 설치본에는 해당 변경이 포함되지 않습니다.

06. 게시 태그 관리와 CSV

사용할 태그를 먼저 정하기

1. 코어 연결 및 계정 설정을 저장합니다.
2. 태그 읽어오기를 눌러 JCore의 프로젝트 태그 목록을 읽습니다.
3. 게시 태그 보기에서 검색·프로젝트 필터로 태그를 찾습니다.
4. 사용할 태그를 선택하고 개수와 라이선스 한도를 확인합니다.
5. 프로젝트를 저장하고 미리보기 태그 준비 또는 게시를 실행합니다.

태그 읽어오기는 목록 조회입니다. 이 동작만으로 JCore 게시 등록을 완료하지 않습니다. JCore에서 태그를 추가·수정·삭제한 후에는 다시 읽어와야 합니다. 새 태그는 기본 선택되며 기존 선택과 바인딩은 유지됩니다. 삭제·이름 변경된 태그는 누락 상태를 확인하고 수정합니다.



태그 목록에서 운영까지의 흐름

태그 경로 입력

상황	예시
단일 프로젝트	Temperature
여러 프로젝트	FactoryA.Temperature
스크립트에서 코어까지 명시	core_1234::FactoryA.Temperature

객체 바인딩은 목록 검색과 직접 입력을 모두 지원합니다. 직접 입력한 태그도 게시 전에 목록에 등록·선택되어 있어야 합니다. 스크립트는 같은 공통 게시 목록을 사용하므로 이벤트별 별도 태그 선언은 필요하지 않습니다.

CSV 가져오기·내보내기

게시 태그 보기에서 CSV를 내보내거나 가져옵니다. 열은 ConnectionId, Project, TagName, Selected, Description, DataType, CanWrite입니다. Selected는 1 또는 0입니다. 가져오기는 기존 목록에 병합하며 중복·오류 행을 표시합니다. 새 항목은 태그 읽어오기로 확인해야 하며 CSV의 CanWrite를 바꾸어도 실제 권한은 늘어나지 않습니다.

등록·선택 해제·게시 해제

일반 등록은 Merge이므로 다른 화면의 기존 게시 태그를 보존합니다. 선택 해제만으로 JCore에서 삭제되지 않습니다. 명시적인 미선택 태그 JCore 게시 해제는 Remove를 사용하며 현재 작화·스크립트·게시본에서 사용 중인 태그는 차단합니다. 다른 JView 서버의 사용 여부는 관리자가 확인해야 합니다.

라이선스 초과나 경로 오류로 등록이 실패하면 기존 설정을 유지합니다. 게시 해제로 세션이 무효화되면 자동 조회는 재연결하며 조작 계정은 다시 로그인해야 할 수 있습니다.

07. 객체 배치와 정렬

객체 생성과 선택

도구 모음에서 원하는 객체 아이콘을 누른 뒤 캔버스에서 마우스 왼쪽 버튼으로 위치를 지정합니다. 심볼도 라이브러리에서 선택한 뒤 배치합니다. 배치를 취소하려면 Esc 또는 선택 도구를 사용합니다.

객체는 화면에서 클릭하거나 왼쪽 객체 트리에서 선택할 수 있습니다. 빈 영역을 드래그하면 선택 영역에 닿는 객체를 함께 선택합니다. 겹친 객체는 객체 트리를 이용하면 선택하기 쉽습니다.

크기와 위치

속성	의미와 입력
Canvas.Left	페이지 왼쪽 기준 X 위치, 소수점 첫째 자리
Canvas.Top	페이지 위쪽 기준 Y 위치, 소수점 첫째 자리
Width / Height	객체의 너비·높이, 정수 입력
Panel.ZIndex	겹치는 객체의 표시 순서

선택 핸들을 드래그해 이동·크기를 조절할 수 있습니다. 원래 크기 모드의 이미지 객체는 크기 변경이 제한되므로 사용자 지정 크기로 바꿉니다.

여러 객체 정리

정렬 도구로 좌측·중앙·우측 또는 상단·중앙·하단을 맞춥니다. 너비 같게·높이 같게는 첫 선택 객체를 기준으로 적용합니다. 가로·세로 간격 일정하게는 여러 객체 사이의 공간을 정리할 때 사용합니다. 그룹을 만들면 연관된 객체를 함께 다룰 수 있습니다.

복사와 변환

같은 페이지에 붙여넣으면 원본과 구분되도록 위치가 이동합니다. 다른 페이지로 복사하면 원래 좌표를 유지하며 대상 화면 범위를 벗어나면 안쪽으로 조정될 수 있습니다.

변환의 좌우 반전, 위아래 반전, 90도 시계방향·반시계방향 회전 버튼을 사용합니다. 회전 각도를 직접 입력할 수도 있습니다. 실행 취소와 다시 실행으로 편집 결과를 되돌리거나 복구할 수 있습니다.

08. 공통 속성과 기능별 바인딩

일반에는 스크립트가 참조할 객체 고유명을 입력합니다. 같은 화면에서 객체를 구분할 수 있는 이름을 사용하세요. 값 태그가 있는 객체는 일반 다음의 값 태그에서 연결합니다. 추가 동작은 하단 기능별 바인딩에 설정합니다.

모양

Background는 배경, Fill / Foreground는 객체별 채움 또는 글자색, Line / BorderBrush는 외곽선입니다. 제공되는 색상 방식에서 단색·선형·방사형을 선택하고 색상이나 HEX를 지정합니다. 실제 적용 대상은 객체 종류에 따라 다릅니다.

