

4장. ER 모델의 고급 기능

2018년 9월 17일 (Mon)

송 병 호, 상명대학교

Prof. **Byoungho Song**, Ph. D.

Sangmyung University

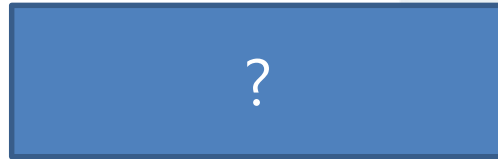
들어가기 전에..

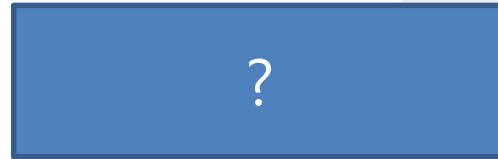
Note : ER의 기본적인 추상화

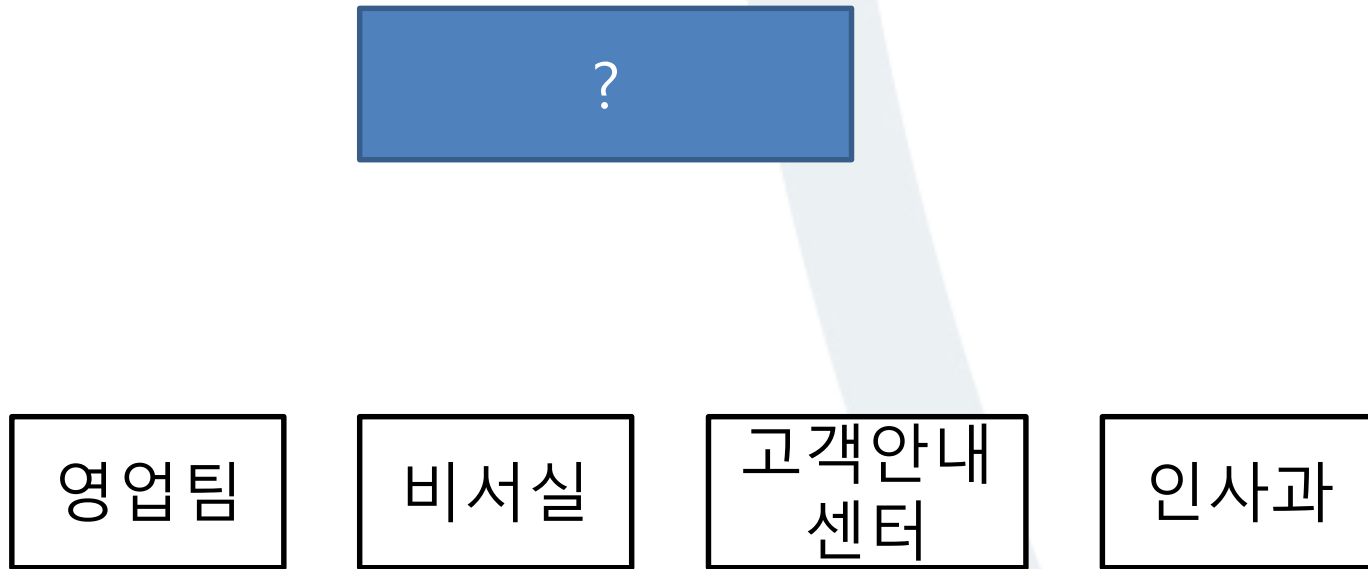
추상화된 개념



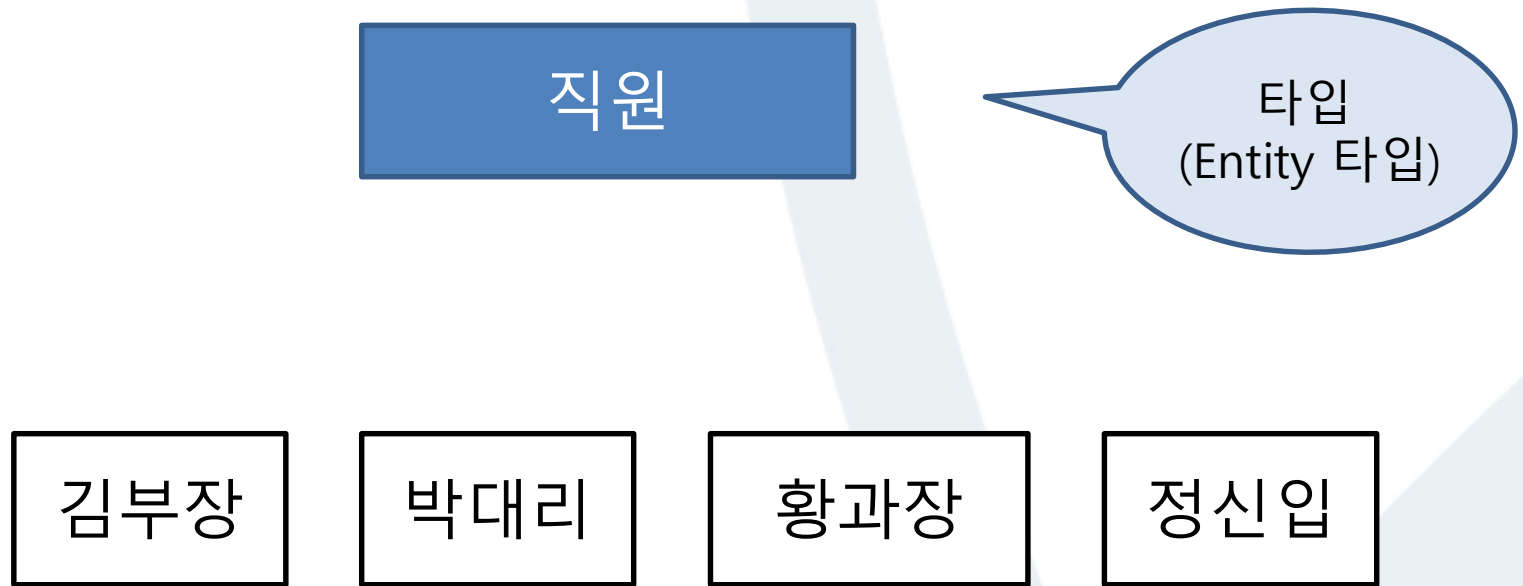
- 추상화 (ch.3)
 - 현실은 값(인스턴스)들로 구성된다
 - 모델링은 추상화된 타입 관점으로 묘사한다





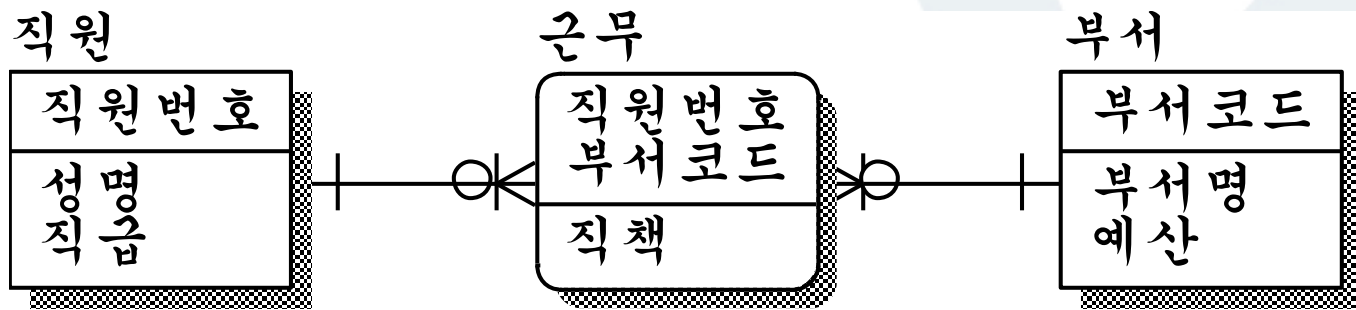
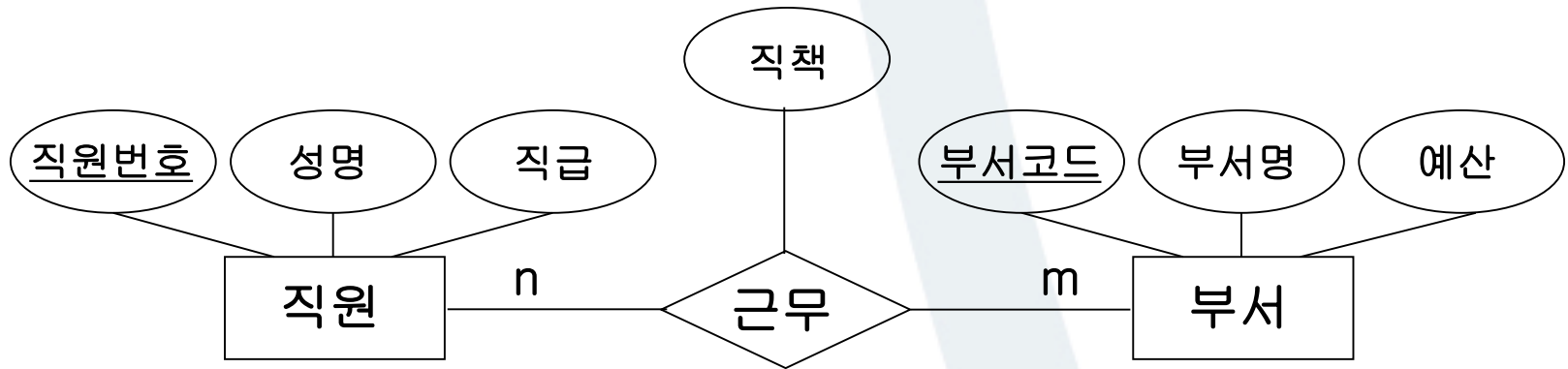


타입 : 어떤 부류의 인스턴스를 총괄

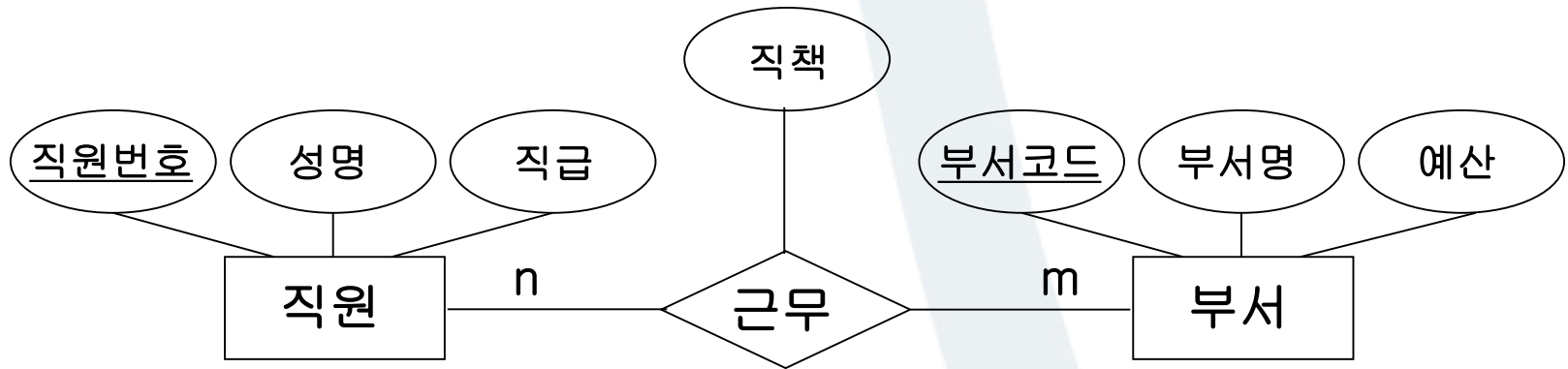


- ERD의 기본적인 추상화 :
 - Class 화 : Classification
 - Instance 화 : Instantiation

Note : Chen 표기법과 IE 표기법

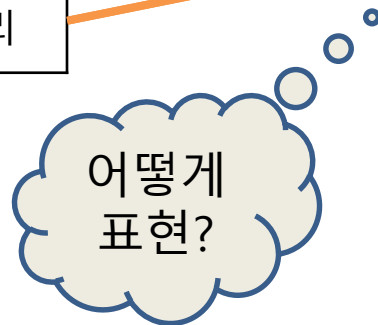


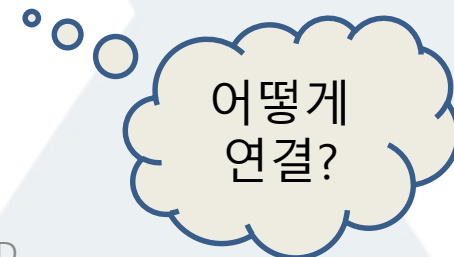
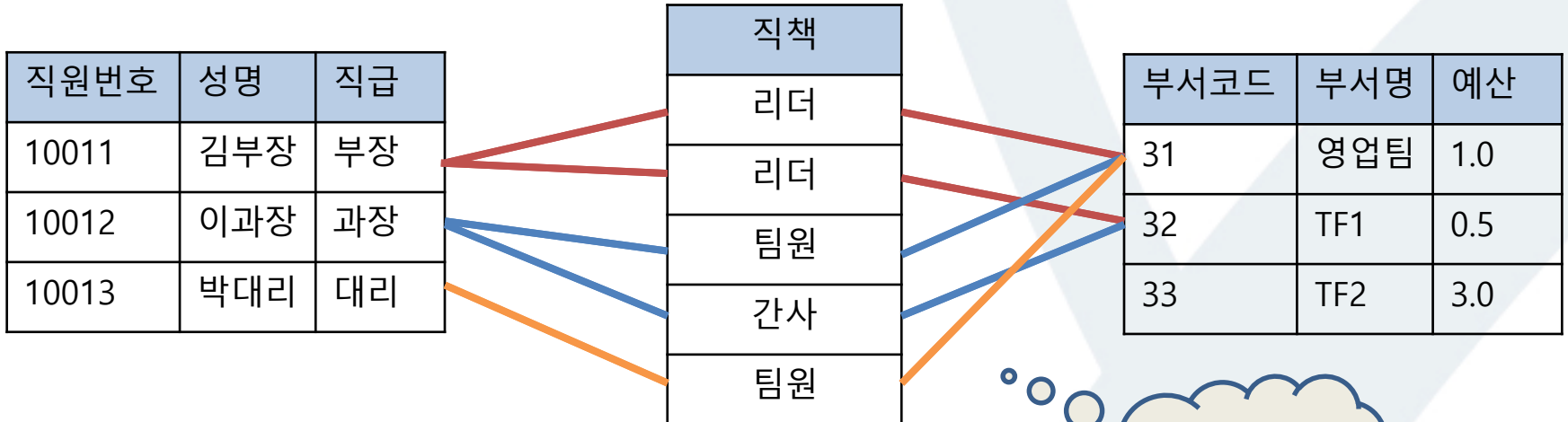
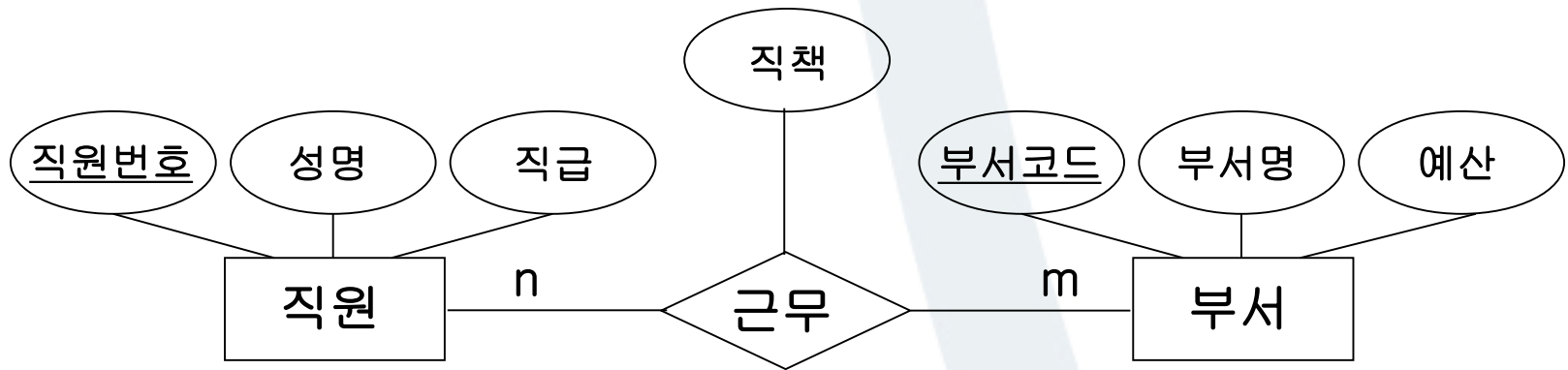
Relationship 타입과 인스턴스

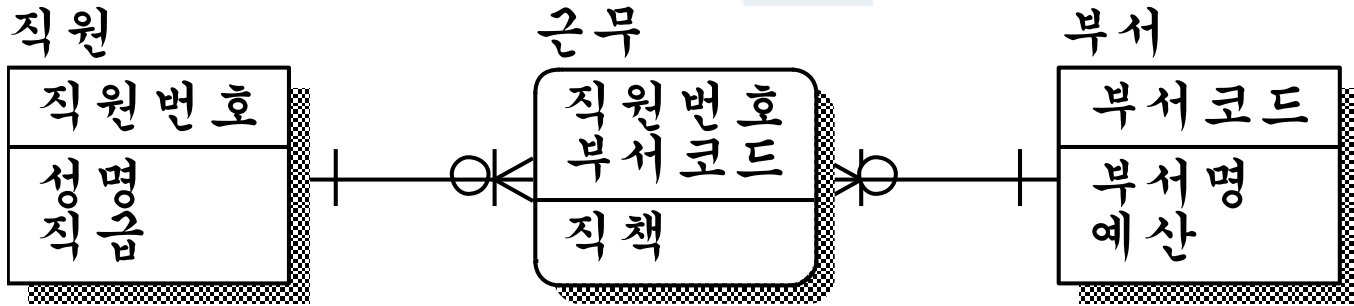


직원번호	성명	직급
10011	김부장	부장
10012	이과장	과장
10013	박대리	대리

부서코드	부서명	예산
31	영업팀	1.0
32	TF1	0.5
33	TF2	3.0







직원번호	성명	직급	직원번호	부서코드	직책	부서코드	부서명	예산
10011	김부장	부장	10011	31	리더	31	영업팀	1.0
10012	이과장	과장	10011	32	리더	32	TF1	0.5
10013	박대리	대리	10012	31	팀원	32	TF2	3.0
			10012	32	간사			
			10013	31	팀원			

