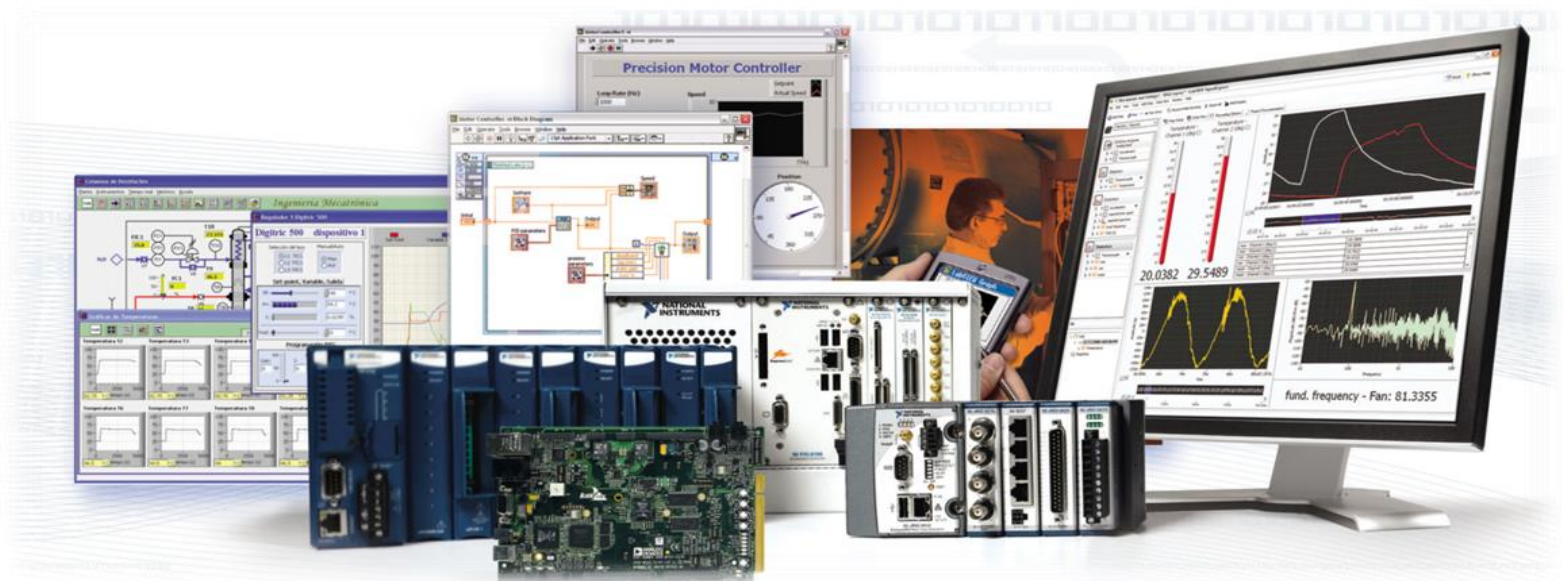


LabVIEW

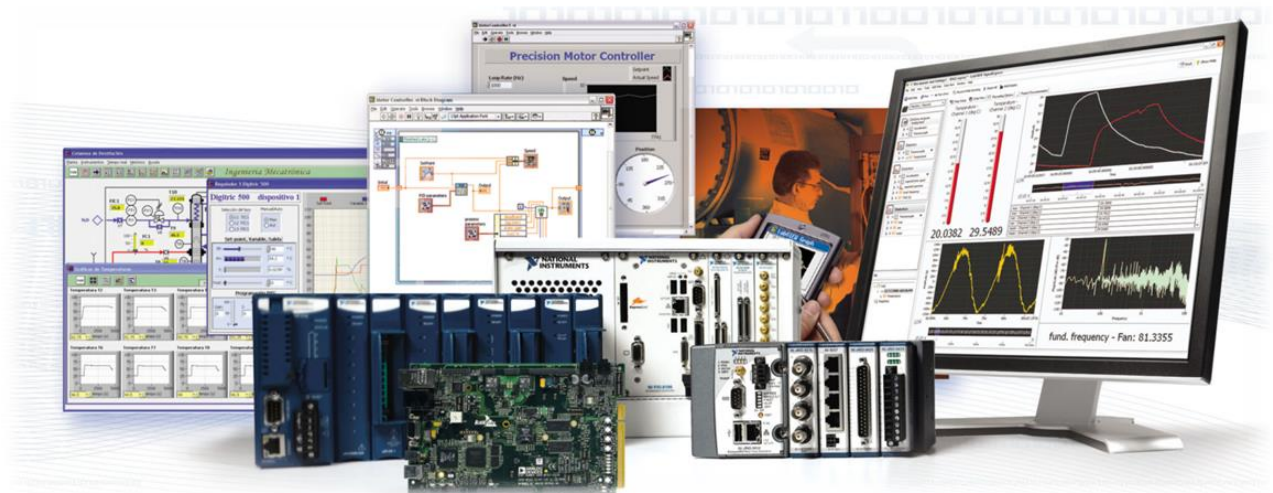
LabVIEW의 정석 기본편



INFINITYBOOKS

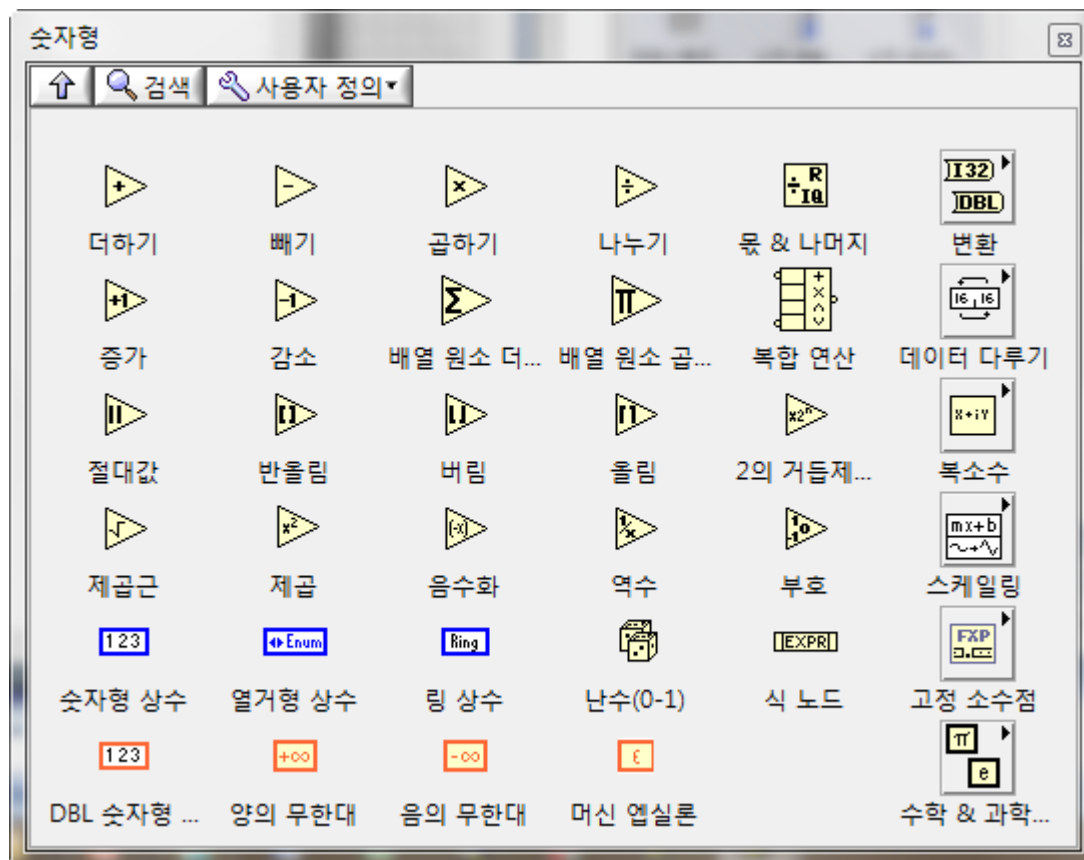
인 피 니 티 북 스

3. 데이터 타입





숫자형(Numeric)



- 숫자형 - 숫자 데이터를 다룸.
- 주황색은 실수형, 파란색은 정수형을 나타냄





숫자형의 형 변경

실수형 0 정수형 0

보이는 아이템
터미널 찾기
인디케이터로 변경

설명과 팁...

생성
대체
데이터 처리
고급
컨트롤을 구획에 맞춤
객체를 구획에 맞춰 스케일

형

데이터 입력...
디스플레이 포맷...

프로퍼티

EXT DBL SGL FXP
I64 I32 I16 I8
U64 U32 U16 U8
CXT CDB CSG

실수형
정수형
양의 정수형
복소수형



숫자형 데이터 타입 - 정수

❖ 부호있는 정수

- 32-bit (I32): -2147483648 ~ 2147483647
- 16-bit (I16): -32768 ~ 32767
- 8-bit (I8): -128 ~ 127

❖ 부호없는 정수

- 32-bit (U32): 0 ~ 4,294,967,295
- 16-bit (U16): 0 ~ 65536
- 8-bit (U8): 0 ~ 256



숫자형 데이터 타입 - 실수형 숫자

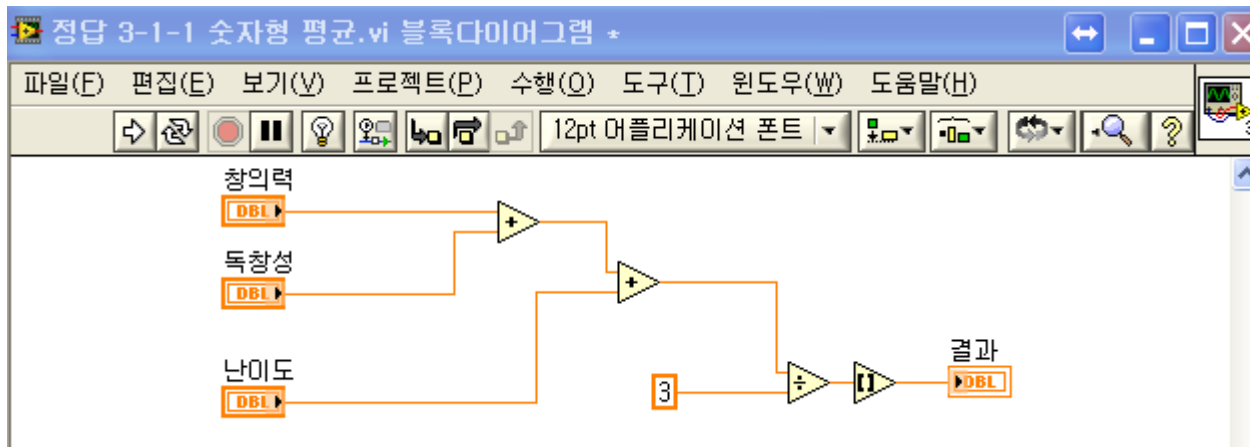
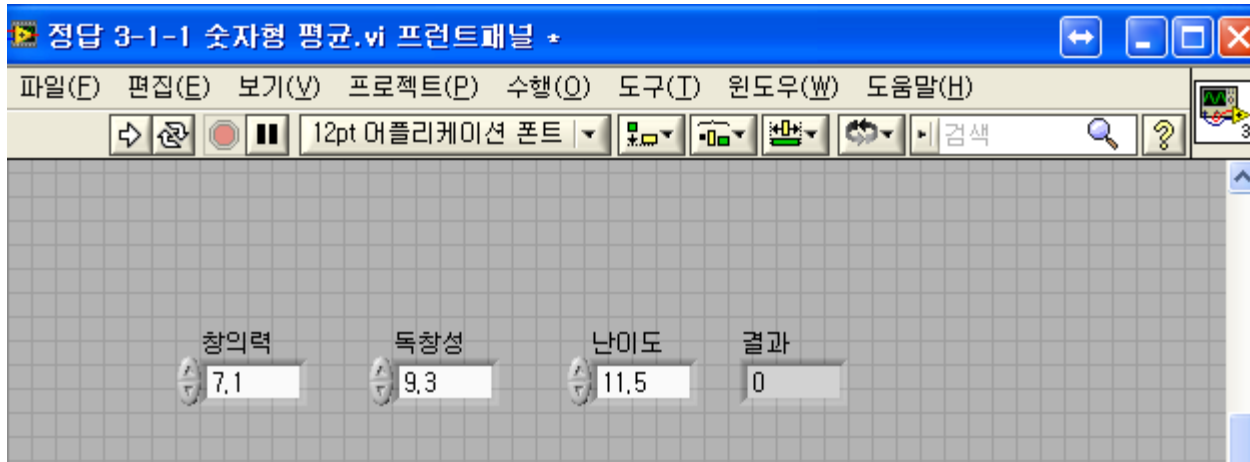
❖ 실수형:

- 확장형 정밀도 [EXT]: $-1.19e+4932 \sim 1.19e+4932$
- 배정도 [DBL]: $-1.79e+308 \sim 1.79e+308$
- 단정도 [SGL]: $-3.40e+38 \sim 3.40e+38$

❖ 복소수 실수형:

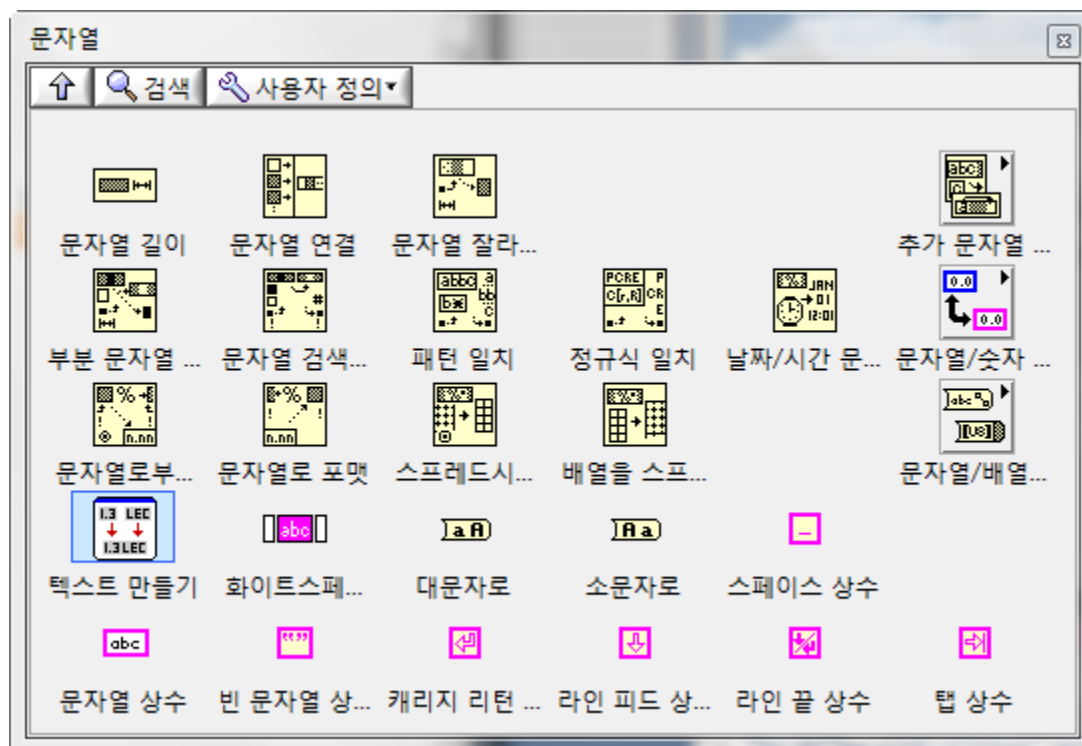
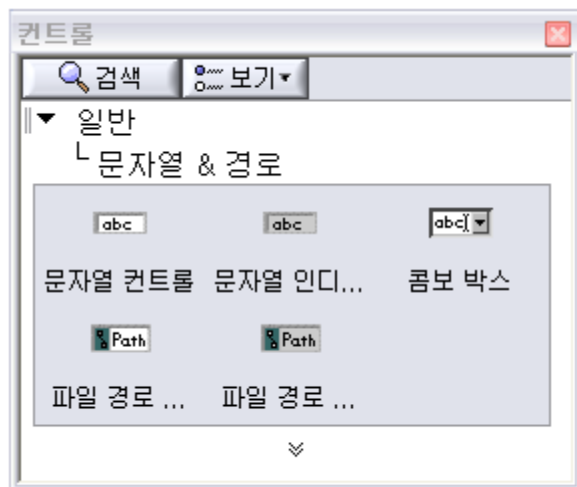
- 복소수 실수형 데이터 타입은 실수형 데이터 타입과 같은 정밀도를 가집니다. 유일한 차이점은 복소수 실수형 데이터 타입은 실수와 허수부를 가진다는 것입니다.

실습3-1-1) 숫자형 사용법





문자열(String)



문자열

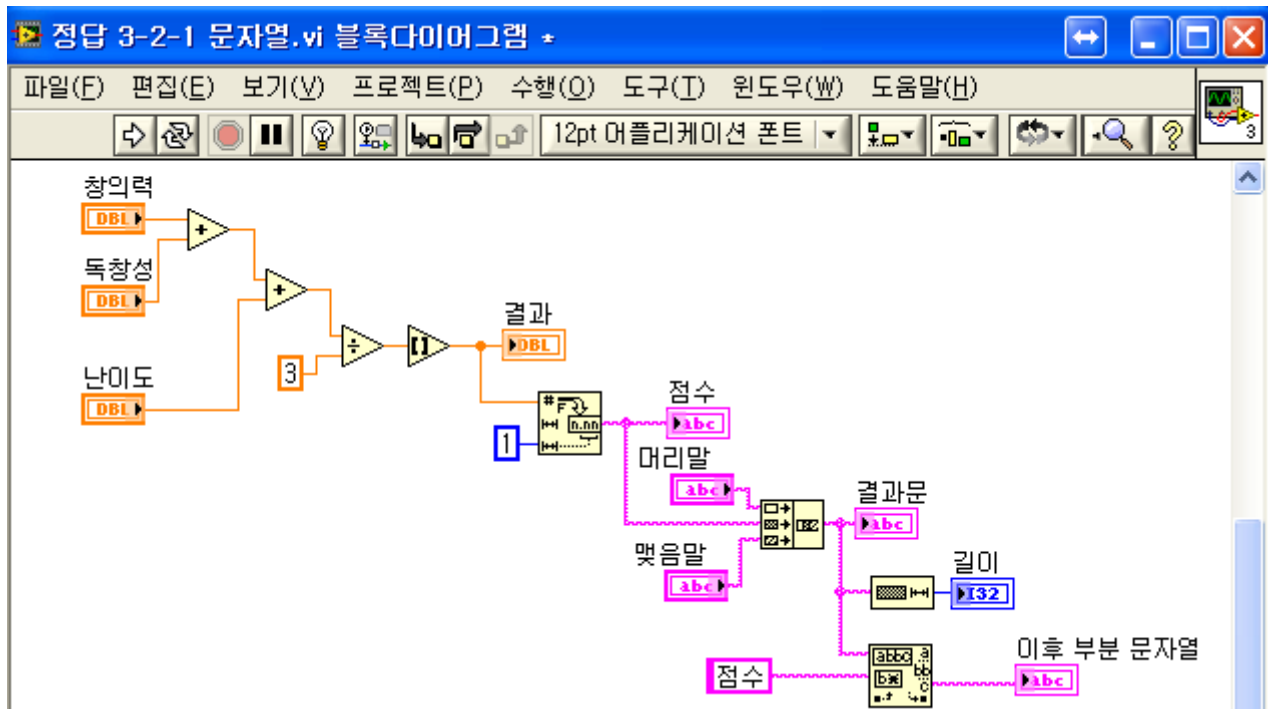
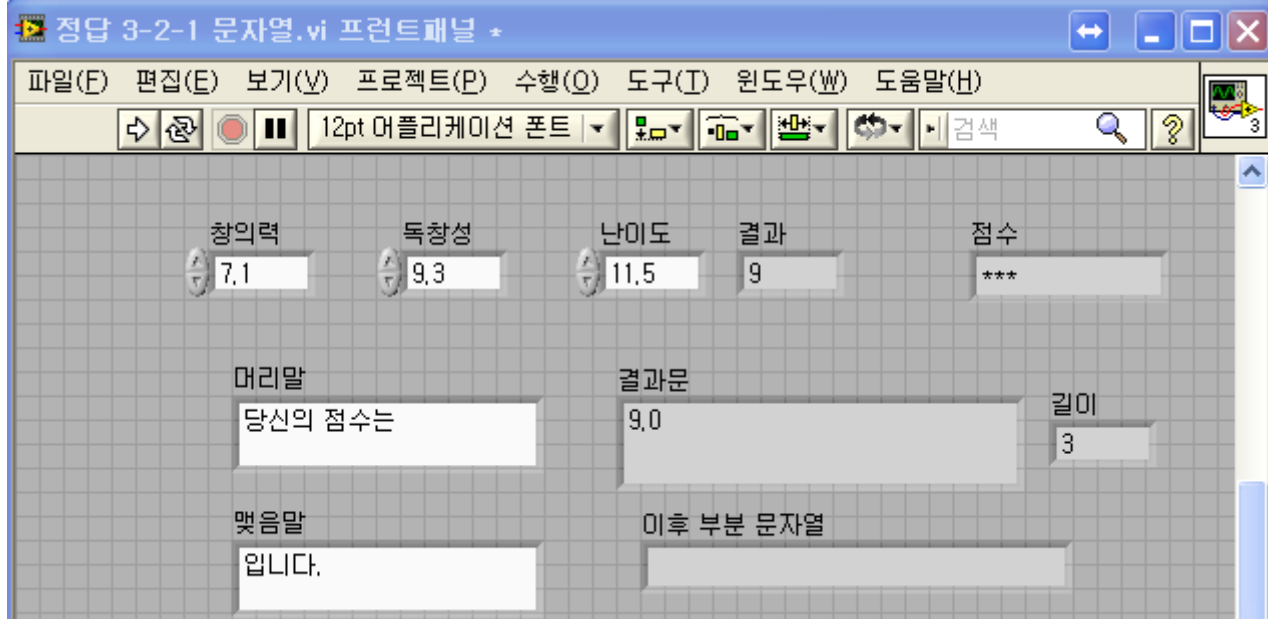
abc

문자열 2

abc

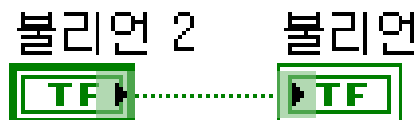
- 문자열 - **ASCII** 코드로 나타낸 문자를 표시하기 위한 데이터 타입. 분홍색을 나타냄.
- 각종 통신(시리얼, **GPIO**, **TCP/IP** 등)에서 사용되는 데이터 타입.

실습3-2-1) 문자열 사용법





불리언(Boolean)

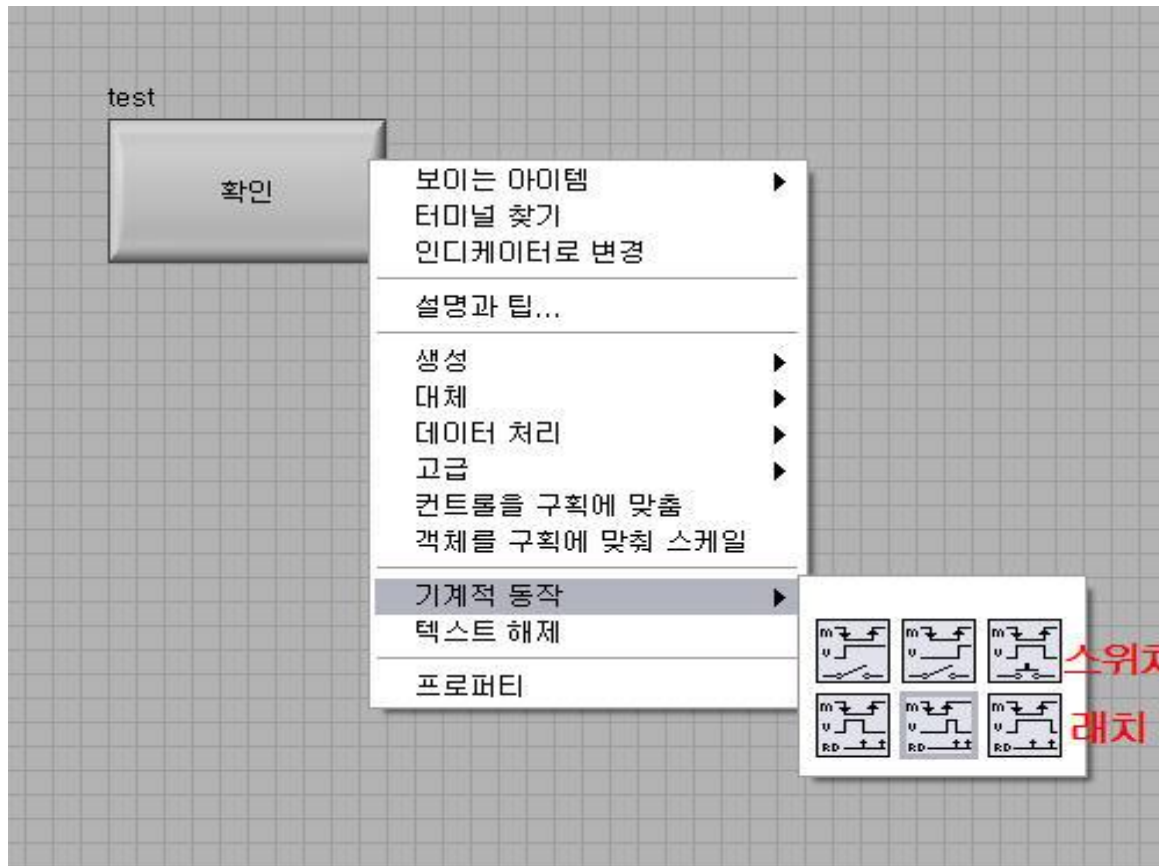


• 불리언 - 참, 거짓 데이터를 표현되는 논리 연산을 위한 데이터, 초록색을 나타냄



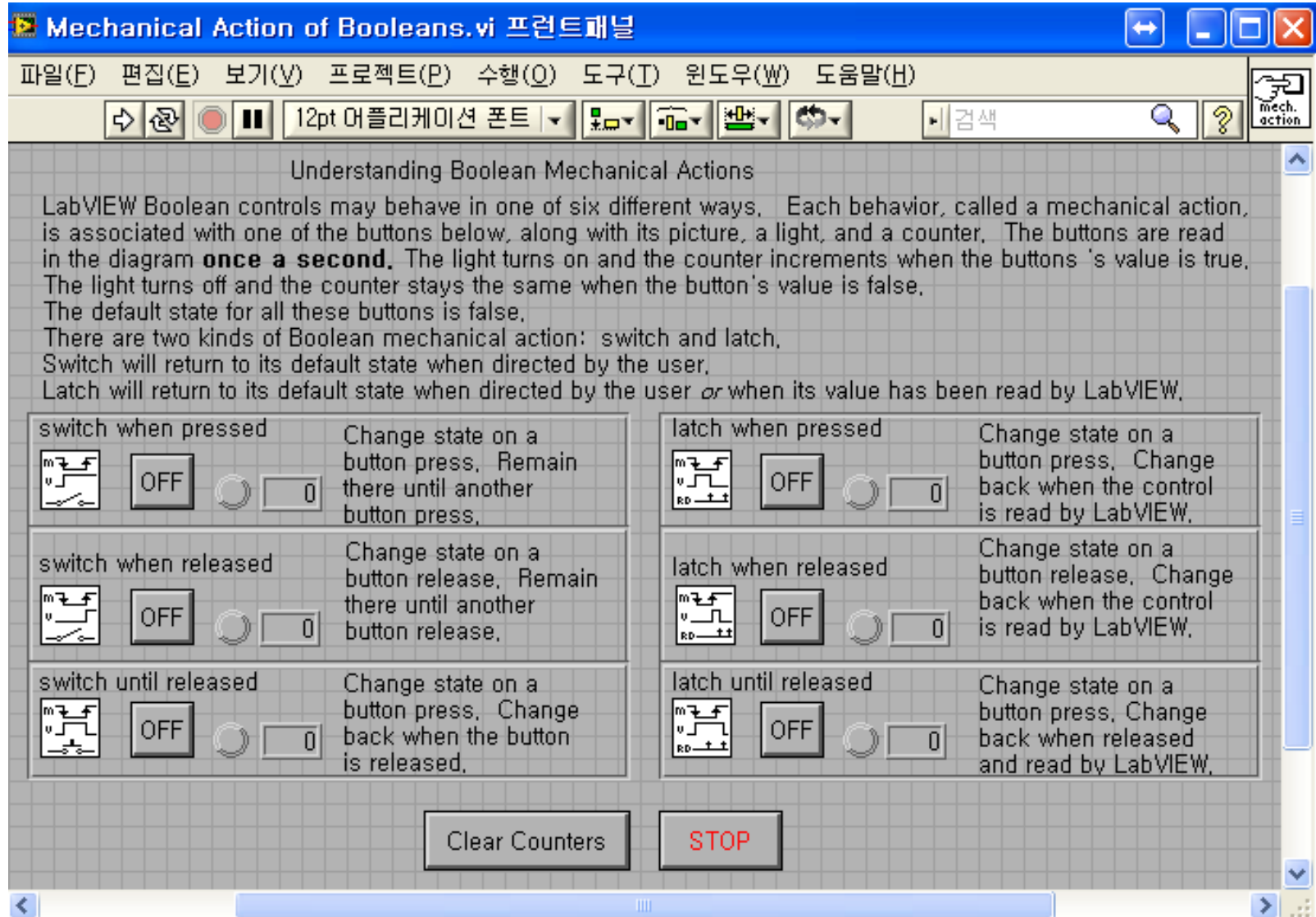
불리언의 기계적 동작

- 스위치 : 상태 유지.전등불의 전원이 같은 동작 원리
- 래치 : 상태 복원. 초인종과 같은 동작 원리.