텍스트·숫자·문자열 등의 배경 사용 안 함 아이콘은 배경을 숨겨 글자 위주로 표시합니다. 이미지 자체의 픽셀 배경을 제거하는 기능은 아닙니다.

Opacity는 0~1이며 0은 투명, 1은 불투명입니다. Visibility는 Visible 또는 Hidden입니다. 디자인에서는 숨긴 객체도 편집할 수 있도록 흐리게 보이며 런타임에서는 숨김 상태를 적용합니다.

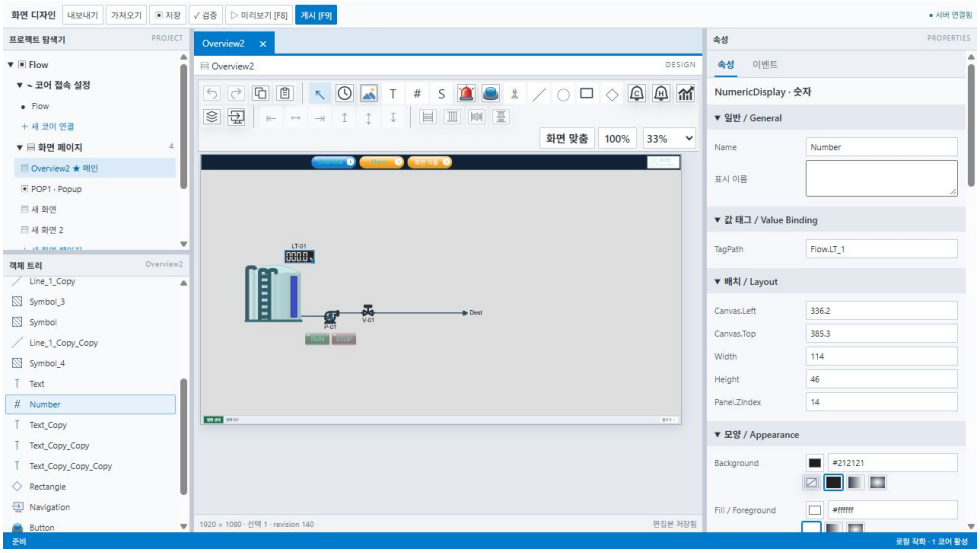
바인딩 설정 절차

1. 객체를 선택하고 기능별 바인딩에서 기능을 고릅니다.
2. 추가 버튼을 누르고 TagPath를 검색하거나 입력합니다.
3. 조건이 필요한 기능은 value == 1, value > 90 등으로 작성합니다.
4. 참·거짓일 때의 색상·문자 또는 동작을 지정합니다.
5. 저장 후 미리보기에서 실제 값을 확인합니다.

기능	표시 동작
Text	조건에 따른 문자열 표시
Calc	지원되는 계산식으로 표시값 계산
Fill / Line	조건에 따른 채움·글자색 또는 선 색상 변경
Blink	조건이 참일 때 두 색상을 지정 간격으로 전환
Visible	조건에 따른 표시·숨김
Rotate	조건에 따른 회전 방향과 속도 적용
Progress	Min~Max에 대응하는 채움 비율 표시

Progress는 별도 조건식 없이 태그 값을 범위에 비례시켜 표시합니다. Min과 Max는 서로 다르게 지정합니다. 객체별로 지원 기능이 다르며 이미지와 시계에는 기능별 바인딩을 제공하지 않습니다. 품질 불량이나 오래된 수신 값은 정상 값과 구분되므로 통신 상태도 확인합니다.

09. 숫자·문자열·램프 표시



숫자 객체의 공통 속성 화면

숫자 표시 만들기

1. NumericDisplay를 배치하고 Name을 지정합니다.
2. 값 태그에 표시할 숫자 태그를 연결합니다.
3. 표시 및 입력에서 폰트 또는 7세그먼트를 선택합니다.
4. 표시 형식 프리셋 또는 NumberFormat을 지정합니다.
5. Unit을 입력하고 전용 글꼴·크기·색상·정렬을 설정합니다.

폰트 방식은 일반 글꼴로, 7세그먼트 방식은 숫자 세그먼트로 표시합니다. 세그먼트는 배경색과 숫자색을 지정할 수 있습니다. 디자인 화면에도 형식을 적용한 샘플이 표시되며 실제 수신값은 미리보기·런타임에서 확인합니다.

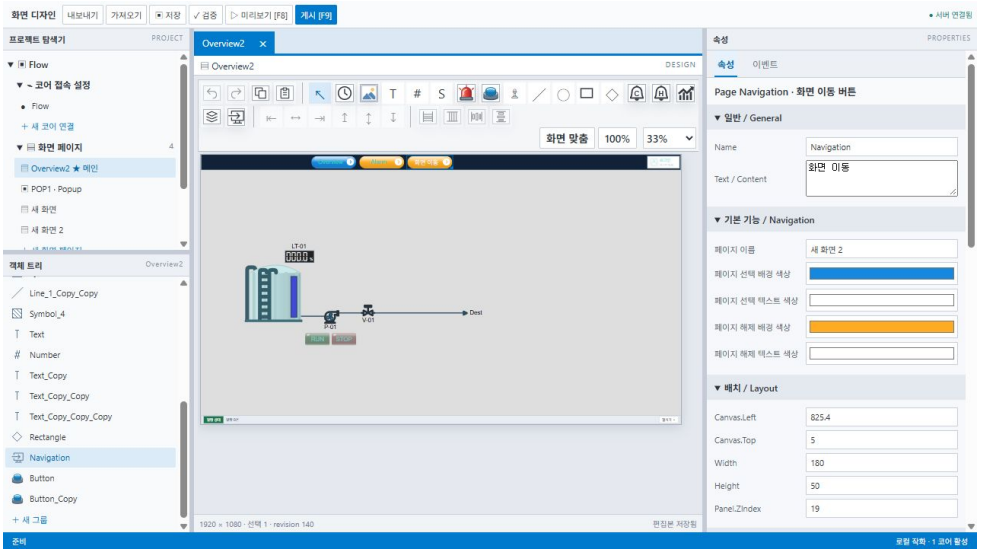
Unit은 숫자 오른쪽에 붙습니다. 상단·중단·하단 정렬은 숫자 텍스트 영역을 기준으로 적용합니다. 숫자와 별도로 Unit 글꼴과 크기를 정할 수 있습니다. 예를 들어 온도에는 °C, 압력에는 bar, 수위에는 %를 사용합니다.