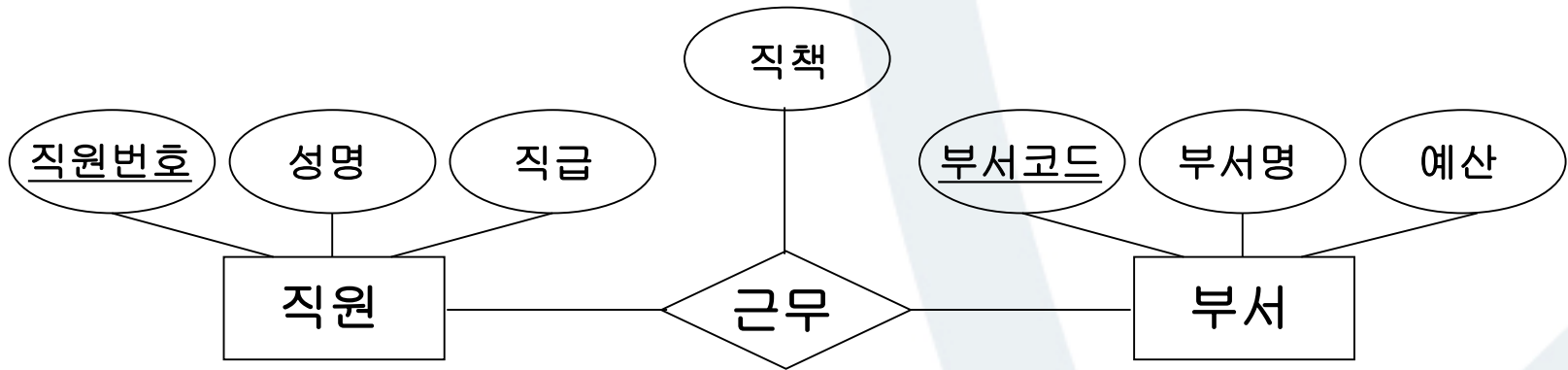
Relational 모델로
논리설계할 것을
가정

ER 모델의 고급 기능

소개할 고급 (추가) 기능

- 3진 관계
- 키 제약조건, 참여제약조건
- 약 개체, 식별관계
- 일반화 (계층구조)
- 군집화

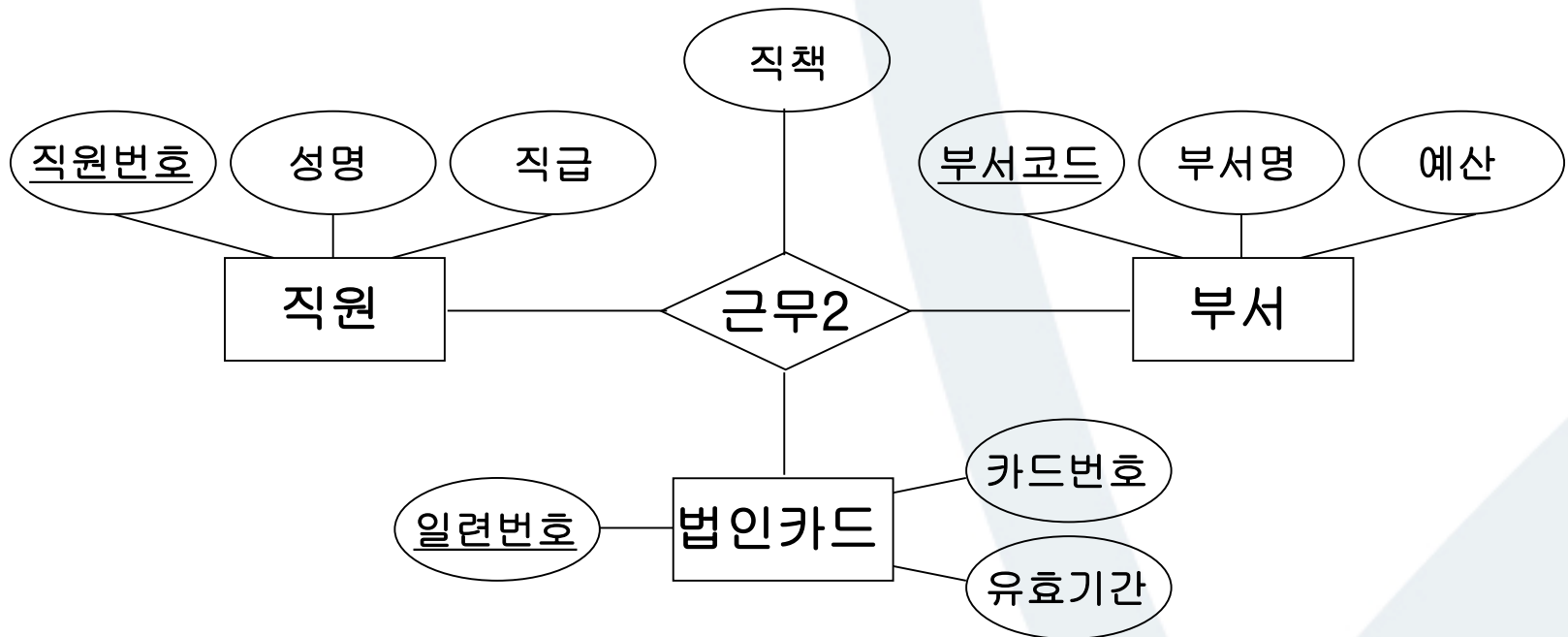
기본 예제 ERD



< 근무 관계 집합 >

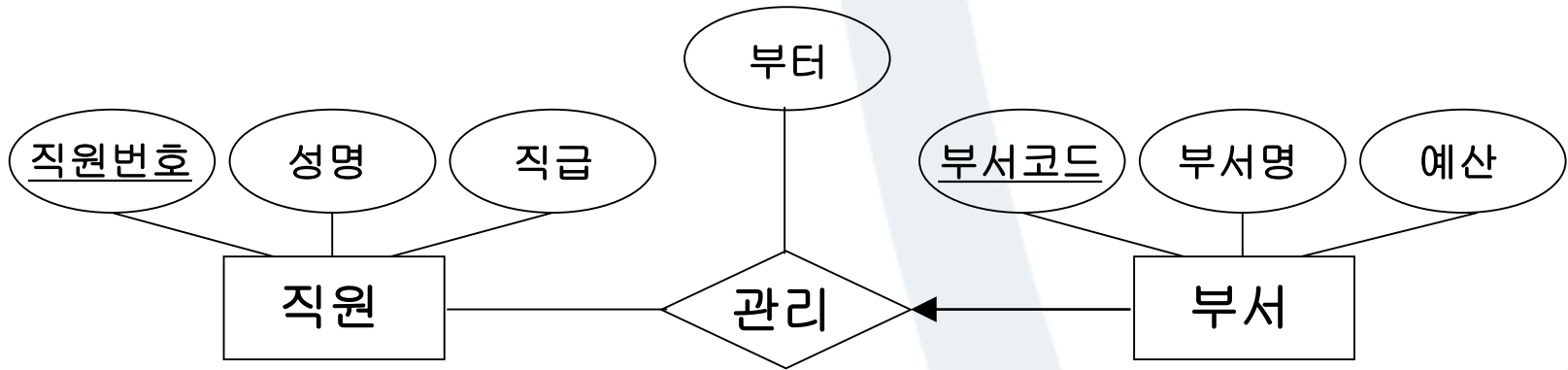
- 4장의 ERD는 표기가 없으면 n:m
키 제약조건이 한쪽에만 있으면 1:n
양쪽에 있으면 1:1이다.

3진 관계



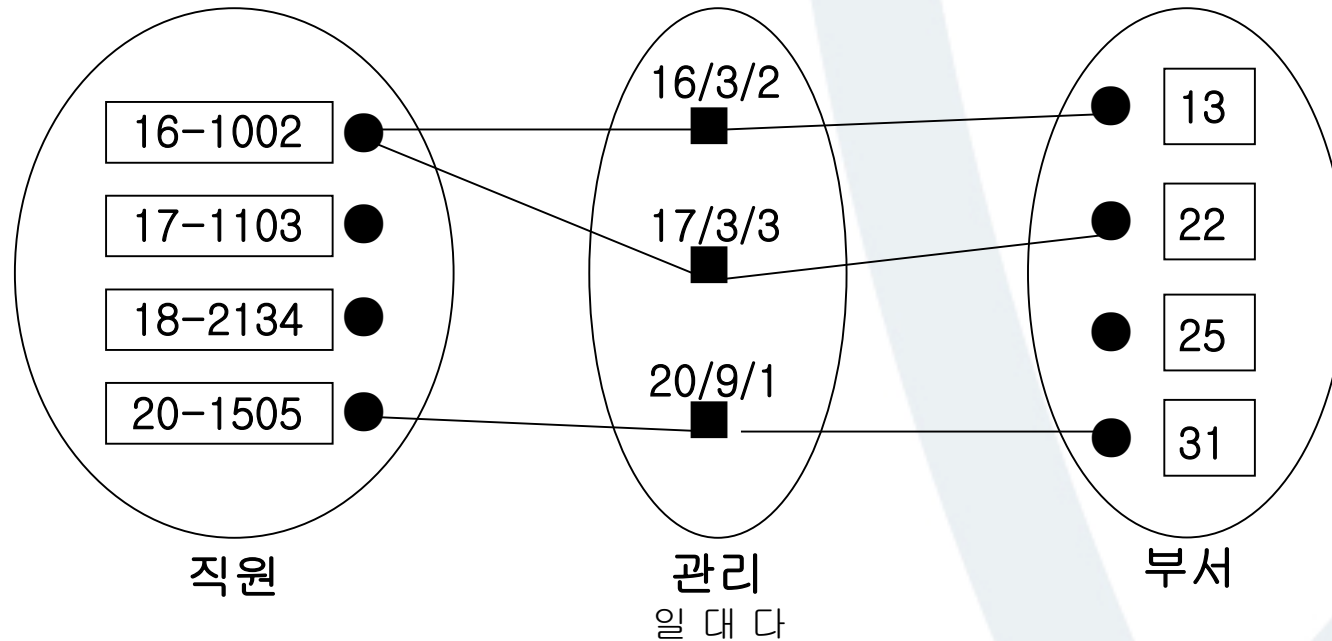
- 한 부서에서 직원이 사용하는 법인카드를 기록하고 싶다고 하자 . 이 관계에는 **직원** 한 명, **부서** 하나, **법인카드** 하나가 관련되어 있으므로 삼진(ternary)관계이다.
- 가정 : 직원별, 부서별로 사용 가능한 법인카드가 다르다.
- Note : **인스턴스 레벨로 그려볼 것!**

키 제약조건 (Key Constraints) (1)



- **직원** 개체집합과 **부서** 개체집합 간의 또 다른 관계집합인 **관리**를 생각해 보자.
- **키 제약조건**의 예
 - 각 부서에는 **많아야** 한 명의 부서장만 있다.
 - Note : **부서** 가 결정되면 **관리** 가 결정된다는 의미!
- 한 명의 직원이 여러 부서를 관리할 수 있다.

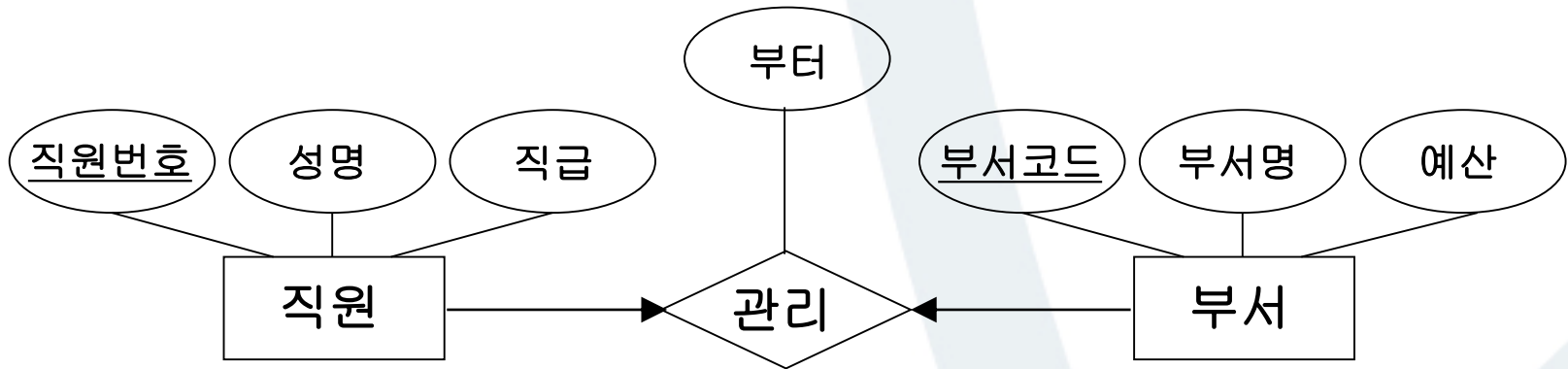
키 제약조건 (Key Constraints) (2)



< 관리 관계 집합의 한 인스턴스 >

- 관리 관계집합 : 일대다
- 근무 관계집합 : 다대다

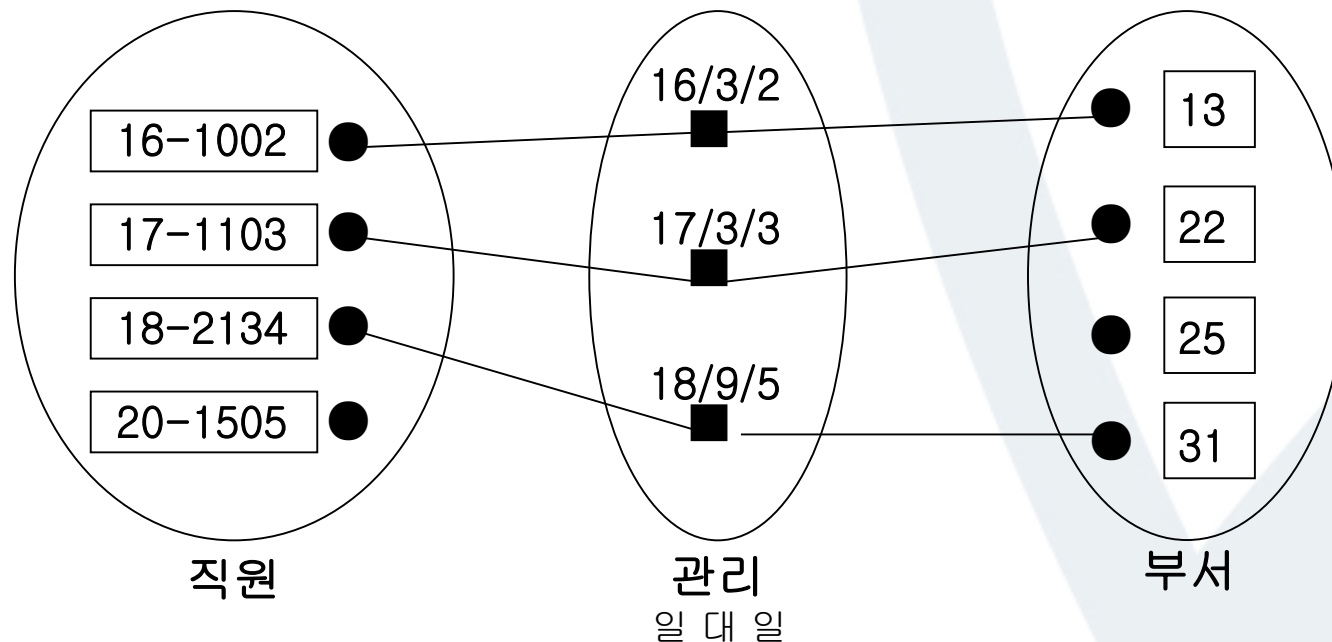
키 제약조건 (Key Constraints) (3)



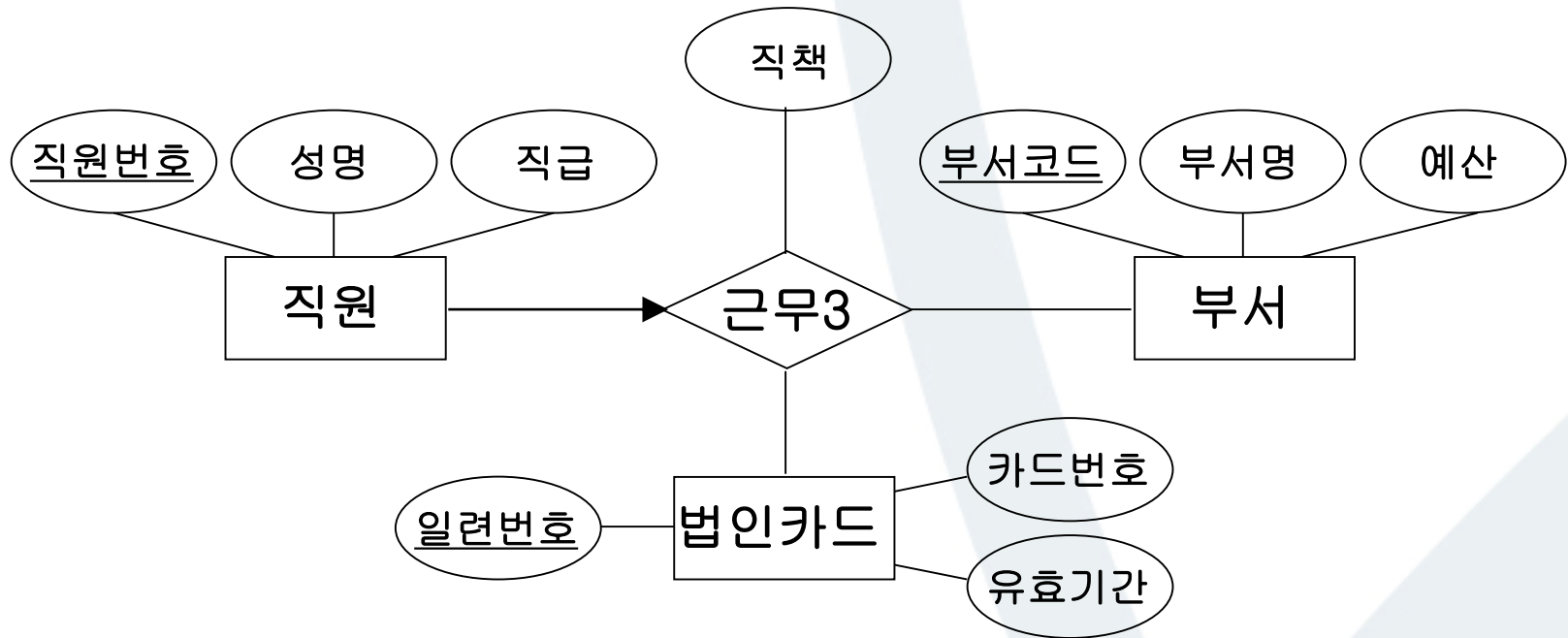
- 만일 한 직원이 많아야 하나의 부서만 관리할 수 있다는 제약을 **관리** 관계집합에 부가한다면 다음과 같이 **직원**에서 **관리**로 가는 화살표도 추가하여 표시
- 일대일(One-to-One)

키 제약조건 (Key Constraints) (4)

- 일대일 관리 관계 집합의 한 인스턴스



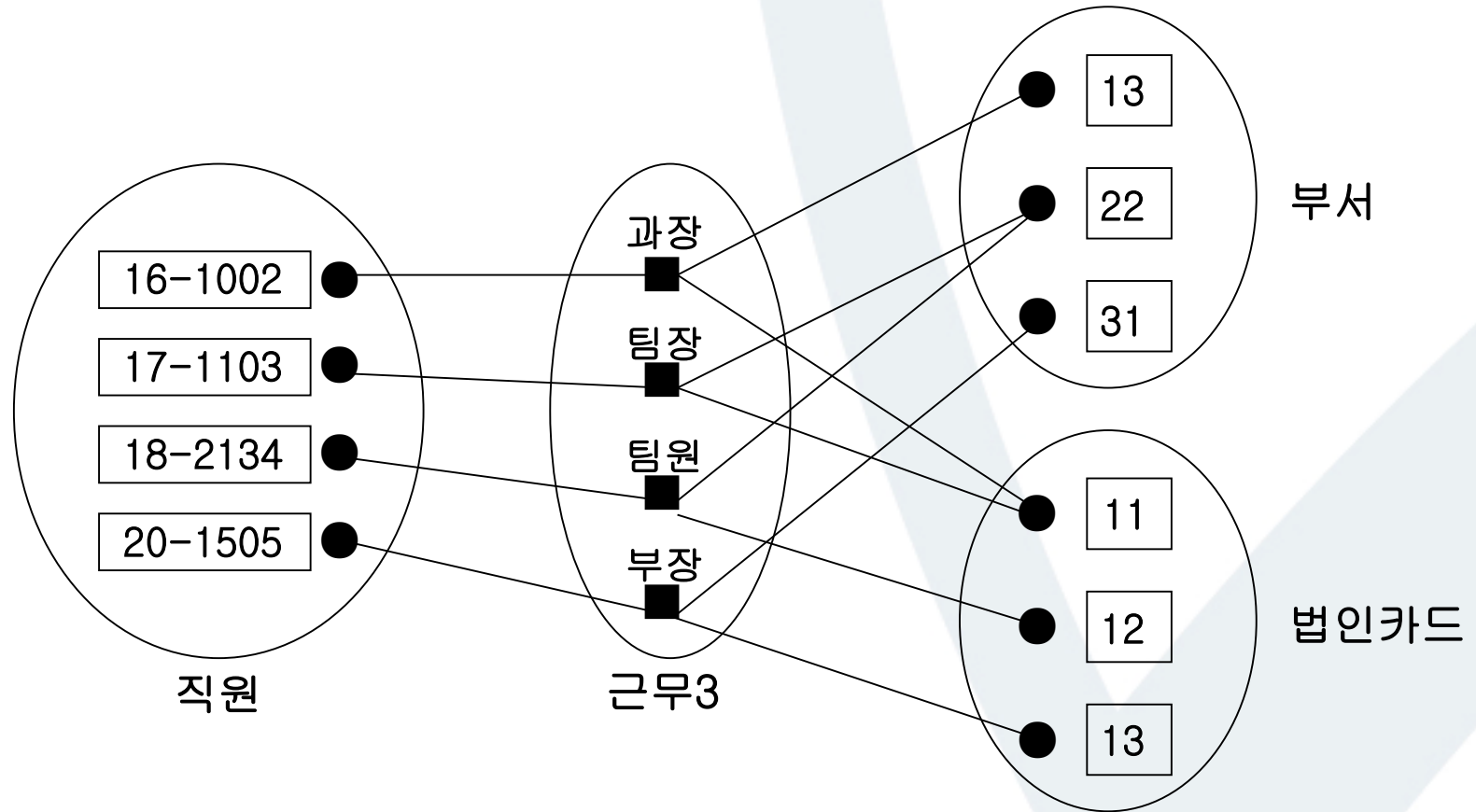
키 제약조건 (Key Constraints) (5)



< 키 제약조건이 있는 3진 관계 집합 >

- 한 부서에서는 여러 직원과 여러 법인카드가 관련될 수 있고, 또 한 법인카드에 여러 부서와 여러 직원이 관련될 수 있지만, 한 직원에게는 단일 부서와 단일 법인카드만 관련될 수 있다.

