

**codingOn x posco**  
**K-Digital Training** 스마트 팩토리 단기 8기

# 5. 사용자 정의 평선



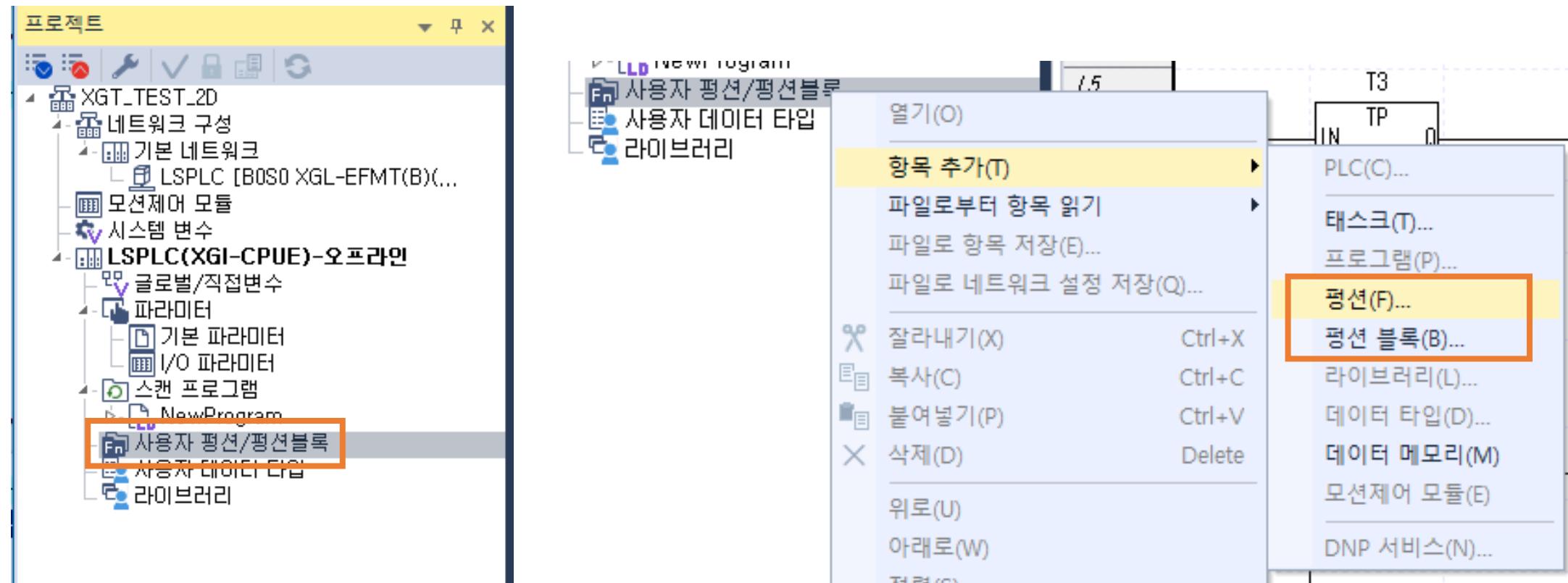
# 사용자 정의 평션

- 기존의 평션/평션블록 외에 사용자가 직접 만드는 평션/평션 블록
  - 사용자가 직접 평션 또는 평션블록을 디자인 하여 동일한 로직을 재활용
  - 사용자 정의로 로직을 숨김으로써 노하우 보호
  - 프로그램 메모리를 효과적으로 활용할 수 있음



# 사용자 정의 평션

- 사용자가 직접 내용을 정의할 수 있는 평션

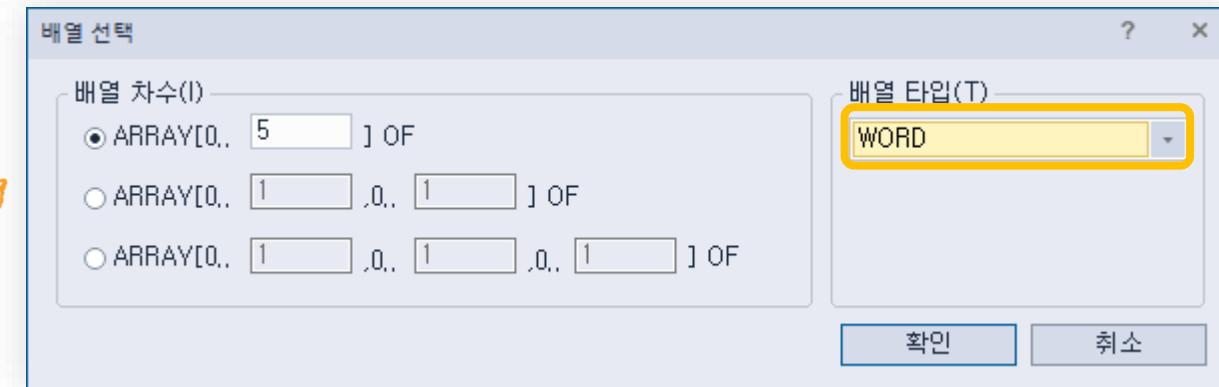


# 사용자 정의 평선

- 이름, 데이터 타입, 리턴 타입 정의



제작하려는 기능별로 자유롭게 선택 가능



# 사용자 정의 평선

- 평선에서 사용할 입력, 출력 변수 정의

The screenshot shows a programming interface with two main sections: a variable declaration table and a block diagram.