문자열과 램프

StringDisplay는 값 태그의 문자열을 표시합니다. TextBlock은 고정 문구나 조건부 안내에 사용합니다. StatusLamp는 상태 태그와 Fill 등 기능별 바인딩을 연결해 정상·동작·이상 상태를 색상으로 표현합니다.

숫자 입력이 필요한 경우 해당 객체의 입력 허용 및 범위를 설정하고 런타임에서 조작 로그인을 수행합니다. 미리보기의 읽기 확인과 실제 쓰기 조작은 구분합니다. 품질 불량·세션 만료·쓰기 권한 없음 상태에서는 입력이 제한될 수 있습니다.

10. 버튼·페이지 이동·로그인



페이지 이동 버튼의 기본 기능과 상태별 색상

일반 버튼

일반 버튼은 필요한 기본 동작이나 객체 OnClick 이벤트를 설정해 사용합니다. 스크립트만 실행하려면 기본 Action을 없음으로 설정합니다. 기본 동작과 이벤트를 함께 설정하면 둘 다 실행될 수 있습니다.

페이지 이동 버튼

페이지 이동 전용 객체를 추가하고 기본 기능의 페이지 이름에 대상 화면을 입력합니다. 현재 페이지 여부에 따라 선택·해제 배경색과 텍스트 색상을 각각 지정할 수 있습니다. 존재하지 않는 대상은 실제 클릭 때 오류가 발생하므로 이름을 확인합니다.

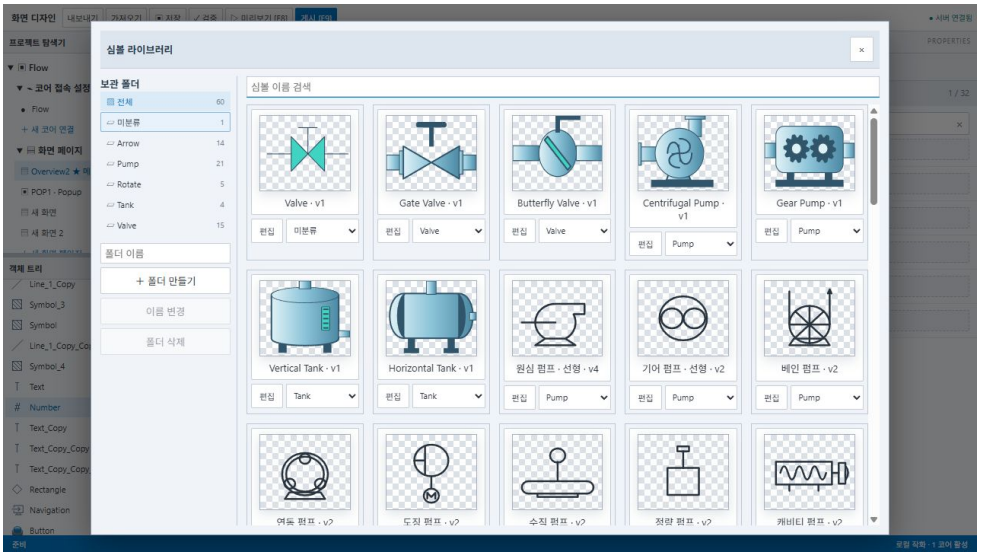
공통 버튼 스타일

일반 버튼과 페이지 이동 버튼은 기본, 입체 프레임, 광택 · 화살표, 글래스, 젤 · 둥근형, 젤 · 사각형 스타일을 제공합니다. 지정한 색상을 바탕으로 음영과 광택을 표현합니다. 일반 버튼의 Fill 바인딩으로 색이 변해도 스타일 형태는 유지됩니다. 페이지 이동 버튼은 페이지 선택·해제 전용 색상을 기준으로 사용합니다.

로그인 객체

로그인 객체를 운영 화면에 배치하면 사용자가 로그인 창을 열 수 있습니다. 자동 조회가 정상이어도 쓰기 작업에는 조작 계정의 권한이 필요합니다. 조작 전 대상 태그, 입력값, 로그인 상태를 확인하고 완료 후 필요하면 로그아웃합니다.

11. 심볼·이미지·시계



폴더별 심볼 라이브러리

심볼

라이브러리에서 Pump, Valve, Tank, Rotate, Arrow 등의 폴더와 검색을 이용합니다. 원하는 심볼을 선택하고 캔버스 위치를 클릭합니다. 원본 모양을 편집할 때는 라이브러리의 편집을 사용합니다. 내부 부품 색상에서 초기 외부 색상을 가져오며, 이후 객체 모양과 지원 바인딩으로 색상을 조절할 수 있습니다.