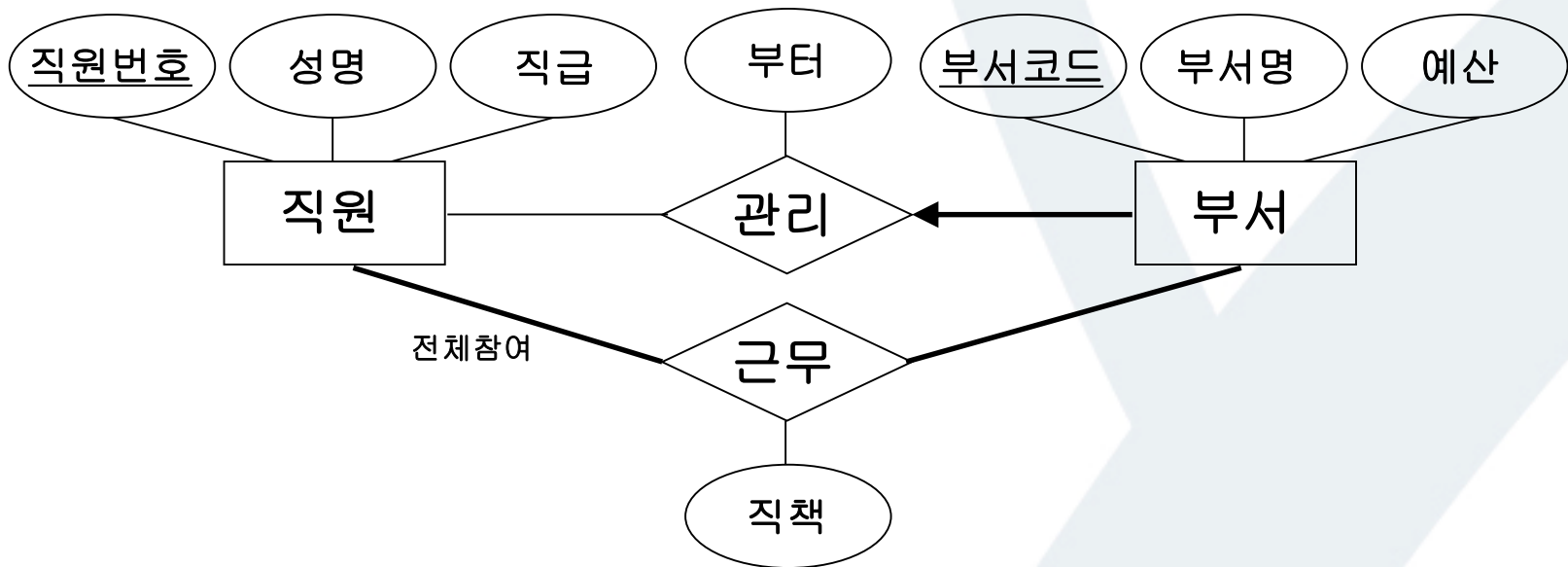
키 제약조건 (Key Constraints) (6)



< 근무 3의 한 인스턴스 >

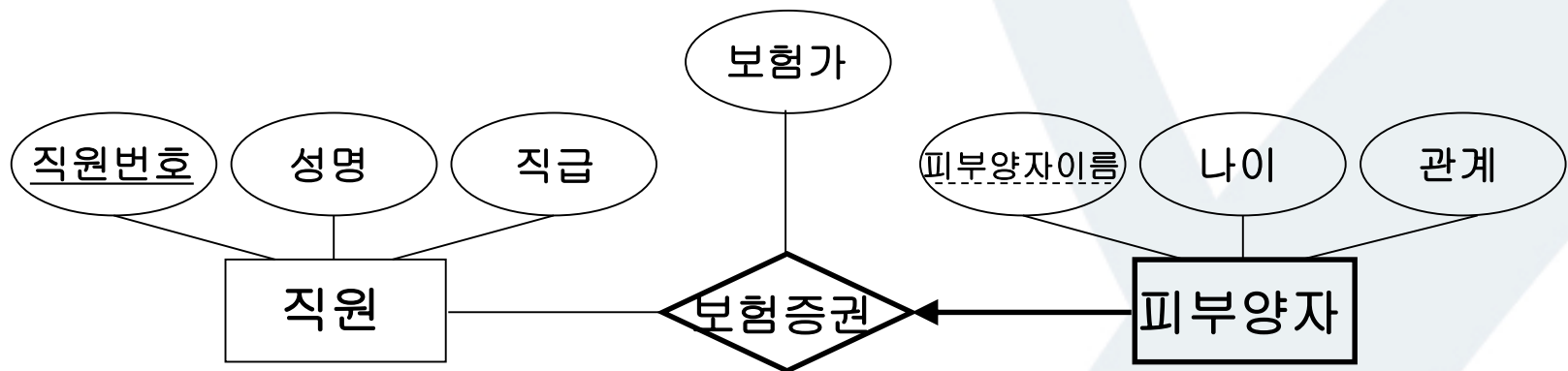
참여 제약조건 (Participation Constraints)

- 모든 부서에 부서장이 있는가?
 - 그렇다면 **참여제약조건**이 된다.
 - 관리에 대한 부서의 참여도를 **전체적(Total)**이라고 한다.
 - 전체적이 아닌 참여도를 **부분적(Partial)**이라고 한다
 - 부서 테이블의 어떤 '부서코드'값도 빠짐없이 관리 테이블에 한 행으로 나타나야 한다.(이때 '직원번호'는 널이 될 수 없다)
 - Note : 필수 / 선택 개념!



약 개체 (Weak Entity)

- 약 개체는 다른(소유자) 개체의 기본 키까지 보아야 유일하게 식별할 수 있다.
 - 소유자 개체집합과 약 개체집합은 **1대다** 관계집합으로 연결된다. (소유자 하나에 여러 개의 약 개체)
 - 약 개체집합은 이 **식별 (Identifying) 관계집합**에 전체적으로 참여한다.

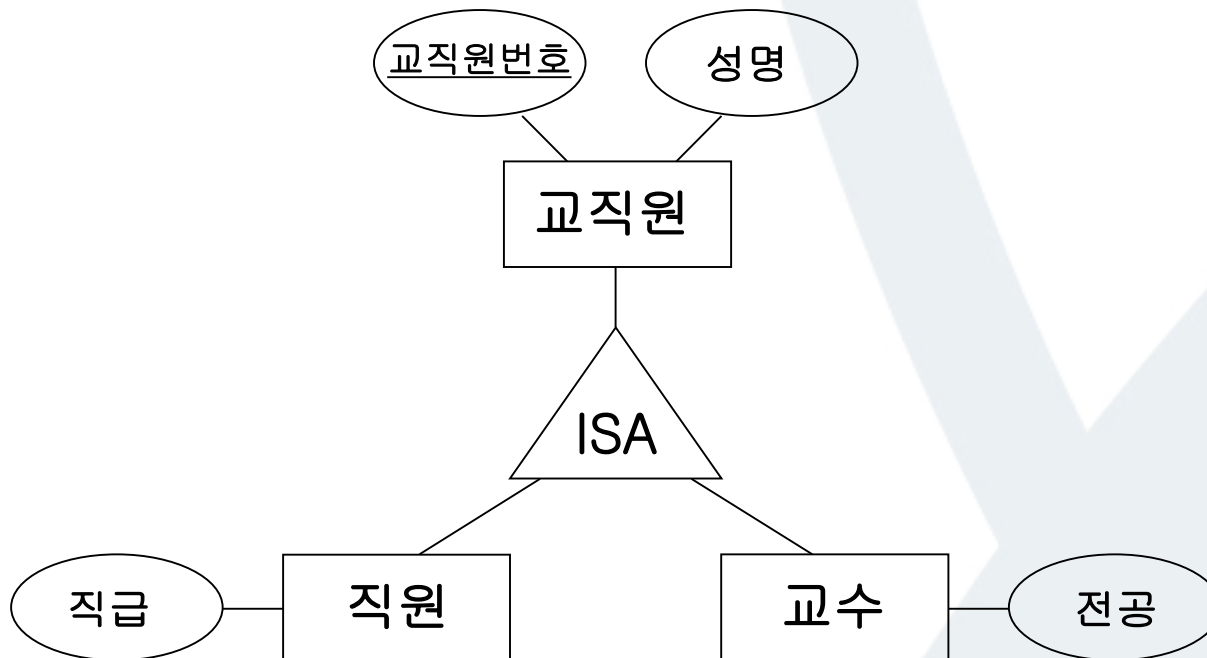


식별 관계 (Identifying Relationship)

- 약 개체는 자기 자신의 일부 애트리뷰트와 다른 개체의 기본 키를 조합하여야만 유일하게 식별 되는데, 이때 이 다른 개체를 **식별 소유자 (Identifying owner)**라고 한다.
- 소유자 개체집합과 약 개체집합 사이에는 일대 다 관계집합이 있어야 한다. 이러한 관계집합을 해당 약 개체집합에 대한 **식별 관계집합 (Identifying Relationship)**이라고 한다.
- 주어진 소유자 개체에 대하여 약 개체 하나를 유일하게 식별해 주는 약 개체집합의 애트리뷰트 집합을 그 약 개체집합에 대한 **부분키 (Partial Key)**라고 한다.
- **피부양자이름**은 **피부양자**에 대한 부분키

일반화 (Generalization) (계층구조) (1)

- ISA ('is a')계층
 - 객체지향언어처럼 애트리뷰트들이 상속(Inheritance)된다.
 - A **ISA** B 라고 선언하면, A개체는 모두 B개체로도 볼 수 있다.
- Note : Classification과 구별할 것!

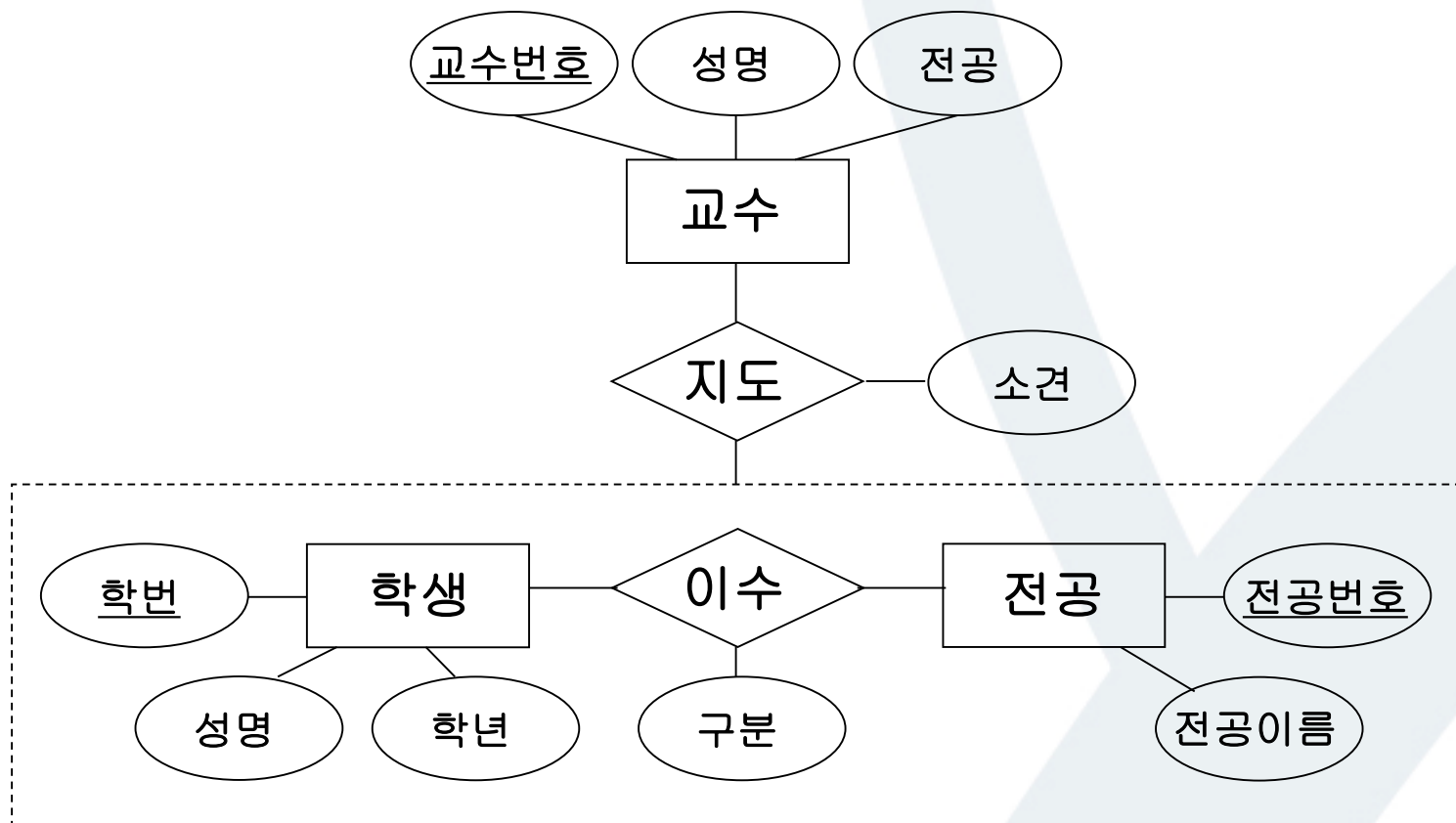


일반화 (계층구조) (2)

- 추가 가능한 조건 :
 - 중첩(Overlap) 제약조건:
홍길동이 직원이면서 교수도 될 수 있는가?
(허용/불허)
 - 포괄(Covering) 제약조건:
교직원들은 모두 직원 아니면 교수인가?
(예/아니오)
- ISA를 사용하는 이유:
 - 세부 부류에 설명용 애트리뷰트를 추가하고 싶어서.
 - 특정 관계에 참여하는 개체들을 구별해 주기 위해서.

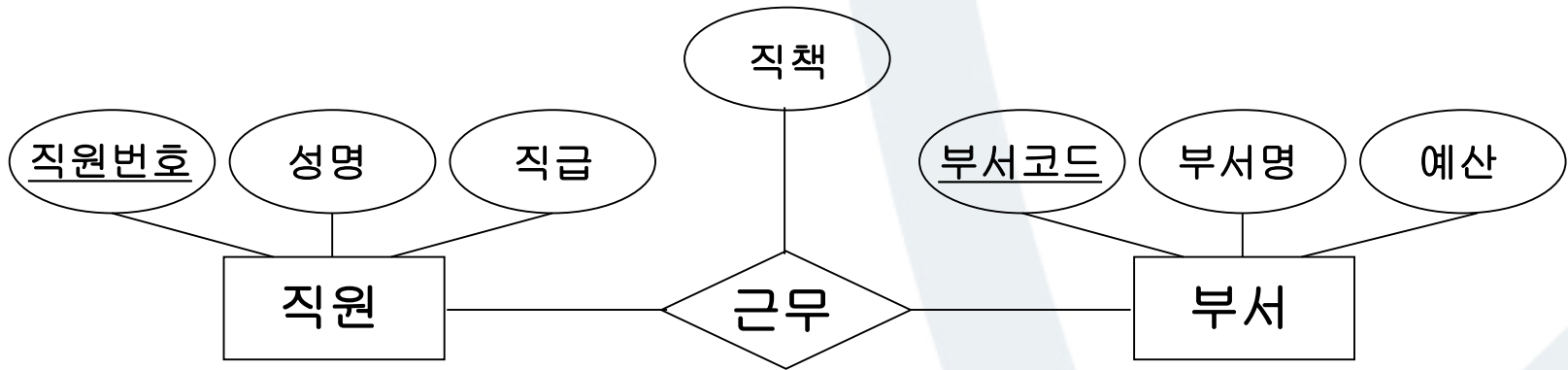
군집화 (Aggregation)

- (개체집합과) 관계집합이 참여하는 (더큰) 관계를 모델링 할 때 사용.
 - **군집화**를 사용하면 관계집합을 일종의 개체집합으로 취급하여 또 다른 관계에 참여 시킬 수 있다.



ERwin 으로 표현하기

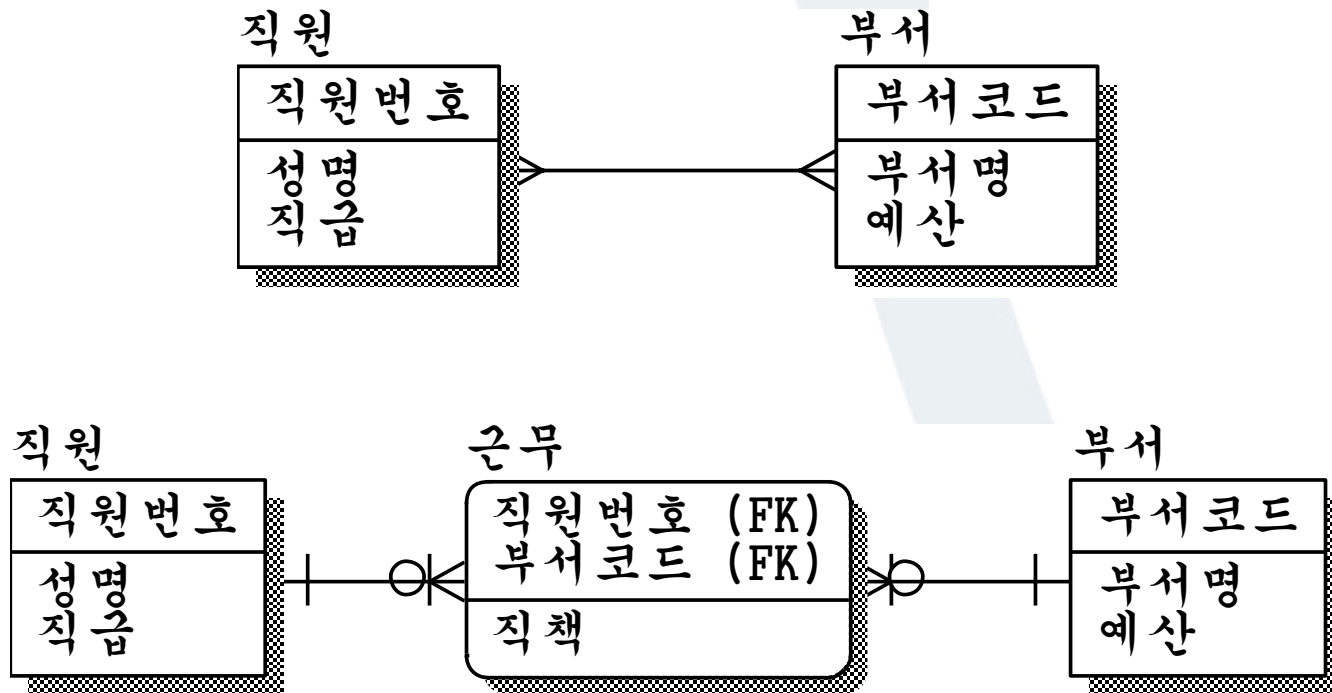
ERwin - 기본 예제 ERD 다시보기

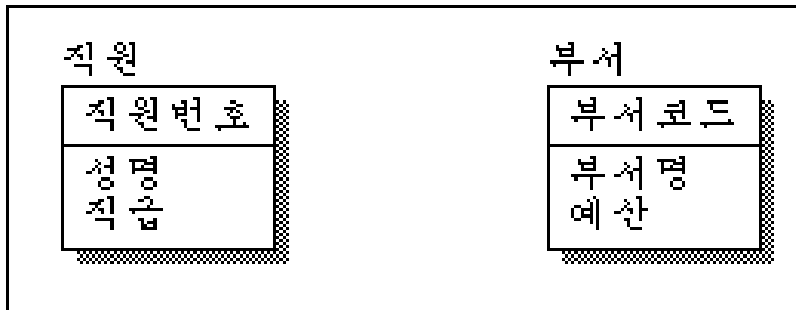
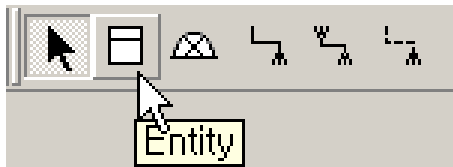


< 근무 관계 집합 >

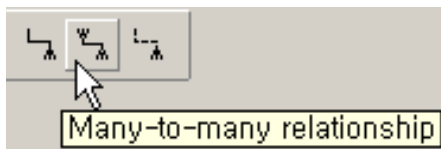
- 4장의 ERD는 표기가 없으면 n:m
키 제약조건이 한쪽에 있으면 1:n
양쪽에 있으면 1:1이다.

ERwin의 기본 예제 ERD 표기법



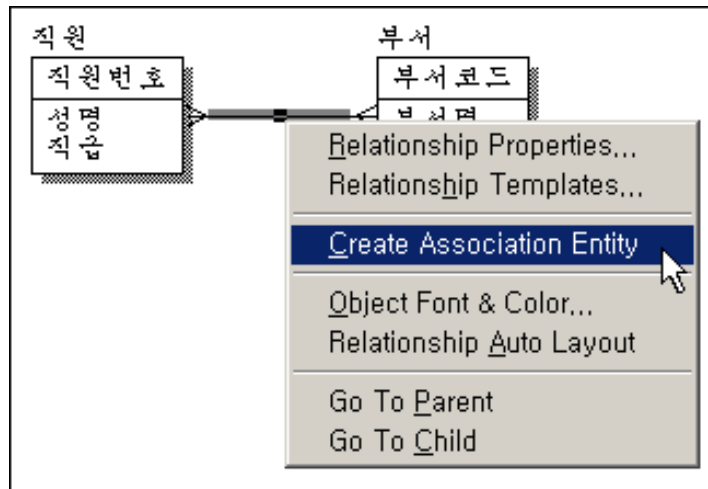


1. Toolbox에서 Entity를 선택하고 직원과 부서 Entity를 만든다.



2. Toolbox에서 Many-to-many relationship을 선택하고 두 Entity를 선택한다. (순서 상관 없음)

3. 생성된 관계선에 오른쪽 마우스 버튼을 클릭하여 Create Association Entity를 선택한다.