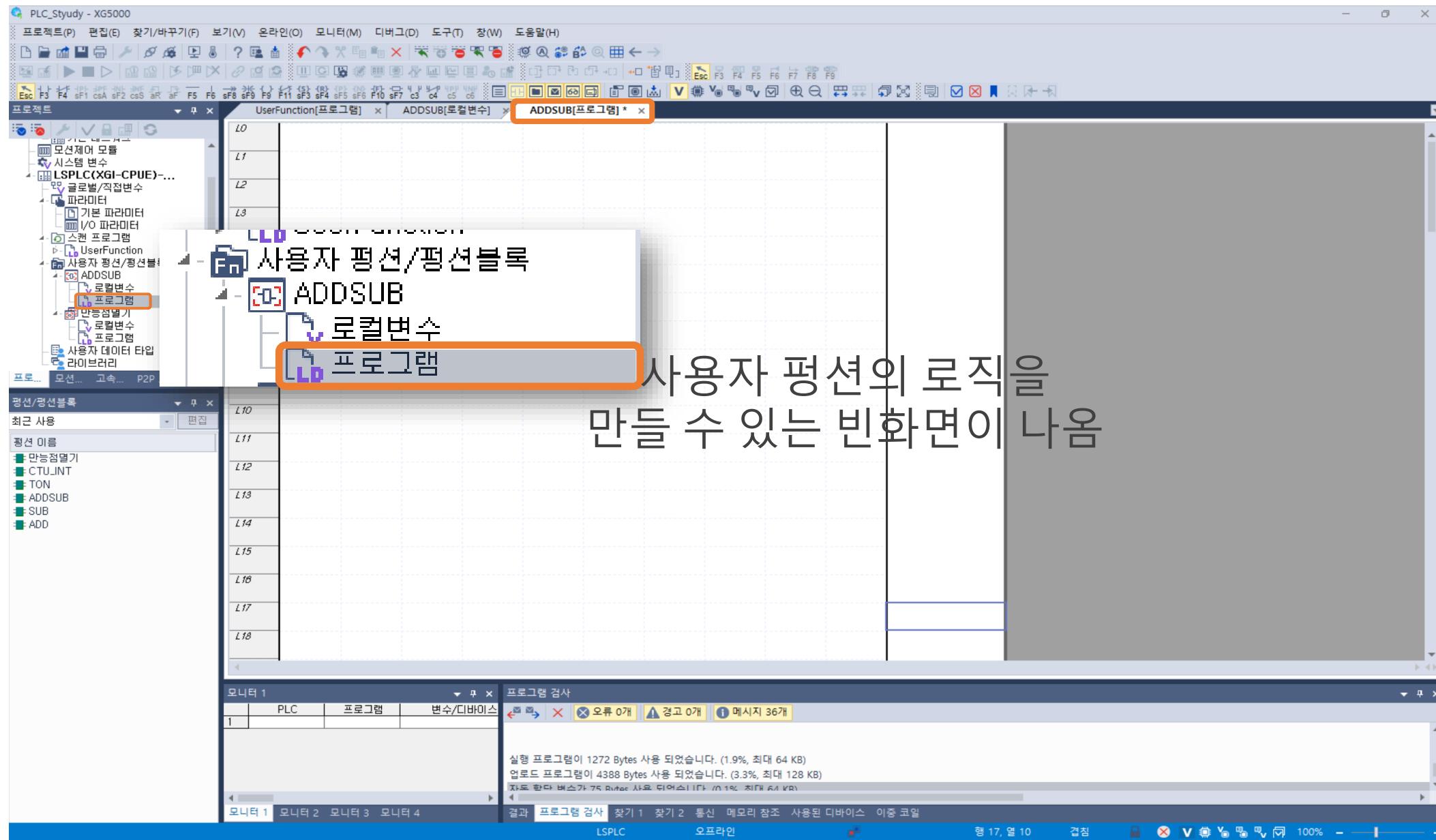
**Variable Declaration Table:**

	변수 종류	변수	타입	사용 유무	설명문
1	VAR_INPUT	입력A	INT	<input checked="" type="checkbox"/>	
2	VAR_INPUT	입력B	INT	<input checked="" type="checkbox"/>	
3	VAR_INPUT	입력C	INT	<input checked="" type="checkbox"/>	
4	VAR_RETURN	ADDSUB	INT	<input checked="" type="checkbox"/>	
5	VAR	임시저장	INT	<input checked="" type="checkbox"/>	

**Block Diagram:**

```
graph LR; EN[BOOL - EN] --> A[INT - 입력A]; A --> ADDSUB[ADDSUB]; ADDSUB --> B[INT - 입력B]; B --> C[INT - 입력C]; EN --- ENO[ENO - BOOL]; ADDSUB --- INTB[INT - B]
```

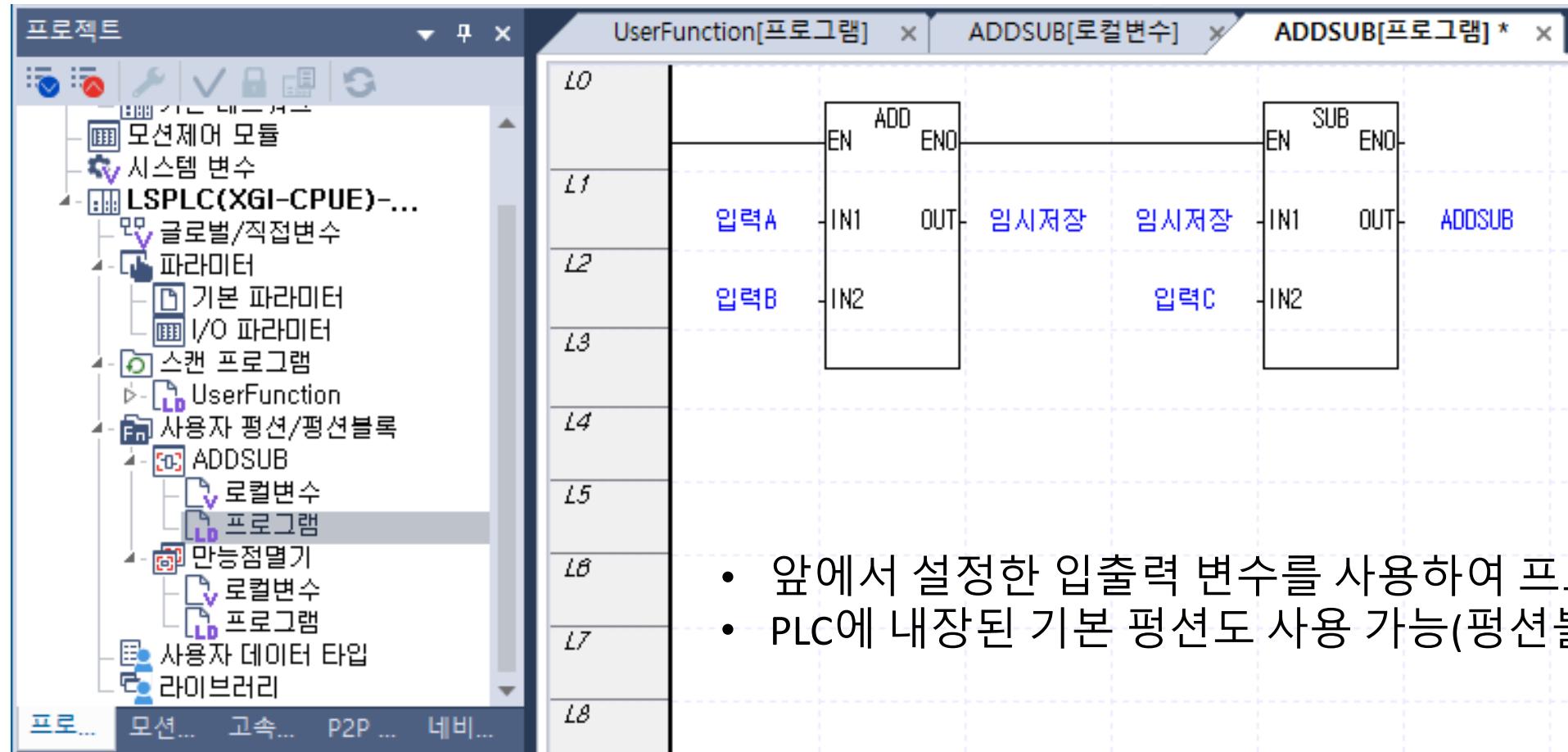
The block diagram illustrates the data flow: an enable signal (EN) triggers the ADDSUB block, which takes three integer inputs (A, B, and C) and produces one integer output (ENO).