이미지

Image 객체를 배치하고 불러오기로 PNG·JPG·WebP 파일을 선택합니다. 파일당 2 MB 이하, 한 변 8192px 이하, 전체 16,777,216픽셀 이하를 지원합니다. 원래 크기는 이미지 해상도를 사용하며 사용자 지정 크기는 드래그 및 Width·Height 조절이 가능합니다. 원본이 페이지보다 크면 사용자 지정 크기로 맞춥니다.

이미지는 프로젝트에 포함됩니다. 이미지에 색상·텍스트·기능별 바인딩을 추가하지 않습니다. 자동 배경 제거는 제공하지 않습니다. 투명 PNG는 투명도를 유지하지만 체크무늬가 픽셀로 그려진 이미지는 체크무늬도 그대로 나타납니다.

서버 시계

Clock 객체는 JView 서버의 시각과 시간대 기준으로 표시합니다. 추천 포맷을 선택하거나 포맷을 직접 수정합니다. 기본 형식은 YYYY-MM-DD hh:mm:ss입니다. MM은 월, mm은 분, ss는 초이며 hh 또는 HH는 24시간, hh12는 12시간입니다. ddd는 요일, A는 오전·오후입니다.

날짜만은 YYYY-MM-DD, 시간만은 hh:mm:ss, 12시간 표시는 A hh12:mm:ss를 사용합니다. 서버와 주기적으로 동기화하며 정밀 계측용 시각 장치로 사용하지 않습니다.

12. 팝업 화면

팝업은 현재 기본 화면 위에 상세 정보나 설정 화면을 띄울 때 사용합니다. 기본 화면과 같은 방식으로 객체를 배치하되 화면 타입을 Popup으로 설정합니다.

팝업 구성 순서

1. Popup 페이지를 만들고 이름을 POP1 등으로 지정합니다.
2. 상세 표시 객체를 배치하고 태그를 연결합니다.
3. 호출 버튼이나 스크립트에 팝업 대상과 위치를 설정합니다.
4. 저장하고 미리보기에서 표시 위치와 크기를 확인합니다.
5. 닫기 동작과 중복 호출 시 동작을 확인합니다.

스크립트 호출 위치

```
var popup = View.Popups.Open("POP1", new {
    Position = "Coordinates", X = 300, Y = 200
});
```

Coordinates의 X·Y는 런타임 표시 영역 왼쪽 위 기준 CSS 픽셀입니다. 화면 밖의 위치는 표시 영역 안으로 제한합니다. Position을 Anchor로 설정하면 TopLeft, TopCenter, TopRight, MiddleLeft, Center, MiddleRight, BottomLeft, BottomCenter, BottomRight 중 하나를 사용합니다.

```
View.Popups.Open("POP1", new {
    Position = "Anchor", Anchor = "BottomRight"
});
```

Mouse는 마지막으로 확인한 포인터 위치를 사용하고 기록이 없으면 중앙에 표시합니다. 옵션을 생략하면 중앙입니다. 따라서 글로벌 이벤트에서 호출하더라도 위치 옵션을 명시하면 의도한 위치로 열 수 있습니다.

호출 결과의 InstanceId를 View.Popups.Close에 전달하면 해당 팝업을 닫습니다. 팝업 자체의 이벤트에서는 View.Close()를 사용할 수 있습니다. 동시 팝업은 최대 16개이며 없는 대상·Popup이 아닌 대상·잘못된 옵션은 오류로 표시됩니다.

13. 현재 알람과 알람 이력

현재 알람

Current Alarm을 배치하고 알람 코어를 선택합니다. 기본값은 모든 코어이며 활성 연결이 대상입니다. 현재 알람은 화면 위 또는 아래의 공통 알람 바와 접이식 리스트로 표시됩니다. 바를 닫아도 알람 수신과 건수는 갱신합니다.

바의 위치·높이·배경·발생 및 정상 색상을 설정하고 펼친 리스트 높이를 조절합니다. 펼친 목록은 작화 위에 겹쳐 표시되므로 화면 구성 시 공간을 고려합니다. 현재 알람에는 기간 조회가 없습니다.

코어 연결에 대상 프로젝트가 있으면 그 프로젝트를 사용합니다. 비어 있으면 바인딩과 보관된 게시 경로에서 프로젝트를 찾습니다. 기대한 프로젝트가 조회되지 않으면 연결의 대상 프로젝트를 명시합니다. JCore는 해당 프로젝트의 게시된 태그에 대한 알람을 반환합니다.

알람 이력

Historical Alarm은 과거 발생 기록을 조회할 때 사용합니다. 조회할 코어·프로젝트 등 객체가 제공하는 항목을 설정하고 저장합니다. 운영 화면에서 발생기간을 선택하고 조회한 뒤 태그명·해제시간 등으로 결과를 확인합니다.