Many-To-Many Relationship Transform Wizard

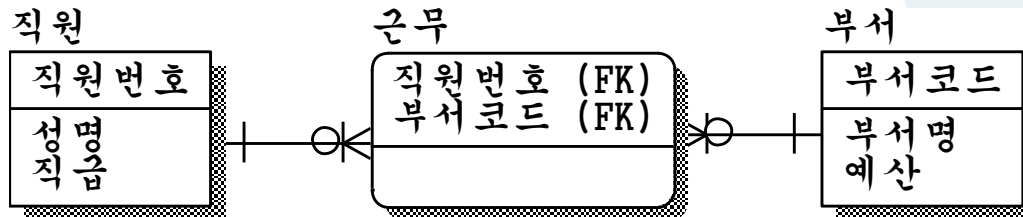
Specify the name of the association entity.

Entity Name
근무

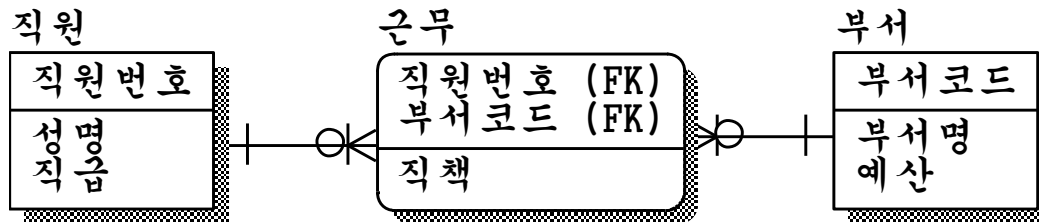
Entity Comment

< 뒤로(B) 다음(N) > 취소 도움말

4. Wizard에서 계속 다음을 클릭하면서 Entity Name을 '근무'로 입력한다.

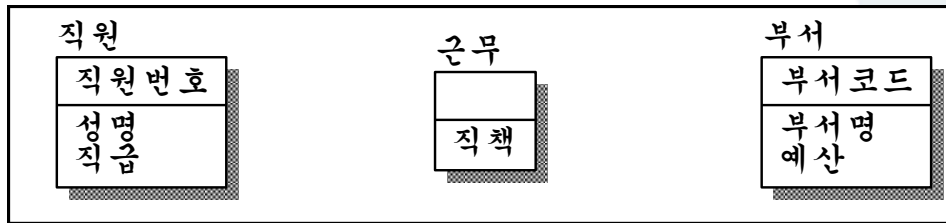


5. 마침 버튼을 누르면 다음과 같이 근무 Relationship이 생성된다.
Attribute를 입력한다.

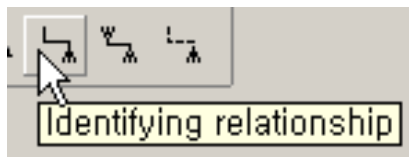


6. 완성된 ERD

■ 다른 방법

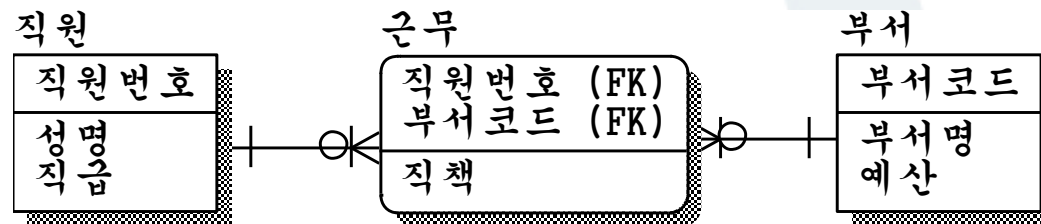


1. Toolbox에서 Entity를 선택하고 직원, 부서, 근무 Entity를 만든다.

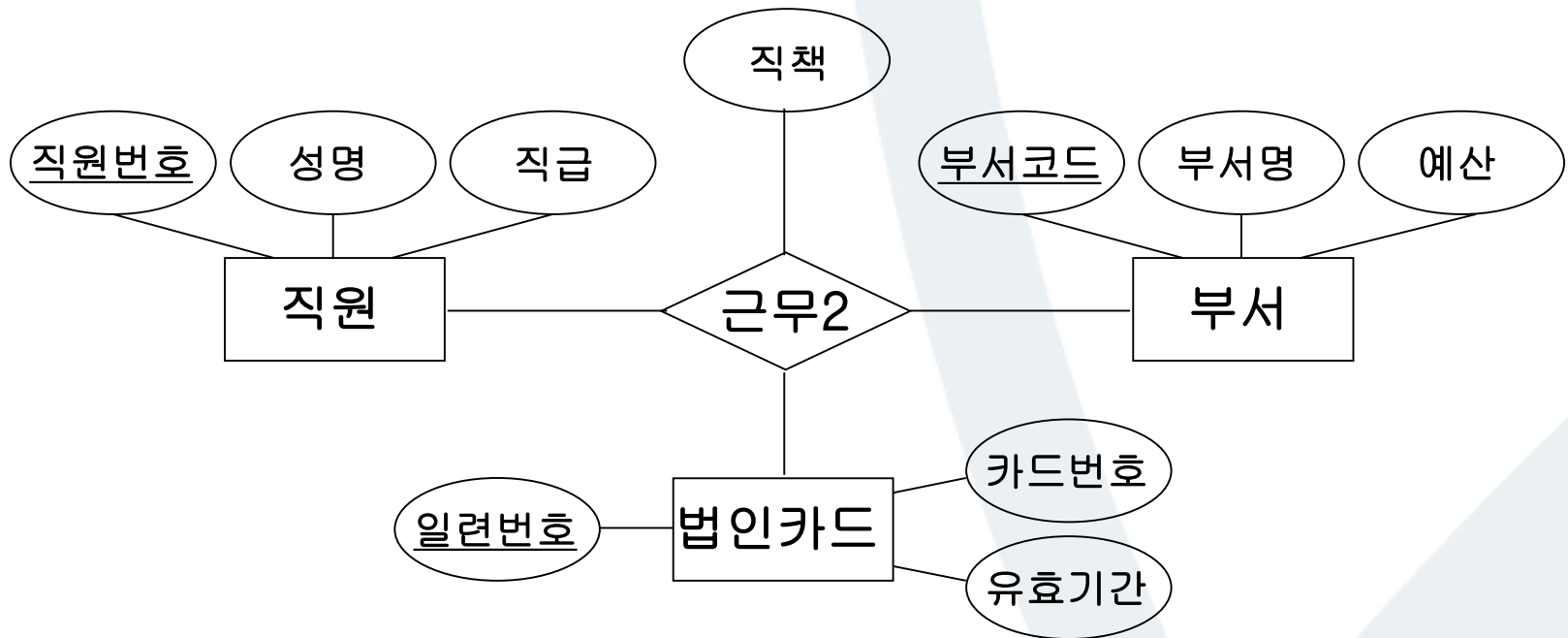


2. Toolbox에서 Identifying relationship을 선택하여 직원, 근무 순서로 클릭하고 다시 Toolbox에서 Identifying relationship을 선택하여 부서, 근무 순서로 클릭한다.

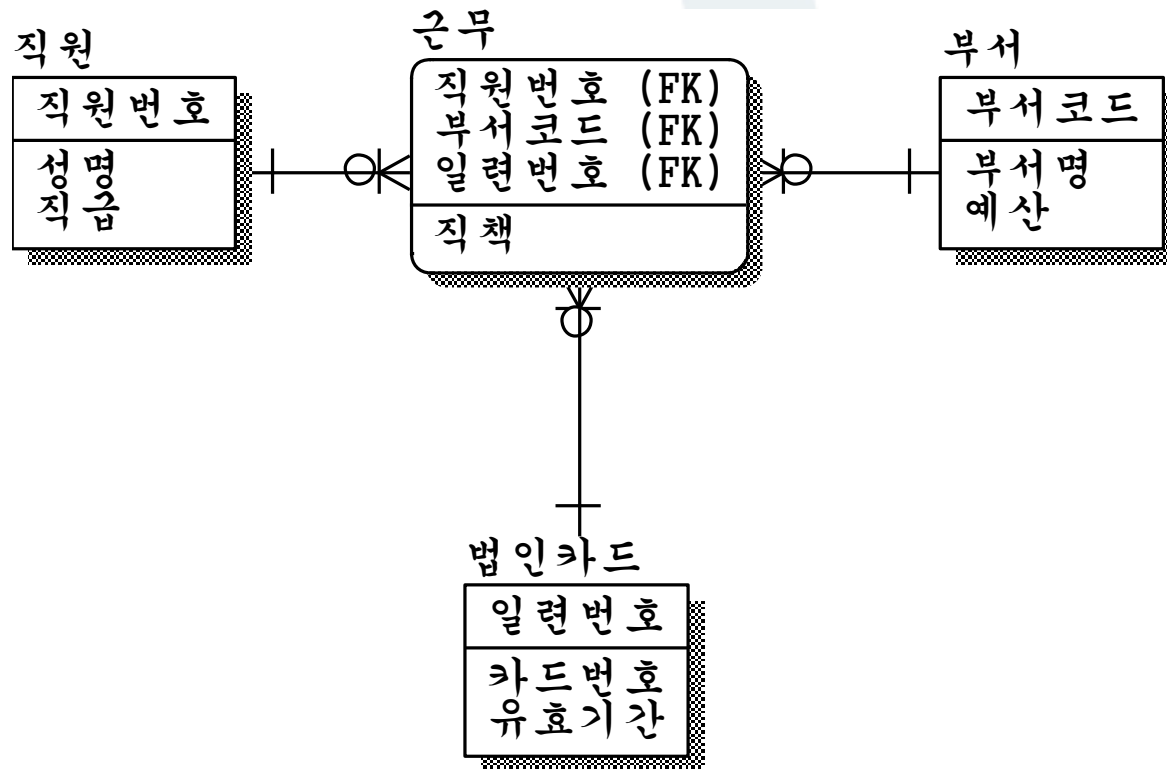
3. 완성된 ERD



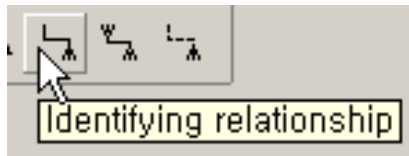
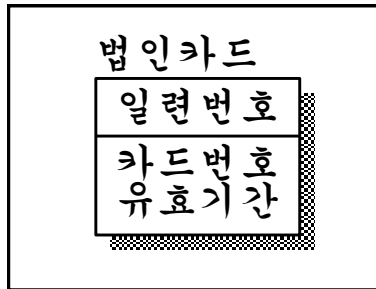
ERwin - 3진 관계 다시보기



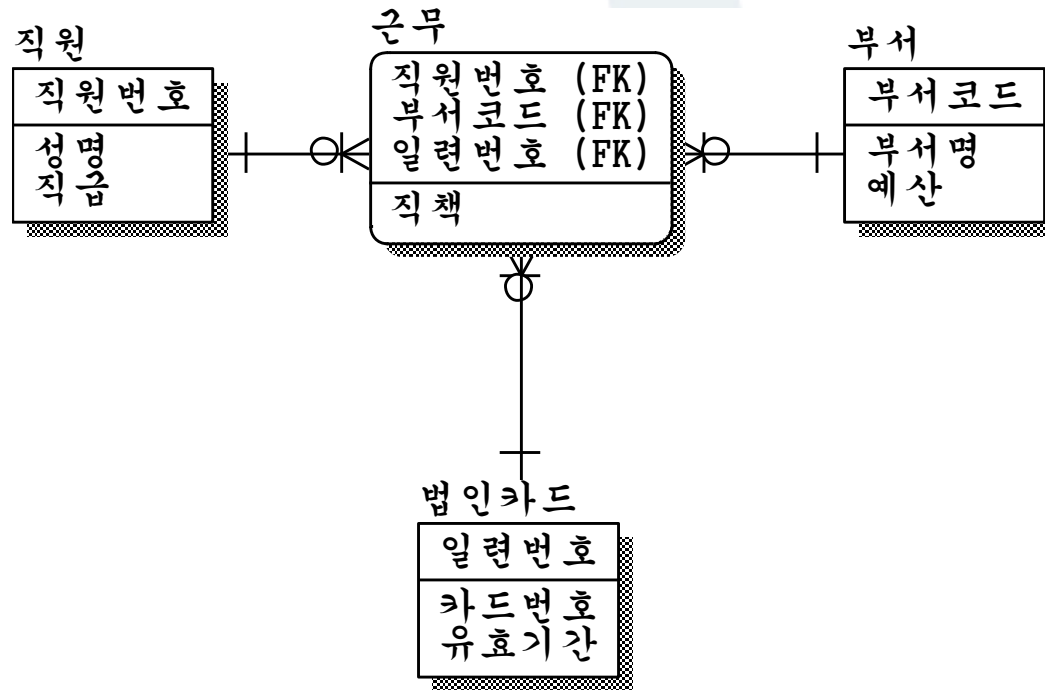
ERwin의 3진 관계 표기법



1. Toolbox에서 Entity를 선택하고 법인카드 Entity를 만든다.

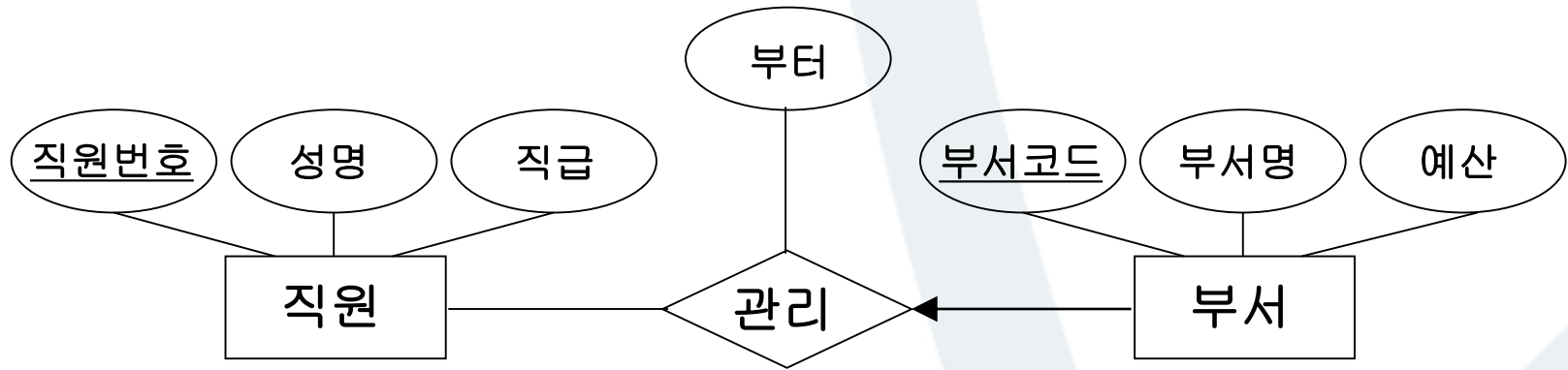


2. Toolbox에서 Identifying relationship을 선택하고 ‘법인카드’, ‘근무’ 순서로 클릭하여 부모, 자식 관계로 만든다.

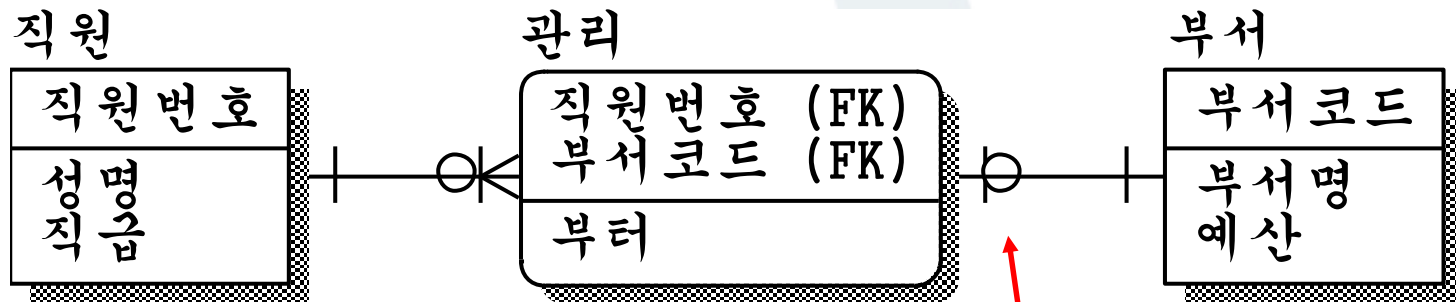


3. ‘법인카드’ Entity의 기본키가 ‘근무’ Relationship의 기본키에 외래키로 추가되면서 3진 관계가 된다.

ERwin - 키 제약조건 다시보기 (1대다)



ERwin의 키 제약조건 표기법 (1대다)



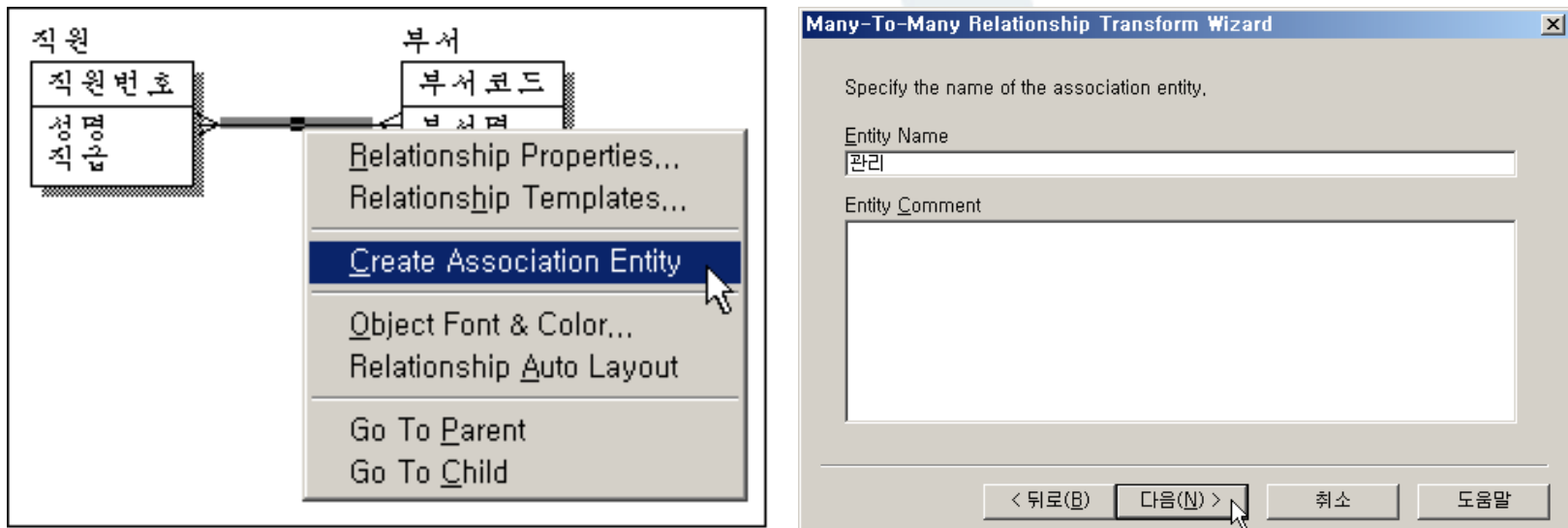
키 제약조건 명시
(‘More’가 빠진다)

1. Toolbox에서 Entity를 선택하고 직원과 부서 Entity를 만든다.

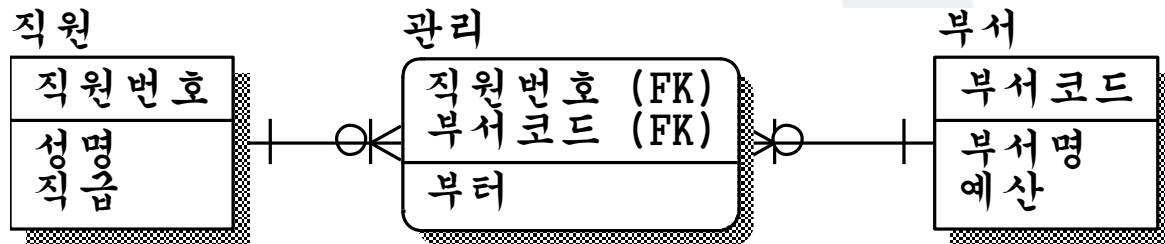


2. Toolbox에서 Many-to-many relationship을 선택하고 두 Entity를 선택한다. (순서 상관 없음)

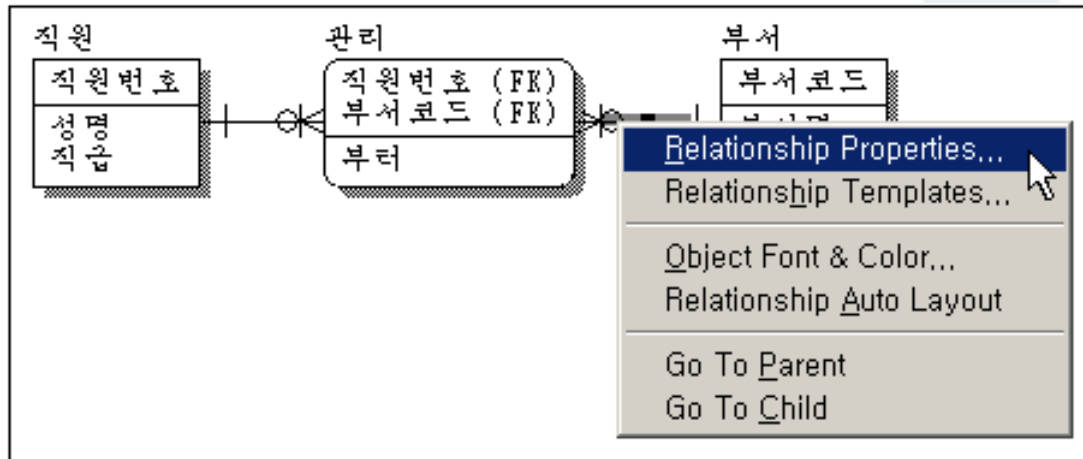
3. 생성된 관계선에 오른쪽 마우스 버튼을 클릭하여 Create Association Entity를 선택한다.