사용자 평선의 로직을  
만들 수 있는 빈화면이 나옴

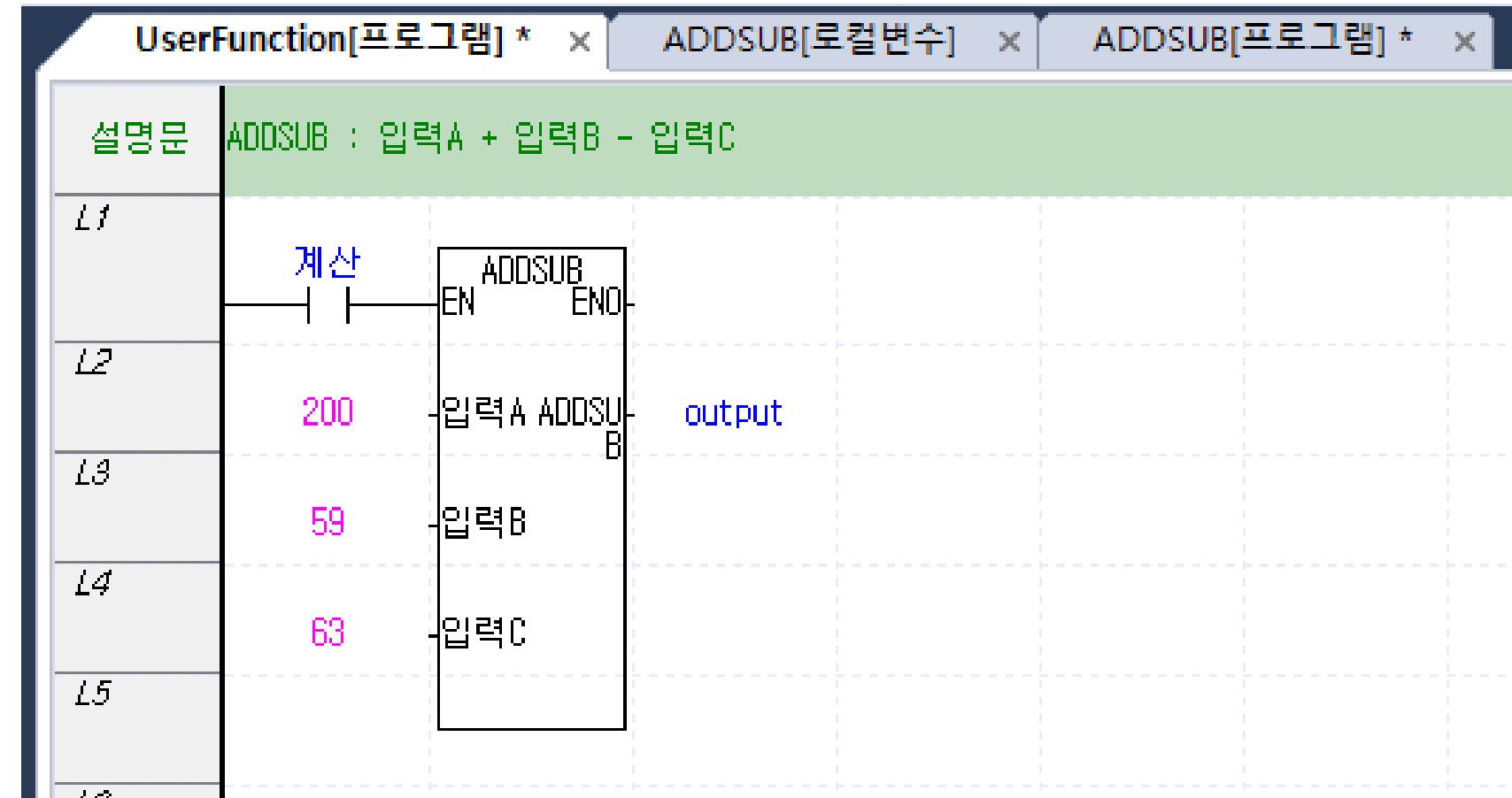
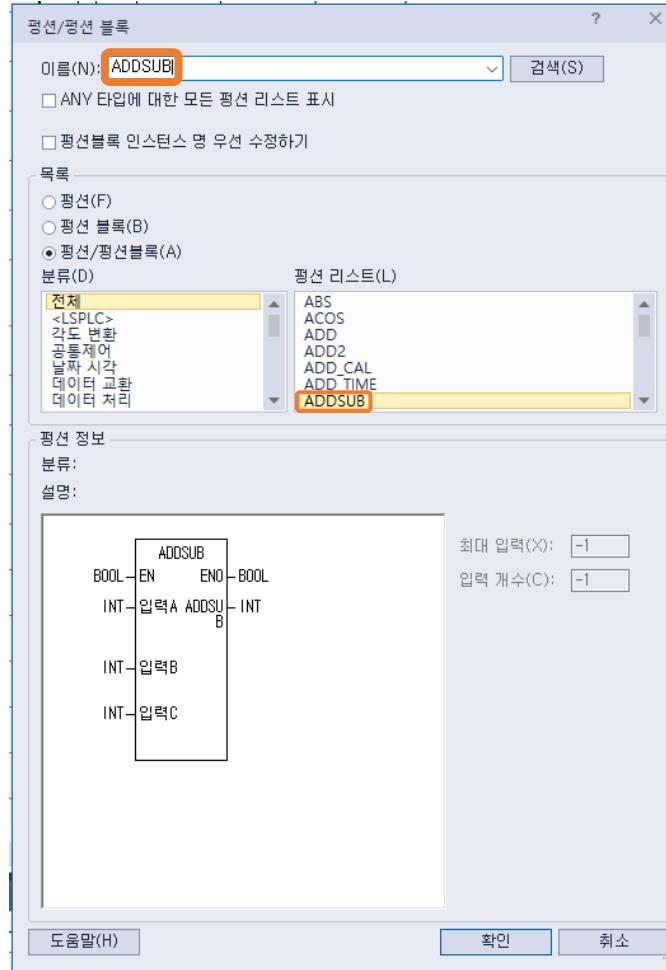
# 사용자 정의 평선

