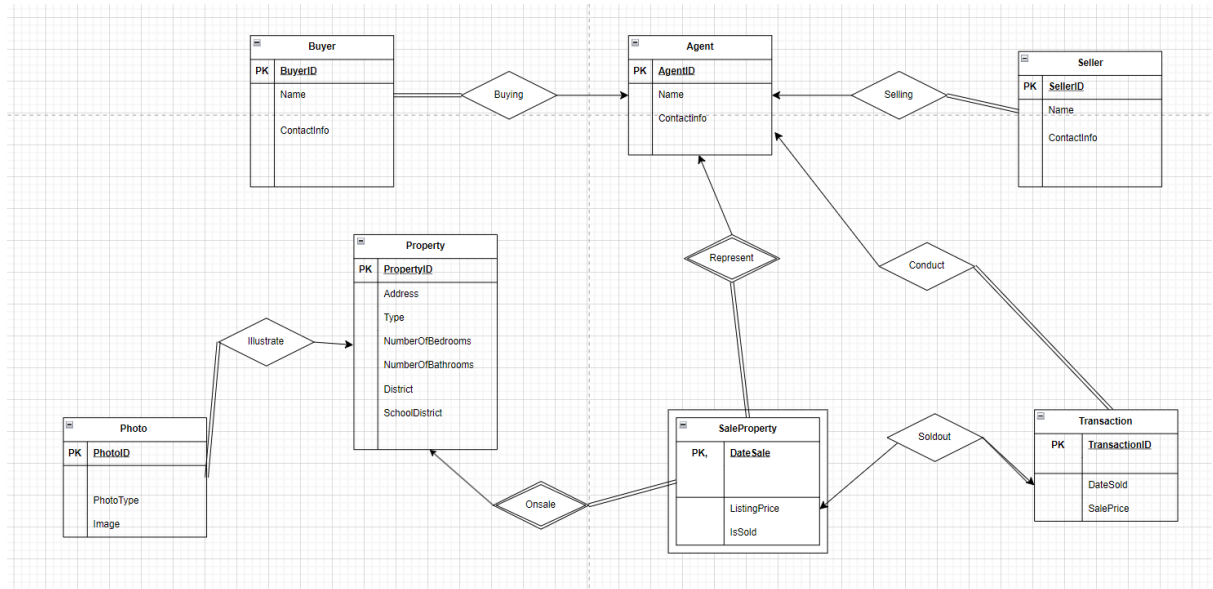


2024 Database Project1

학번 : 20191098

이름 : 백승주

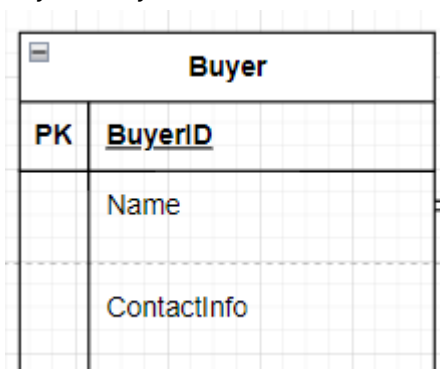
1. ER Model



Entity

프로젝트 개요 및 시행착오 : 처음에는 6개의 entity, 그러니까 property와 sale property

1) Buyer entity

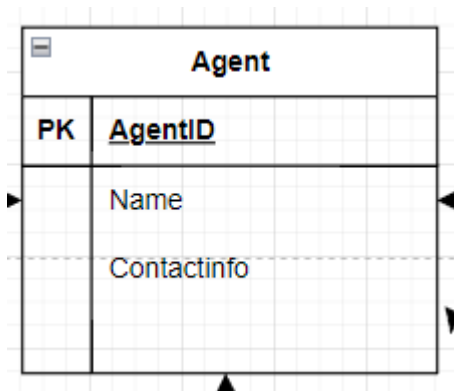


부동산의 구매자 정보가 저장된 entity다.