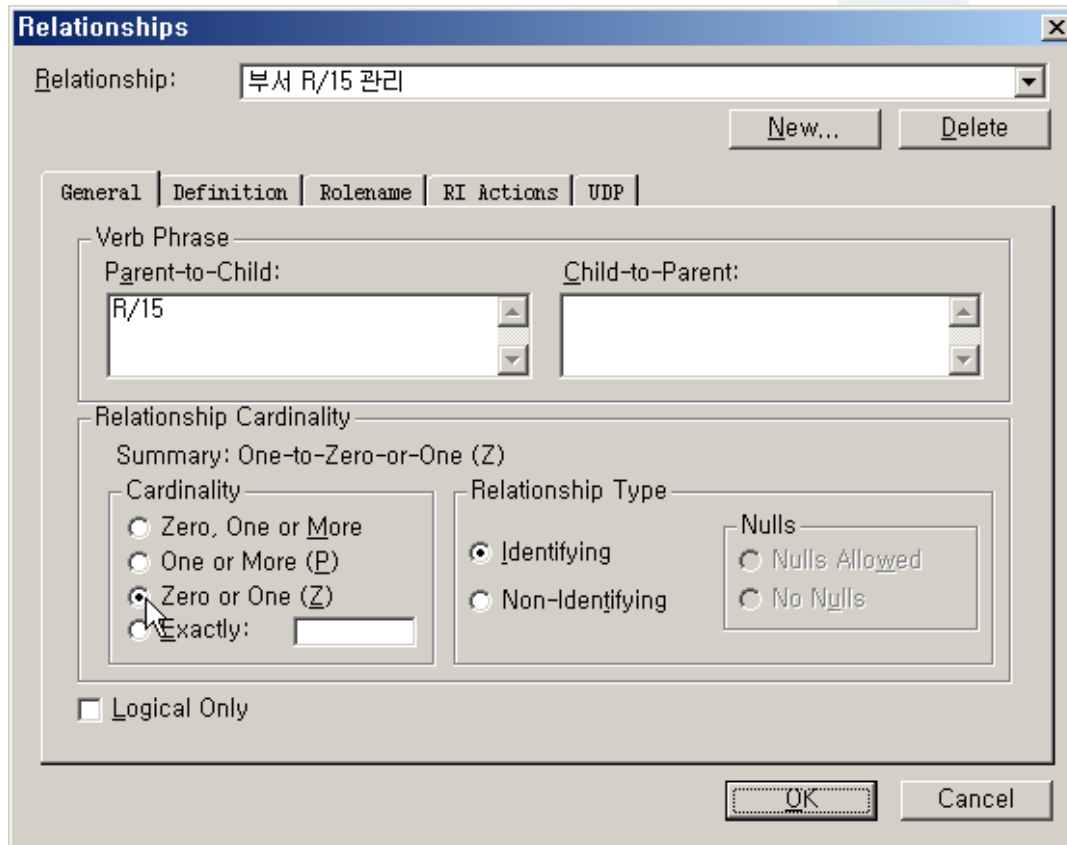
4. Wizard에서 계속 다음을 클릭하면서 Entity Name을 '관리'로 입력한다.



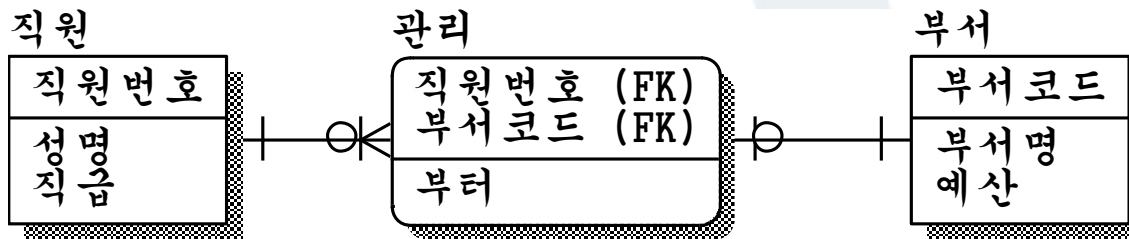
5. ‘관리’ Relationship이 생성되면 Attribute를 추가한다.



6. 키 제약조건을 명시할 관계선의 팝업 메뉴로 Relationship Properties를 선택한다.



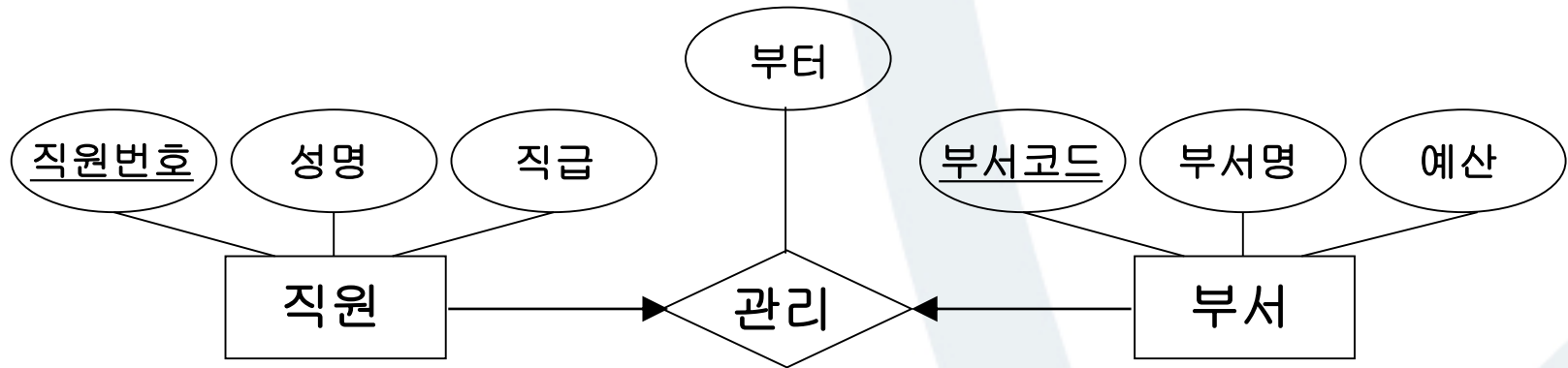
7. Cardinality에서 'Zero or One'을 선택한다. 이는 부서를 관리하는 사람은 없거나 한명이라고 규정하는 것으로 부서장은 많아야 한명이라는 키 제약조건을 명시한다.



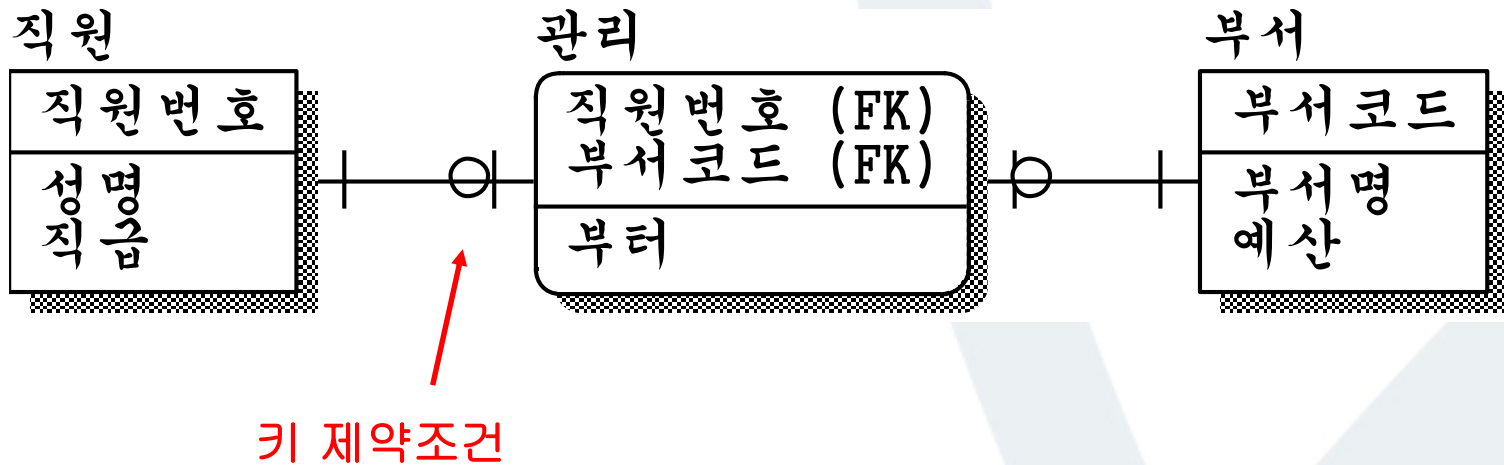
8. 완성된 ERD

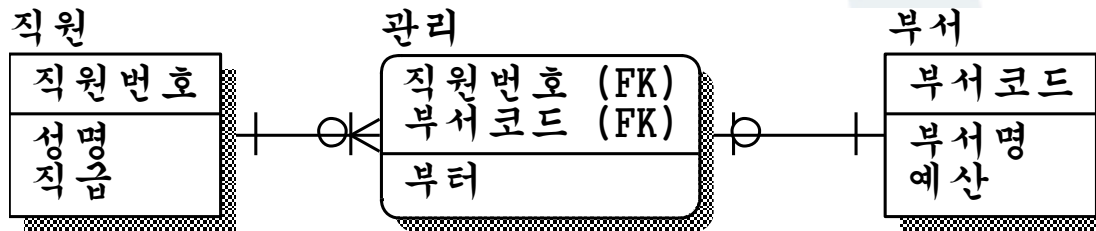
관리 Relation과 부서 Entity 사이의 관계에 변화가 생긴 것을 볼 수 있다.

ERwin - 키 제약조건 다시보기 (1대1)

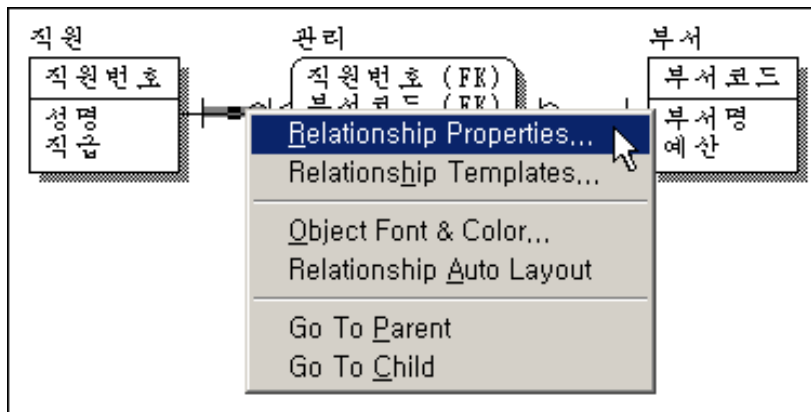


ERwin의 키 제약조건 표기법 (1대1)





1. 키 제약조건을 명시할 직원 Entity와 관리 Relationship의 관계선 팝업 메뉴로 Relationship Properties를 선택한다.



Relationship Cardinality

Summary: One-to-Zero-or-One (Z)

Cardinality

☐ Zero, One or More
☐ One or More (P)
☒ Zero or One (Z)
☐ Exactly:

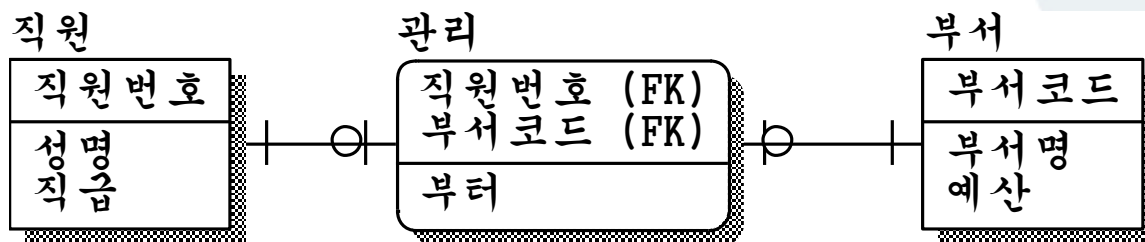
Relationship Type

☒ Identifying
☐ Non-Identifying

Nulls

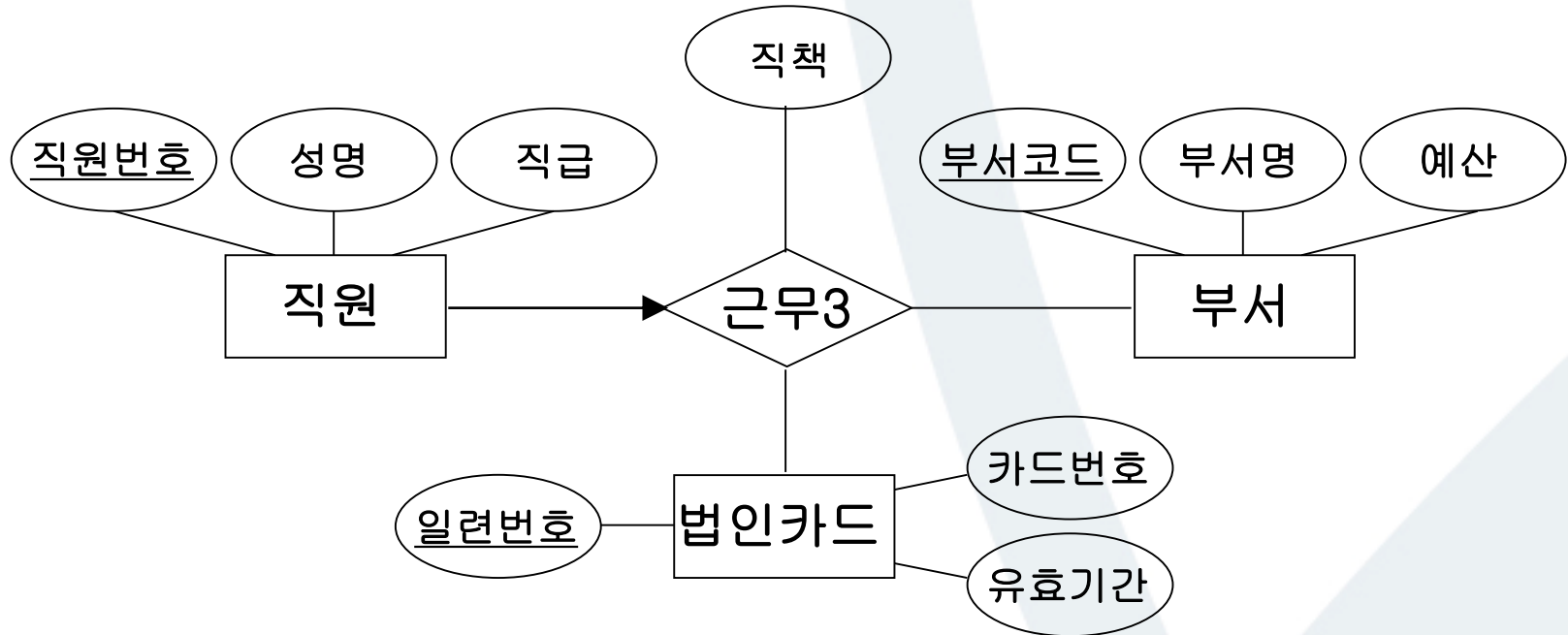
☐ Nulls Allowed
☐ No Nulls

2. Cardinality에서 'Zero or One'을 선택한다. 이는 한 직원은 부서를 관리하지 않거나 하나의 부서만을 관리할 수 있다는 것을 규정하는 것으로 한 직원은 많아야 한 부서만 관리할 수 있다는 키 제약조건을 명시한다.



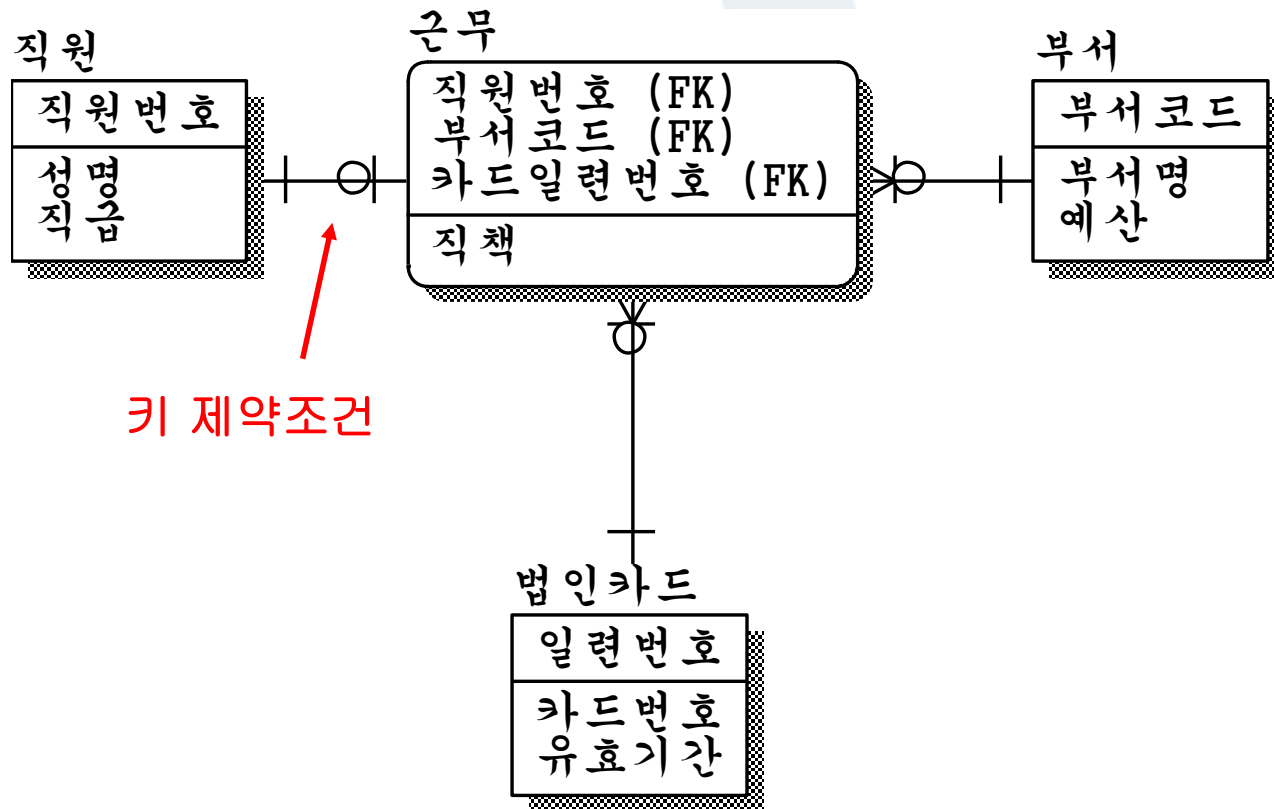
3. 완성된 ERD

ERwin - 키 제약조건 다시보기 (3진)

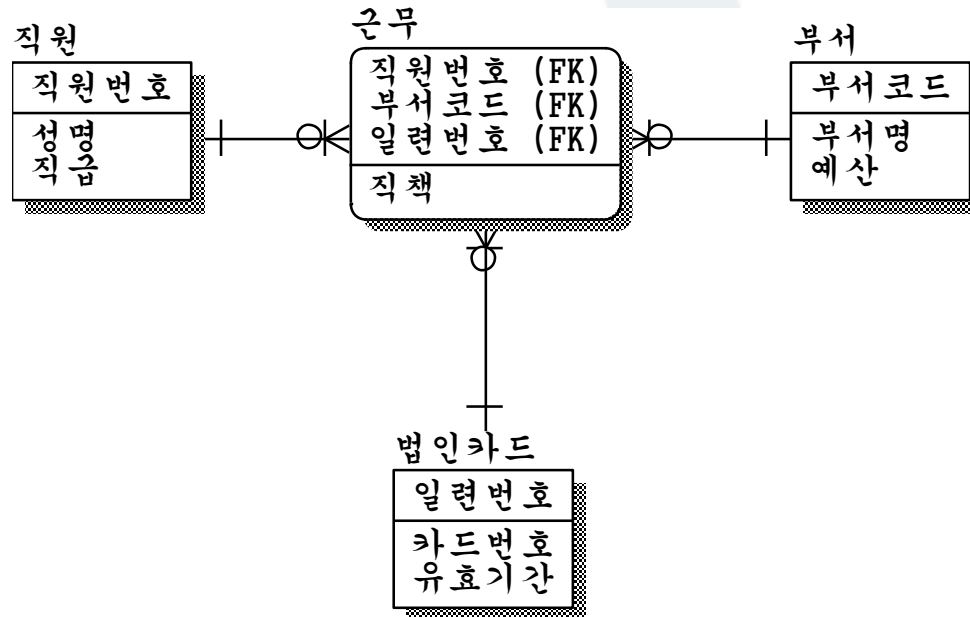


< 키 제약조건이 있는 3진 관계 집합 >

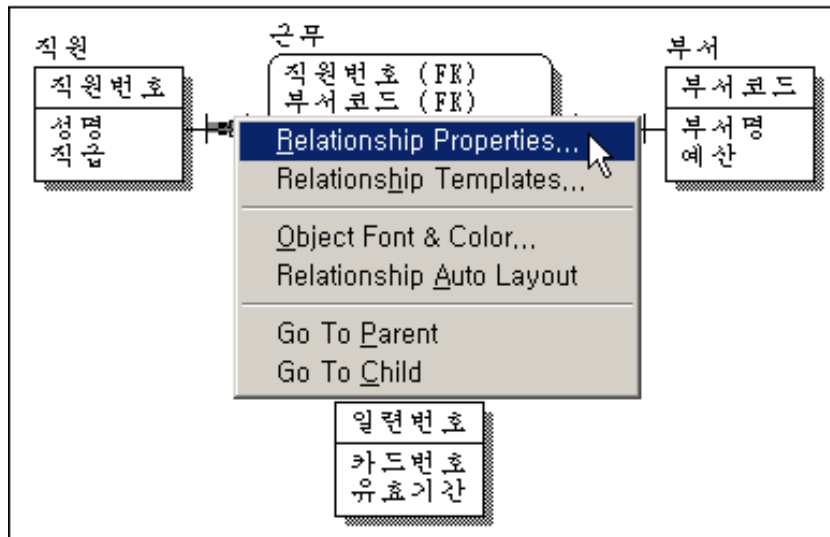
ERwin의 키 제약조건 표기법 (3진)



■ 앞에서 다루었던 3진 관계에 키 제약조건을 추가한다.