- 평선 내용은 래더 프로그램으로 작성



# 사용자 정의 평선

- 프로그램에서 불러와서 사용 가능



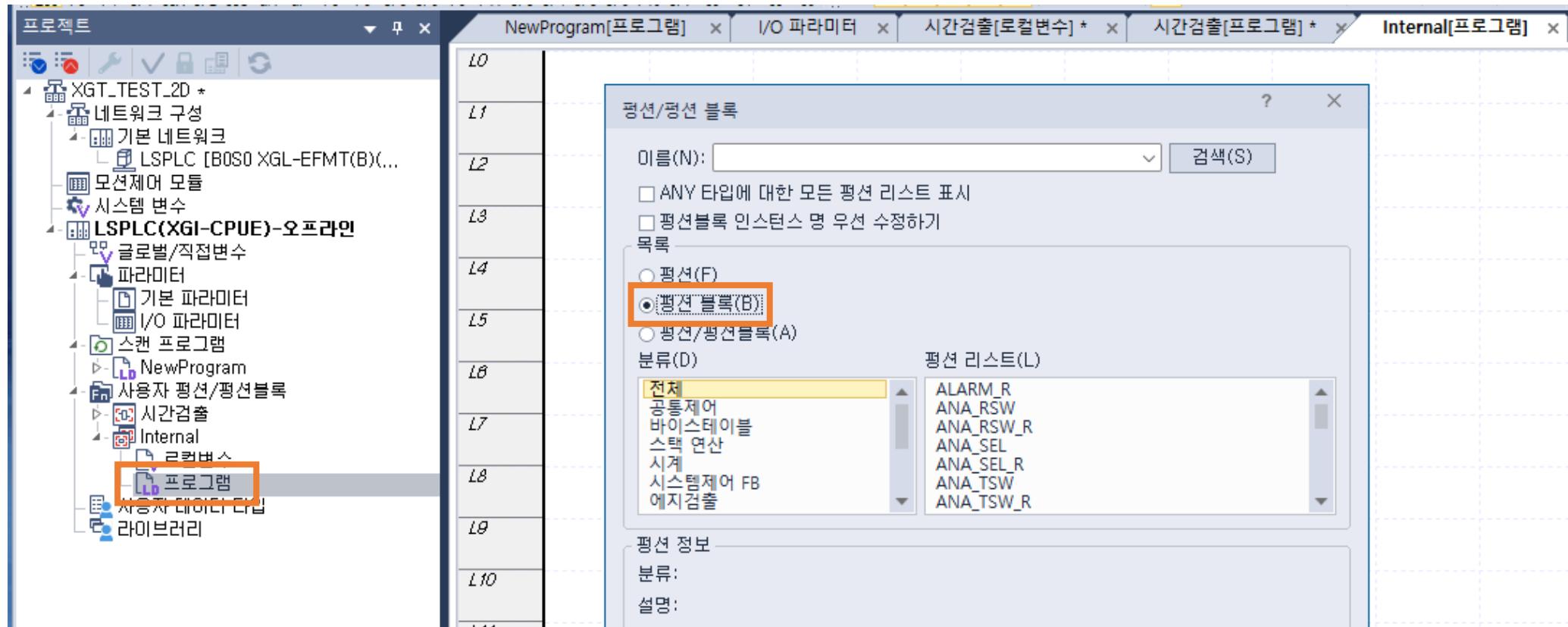
# 실습1. 피타고라스 정리

- PITA라는 이름의 사용자 정의 평션 만들기
- A변, B변이라는 입력에 값을 넣으면, 피타고라스 정리에 의해 Result라는 결과값을 출력
  - ※ 참고 : 피타고라스 정리  
- 밑변 a, 높이 b인 직각삼각형의 빗변 c를 구할 때
- 데이터 타입은 모두 Real로 진행
- SQRT(루트) 평션 사용
- 결과 예시 : A변 = 3, B변 = 4 입력 → Result = 5 출력

$$a^2 + b^2 = c^2$$

# 사용자 정의 평션 블록

- 사용자 정의 평션과 동일하나 프로그램 작성시 내장된 평션 블록도 사용 가능



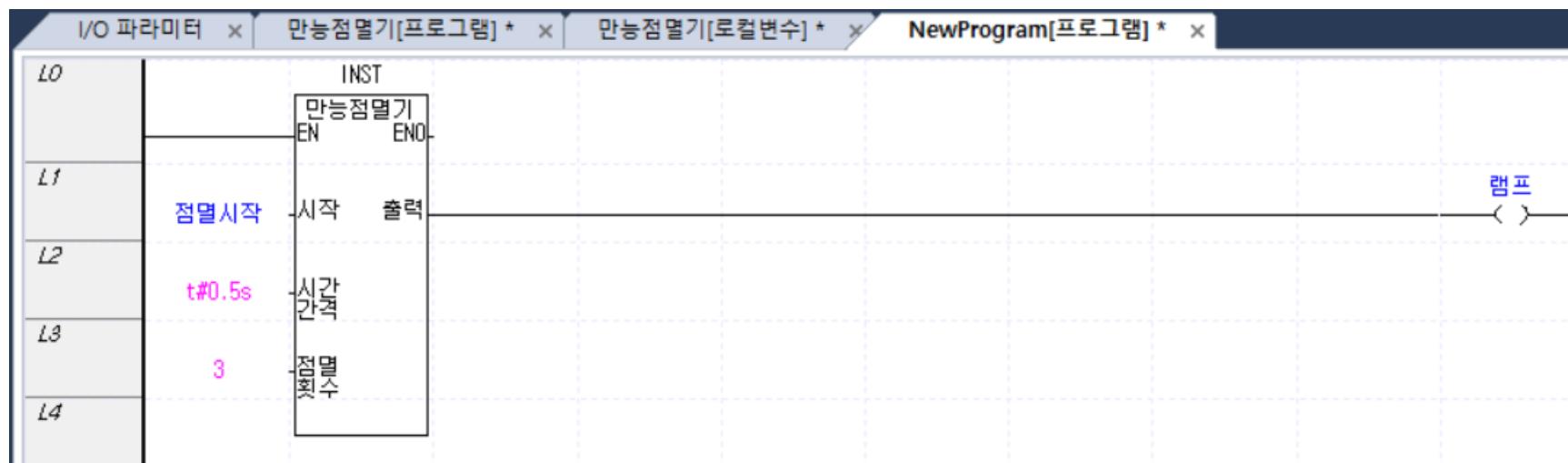
# 실습2. 사칙연산

- “사칙연산”이라는 이름의 평션 블록을 생성
- 입력A, 입력B를 정의하고
- 두입력에 대해 사칙연산을 각각 적용한 결과를 출력하는 평션 블록을 정의
- 입/출력 타입은 INT로



# 실습3. 만능점멸기

- “만능점멸기”라는 이름의 평션블록을 생성
- 아래와 같이 인풋을 정의하고
  - 시작, 시간간격, 점멸횟수
- 정의된 점멸 횟수만큼 10이 출력되도록 구현
- 예를 들어 평션블록에 램프를 연결하면 점멸 횟수만큼 램프가 깜빡여야 함



# 실습4. 깜박이 평션 블록 제작

## ■ 3개 입력, 2개 출력을 가지는 평션 블록

- 입력 FL : LAMP1이 5초 동안 켜지고, 이후 1초 단위로 깜박임
- 입력 NFL : LAMP1이 꺼지고, LAMP2가 켜짐
- 입력 OFF : 모두 꺼짐
- 출력 LAMP1
- 출력 LAMP2
- **\_T1S 플래그를 사용할 것**

변수 선택

변수(V): \_T1S [필터] [A] [검색]

변수 종류  
 로컬 변수(L)  글로벌 변수(G)  직접변수 설명문(I)  **플래그(F)**

플래그 보기  
 할복별 보기(O)  전체 그룹 보기(T)

변수

변수	타입	메모리 할당	설명문
T10S	BOOL	%FX140	10s 주기 CLOCK
<b>_T1S</b>	BOOL	%FX147	1s 주기 CLOCK
T200MS	BOOL	%FX140	200ms 주기 CLOCK
T20MS	BOOL	%FX144	20ms 주기 CLOCK
T20S	BOOL	%FX150	20s 주기 CLOCK
T2S	BOOL	%FX148	2s 주기 CLOCK
T60S	BOOL	%FX151	60s 주기 CLOCK
PLC_OPERA...	DWORD	%FD498	전원 가동시간(단위: 초)
상태			

# 실습4. 깜박이 평선 블록 제작

Heatmap showing gene expression levels across various conditions. The y-axis categories are FL, NFL, OFF, LAMP1, and LAMP2. The x-axis shows 10 time points from -5 to 5 minutes. Dark blue indicates high expression, while red indicates low expression.