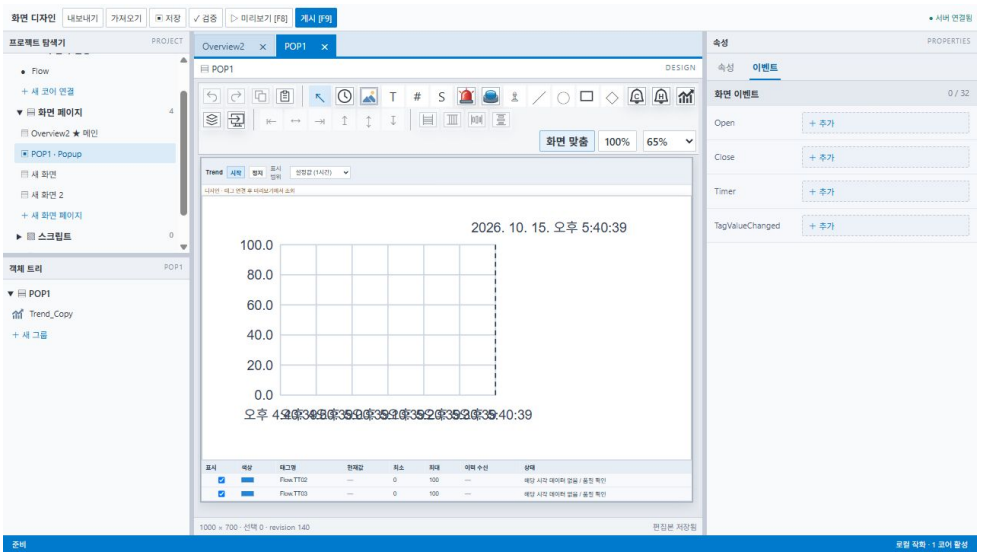
이력은 JCore에 기록된 데이터를 조회합니다. 현재 알람이 정상화되었다고 해서 과거 발생 이력이 자동으로 사라지는 것은 아닙니다. 예상 기록이 없으면 조회기간, 프로젝트, 기록 여부, 조회 오류를 순서대로 확인합니다.

결과 해석

알람 0건과 조회 실패는 다릅니다. 일부 조회 실패·조회 한도 도달 안내가 있으면 전체 기록을 다 받은 것으로 판단하지 않습니다. 목록의 정렬은 불러온 결과를 기준으로 합니다. 기간을 나누어 조회하면 필요한 기록을 찾기 쉽습니다.

현재 알람을 여러 개 구성해도 공통 바와 합산 목록을 사용합니다. 변경한 알람 코어와 조회 설정은 저장 후 미리보기에, 게시 후 운영 화면에 적용됩니다.

14. 실시간·이력 트렌드



트렌드를 배치한 팝업의 디자인 예시. 실제 수신 전 화면

Trend 객체는 여러 태그 값을 펜으로 표시합니다. 실제 태그 경로와 색상·굵기·축 범위를 설정한 뒤 저장합니다. 같은 프로젝트·태그 이름이라도 코어 연결이 다르면 다른 데이터입니다.

설정 순서

1. Trend 객체를 배치하고 Real 또는 Historical 모드를 정합니다.
2. 펜마다 코어 연결·프로젝트·태그를 지정합니다.
3. 색상·굵기·Min·Max 및 표시 여부를 설정합니다.
4. 프로젝트를 저장하고 미리보기에서 수신을 확인합니다.
5. 표시 기간과 펜 수를 실제 사용 목적에 맞게 조정합니다.

조회와 탐색

시작을 누르면 최신 시간으로 이동하며 데이터를 보충합니다. 정지는 현재 표시 시간과 데이터를 유지하고 JCore의 기록 동작을 중지하지 않습니다. 정지 상태에서는 기간 검색을 사용할 수 있습니다. 실시간 모드에서도 표시 기간을 조절할 수 있습니다.

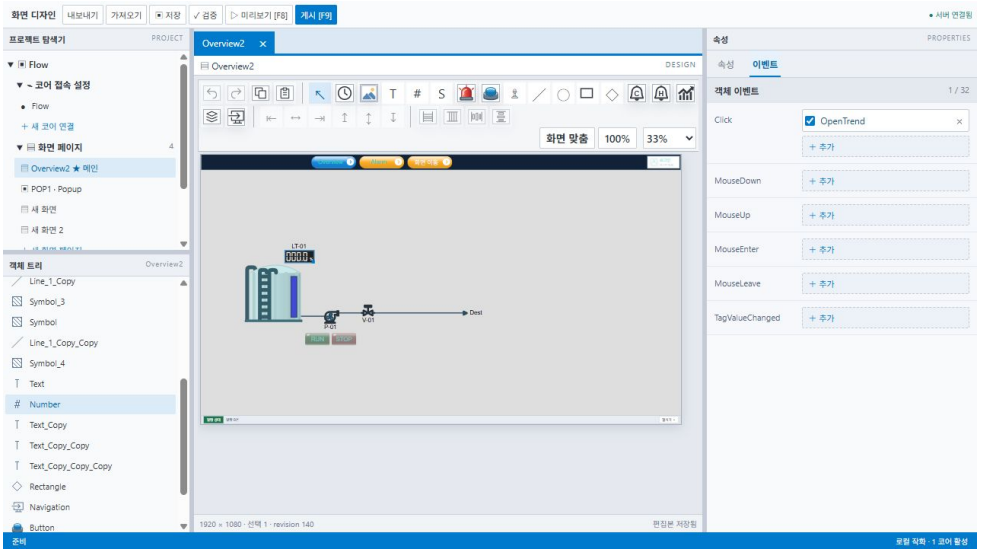
그래프를 드래그하면 자동으로 정지한 후 선택 구간을 확대합니다. 우클릭 메뉴의 확대 초기화로 원래 범위로 돌아갑니다. 하단 펜 그리드에서 표시 여부, 색상, 경로, 현재값·커서값, 축 범위 및 오류를 확인합니다.

표시 품질과 성능

현재 펜 라인은 샘플 점 마커 없이 선으로 표시합니다. 점이 없다는 것과 데이터가 연속으로 수신되었다는 것은 별개입니다. 통신 중단 구간이나 품질 오류는 시간축과 조회 상태를 함께 확인합니다.