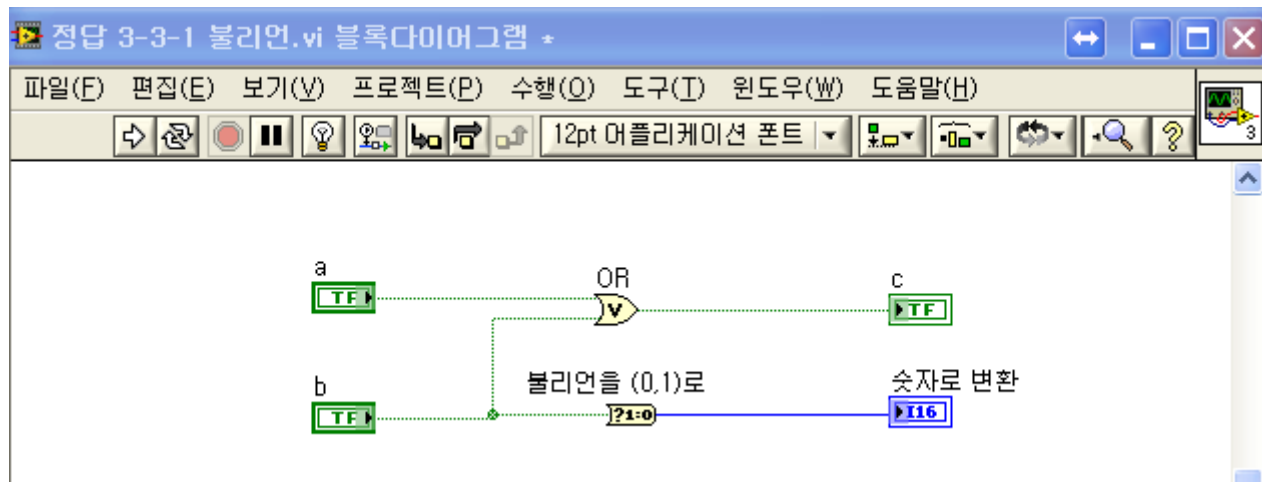


예제를 통한 기계적 동작 확인

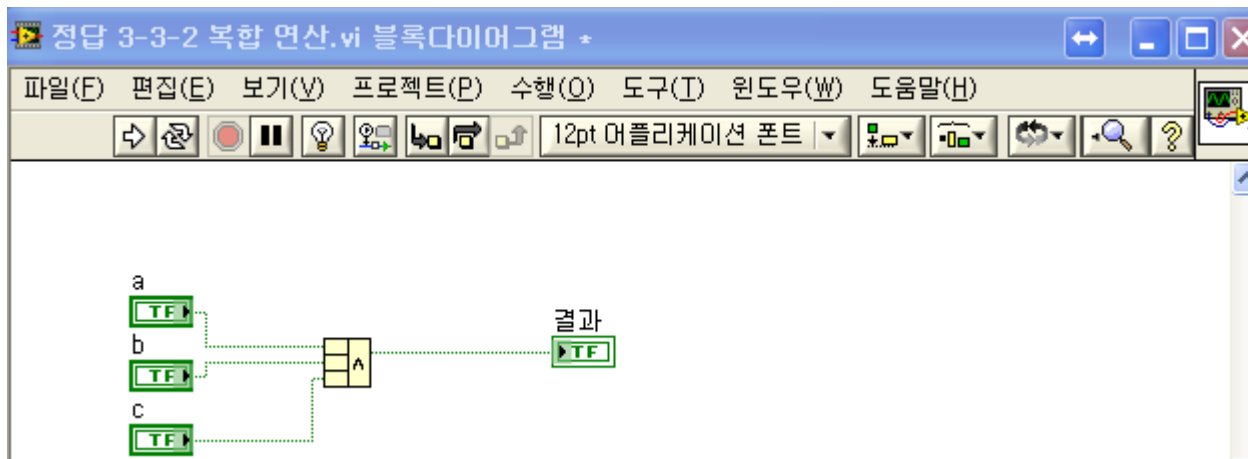
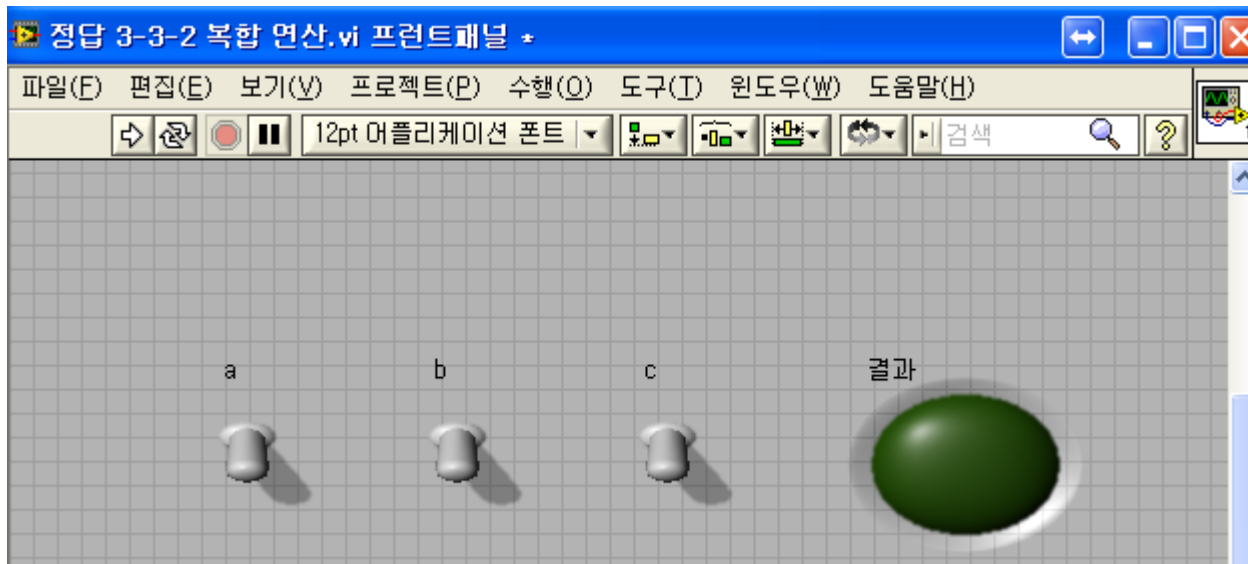
파일>폴다운메뉴>도움말>예제찾기에서 “기계적”이라고 해서 검색



실습3-3-1) 불리언 사용법

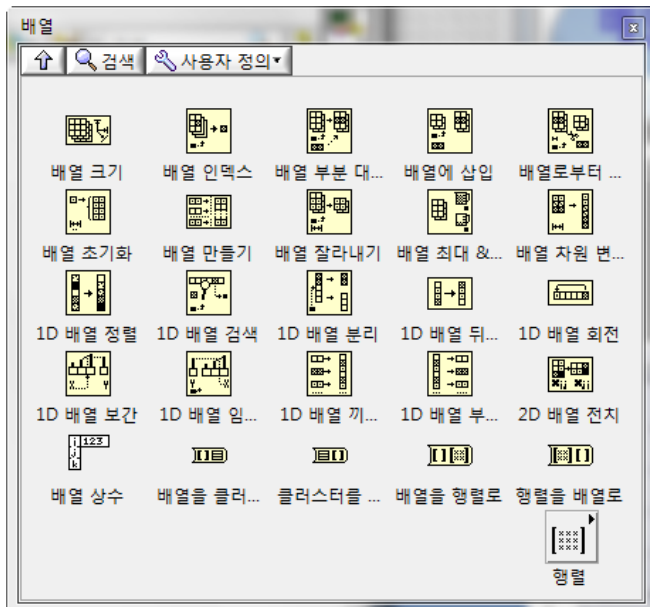
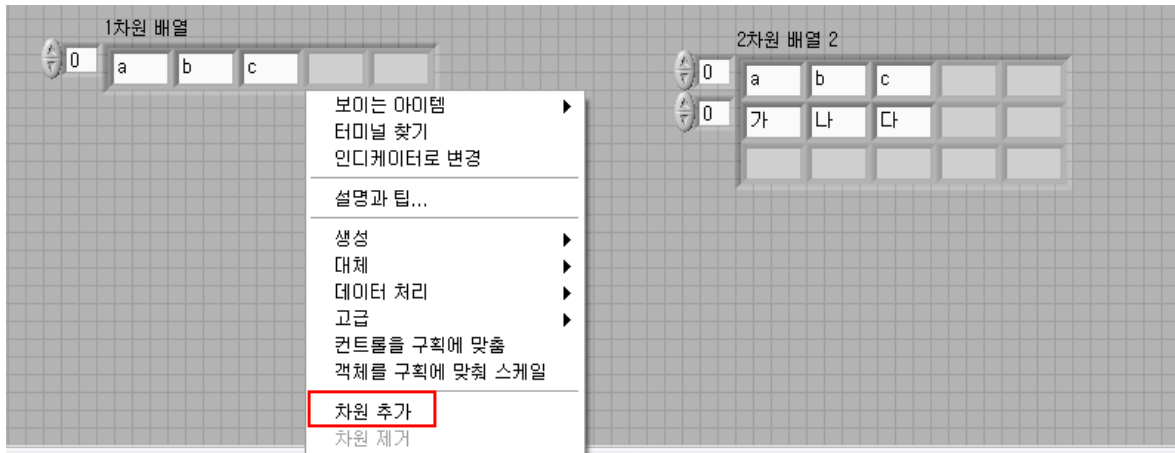


실습3-3-2) 복합연산 사용법





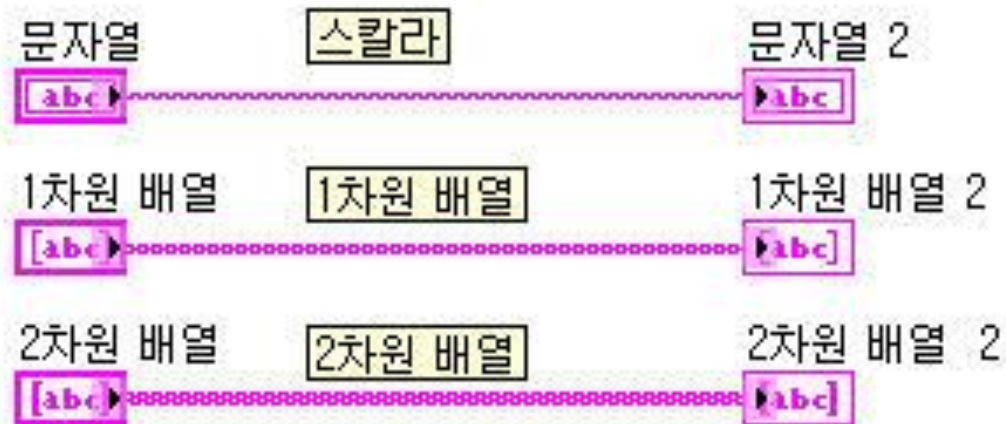
배열(Array)