	-5	-4	-3	-2	-1	0	1	2	3	4	5
FL	High	Low									
NFL	Low	High	Low								
OFF	Low	Red									
LAMP1	High	Low									
LAMP2	Low	High	High	High							

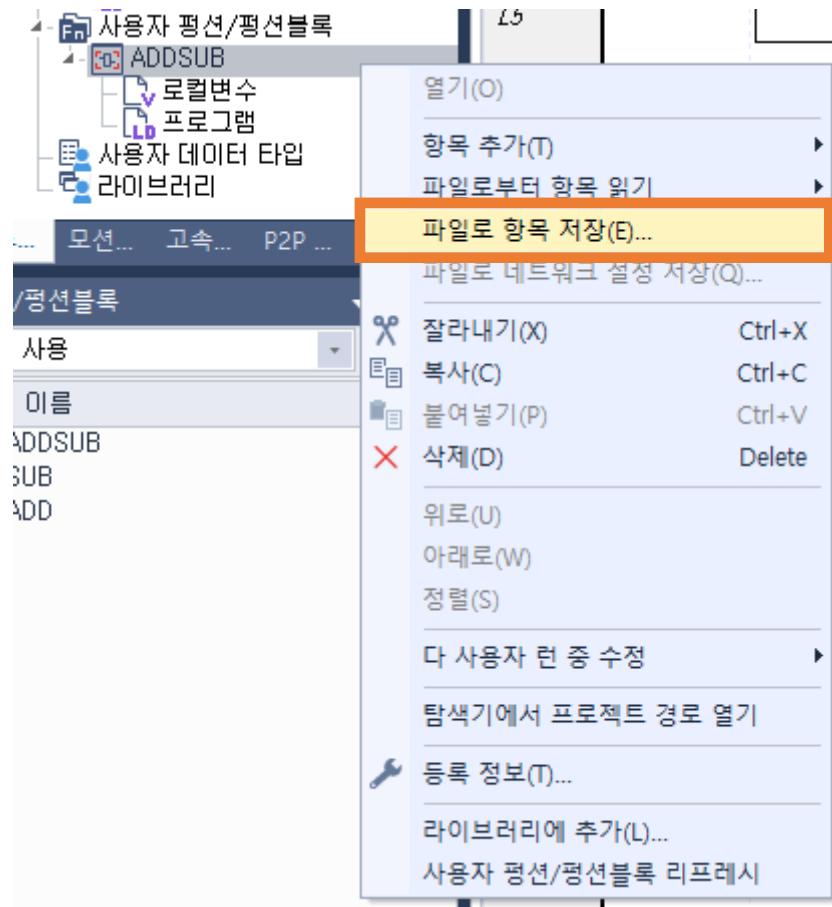
- “FL” 신호 입력 → LA1 5초간 켜지고, 5초 이후에 1초 주기로 깜빡 거림
  - “NFL” 신호 입력 → 켜졌던 LA1이 꺼지고, LA2가 On된다.
  - “OFF” 신호 입력 → 모든 출력값(LA1, LA2)이 OFF
  - “EN” 과 “ENO”는 입/출력 모션과 연결되는 변수 (기본제공)