- **BuyerID**: 고유 식별자로서 각 구매자를 구별하는 데 사용된다. 이는 데이터베이스에서 주요 키로 작용한다. 각 Buyer의 Name이나 Buyer가 거래한 Agent의 AgentID만으로 Buyer를 식별할 수 없기 때문에 BuyerID 속성을 만들었다.

- **Name:** 구매자의 이름을 나타낸다.
- **ContactInfo:** 부동산 업자가 거래를 위해서는 구매자의 연락처 정보를 저장하고 있어야 한다.
- **Buying 관계(Buyer(1)-<buying>- (N)Agent) :** 구매자는 대리하는 부동산 중개인. 다시 말해 BuyingAgent와의 관계가 필요하다. Buyer는 항상 agent와 연결되어야 하기 때문에 buyer는 total participation을 해야 한다. 또한 한 명의 agent가 여러 명의 buyer와 연결될 수 있기 때문에 일 대 다 관계이다. 그리고 **Agent** 엔티티의 **AgentID** 속성을 Buyer entity가 외래 키로 가져오게 될 것이란 것을 알 수 있다

2) Agent entity



- AgentID (기본 키): 부동산 에이전트를 고유하게 식별하는 식별자이다.
- Name: 에이전트의 전체 이름이다.
- ContactInfo: 에이전트와 연락할 수 있는 정보다.
- **Buying 관계(Buyer(1)-<buying>- (N)Agent) :** buying 관계에서 agent entity는 모든 agent가 buyer와 관계를 맺을 필요가 없기 때문에 partial participation을 하며 하나의 agent가 여러 명의 buyer와 대응할 수 있기에 화살표를 buyer 쪽에 그린다.
- **Selling 관계(Seller(1)-<Selling>- (N)Agent) :** selling 관계에서 agent entity는 buying 에서와 마찬가지로 partial participation을 하며 한 명의 agent가 여러 명의 selling agent와 대응하기에 일 대 다 관계를 가진다.
- **Represent 관계(Agent(1)-<represent>- (N)SaleProperty, identifying relationship) :** Saleproperty는agent와 관계를 갖는다. 이 때의 agent는 SellerAgent에 해당한다. Buyer agent의 경우는 후술할 다른 entity와 관계를 가질 것이다. 이 관계를 통해 SellerAgent가 SaleProperty, 다시 말해 부동산 매물을 올린다. 한 명의 Agent는 여러 개의 property와 대응하기 때문에 일 대 다 관계다.

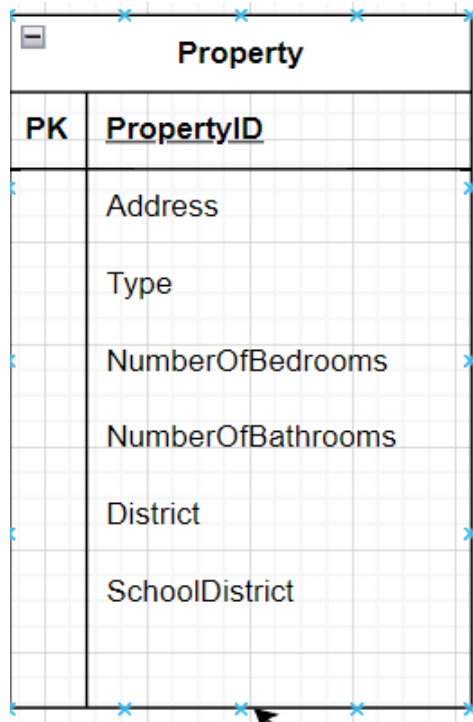
- **Conducts 관계(Agent(1)-<conducts>- (1)Transaction)** : agent는 transaction과 관계를 맺는다. 이 때의 agent는 BuyerAgent에 해당한다. 이 관계를 통해 BuyerAgentID가 transaction의 외래키로 들어갈 것이다. 한 명의 Agent는 여러 개의 transaction과 대응하기 때문에 일 대 다 관계이며 모든 agent가 transaction과 관계를 맺을 필요가 없기에 partial participation을 한다.