- 배열 : 같은 데이터 타입의 묶음
- 원소 : 배열 구성원
- 인덱스 : 원소의 주소를 의미
- 차원



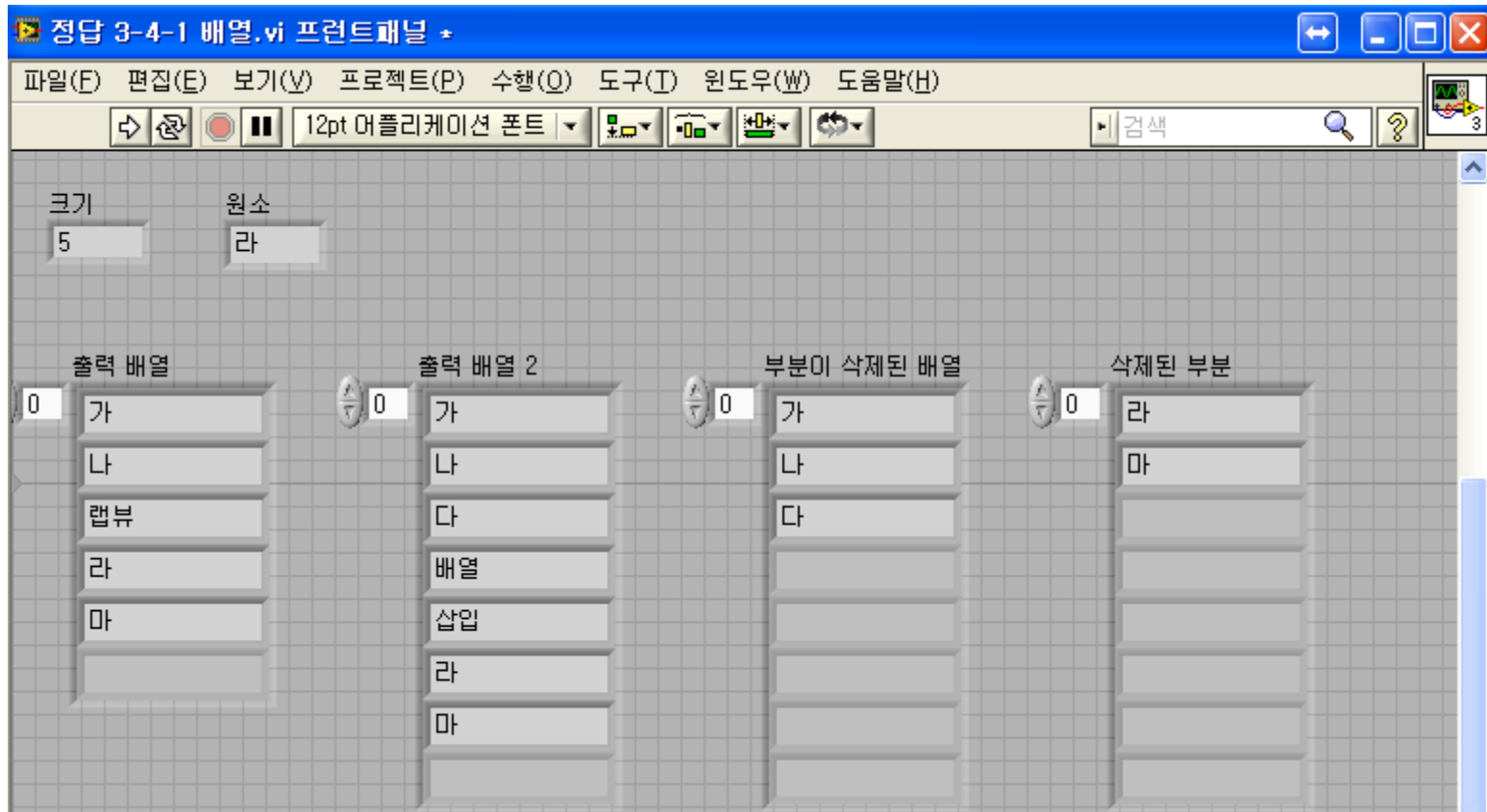
배열 와이어

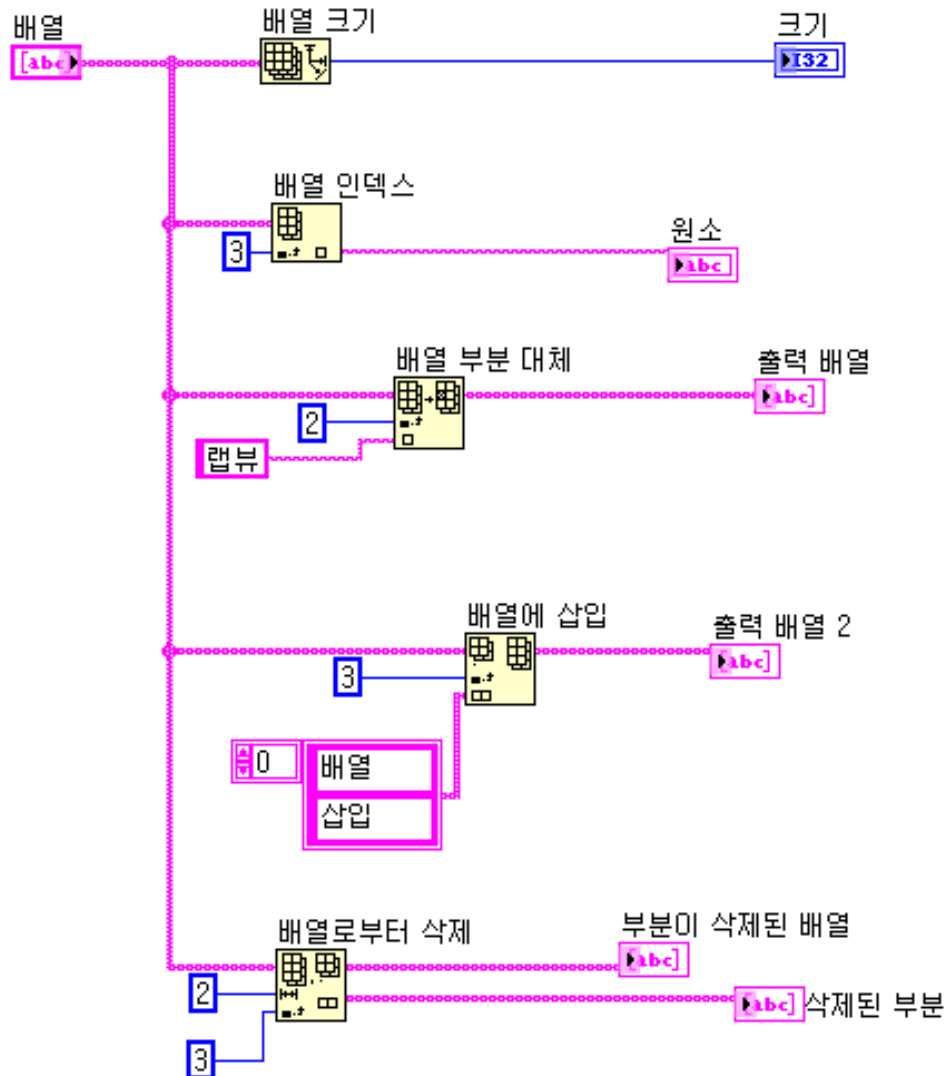


- 와이어의 굵기가 배열의 차원에 따라 달라짐.

실습3-4-1)

배열 생성 및 노드 사용법



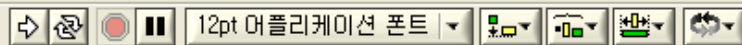


실습3-4-2]

배열 생성 및 노드 사용법

정답 3-4-2 배열.vi 프런트패널 *

파일(F) 편집(E) 보기(V) 프로젝트(P) 수행(Q) 도구(T) 윈도우(W) 도움말(H)



검색

숫자형 배열

0

1.1

10.3

11.9

5.7

6.3

0

0

0

정렬된 배열

0

0

0

0

0

0

0

0

원소의 인덱스

-1

첫번째 부분배열

0

0

0

0

0

0

0

0

두번째 부분배열

0

0

0

0

0

0

0

0

뒤집힌 배열

0

0

0

0

0

0

0

0

최대값

0

최대 인덱스

-1

최소값

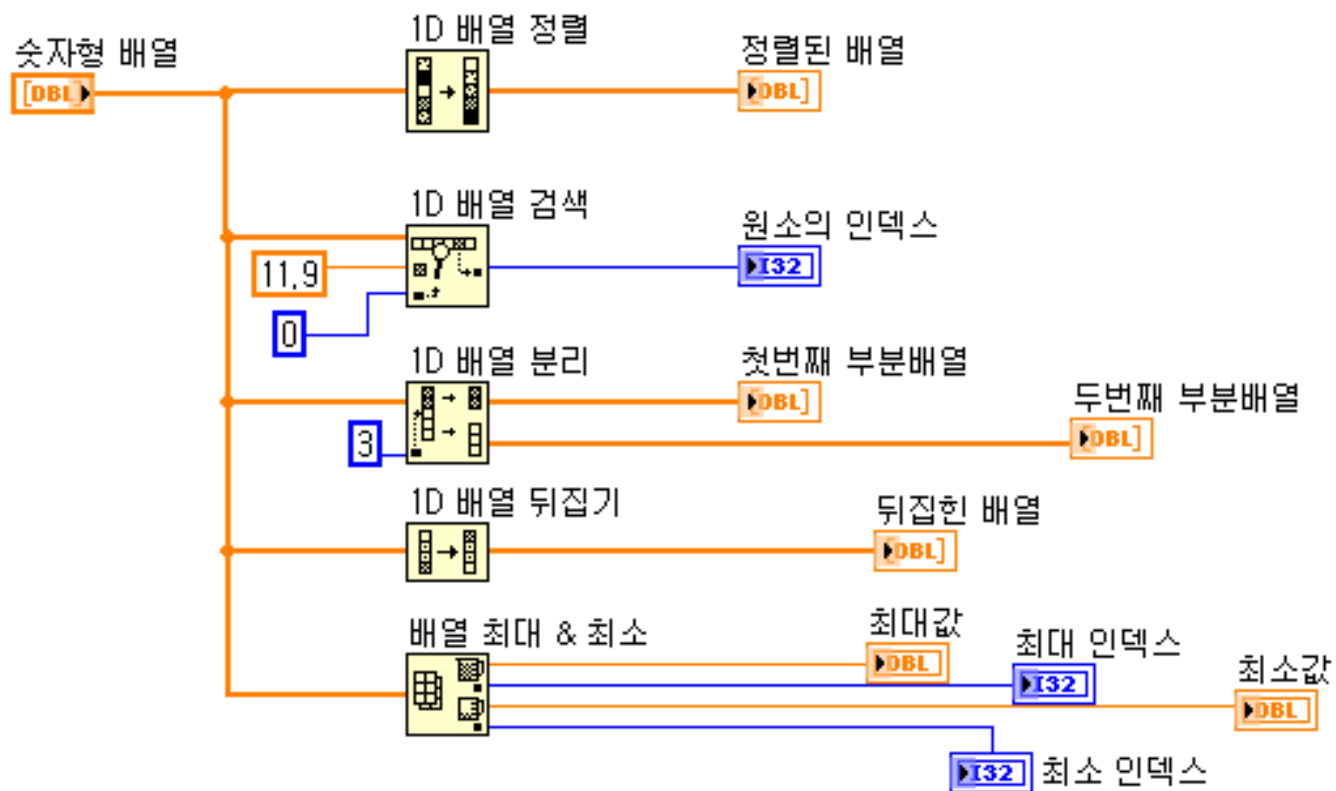
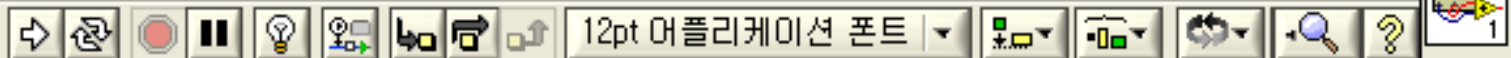
0