조회기간·펜 수·샘플 수를 늘리면 브라우저 메모리와 통신·그리기 부하가 증가할 수 있습니다. 필요한 기간으로 좁히고 사용하지 않는 펜을 정리합니다. 장시간 기록의 원본 보관과 기록 주기는 JCore에서 관리합니다.

15. 이벤트와 C# 스크립트



객체의 이벤트 탭

범위	이벤트	실행 수명
객체	OnClick, OnMouseDown/Up/Enter/Leave, OnTagValueChanged	해당 페이지·팝업 인스턴스
화면	OnOpen, OnClose, OnTimer, OnTagValueChanged	해당 페이지·팝업 인스턴스
글로벌	OnStart, OnStop, OnTimer, OnTagValueChanged	런타임 브라우저 탭

객체는 선택 후 이벤트 탭에서 추가합니다. 화면 이벤트는 페이지를 선택하고 이벤트 탭에서 편집합니다. 글로벌 이벤트는 왼쪽 스크립트 노드에서 추가합니다. 기존 이벤트 이름을 누르면 코드 편집기가 열립니다.

작성과 저장

이벤트 이름·종류·사용 여부·실행 제한을 지정하고 코드를 작성합니다. 타이머는 인터벌을 설정합니다. 적용 후 프로젝트 저장을 해야 디스크에 보관되며 운영 반영에는 게시가 필요합니다. 별도 .cs 파일은 생성하지 않습니다.

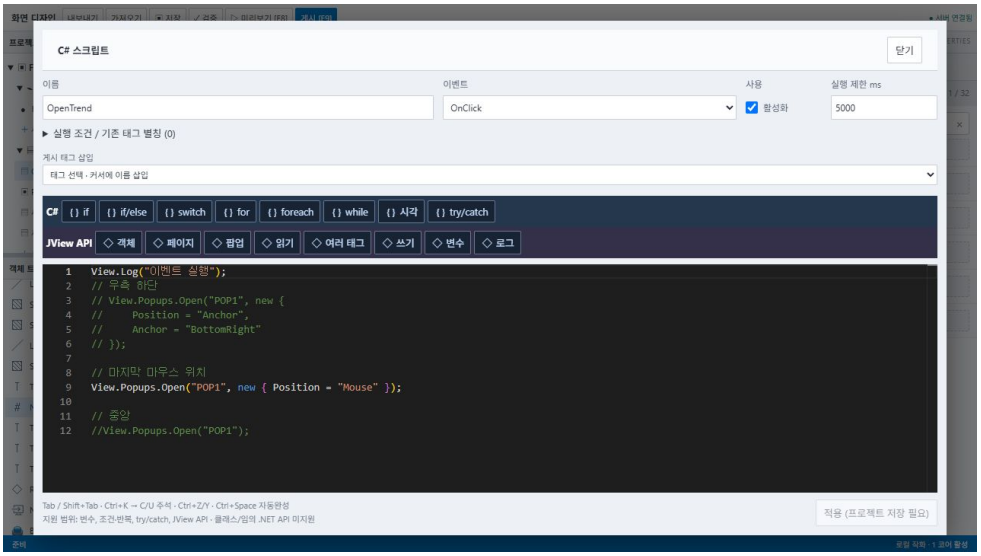
글로벌 타이머 주의점

글로벌 타이머는 페이지를 이동해도 유지되지만 서버에서 한 번만 실행되는 공통 작업이 아닙니다. 브라우저 탭마다 독립적으로 실행합니다. 탭을 닫거나 새로 고침하면 변수와 임시 상태가 초기화됩니다.

인터벌은 100~86,400,000ms, 실행 제한은 100~30,000ms입니다. 같은 이벤트가 실행 중이면 다음 호출을 건너뛰며 동시 실행에도 제한이 있습니다. 브라우저 비활성화 시 타이머가 지연될 수 있으므로 정밀 제어와 인터록에는 사용하지 않습니다.

OnClose·OnStop은 종료 시 완료를 보장하지 않습니다. 종료 이벤트에 태그 쓰기를 의존시키지 마세요. 태그 변경 이벤트는 정상 품질의 값이 바뀔 때 발생하며 최초 샘플은 변경으로 처리하지 않습니다.

16. 스크립트 API와 예제



C# 문법을 사용하는 이벤트 편집기

편집기는 C# 구문 강조·줄 번호·자동 들여쓰기·예제 삽입·자동완성을 제공합니다. 적용·저장 시 오류의 행과 열을 확인하고 오류 항목으로 이동할 수 있습니다. 전체 .NET C# 실행기가 아니라 지원되는 이벤트 본문 문법과 API를 해석합니다. 임의 DLL·using·클래스·LINQ 등은 지원하지 않습니다.

API	용도
View.Objects.Get / Set / Reset	현재 범위 객체 조회·임시 속성 변경·초기화
View.Pages.Open	기본 화면 이름 또는 ID로 이동
View.Popups.Open / Close	팝업 열기·인스턴스 닫기
View.Tags.ReadAsync / ReadManyAsync	정상 품질의 최신 수신 샘플 읽기
View.Tags.WriteAsync	게시 태그에 권한 검증 후 쓰기
View.Vars.Get / Set	현재 실행 범위의 변수
View.Global.Get / Set	같은 런타임 탭에서 공유하는 변수
View.Log	하단 스크립트 로그 표시, 최근 100건

값 읽기와 모양 변경

```
var sample = await View.Tags.ReadAsync("Temperature");
if (sample.Value > 80) {
    View.Objects.Set("Lamp1", new {
        Fill = "#ff0000", Opacity = 1,
        Visibility = "Visible"
    });
} else {
    View.Objects.Set("Lamp1", new { Fill = "#00aa00" });
}
```

Temperature는 선택된 게시 태그, Lamp1은 현재 화면의 실제 객체 이름으로 바꿉니다. 다중 프로젝트라면 FactoryA.Temperature처럼 작성합니다. 객체 속성 변경은 런타임 임시 변경이며 원본 작화를 저장하지 않습니다.