다른 프로젝트에서 불러오기



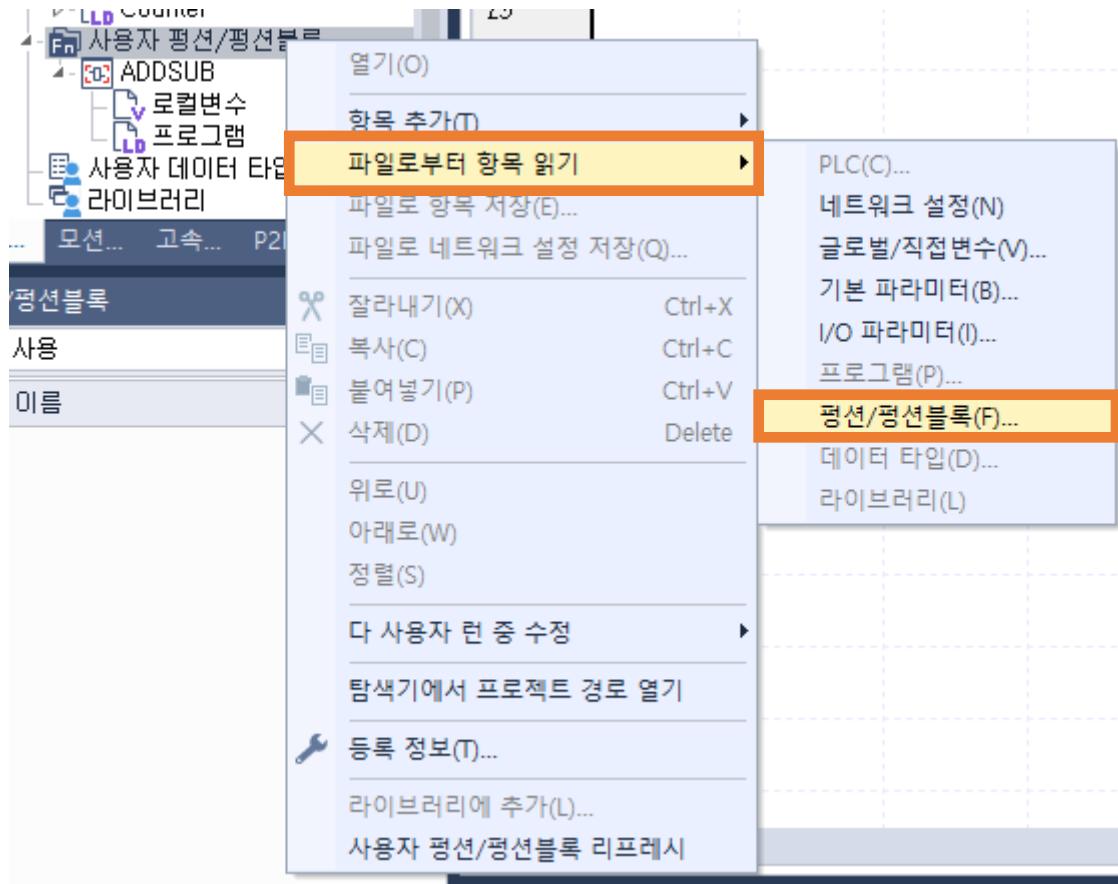
# 다른 프로젝트에서 불러오기

## ▪ 파일로 함수 저장



# 다른 프로젝트에서 불러오기

## ▪ 다른 프로젝트에서 불러오기



# 라이브러리 만들기



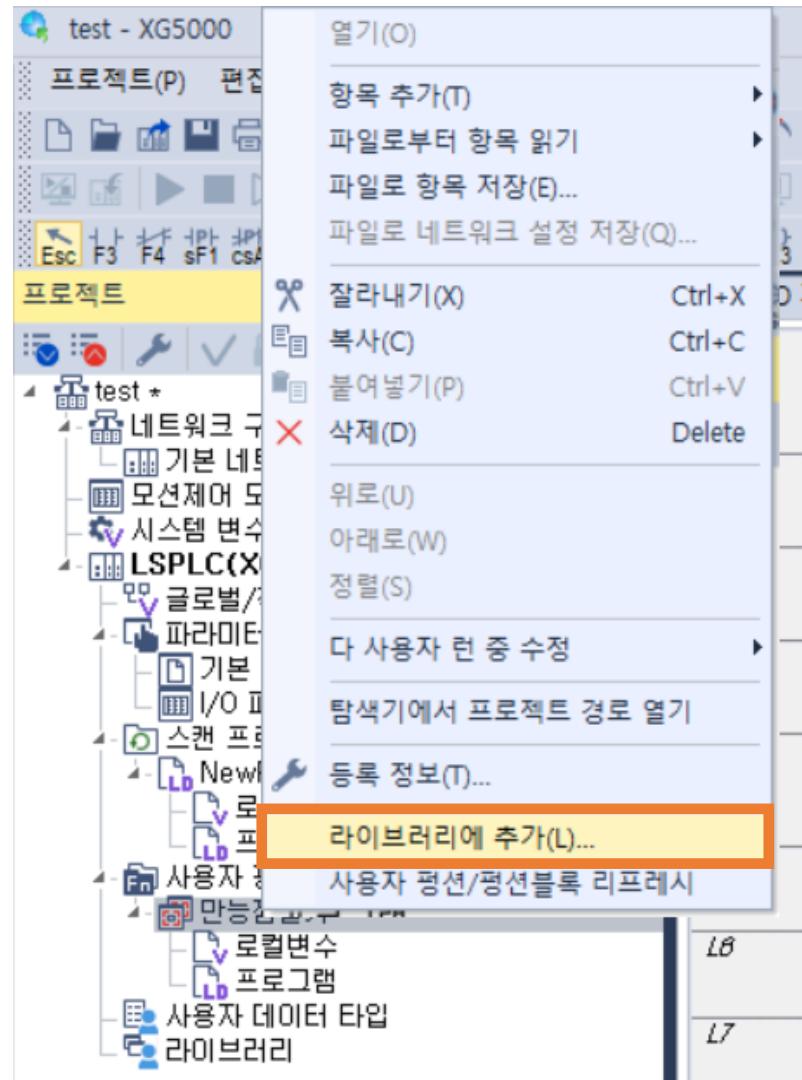
# 라이브러리 만들기

## ■ 라이브러리

- 자주 사용하는 기능이나 코드를 모듈화하여 저장하고 재사용할 수 있도록 하는 기능
- 개발 효율성을 높이고, 코드 일관성을 유지하며, 반복 작업을 최소화할 수 있도록 돋는 기능
- 내부 구현을 볼 수 없기 때문에 보안 유지가 가능함



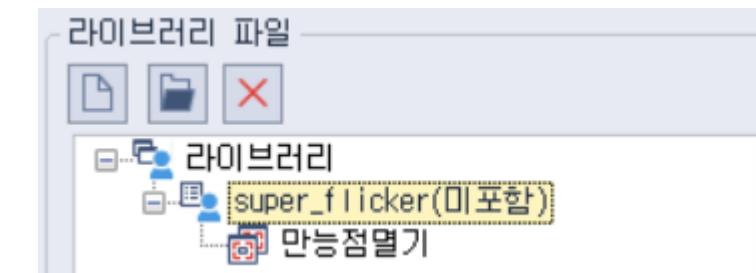
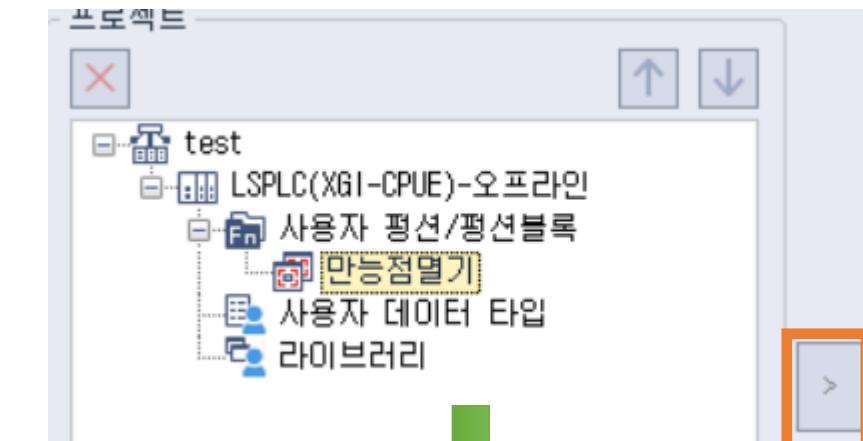
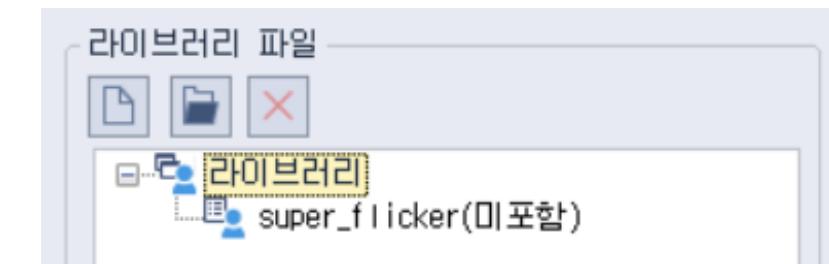
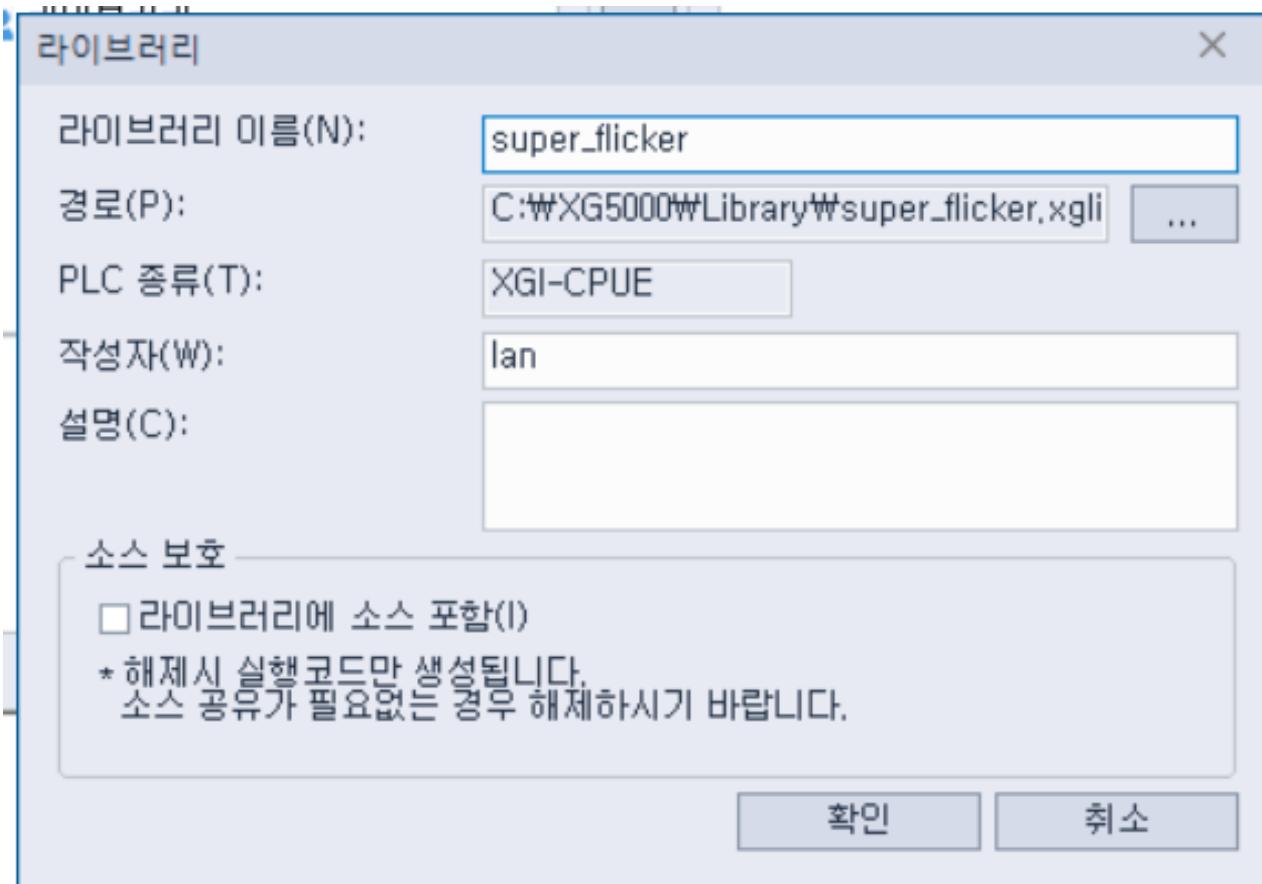
# 라이브러리 만들기