3) Seller entity

Seller	
PK	<u>SellerID</u>
	Name
	ContactInfo

- SellerID (기본 키): 판매자를 고유하게 식별하는 식별자다.
- Name: 판매자의 전체 이름이다.
- ContactInfo: 판매자의 연락처 정보로, 이는 구매자의 연락처 정보와 유사한 속성이다.
- **Selling 관계(Seller(1)-<Selling>- (N)Agent)** : 판매자는 대리하는 부동산 중개인. 다시 말해 SellingAgent와의 관계가 필요하다. Buyer는 항상 agent와 연결되어야 하기 때문에 seller는 total participation을 해야 한다. 또한 한 명의 agent가 여러 명의 seller와 연결될 수 있기 때문에 일 대 다 관계이다. 그리고 seller entity가 **Agent** 엔티티의 **AgentID** 속성을 Seller entity가 외래 키로 가져오게 될 것임을 알 수 있다

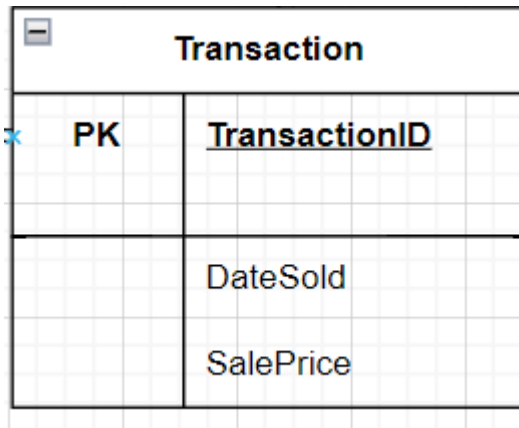
4) Property entity



- PropertyID (기본 키): 부동산을 고유하게 식별하는 식별자다. Property entity를 Sale Property와 구분짓고 따로 Property를 만들어 그 ID를 부여한 덕분에 동일한 매물이 여러 번 거래될 때 데이터 중복을 피할 수 있다.
- Address: 부동산의 구체적 주소를 나타낸다.
- Type : 건물의 타입을 나타낸다. 예를 들어 apartment인지, detached house인지 등의 정보를 저장한다. 쿼리 e를 수행하기 위해 필요하다.
- NumberOfBedrooms: 부동산 내의 침실 수입니다. 이 속성은 4번 쿼리를 처리하기 위해 필요하다.
- NumberOfBathrooms: 부동산 내의 욕실 수다. 4번 쿼리를 위해 필요
- District: 부동산이 위치한 구를 나타낸다.
- School District : 부동산이 위치한 학군을 나타낸다. b쿼리를 위해 필요
- **Illustrate 관계(Property(1)-<illustrate>- (N)Photo) :** 하나의 property를 찍은 여러 개의 사진이 존재할 것이다. Property는 photo와 관계를 갖는다. 이 관계를 통해 PropertyID가 Photo의 외래키로 들어갈 것이다. 하나의 Property에 대해 여러 개의 photo가 대응하기 때문에 일 대 다 관계이며 모든 property가 photo와 관계를 맺을 필요가 없기에 partial participation을 한다.
- **Onsale 관계(Property(1)-<Onsale>- (N)SaleProperty,identifying) :** Property는 매물로 나올 수 있다. 이 때 하나의 부동산은 여러 개의 매물로 나오는 것이 가능하므로

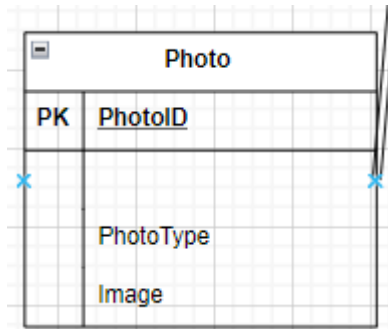
일 대 다 관계다. 또한 모든 Property가 SaleProperty일 필요는 없기에 partial participation한다.