1. 키 제약조건을 명시할 직원 Entity와 근무 Relationship의 관계선 팝업 메뉴로 Relationship Properties를 선택한다.



Relationship Cardinality

Summary: One-to-Zero-or-One (Z)

Cardinality

☐ Zero, One or More

☐ One or More (P)

☒ Zero or One (Z)

☐ Exactly:

Relationship Type

☒ Identifying

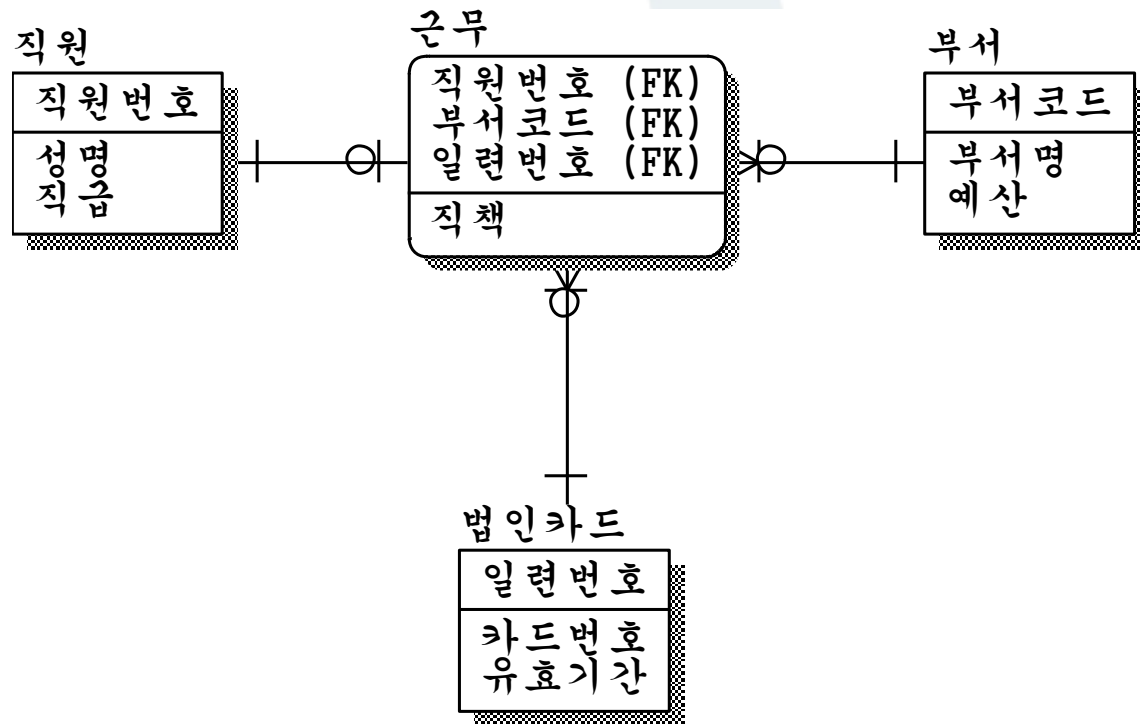
☐ Non-Identifying

Nulls

☐ Nulls Allowed

☐ No Nulls

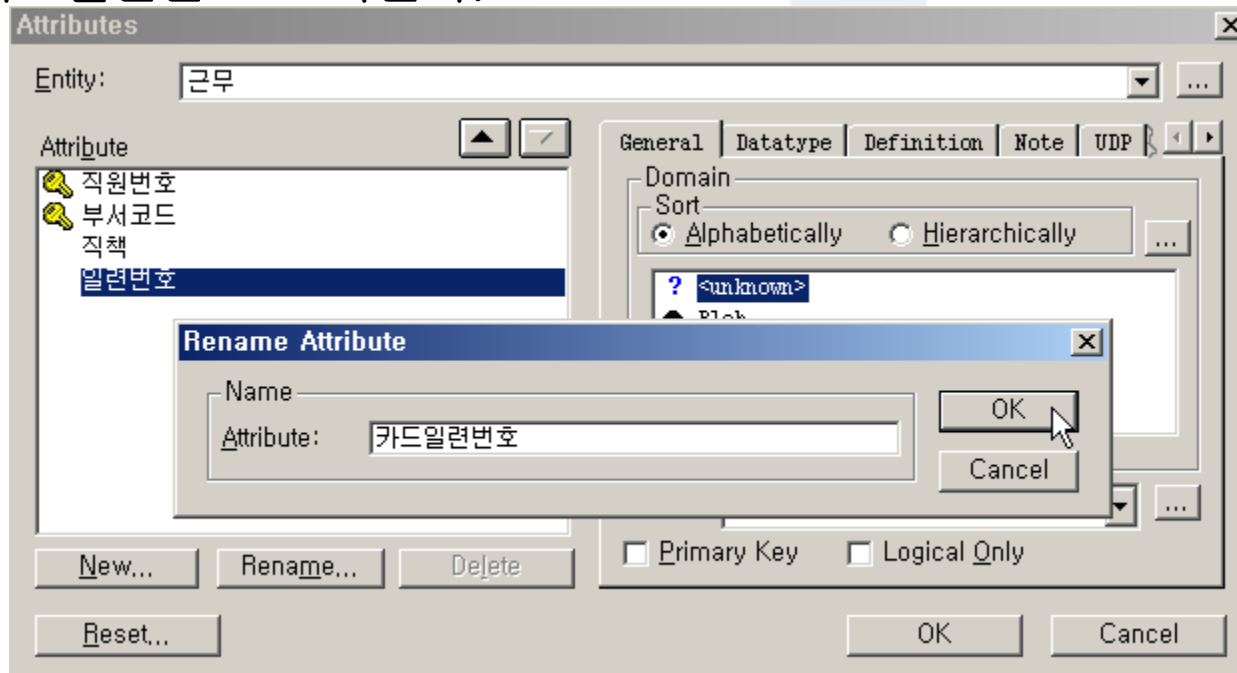
2. Cardinality에서 'Zero or One'을 선택한다. 이는 한 직원은 많아야 한 부서에서만 근무할 수 있고 많아야 하나의 법인카드만 소유할 수 있다는 키 제약조건을 명시한다.

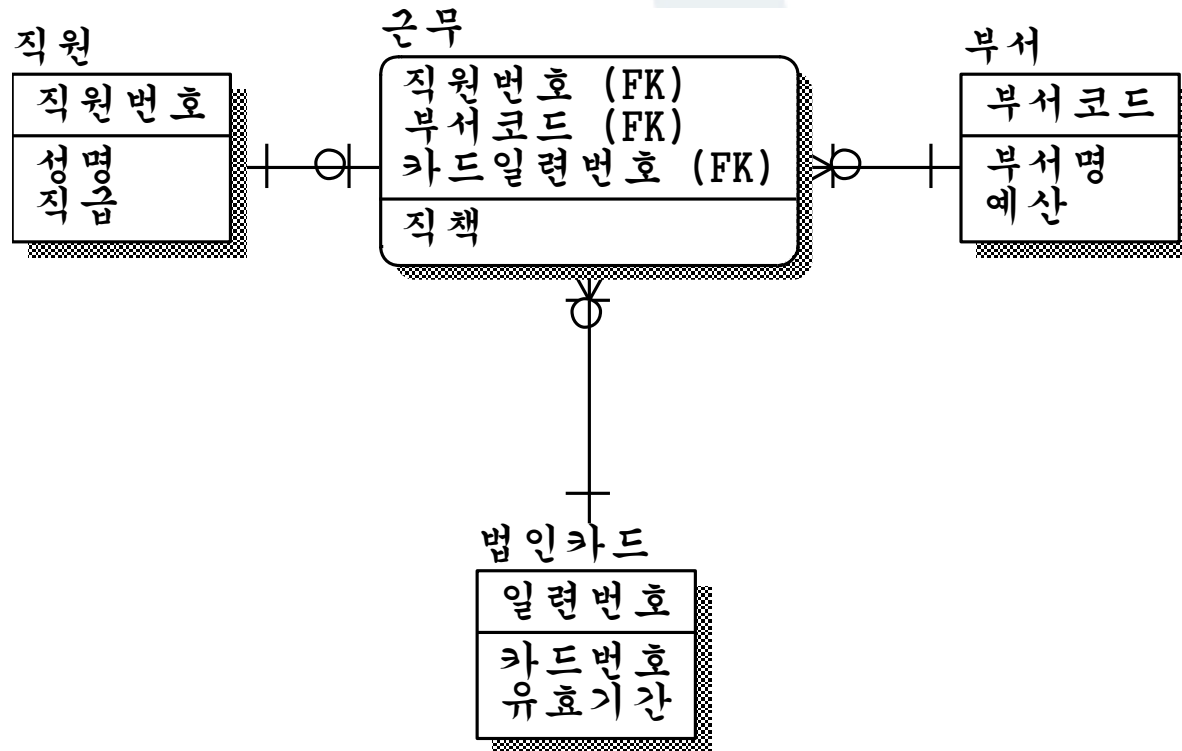


3. 완성된 ERD

Tip. 현재 근무 Relation의 Attribute로 있는 법인카드의 외래키 ‘일련번호’는 근무 Relation으로 들어오면서 Readability가 약해졌다. 이를 보완하기 위해 근무 Relation에 보이는 ‘일련번호’를 다른 이름으로 지정할 수 있다.

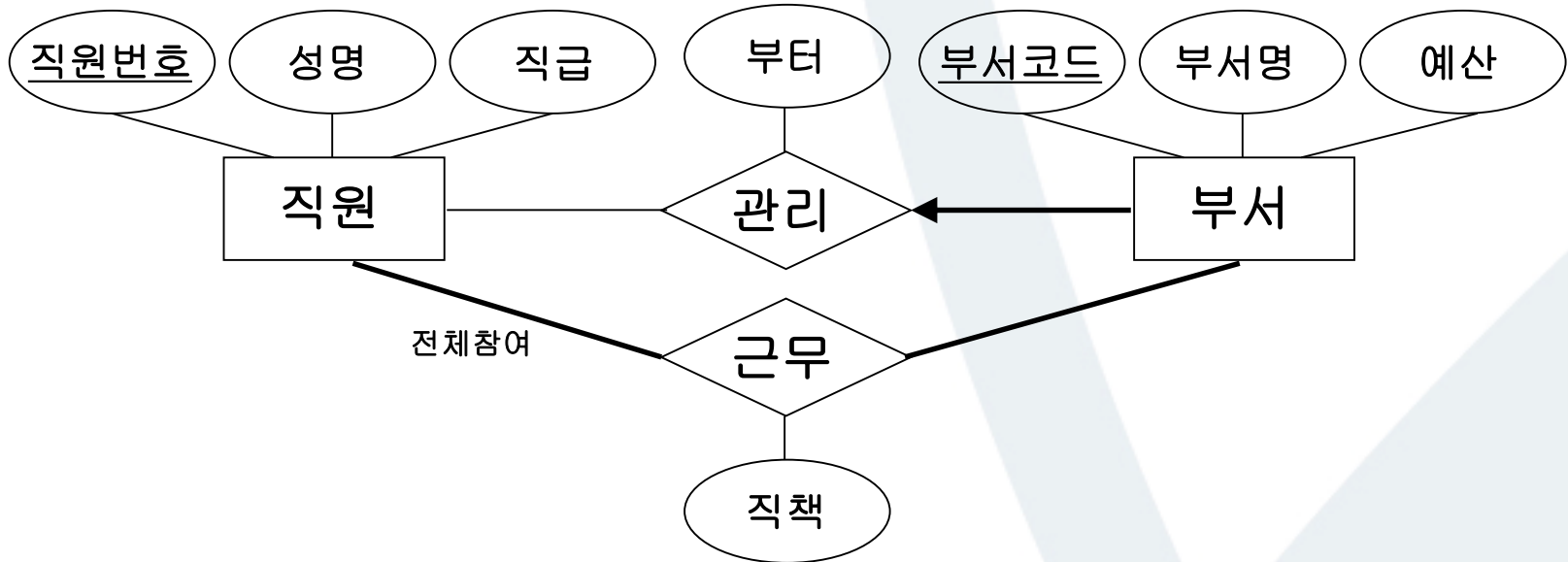
1. 근무 Relation의 팝업 메뉴로 Attributes를 선택하여 ‘일련번호’를 Rename하여 ‘카드일련번호’로 바꾼다.



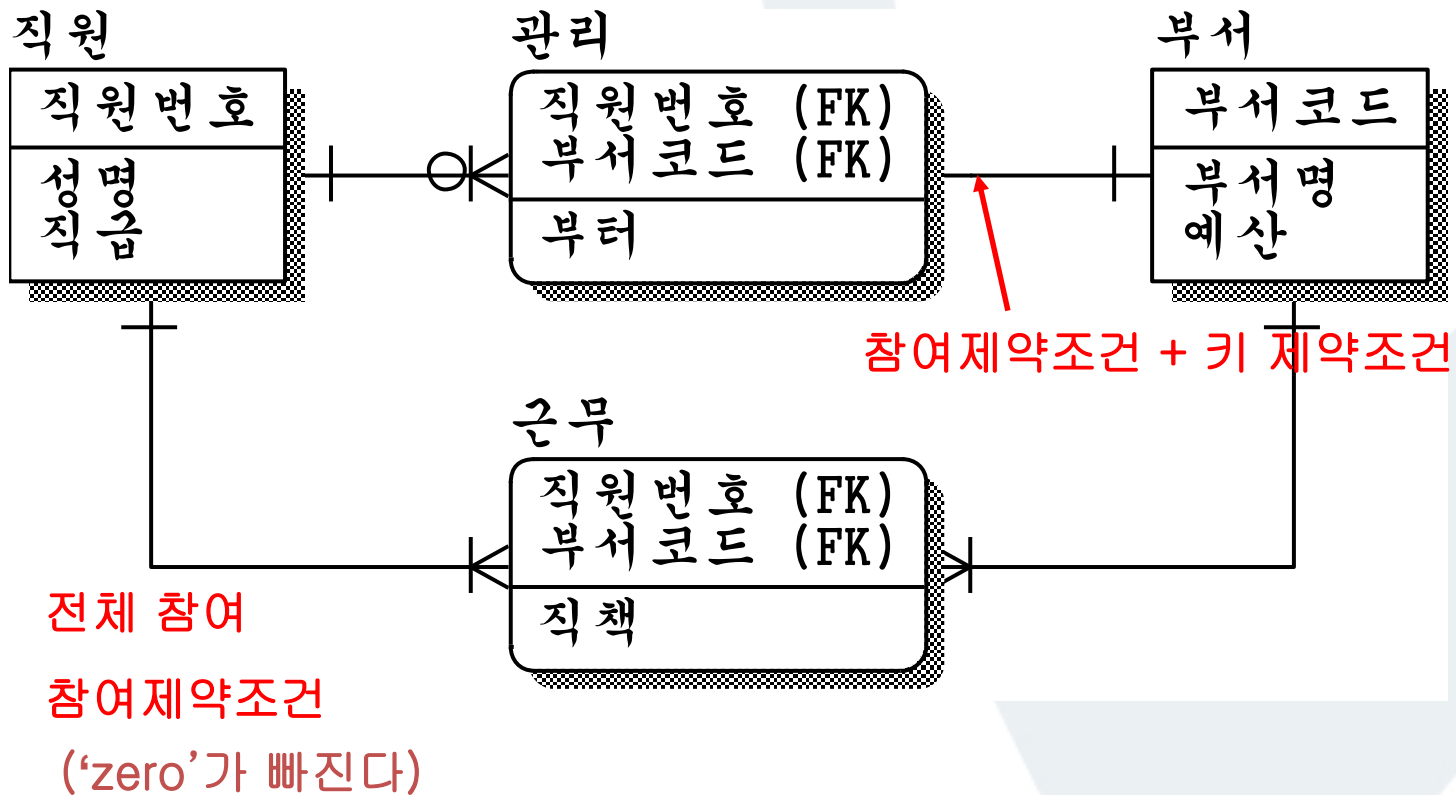


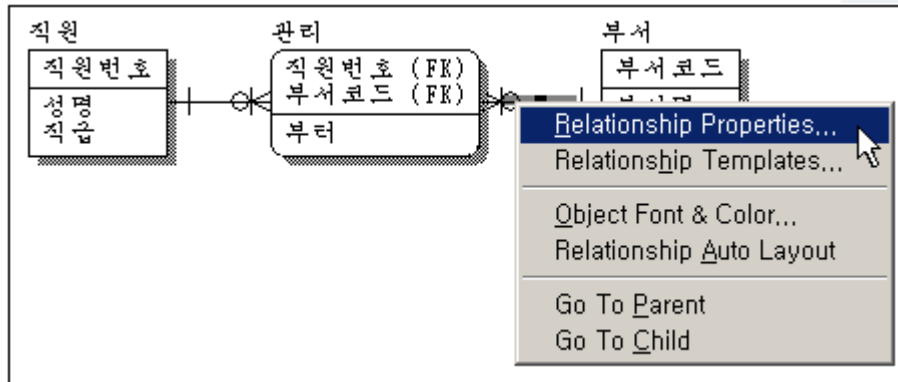
2. 개선된 ERD

ERwin - 참여 제약조건 다시보기



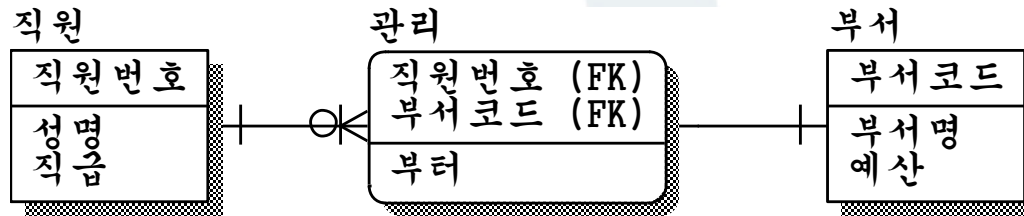
ERwin의 참여 제약조건 표기법



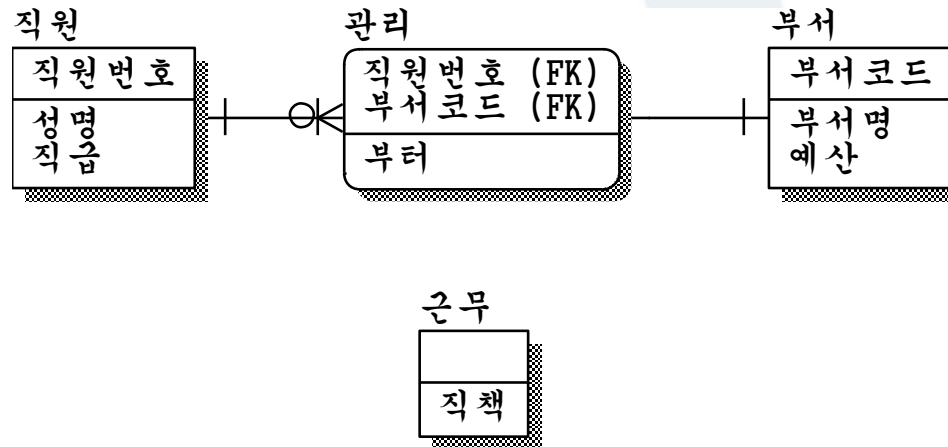


1. 관리 관계에서 참여제약조건과 키 제약조건을 명시할 관리 Relationship과 부서 Entity의 관계선 팝업 메뉴로 Relationship Properties를 선택한다.