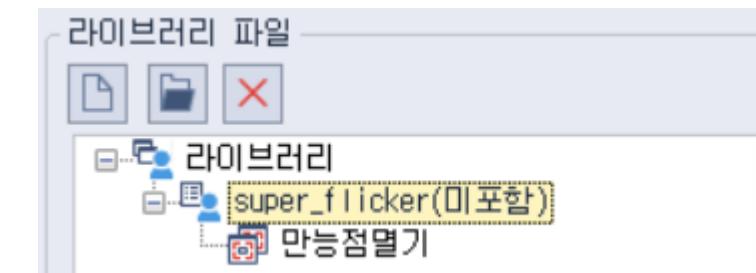
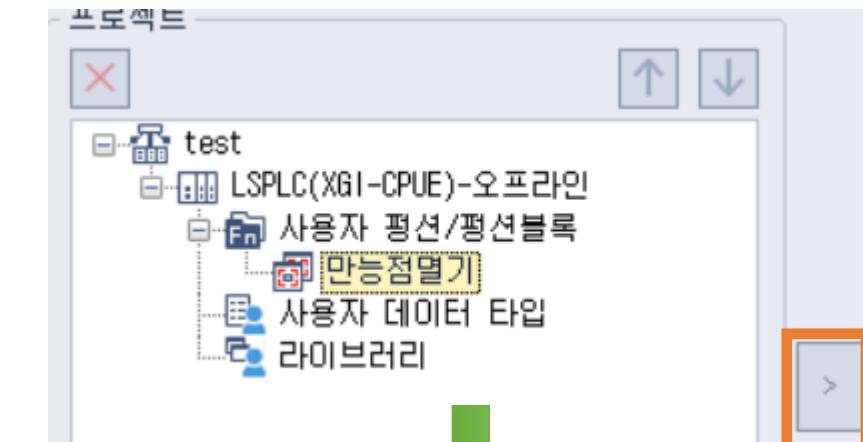
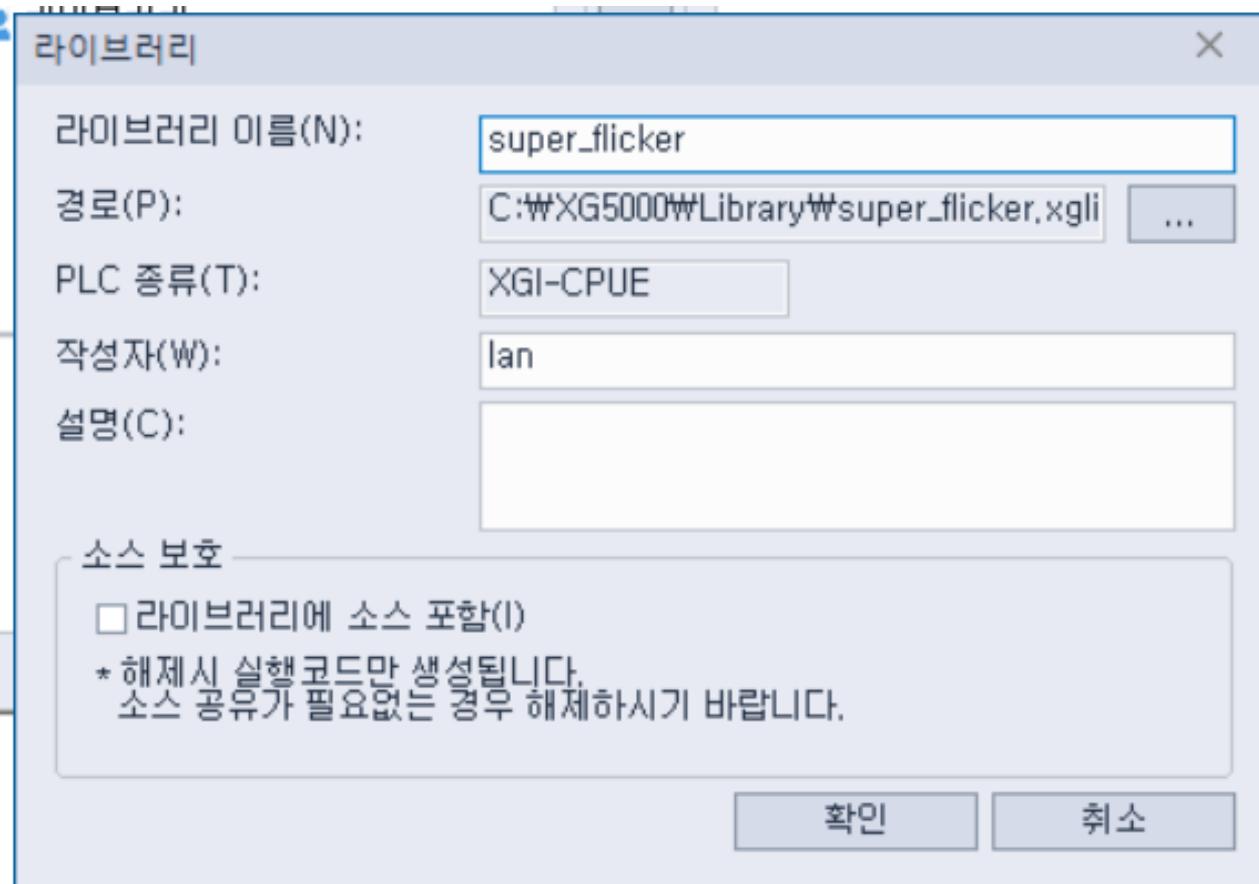
# 라이브러리 만들기