5) Transaction entity



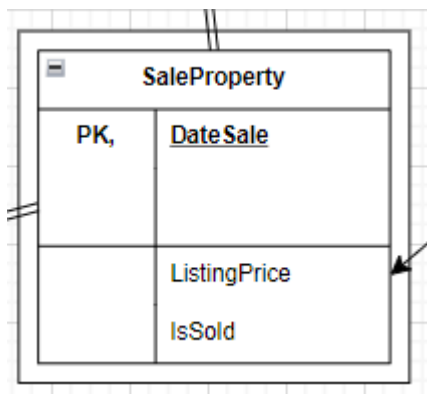
- TransactionID(기본 키): 기본 식별값이다.
- SalePrice: 부동산이 실제로 판매된 가격이다.
- DateSold : 부동산이 판매된 날짜.
- **Conducts 관계(Agent(1)-<conducts>- (N)Transaction)** : agent는 transaction과 관계를 맺는다. 이 관계를 통해 BuyerAgentID가 transaction의 외래키이자 주키로 들어갈 것이다. 한 명의 Agent가 여러 개의 거래를 수행할 수 있다, 다시 말해 한 명의 agent와 여러 개의 transaction가 대응하기 때문에 일 대 다 관계이다. 또한 모든 transaction은 agent가 있기 때문에 total participation한다..
- **SoldOut 관계(SaleProperty(1)-<soldout>- (1)Transaction)** : Saleproperty는 transaction과 관계를 갖는다. SaleProperty, 부동산 매물 중 거래가 완료된 매물은 transaction entity로 포함된다. 하나의 property가 하나의 transaction과 대응하기 때문에 일 대 일 관계이며 모든 transaction이 관계에 참여한다. .

6) Photo entity



- PhotoID (기본 키): 사진을 고유하게 식별하는 식별자다.
- PropertyID: 사진이 나타내는 부동산의 식별자다.
- PhotoType: 사진이 외부를 찍고 있는지 내부를 가리키고 있는지 알려준다.
- Image: 사진의 정보를 저장하는 데이터다.
- **Illustrate 관계(Property(1)-<illustrate>- (N)Photo)** : 하나의 property를 찍은 여러 개의 사진이 존재할 것이다. Property는 photo와 관계를 갖는다. 이 관계를 통해 PropertyID가 Photo의 외래키로 들어갈 것이다. 하나의 Property에 대해 여러 개의 photo가 대응하기 때문에 일 대 다 관계이며 모든 photo가 property와 연관되어야 하기에 total participation한다.