2. Cardinality에서 'Exactly'를 선택하고 1을 입력한다. 이는 한 부서에는 반드시 한 명의 관리자가 있어야 한다는 것을 규정하는 것으로 한 부서에는 한 명의 부서장만 존재할 수 있다는 참여제약조건과 키 제약조건을 명시한다.

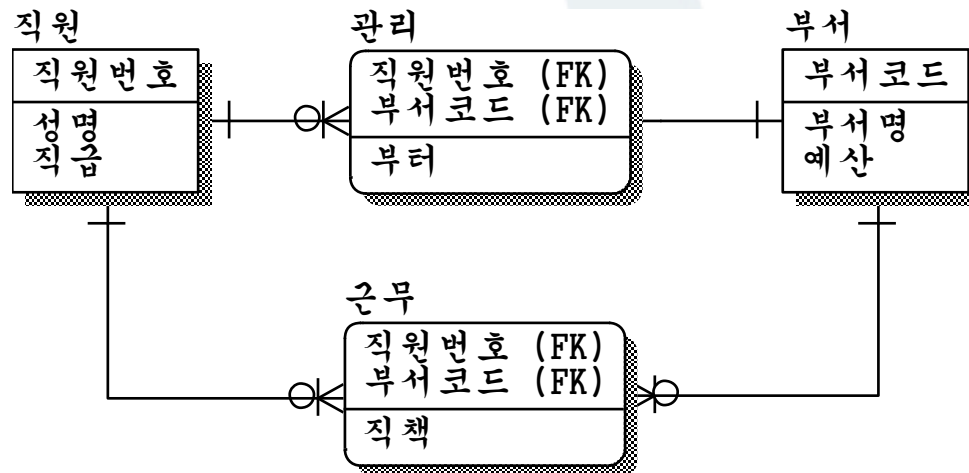


3. 참여제약조건과 키 제약조건이 명시된 ERD



4. 근무 관계를 추가하기 위해 Entity를 생성하여 다음과 같이 입력한다.
(Toolbox에서 Many-to-many relationship을 선택하여 팝업 메뉴로 Create Association Entity를 생성하여도 상관 없다.)

4. Toolbox의 Identifying relationship을 선택하여 직원, 근무의 순서로 클릭 하고 다시 Identifying relationship을 선택하여 부서, 근무의 순서로 클릭한다.



4. 참여제약조건을 명시할 직원 Entity와 근무 Relationship의 관계선을 더블 클릭하여 Cardinality를 'One or More'로 선택한다. 이는 한 직원은 한 부서에는 꼭 근무하여야 하고 여러 부서에도 근무할 수 있다는 것을 규정하며 한 직원은 적어도 한 부서에는 근무하여야 한다는 참여제약조건을 명시한다.

Relationship Cardinality

Summary: One-to-One-or-More (P)

Cardinality

☐ Zero, One or More

☒ One or More (P)

☐ Zero or One (Z)

☐ Exactly:

Relationship Type

☒ Identifying

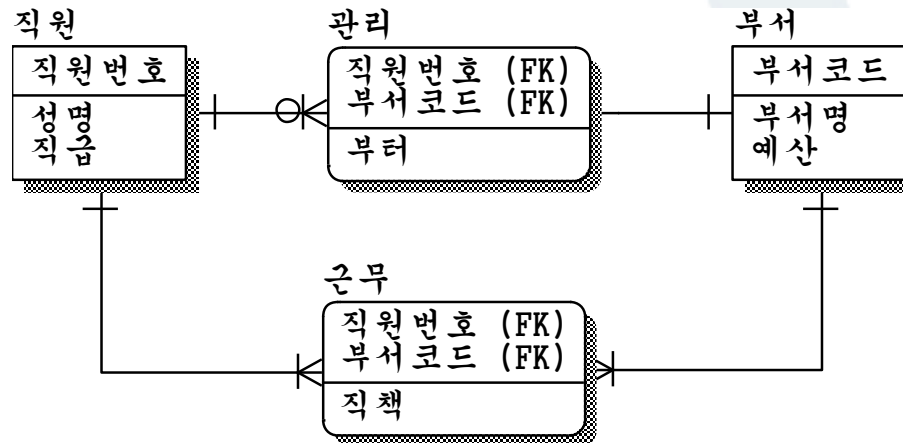
☐ Non-Identifying

Nulls

☐ Nulls Allowed

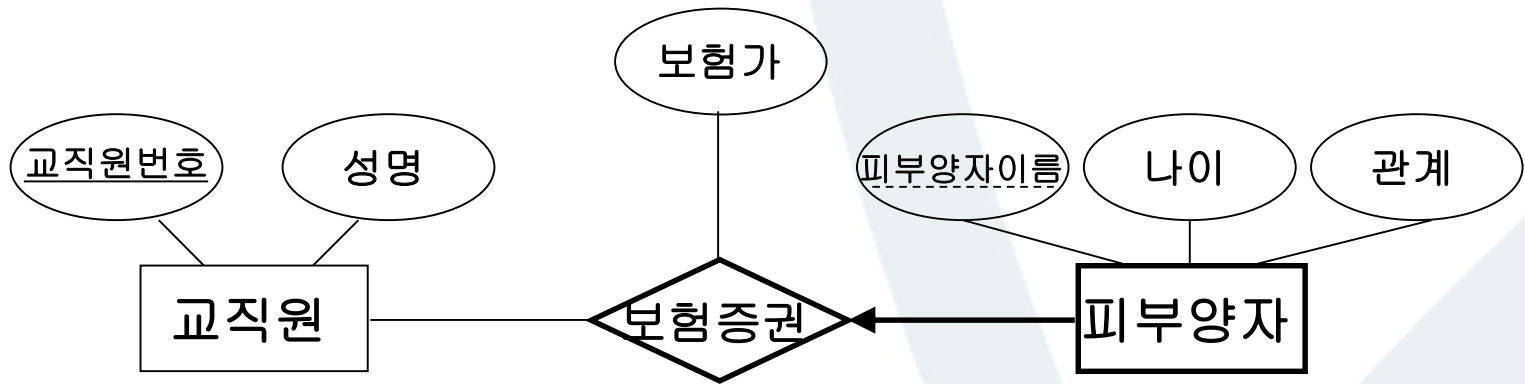
☐ No Nulls

4. 참여제약조건을 명시할 직원 Entity와 근무 Relationship의 관계선을 더블 클릭하여 Cardinality를 'One or More'로 선택한다. 이는 한 부서에는 적어도 한 직원은 근무해야 하고 여러 직원도 근무할 수 있다는 것을 규정하며 한 부서에는 적어도 한 직원은 근무해야 한다는 참여제약조건을 명시한다.

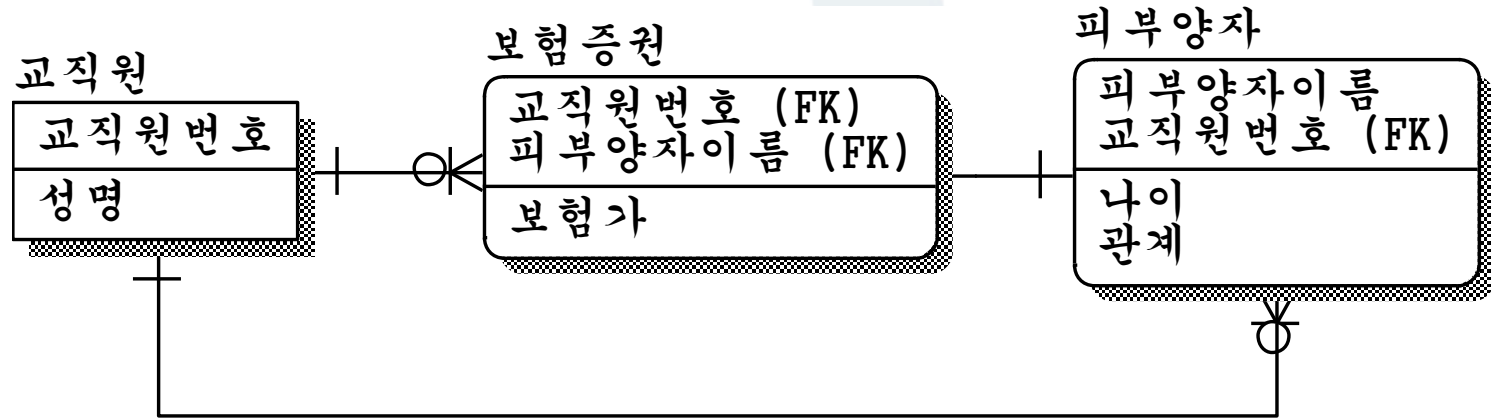


5. 완성된 ERD

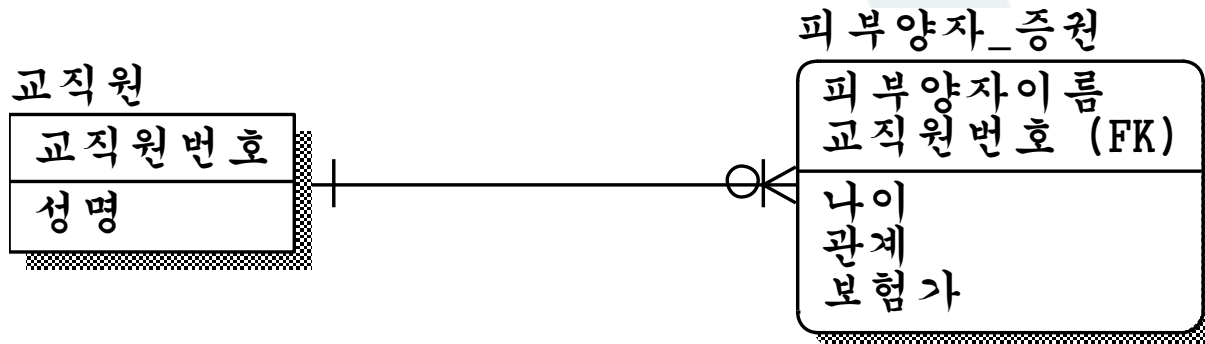
ERwin - 약 개체 다시보기



ERwin의 약 개체 표기법

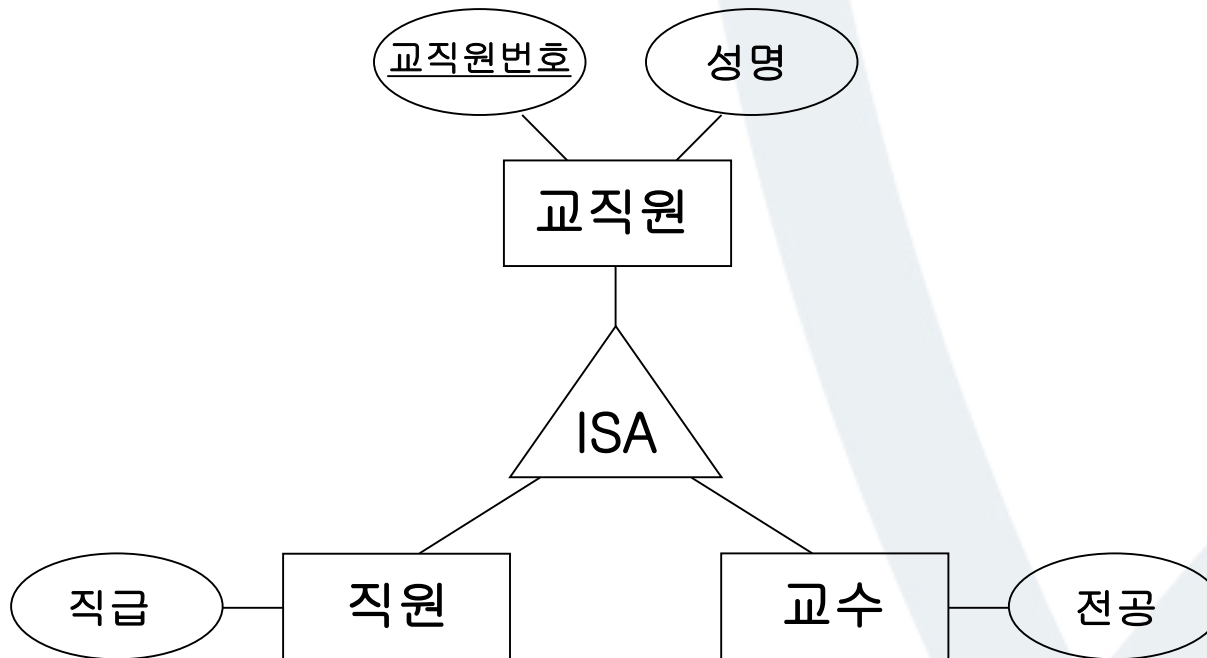


...부자연스럽다

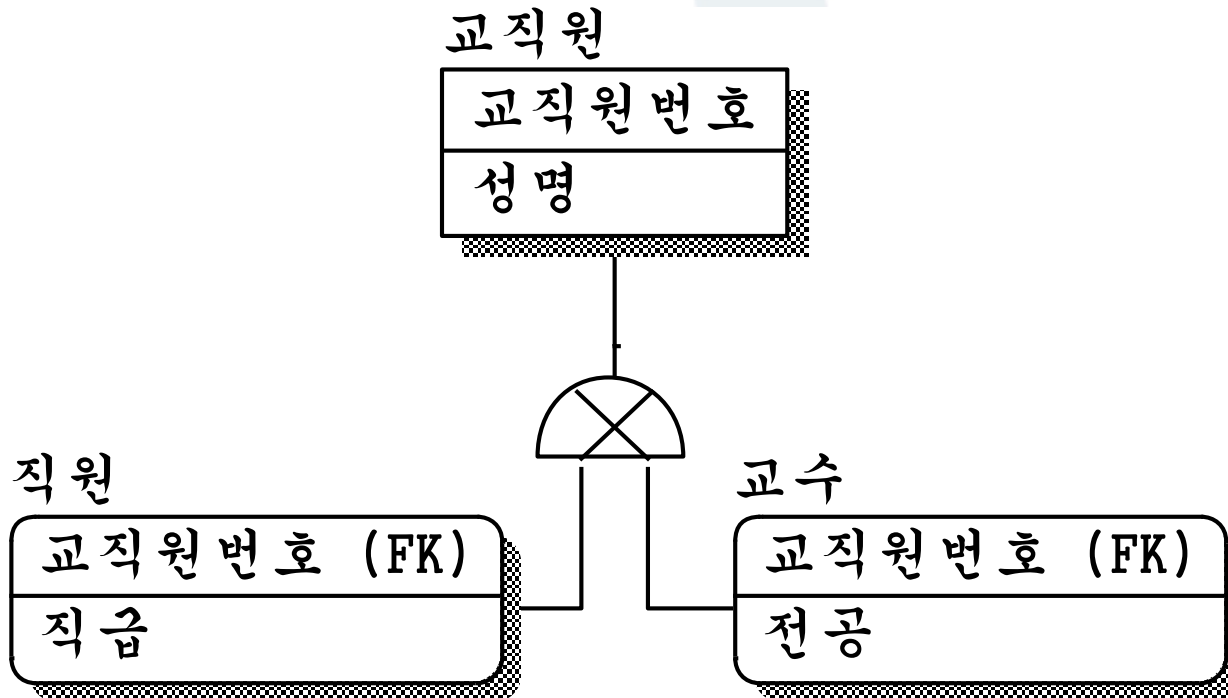


...좀 더 자연스럽다

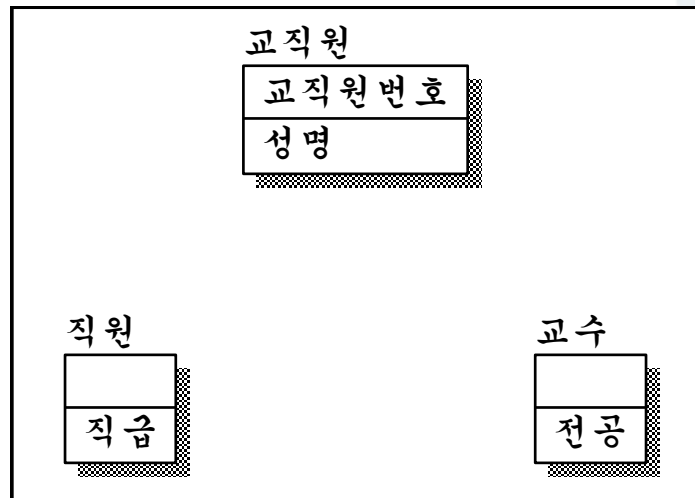
ERwin - 일반화 (계층구조) 다시보기



ERwin의 일반화 (계층구조) 표기법

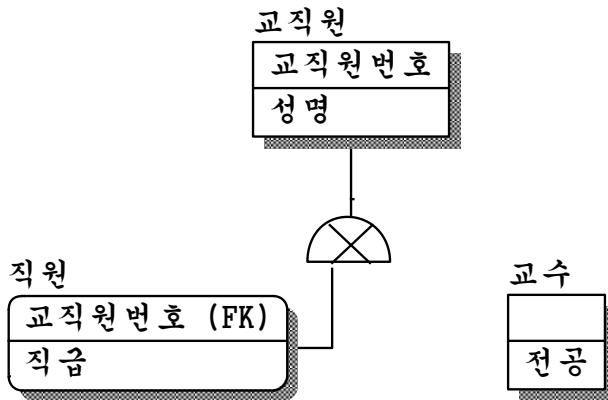


1. 일반화를 표현할 Entity를 생성한다.

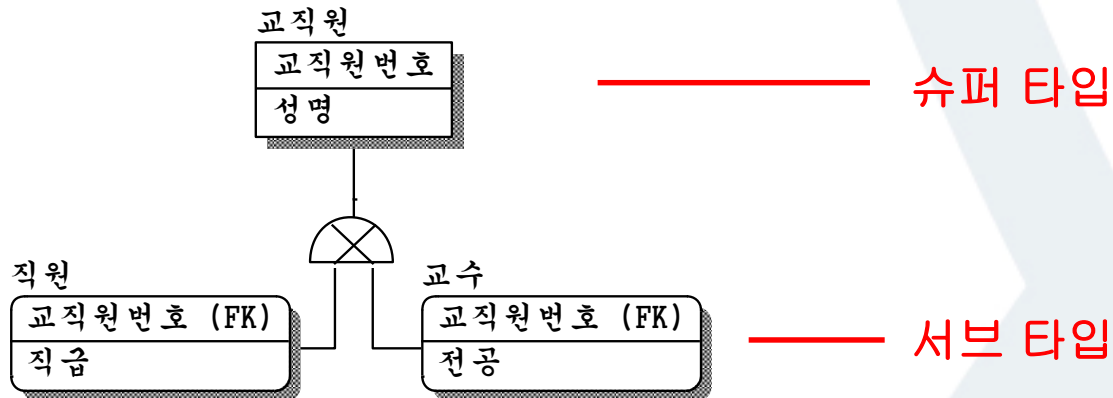


2. Toolbox에서 Exclusive sub-category를 선택한 뒤 교직원을 클릭하고 직원을 클릭한다.

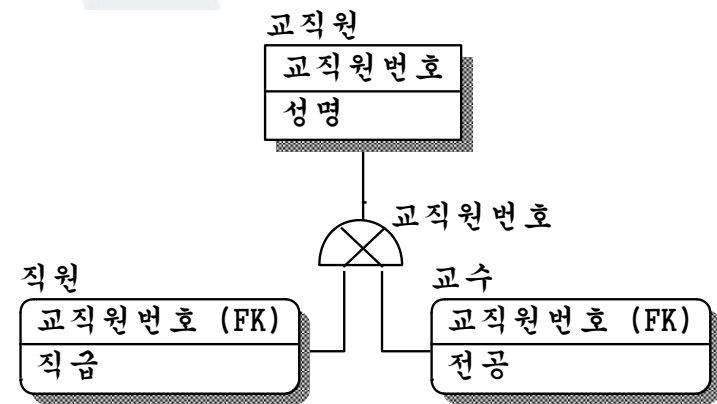
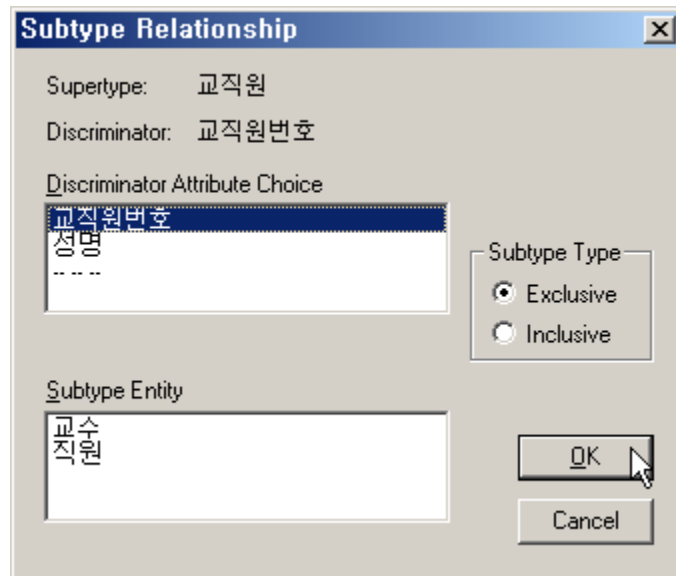




3. 마찬가지로 Toolbox에서 Exclusive sub-category를 선택한 뒤 Exclusive Icon을 클릭하고 교수를 클릭한다.