최소 인덱스

-1

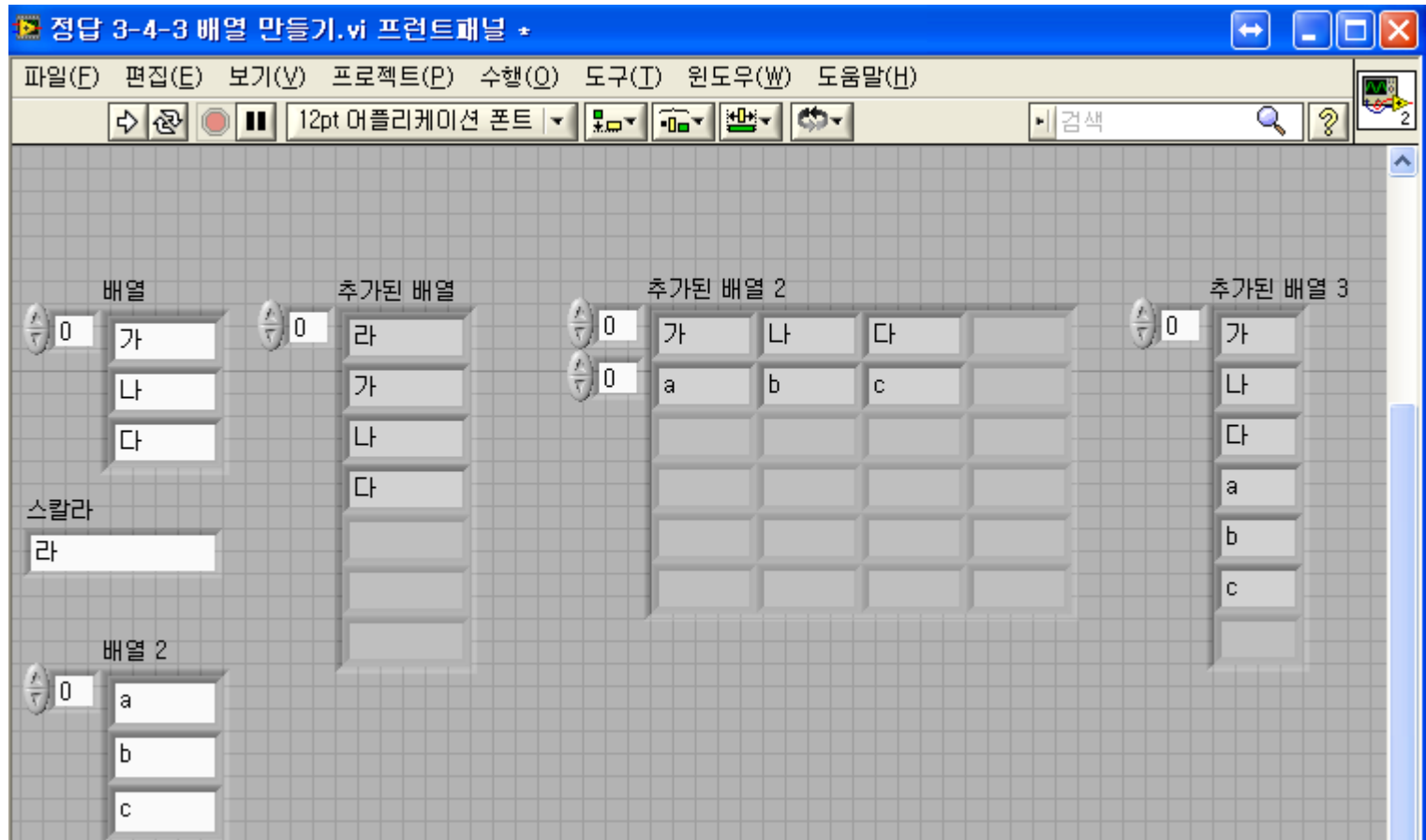
정답 3-4-2 배열.vi 블록다이어그램 *

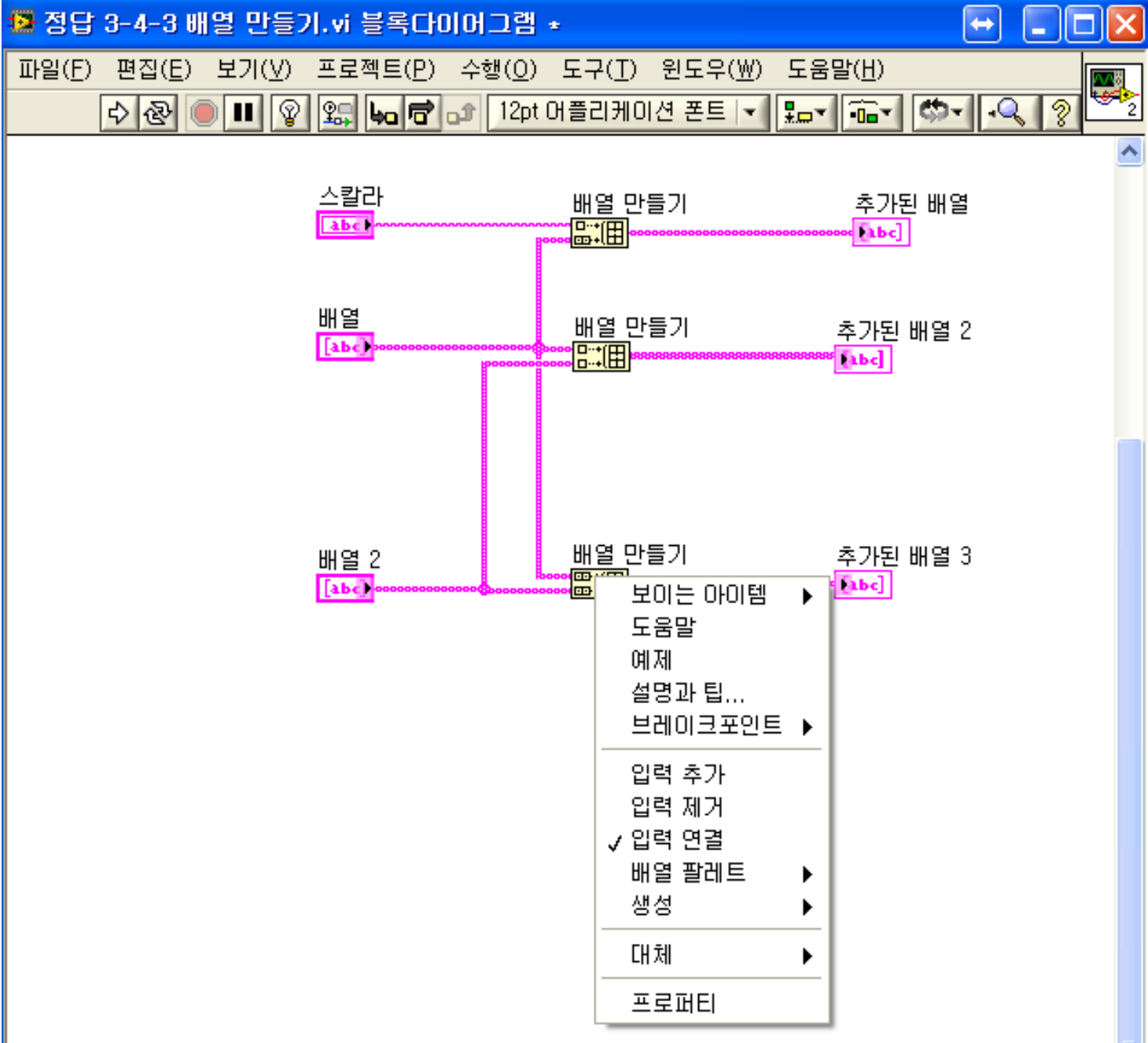
파일(F) 편집(E) 보기(V) 프로젝트(P) 수행(Q) 도구(T) 윈도우(W) 도움말(H)



실습3-4-3)

배열 만들기 노드 사용법







클러스터(Cluster)

학생

이름	홍길동	성별
영어 성적	77	



- 클러스터 : 다양한 데이터 타입의 묶음



클러스터 내의 순서 설정

클러스터

이름

학번

영어성적

0

보이는 아이템
터미널 찾기
인디케이터로 변경

설명과 팁...

생성
대체
데이터 처리
고급

컨트롤을 구획에 맞춤
객체를 구획에 맞춰 스케일

클러스터 내의 컨트롤 순서 재설정...

자동 크기 조정

프로퍼티

제목없음 2 프론트패널 *

파일(F) 편집(E) 보기(V) 프로젝트(P) 수행(Q) 도구(T) 윈도우(W)

클릭하여 설정합니다

0

순서 재설정

클러스터

이름

학번

영어성적

0

변경 전

변경 후



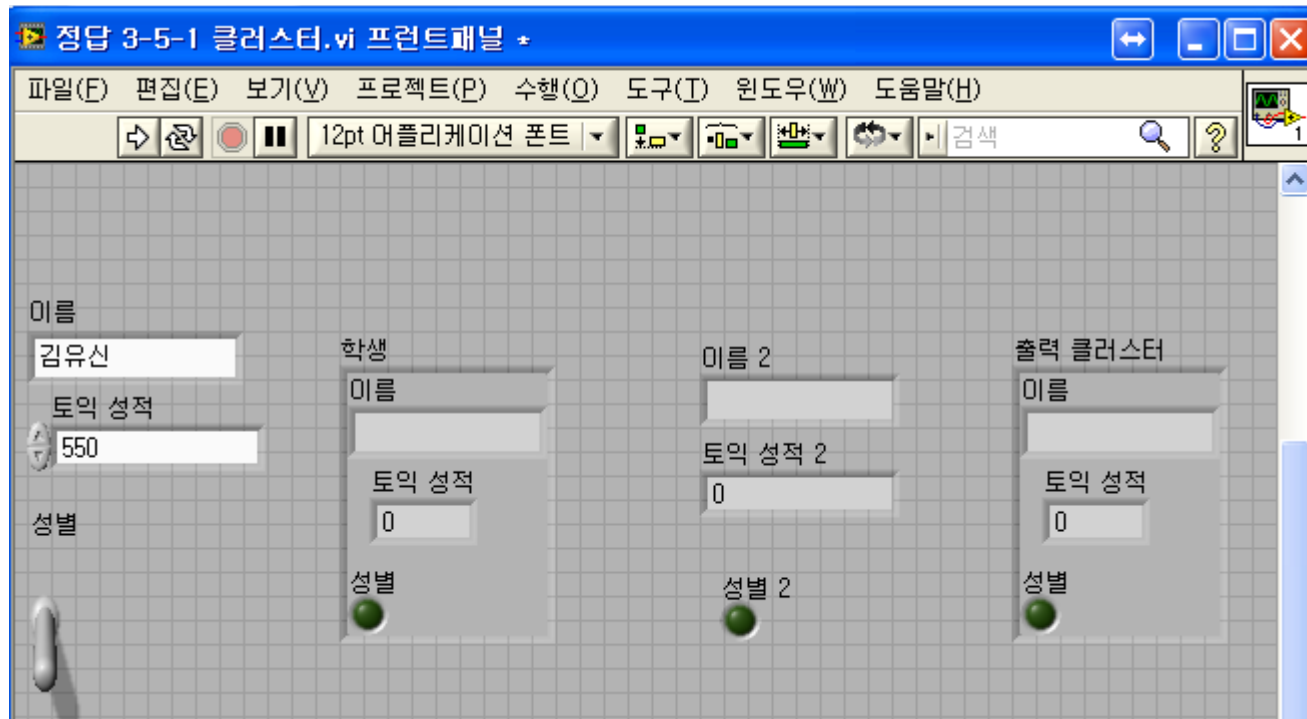
클러스터 노드

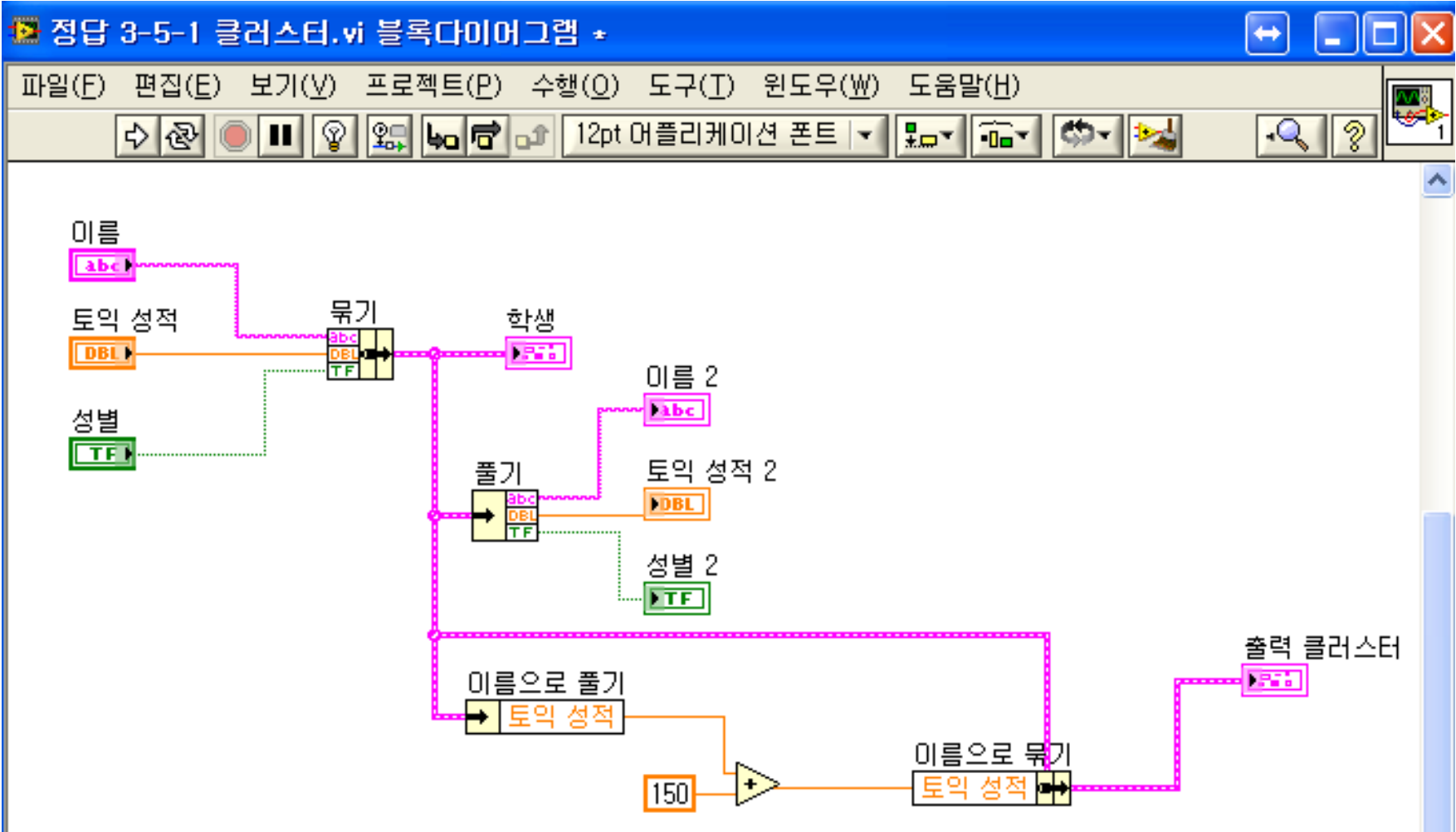


- 묶기 : 클러스터 생성할 때 사용
- 이름으로 묶기 : 클러스터 업데이트할 때 사용
- 풀기 : 클러스터 원소 전체 풀어낼 때 사용
- 이름으로 풀기 : 클러스터 원소 선택적으로 풀어낼 때 사용

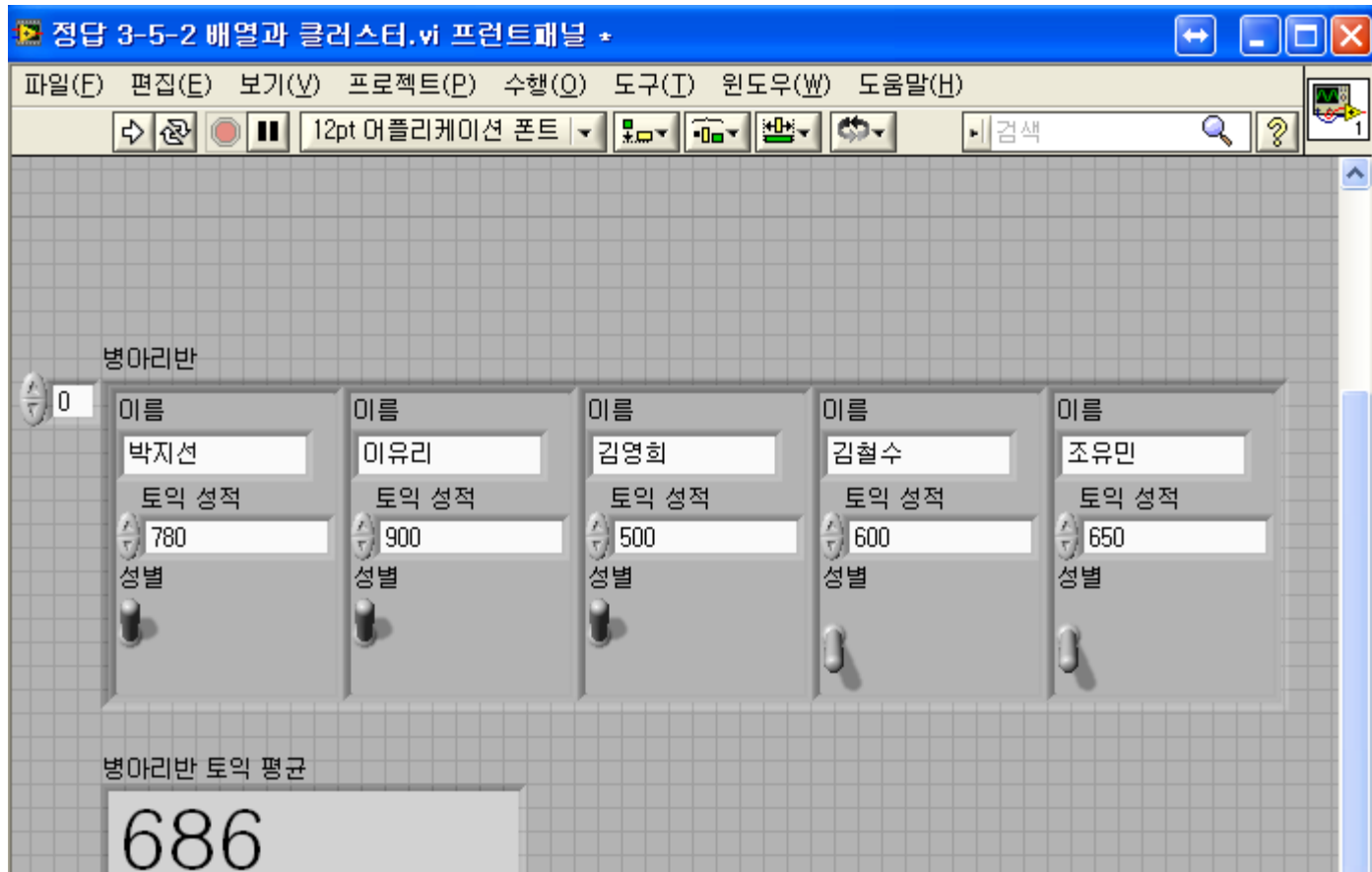
실습3-5-1)