쓰기 예: await View.Tags.WriteAsync("Setpoint", 50); 실제 쓰기는 조작 로그인·권한·품질·최신 게시 상태 검증을 통과해야 하며 미리보기에서는 허용하지 않습니다. 결과를 알 수 없는 쓰기를 타이머로 반복 재시도하지 말고 실제 태그 상태를 확인합니다.

17. 미리보기·게시·운영

미리보기

1. 프로젝트를 저장하고 검증합니다.
2. 미리보기 [F8]를 누릅니다.
3. 배치·색상·태그 표시·페이지 및 팝업 동작을 확인합니다.
4. 미등록 태그 안내가 나타나면 미리보기 태그 준비를 실행합니다.
5. 조회 오류를 해결한 뒤 운영 게시를 진행합니다.

미리보기 태그 준비는 저장된 선택 태그를 JCore에 등록하며 운영 화면 게시 버전을 변경하지 않습니다. 태그 읽어오기만 했을 때 미등록 안내가 나오는 것은 이 등록 단계가 남아 있기 때문일 수 있습니다.

게시 및 HTML 저장

저장 완료 후 게시 [F9]를 누르고 서버 IP 또는 도메인, 포트, 프로토콜을 확인합니다. 다른 PC에서 사용할 파일이라면 localhost 대신 실제 접근 가능한 서버 주소를 지정합니다.

지원 브라우저는 HTML 저장 위치 선택 창을 열며 기본 위치로 바탕화면을 제안합니다. 저장 위치 선택을 지원하지 않는 환경에서는 브라우저 다운로드 설정을 따릅니다. 원격 HTTP 등 브라우저 조건에 따라 동작이 달라질 수 있습니다. 게시 후 저장만 실패한 경우 HTML 다시 저장을 사용합니다.

운영과 전체화면

HTML 또는 /runtime 주소로 접속합니다. 서버가 실행 중이어야 하며 자동 조회 설정과 필요 시 운영 로그인이 적용됩니다. 브라우저가 허용하는 전체화면 진입 동작을 사용합니다. 일반 웹 페이지는 브라우저 정책상 사용자 조작 없이 전체화면 진입을 보장할 수 없습니다.

전체화면에서는 상단 끝으로 마우스를 이동해 종료 바의 X로 웹 화면으로 돌아갈 수 있습니다. 브라우저의 Esc도 사용할 수 있습니다.

디자이너 웹 아이콘은 파란 D, 런타임과 게시 HTML의 웹 아이콘은 빨간 R입니다. Windows 탐색기에서 HTML 파일에 표시되는 아이콘은 기본 브라우저 연결 설정을 따르며 문서 안의 웹 아이콘과 다를 수 있습니다.

18. 백업·이전·업데이트

프로젝트 파일 백업

디자이너 상단 내보내기로 .jview 파일을 저장합니다. 화면·객체·심볼·이벤트·게시 태그 목록과 프로젝트의 연결 설정이 포함됩니다. 비밀번호, 개인키, 세션 토큰, 서버 로그는 포함되지 않습니다. 현재 형식의 프로젝트 파일 크기 제한은 8 MiB입니다. 이미지가 많으면 파일 크기를 확인합니다.

다른 PC로 이전

1. 원본 PC에서 프로젝트를 내보냅니다.
2. 대상 PC에 JView를 설치하고 디자이너를 엽니다.
3. 기존 대상 프로젝트가 있다면 먼저 내보내 백업합니다.
4. 가져오기에서 .jview 파일을 선택하고 편집본 교체를 확인합니다.
5. 대상 환경에 맞게 코어 Host·지문·계정 비밀번호를 재확인합니다.
6. 태그 읽어오기·미리보기·게시를 차례로 실행합니다.

저장된 비밀번호는 Windows 계정의 DPAPI 보호를 사용합니다. 개발 계정에서 만든 암호 파일을 LocalService의 설치 환경에 복사하는 것으로 대체하지 않습니다. PC 또는 서비스 계정이 바뀌면 비밀번호를 다시 입력합니다.

전체 운영 데이터 백업

프로젝트 파일 외에 게시 이력·서버 설정 등을 보관하려면 데이터 폴더 전체를 별도로 백업합니다. 파일 일관성이 필요한 백업은 점검 시간에 서비스를 중지한 상태에서 수행합니다. 서비스 계정과 PC에 종속된 암호 정보는 단순 복원만으로 사용 가능하다고 가정하지 않습니다.

업데이트

1. 현재 버전과 프로젝트 백업을 확인합니다.
2. 점검 시간에 더 높은 버전의 설치파일을 실행합니다.
3. 설치 중 서비스가 중지·교체·재시작되므로 일시 통신 중단을 예상합니다.
4. 완료 후 서비스, 디자이너, 런타임, 태그 수신을 확인합니다.
5. JSolution 진단 수집 상태도 확인합니다.