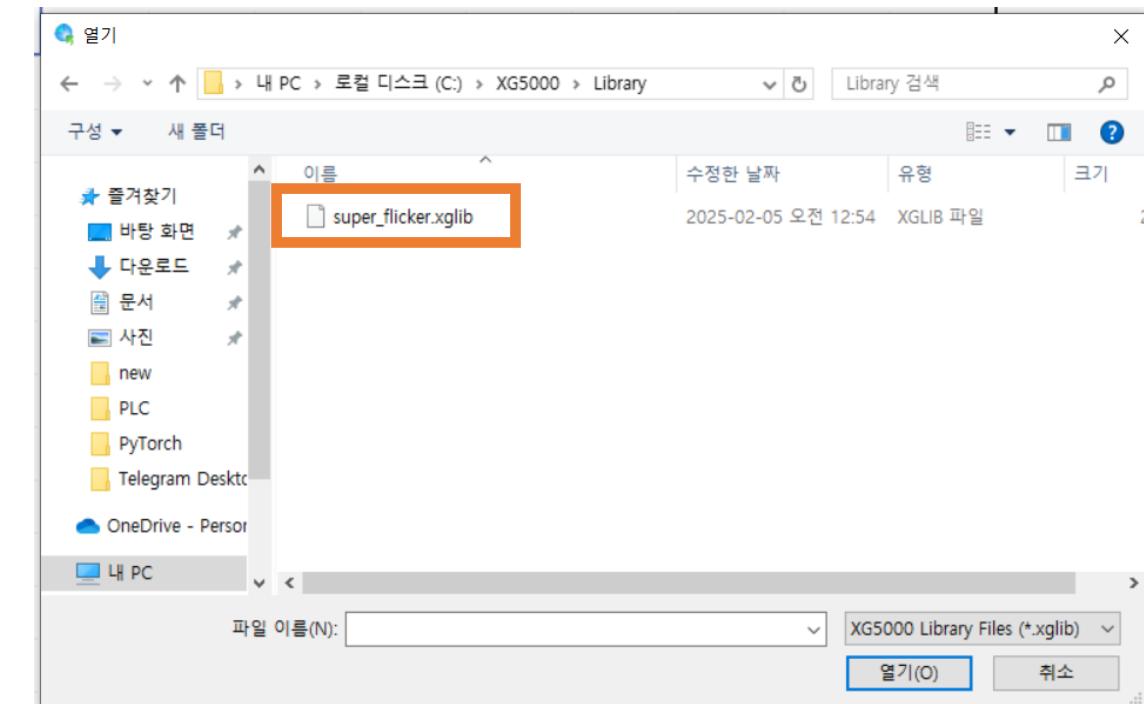
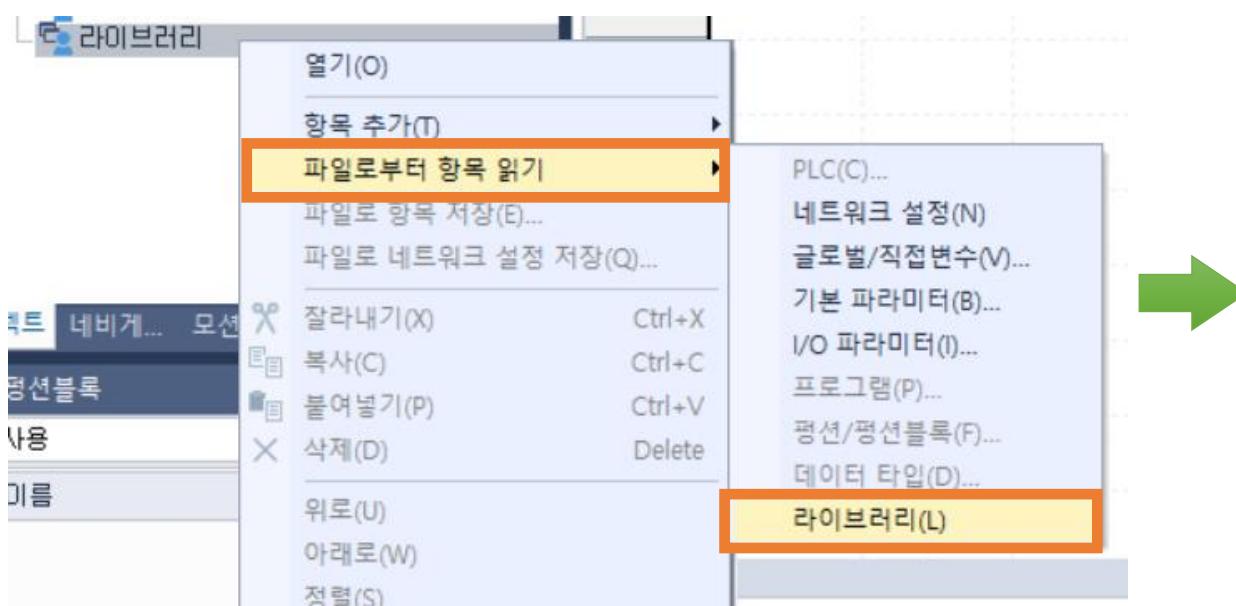
# 라이브러리 만들기



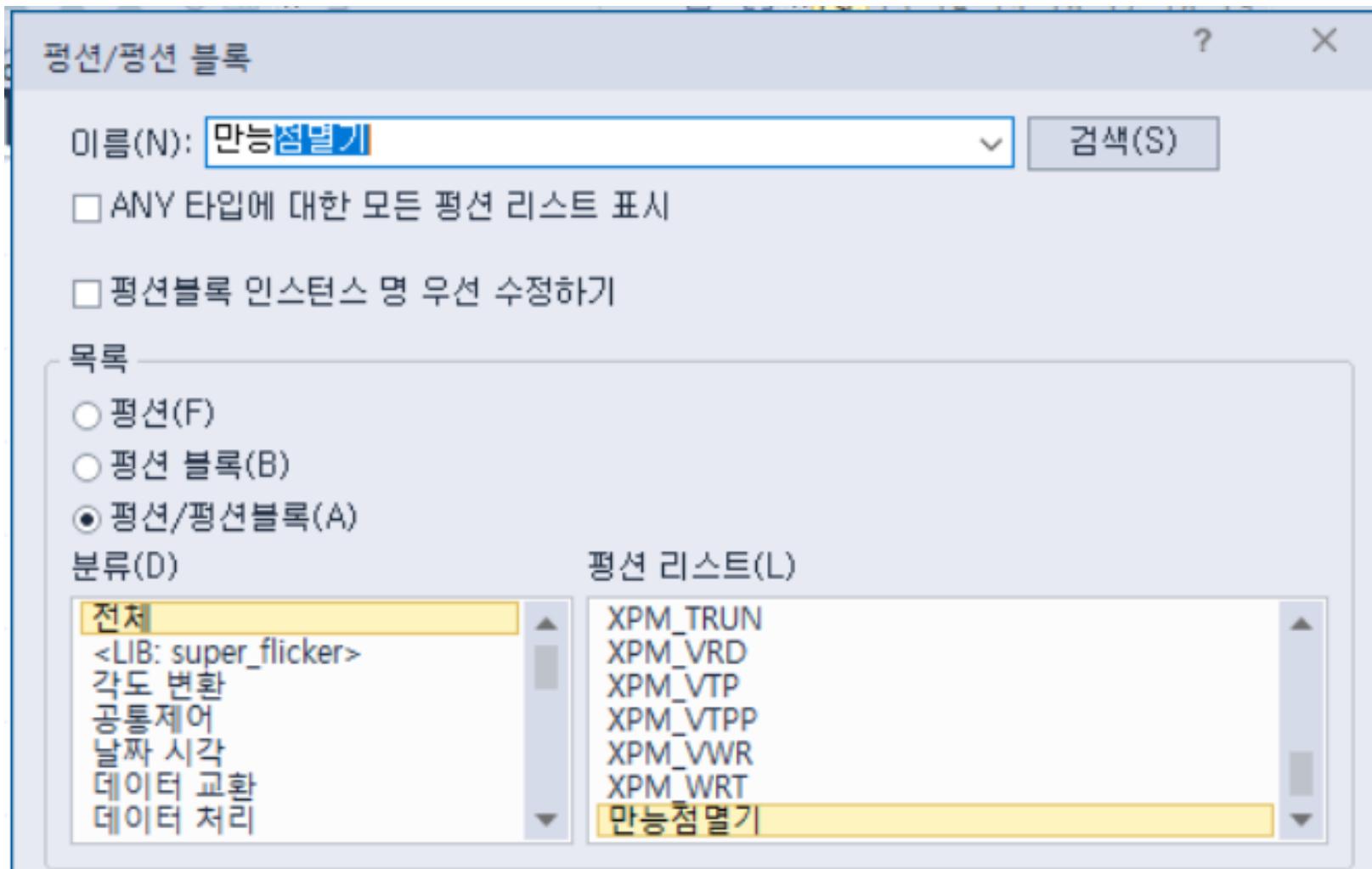
# 라이브러리 만들기



# 라이브러리 불러오기



# 라이브러리 불러오기



감사합니다