클러스터 생성 및 노드 사용법





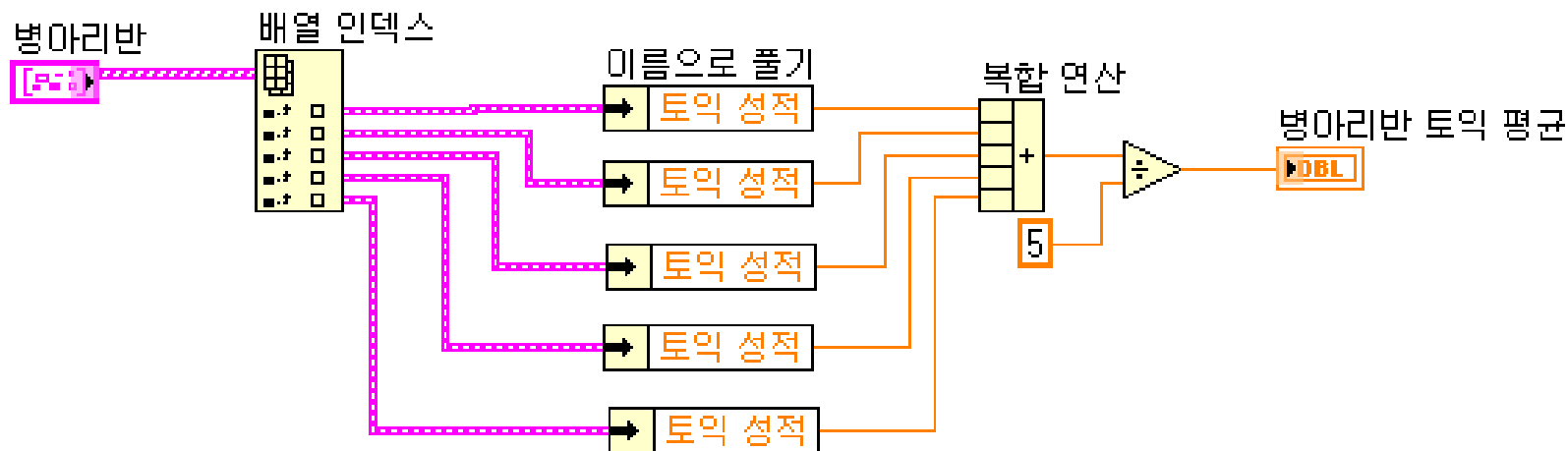
실습3-5-2) 클러스터와 배열



정답 3-5-2 배열과 클러스터.vi 블록다이어그램

파일(F) 편집(E) 보기(V) 프로젝트(P) 수행(Q) 도구(T) 윈도우(W) 도움말(H)

12pt 어플리케이션 폰트





웨이브폼 데이터

DAQ(Data Acquisition) 하드웨어를 통해 실제 신호를 받으면 t_0 , dt , $Y[]$ 로 구성된 데이터를 수신

웨이브폼

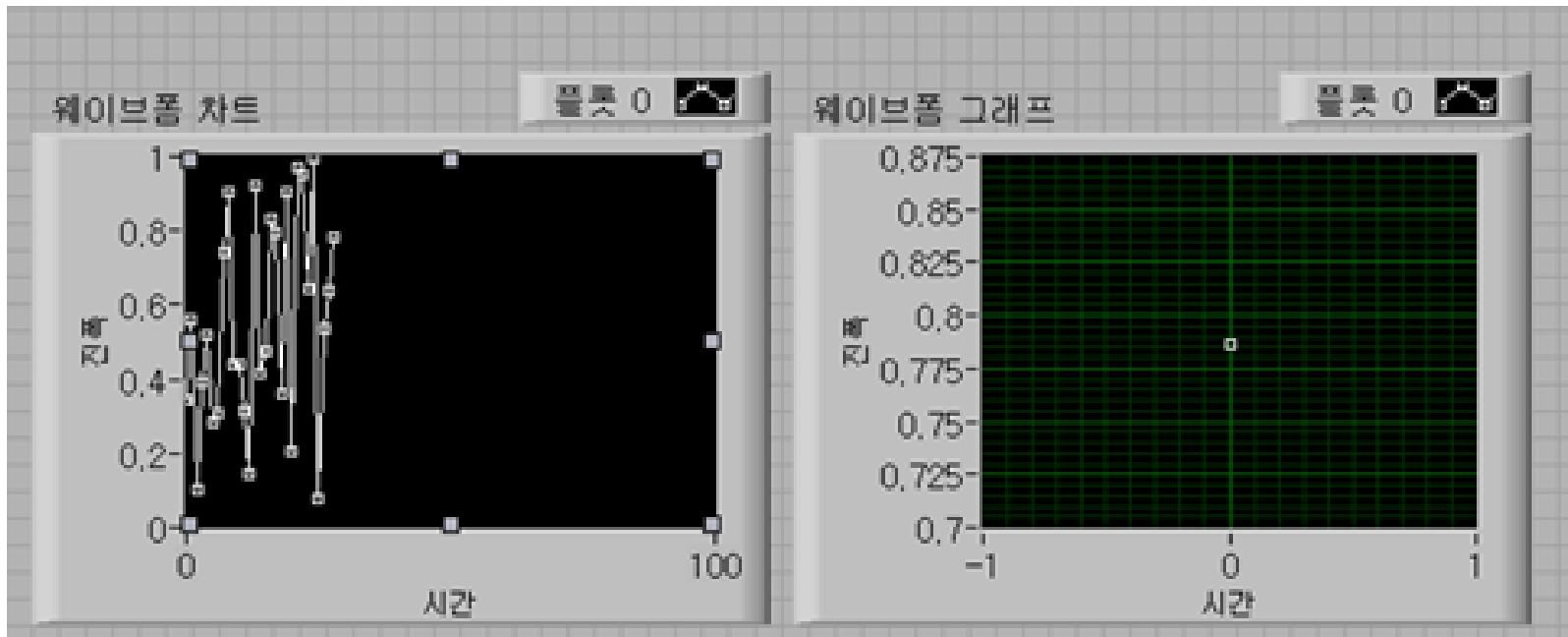
t_0	Y  0
오후 00:00:00 YYYY-MM-DD	0,132
	0,3455
dt	0,75645
1,000000	0,7987
	1,0211

- t_0 : 데이터 획득 시작 시간
- dt : 데이터 획득 간격
- $Y[]$: 획득한 데이터

웨이브폼 데이터는 주로 웨이브폼 차트나 웨이브폼 그래프를 통해 데이터를 보는 것이 일반적



웨이브폼 차트/웨이브폼 그래프



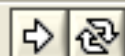
- 웨이브폼 차트 : 데이터값 누적
- 웨이브폼 그래프 : 현재 데이터만 디스플레이

실습3-6-1)

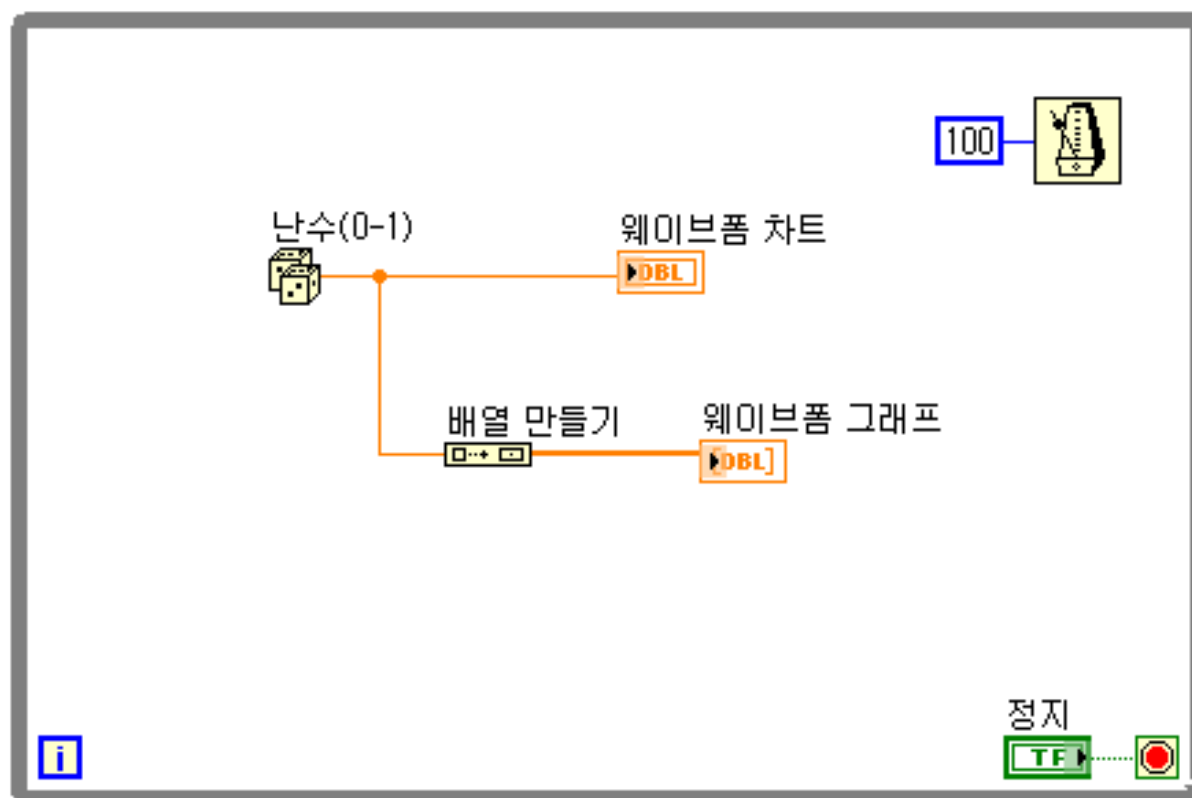
웨이브폼 차트와 그래프

정답 3-6-1 그래프와 차트.vi 블록다이어그램 *

파일(F) 편집(E) 보기(V) 프로젝트(P) 수행(O) 도구(T) 윈도우(W) 도움말(H)

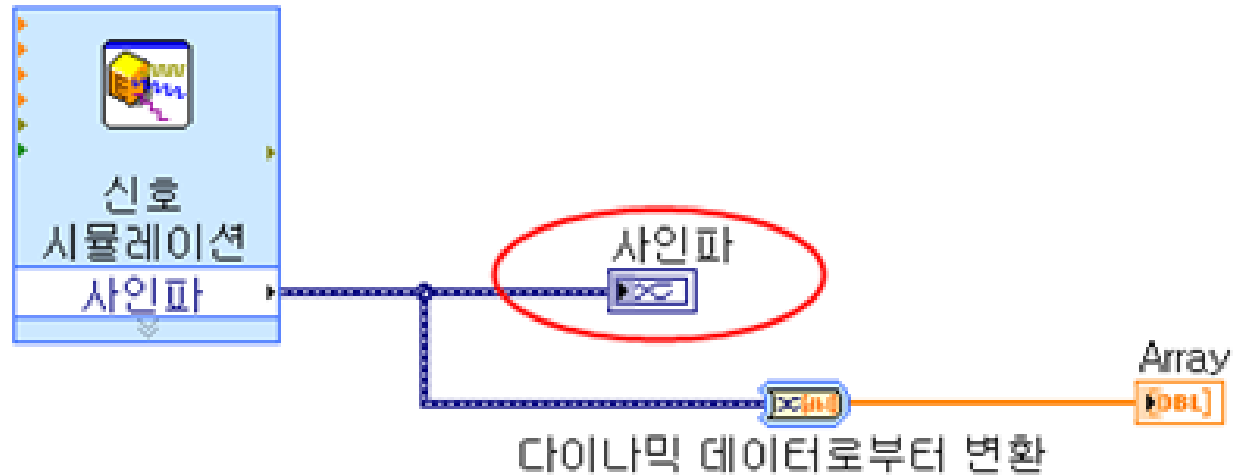


12pt 머플리케이션 폰트





다이나믹 데이터 타입

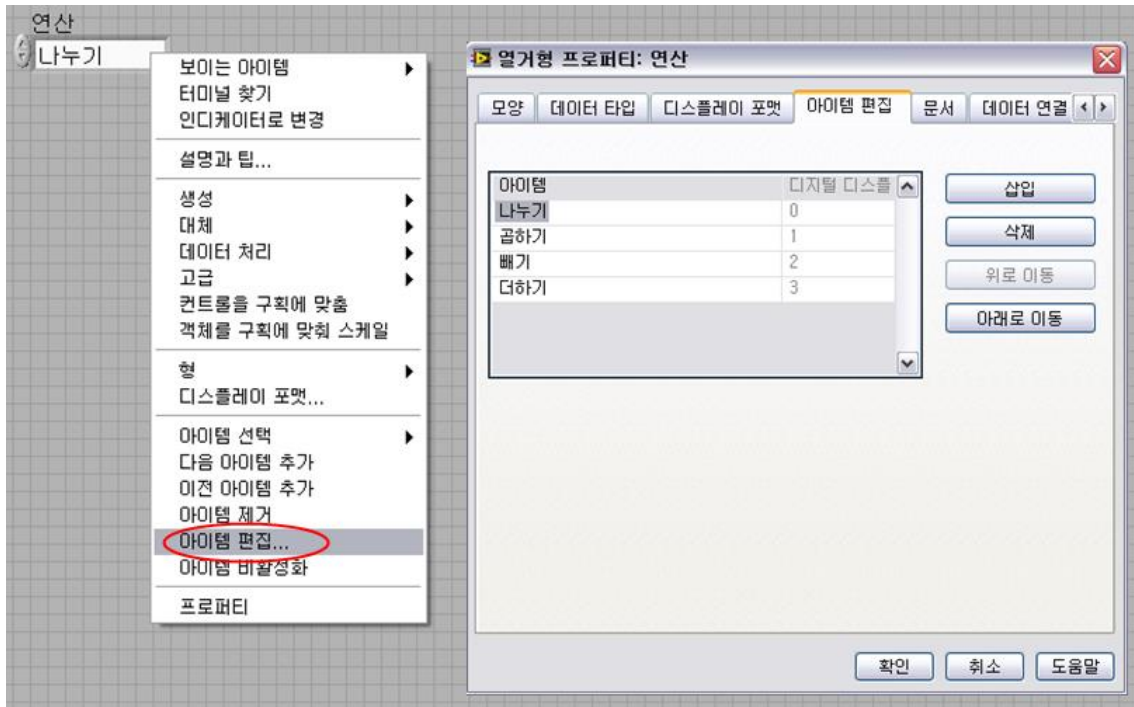


- 다이나믹 데이터 : 익스프레스 전용 데이터 타입
- 터미널 색상은 짙은 파란색

다이나믹 데이터를 다른 데이터 타입과 연결하면 자동으로 데이터 타입을 변환하는 노드가 추가



열거형(Enum)