운영 데이터는 업데이트 시 보존됩니다. 설치 실패 시 롤백이 수행될 수 있으며 실패 원인은 서비스 로그와 설치 로그로 확인합니다. DLL만 일부 수동 교체하면 버전 혼합이 생길 수 있으므로 정식 설치 패키지를 사용합니다.

19. JSolution 진단과 문제 해결

1.0.9에는 Diagnostic SDK 1.1 Provider가 포함됩니다. JSolution의 수집 성공과 제품의 상태 표시는 서로 다른 정보입니다.

항목	설치 환경의 기본값
ProductId / SchemaVersion	JView / 1.1
API 기본 주소	http://127.0.0.1:5080/
Manifest	/api/diagnostics/v1/manifest
Health	/api/diagnostics/v1/health
보호 키	%ProgramData%₩JAYON STUDIO₩JView₩Diagnostics₩Keys₩read-key.dat
진단 저널	데이터 폴더의 Diagnostics₩Journal
서비스 로그	데이터 폴더의 logs₩service.log 및 service.log.1

API는 동일 PC의 loopback 요청과 보호 키 인증이 필요합니다. 브라우저로 주소만 열어 인증 거부가 나오는 것은 정상일 수 있습니다. 키 평문을 문서나 외부 PC로 전달하지 않습니다. 개발 실행은 저장소 data 아래의 키를 사용하며 데이터 폴더를 별도 지정했다면 그 경로를 사용합니다.

기능 선언은 health.snapshot, events.incremental, events.history, events.access입니다. 서버·프로젝트·웹·운영 세션·JCore 관측 상태와 이벤트를 제공합니다. 개발 실행을 수집하려면 JSolution에서 Enabled와 CollectWhenUninstalled 설정을 확인합니다.

확인 불가의 의미

JCore 활성 lease 미관측은 아직 유효한 JCore 운영 세션을 확인하지 못했다는 뜻입니다. 진단 수집이 성공해도 JCore 상태는 Unknown일 수 있습니다. JView 서버의 RUNNING/Healthy와 구분합니다. Windows 서비스의 중지 여부는 JView 자체 API가 아닌 서비스 관리자에서도 확인해야 합니다.

자주 발생하는 문제

증상	확인 순서
태그 읽어오기 비활성	설정·프로젝트 저장, 연결 활성화, 공통 계정과 저장 비밀번호 확인
미리보기 미등록 태그	선택 목록 저장 후 미리보기 태그 준비 실행
태그 쓰기 불가	조작 로그인, 쓰기 권한, 태그 품질·신선도, 최신 게시본 확인
외부 PC 접속 실패	서버 IP·5080 포트·방화벽 확인. localhost 사용 여부 확인
HTTP 403 / 503	실제 요청 주소·서비스 상태·URL 예약·로그 확인
서비스 시작 예외	service.log의 전체 예외와 DLL 버전·해시 확인
진단 API 503	진단 초기화·키 파일 접근 오류와 서비스 로그 확인

관리자 PowerShell에서 Get-Service JView.Server, netsh http show servicestate, netsh http show urlacl로 상태를 조회할 수 있습니다. 오류 원인을 확인하기 전에 URL 예약을 일괄 삭제하지 마세요. 디버그 실행의 초기화 메시지는 서비스 파일 로그 대신 콘솔로 나올 수 있습니다.

20. 단축키와 운영 점검표

디자이너 단축키

키	기능
Ctrl+S	프로젝트 저장
F8	미리보기
F9	게시 설정 열기
Ctrl+Z / Ctrl+Y	실행 취소 / 다시 실행
Ctrl+C / Ctrl+V	객체 복사 / 붙여넣기
Delete	선택 객체 삭제
방향키 / Shift+방향키	선택 객체 1 / 10 단위 이동
Ctrl+휠	확대·축소
Esc	객체 배치 취소 등 현재 편집 동작 종료

입력란이나 스크립트 편집기에 포커스가 있으면 해당 편집 동작이 우선할 수 있습니다. F5는 미리보기 단축키가 아닙니다.

스크립트 편집기 단축키

키	기능
Enter	줄바꿈과 자동 들여쓰기
Tab / Shift+Tab	들여쓰기 / 내어쓰기
Ctrl+K → Ctrl+C	주석 처리
Ctrl+K → Ctrl+U	주석 해제
Ctrl+Space	자동완성
Ctrl+Z / Ctrl+Y	편집 실행 취소 / 다시 실행

게시 전 확인

- 코어 주소·인증서·계정이 실제 운영 서버와 일치한다.
- 사용할 게시 태그가 선택되어 있고 라이선스 한도 이내다.
- 페이지 이동 대상·팝업 위치·숫자 형식·Unit을 확인했다.
- 객체 이름과 스크립트 태그 경로가 일치한다.
- 저장·검증·미리보기를 완료했다.
- 운영 접속 PC에서 사용할 서버 주소로 게시했다.

운영 및 업데이트 후 확인

- 서비스 실행과 브라우저 접속이 정상이다.
- 태그 수신·알람·트렌드의 조회 오류가 없다.
- 필요한 조작 계정으로 로그인하고 권한을 확인했다.
- 진단 수집 성공 시각과 제품 상태를 구분해 확인했다.
- 현재 릴리즈의 프로젝트 백업을 보관했다.

문서 기준

작성 기준: JView 1.0.9의 디자이너·서버·계약 모델 및 설치 스크립트, 게시 태그·스크립트·진단 연동 문서. 현장별 IP·계정·태그·라이선스·JCore 구성은 운영 환경에 맞게 적용합니다. 그림과 함께 이동하려면 Markdown 파일과 images 폴더를 같은 위치에 보관하세요.