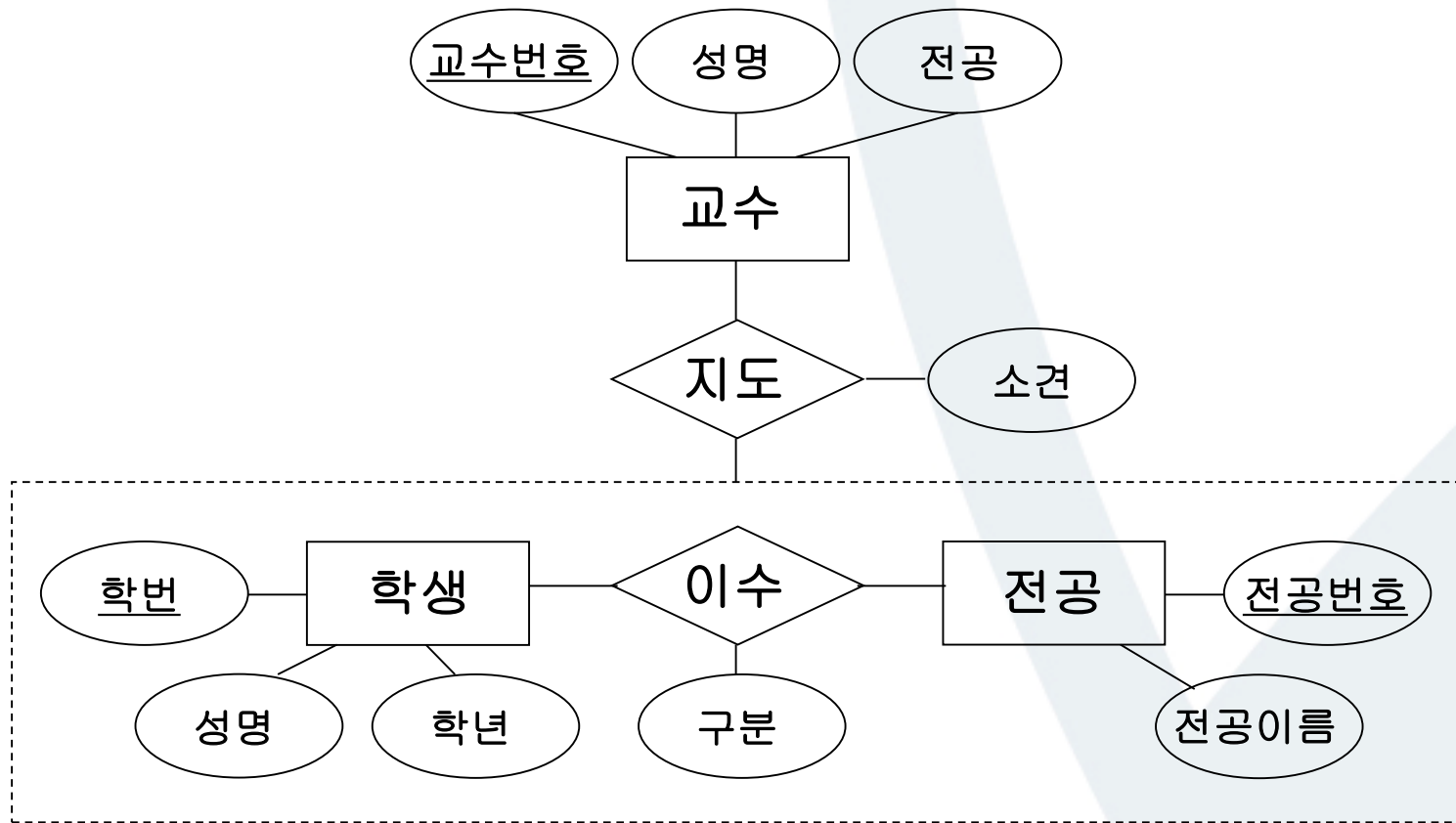
4. Exclusive 모양을 더블 클릭하여 Discriminator Attribute Choice를 ‘교직원 번호’로 선택한다. 이는 일반화의 기준이 되는 Attribute를 선택한다.
(생략 가능)



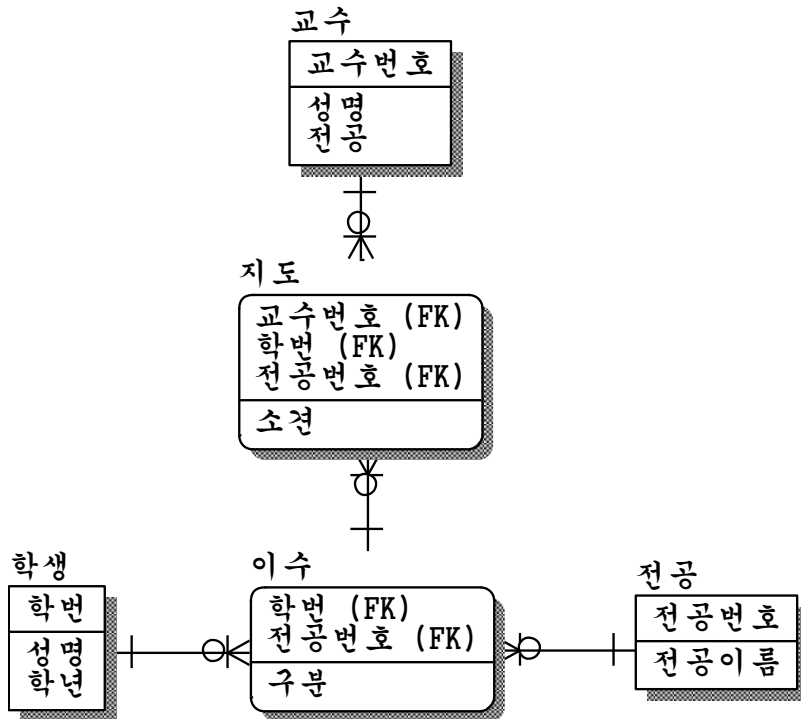
5. 완성된 ERD

- Exclusive : 중복 불허
- Inclusive : 중복 허용

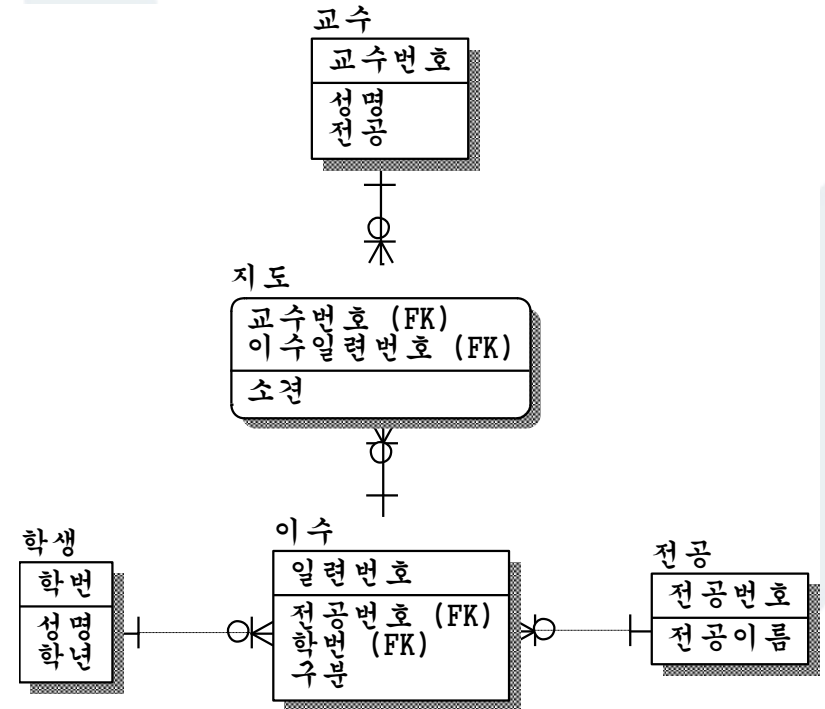
ERwin - 군집화 다시보기



ERwin의 군집화 표기법

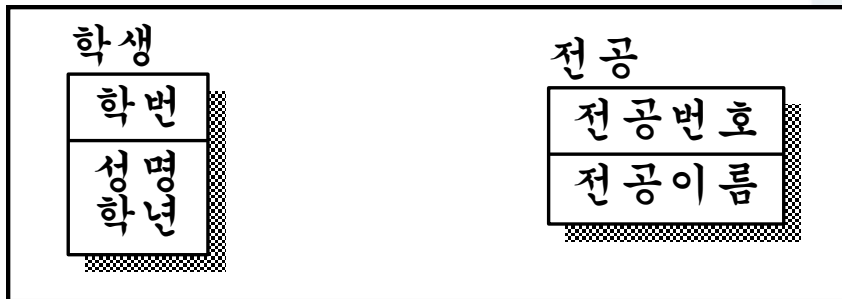


(O)

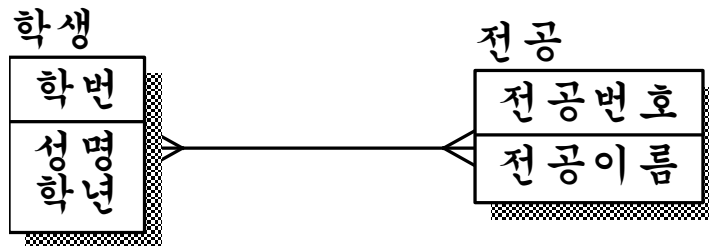


(X)

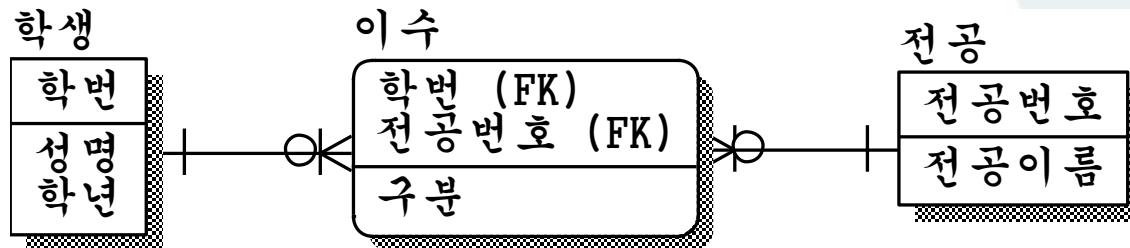
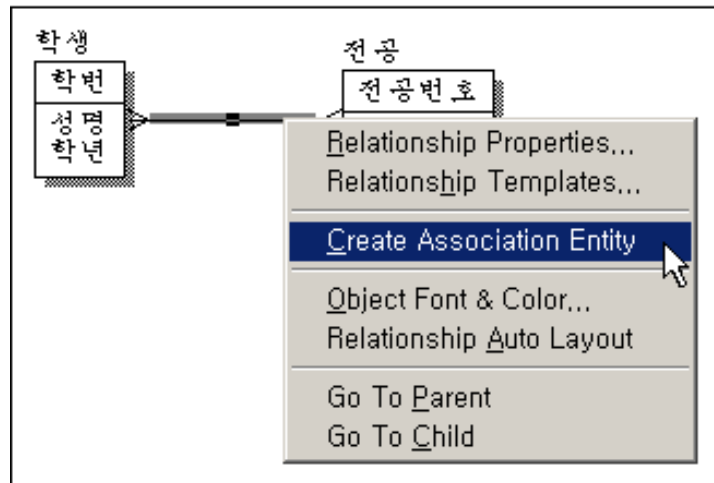
1. 학생과 전공 Entity를 생성한다.



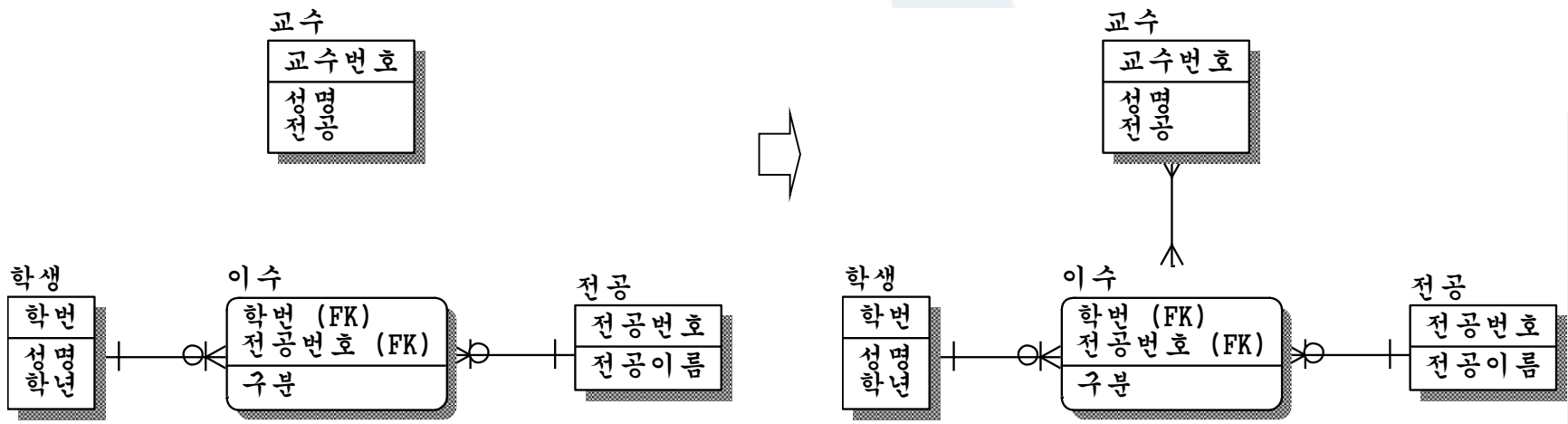
2. Toolbar의 Many-to-many relationship을 선택하여 두 Entity를 클릭한다.



3. 관계선의 팝업 메뉴로 Create Association Entity를 선택하여 생성한다.
Entity이름은 '이수'로 한다.

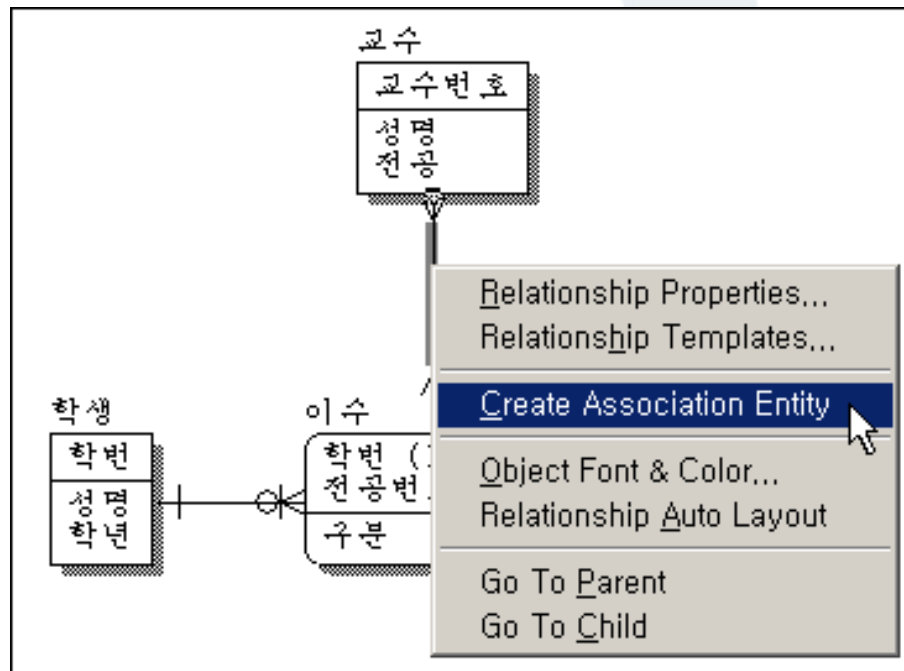


4. 교수 Entity를 생성한다.

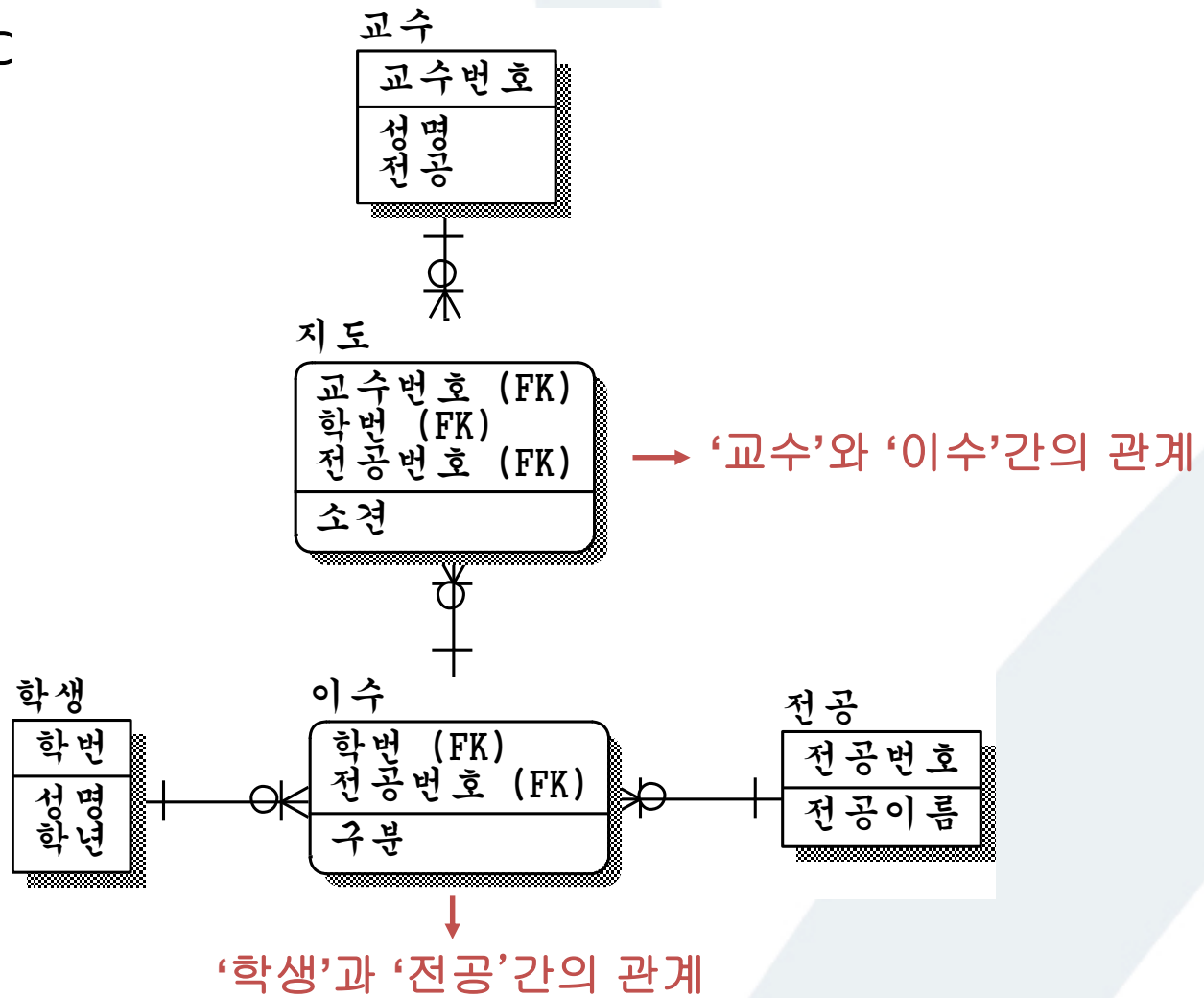


5. Toolbar의 Many-to-many Relationship을 선택하여 교수 Entity와 이수 Relationship을 클릭한다.

6. 관계선의 팝업 메뉴로 Create Association Entity를 선택하여 생성한다.
Entity이름은 '지도'로 한다.



7. 완성된 ERC



실습 :

- 지난 주의 실습 내용을 확장하여
 - 3진 관계를 모델링하라.
 - 키 제약조건 및 참여 제약조건을 모델링하라.
 - 약 개체, 식별관계를 모델링하라.
 - 일반화(계층구조)를 모델링하라.
 - 집단화를 모델링하라.
- P.29의 예를 다음처럼 일부 고쳐서 (리포트#2. 1주일 기한)
 - 직원을 임기직, 정년직으로 일반화
 - 부서를 태스크포스(T/F), 정규부서로 일반화
 - 정규부서에는 정년직만, T/F에는 정년직과 임기직이 모두 근무 가능
 - ERwin으로 그려라.