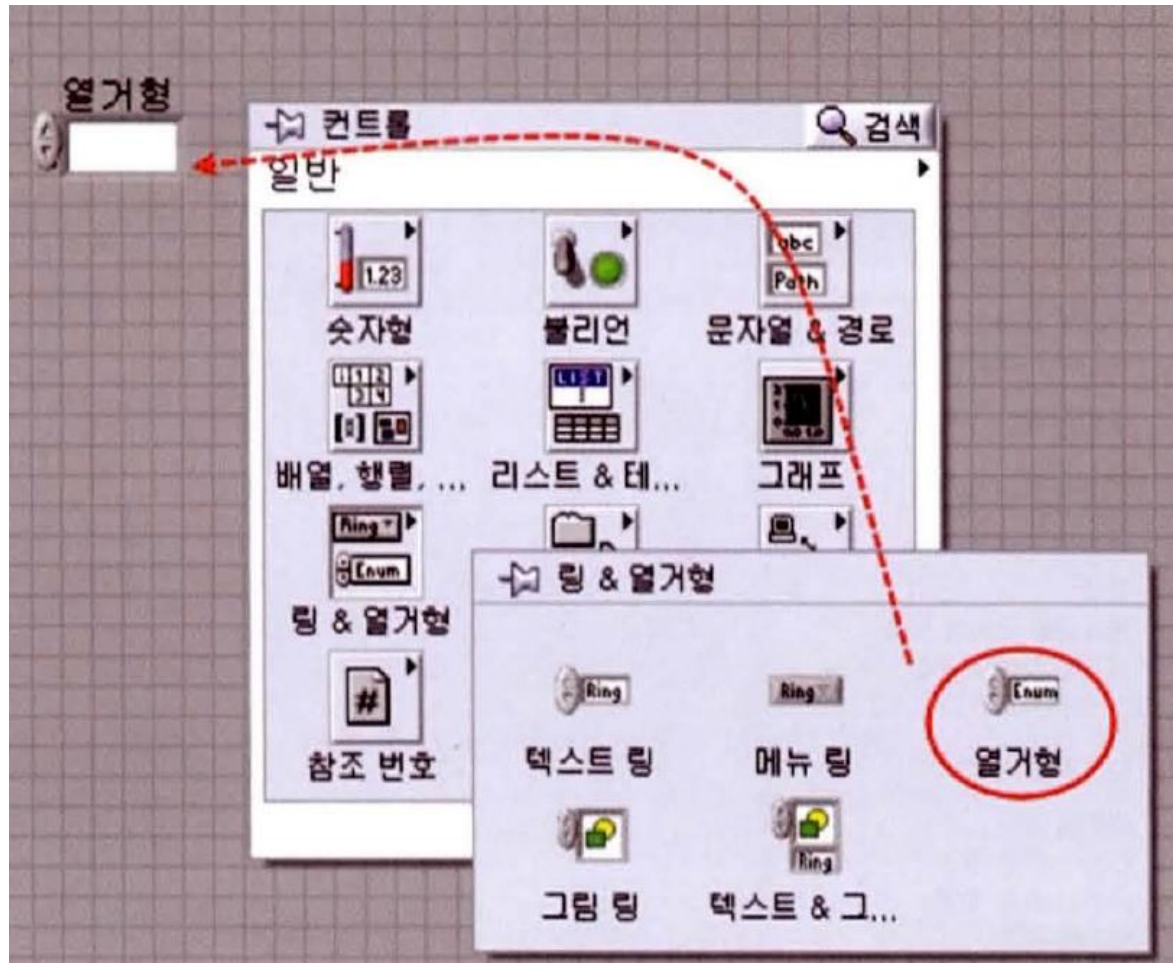
- 정의된 아이템 중 하나를 선택해서 사용
- 4장에서 배울 케이스 구조와 함께 많이 사용

실습3-8-1) 열거형



실습 3-8-1

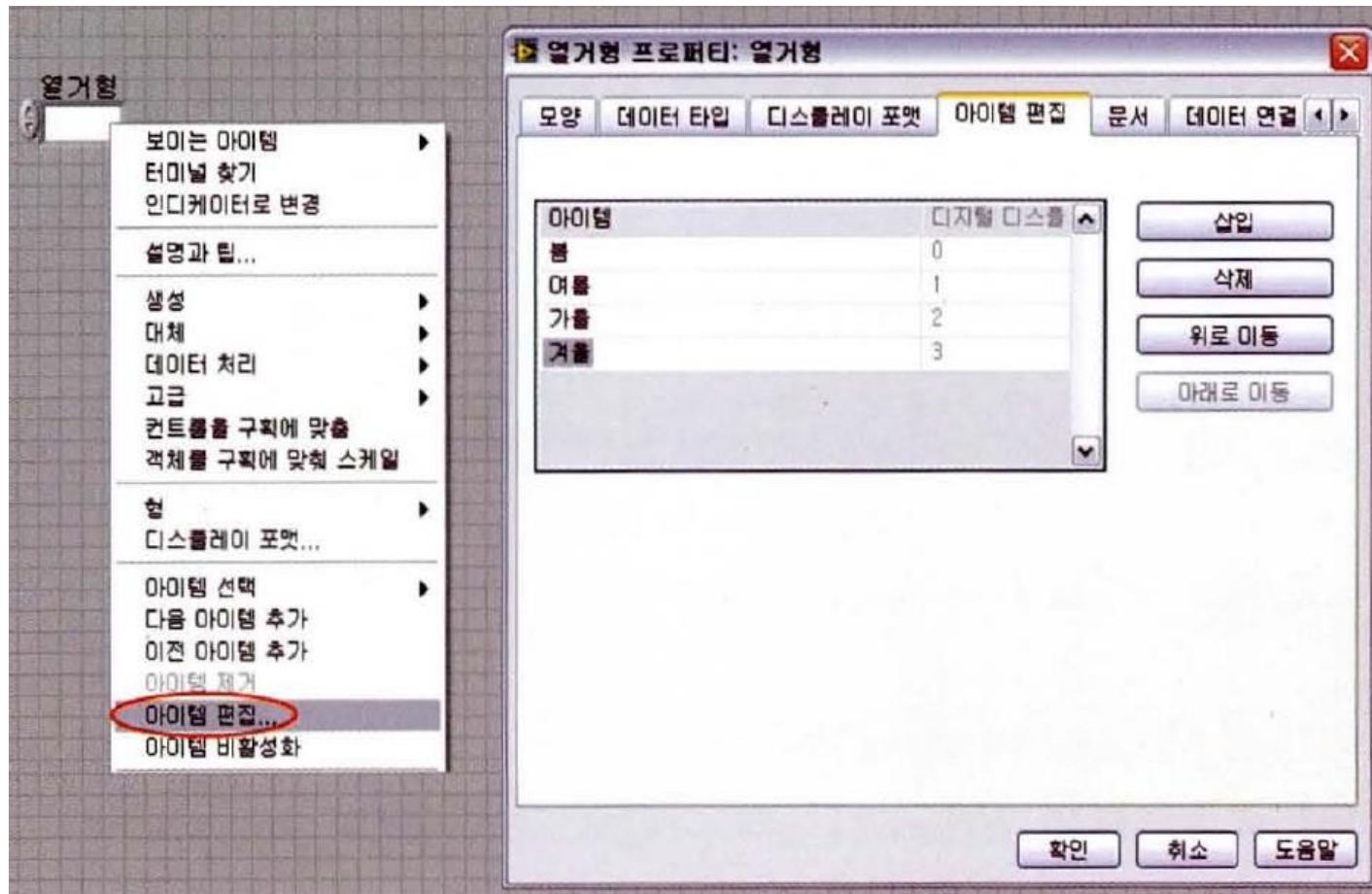
- ❖ 1. 새 VI를 열어 프런트 패널에 열거형을 가져다 놓습니다





실습 3-8-1

- ❖ 2. 열거형의 바로가기메뉴 > 아이템 편집 ... 을 통해 그림처럼 '봄', '여름', '가을', '겨울'을 입력합니다





실습 3-8-1

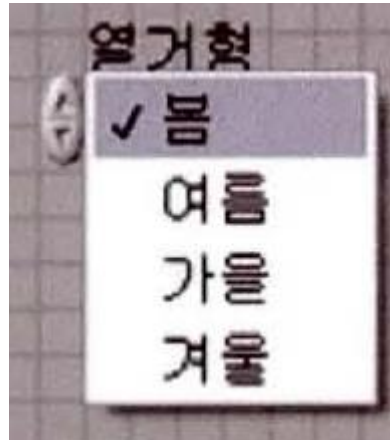
- ❖ 2. 열거형의 바로가기메뉴 > 아이템 편집 ... 을 통해 그림처럼 '봄', '여름', '가을', '겨울'을 입력합니다





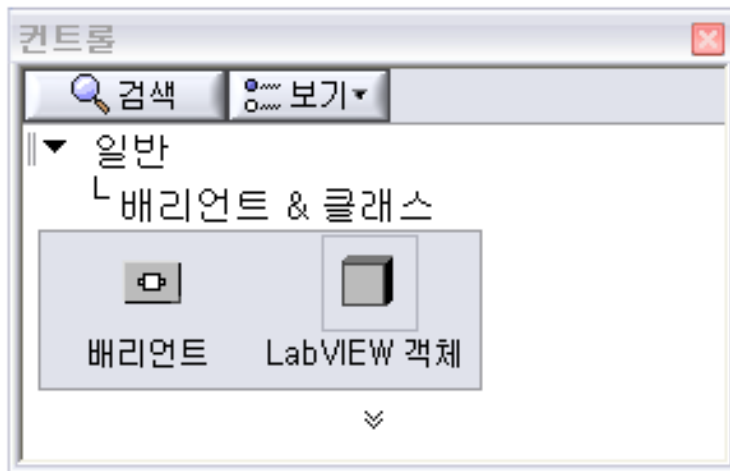
실습 3-8-1

❖ 그림처럼 사계절의 메뉴가 나타나는지 확인합니다





배리언트(Variant)

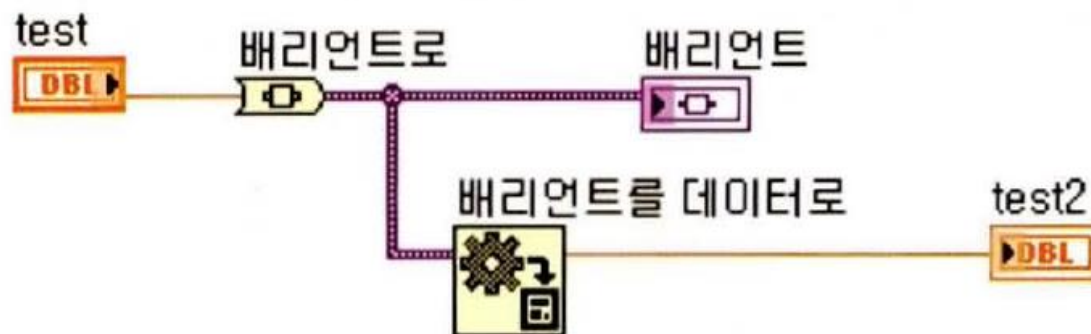


- 프로그램의 유연성을 제공함. 프로그램 유지 관리 시 용이함.
- 단 **RTOS**에서는 사용할 수 없음.

숫자형, 문자형 등과 같이 하나로 고정된 데이터 타입이 아니라 어떠한 데이터 타입도 배리언트로 변환이 가능하고 반대로 배리언트는 원래의 데이터 타입으로도 복원이 가능

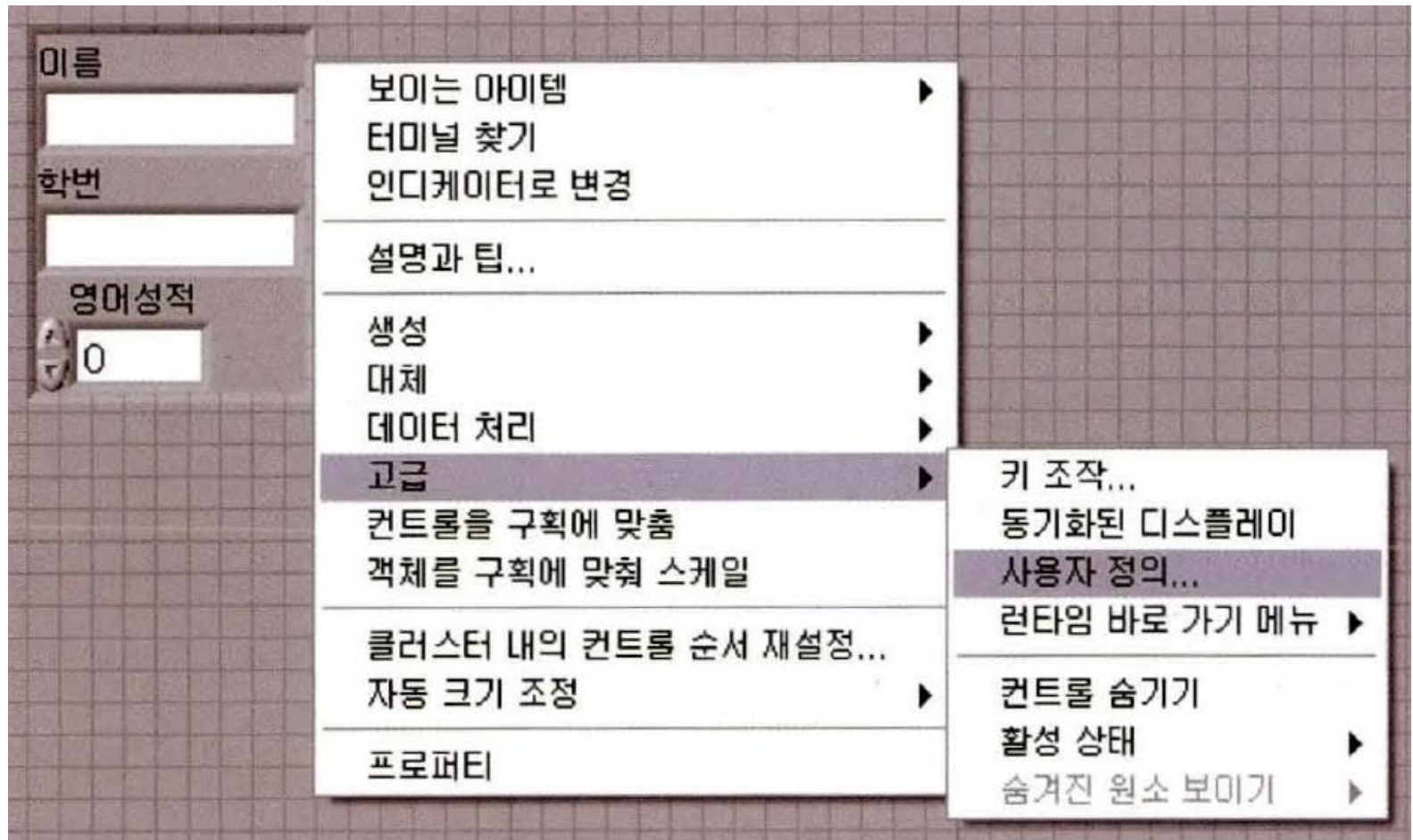


배리언트(Variant)



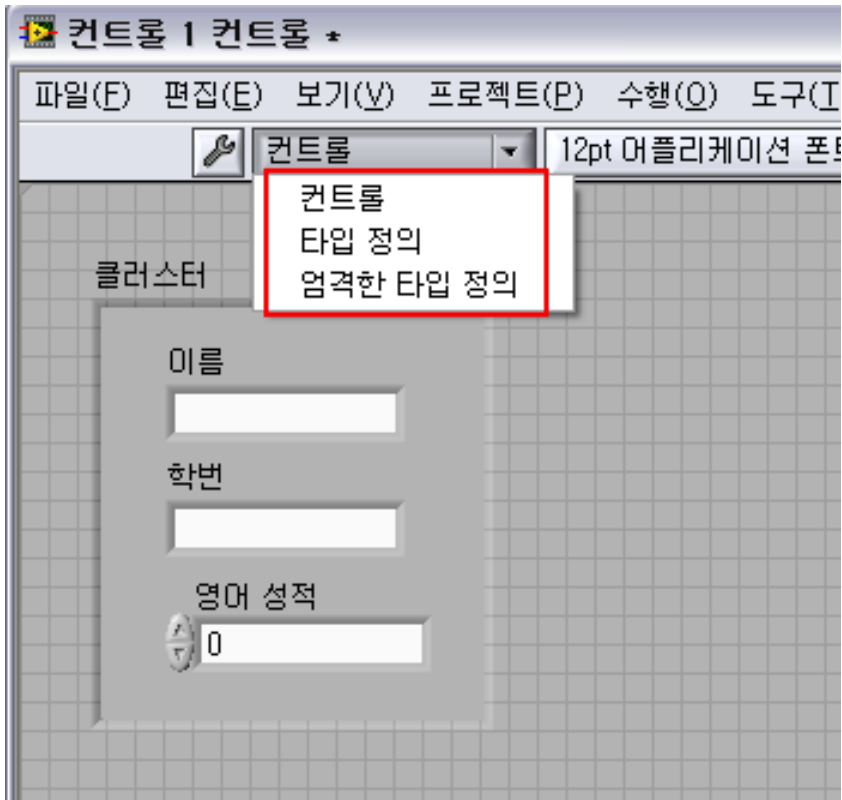


타입정의(Type Definition)





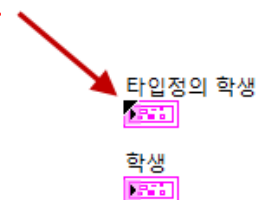
타입정의(Type Definition)



•사용자가 정의하는 컨트롤로써
여러 프로젝트에서 손쉽게 가져다가
재사용 가능함.

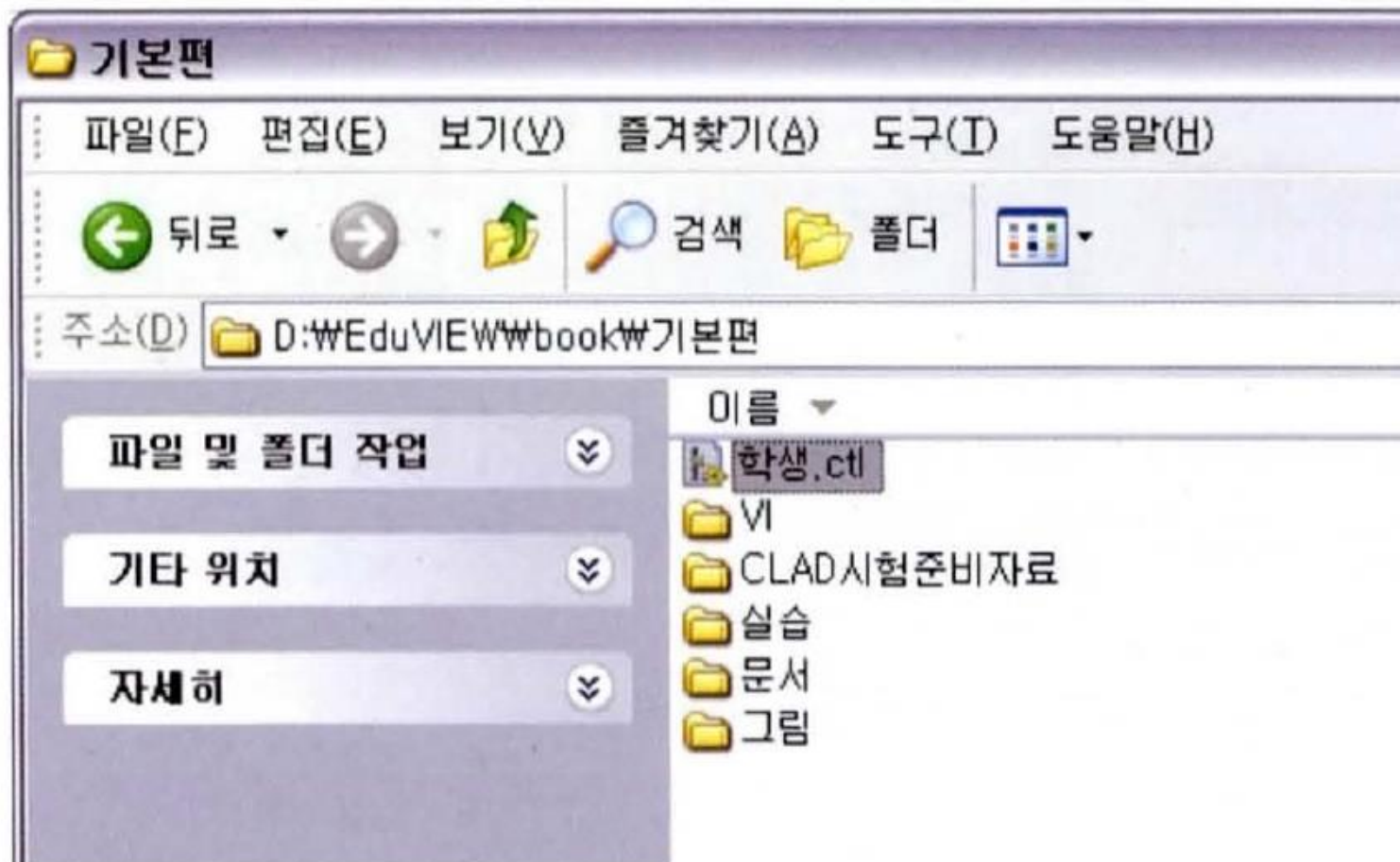
- ✓컨트롤 - 단순한 사용자 정의
- ✓타입 정의 - 파일을 변경하면 사용하는
모든 컨트롤들의 프로퍼티가 함께 변경됨.
크기 등 제외
- ✓엄격한 타입 정의 - 크기를 비롯해 모든
컨트롤들의 프로퍼티가 함께 변경됨.
이름, 캡션, 초기값 등 제외

*터미널에 타입 정의된 의미로 검정색 삼각형 표시가 됨.



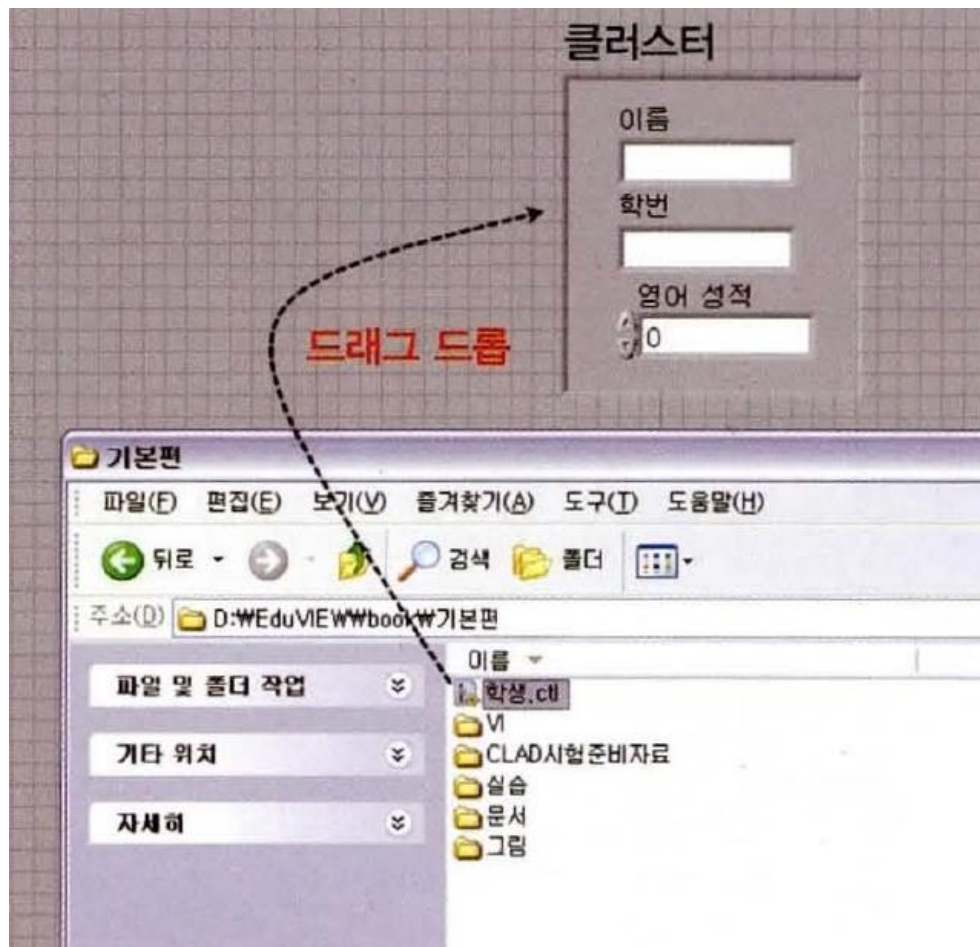


❖ 세가지 중 하나를 선택하여 파일폴다운메뉴 > 파일 > 저장
을 하게 되면 확장자가 '.ctl'로 생성.



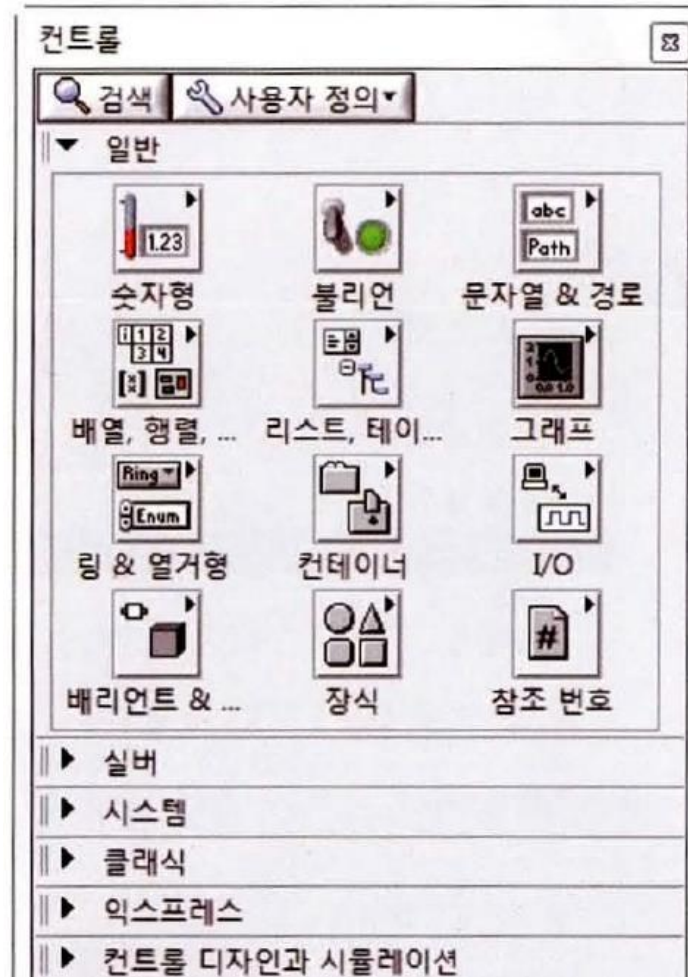


❖ 이를 사용하는 방법은 파일을 선택하여 프런트패널에 드래그 드롭하는 방법이 있다.





❖ 또는 컨트롤 팔레트 > 컨트롤 선택을 이용





❖ 원본 파일을 여는 방법은 직접 '.ctl'을 선택하거나 복사본의 바로가기메뉴 > 타입 정의 열기를 선택하면 원본파일이 열립니다