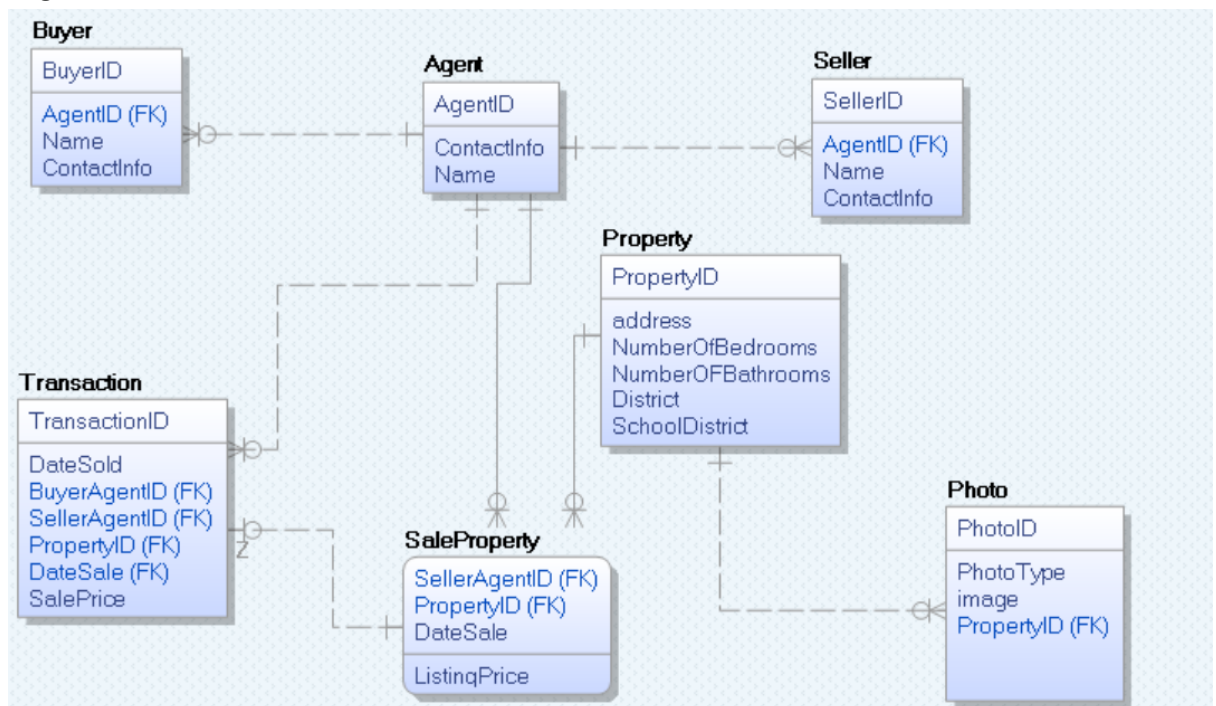
7) SaleProperty



- DateSale : 위 entity는 weak entity로 DateSale은 주 키 속성 중 하나이다. 강한 entity로부터 받아온 entity들과 함께 주 키를 형성한다. DateSale은 매물을 팔기 시작한 날짜를 의미한다.
- ListingPrice : 희망 판매가
- IsSold : 현재 팔린 상태인지를 나타낸다.

- **SoldOut 관계(SaleProperty(1)-<soldout>- (1)Transaction) :** Saleproperty는 transaction과 관계를 갖는다. SaleProperty, 부동산 매물 중 거래가 완료된 매물은 transaction entity로 포함된다. 하나의 property가 하나의 transaction과 대응하기 때문에 일 대 일 관계이며 모든 Saleproperty가 관계에 참여하진 않는다. 거래가 완료된 것들만 참여한다.
- **Represent 관계(Agent(1)-<represent>- (N)SaleProperty, identifying relationship) :** Saleproperty는agent와 관계를 갖는다.. 이 관계를 통해 SellerAgent가 SaleProperty,의 외래키이자 주키가 된다. 한 명의 Agent는 여러 개의 property와 대응하기 때문에 일 대 다 관계이고 weak entity이므로 자동으로 SaleProperty는 total participation한다.
- **Onsale 관계(Property(1)-<Onsale>- (N)SaleProperty,identifying) :** Property는 매물로 나올 수 있다. 이 관계를 통해 SaleProperty는 PropertID를 외래키이자 주키로 얻는다. 이 때 하나의 부동산은 여러 번 매물로 나오는 것이 가능하므로 일 대 다 관계다. 또한 weak entity이기 때문에 total.participation한다.
-

2. Logical Mode



이전 ERmodel 그림에서 직접적으로 나타나지 않았던 foreign key 값들이 각각의 entity들에 추가되었다.

- 1) Buying(Buyer – Agent) : 먼저 하나의 Agent에 대해 Buyer가 관계에 참여하지 않을 수 있다. 있다.(zero, one or more)(쿼리 중 새로운 agent를 추가하는 쿼리가 문제 조건에 있는데 이렇게 새로운 agent를 추가할 경우 새로운 agent는 어떠한 거래도 진행하지 않았기 때문에), 그리고 하나의 agent와 여러명의 Buyer, 일대다 관계 , agentID가 Buyer의 외래 키가 되는 non-identifying관계다.
- 2) Selling(Seller-Agent) : buyer-agent관계와 마찬가지로.
- 3) Illustrates : 먼저 하나의 Property에 대해 Photo가 아예 존재하지 않을 수 있다. 즉 관계에 참여하지 않는 property가 있을 수 있다. (zero,one or more) 반대로 모든 photo는 관계에 참여해야 한다. 한 그리고 하나의 property는 여러 개의 photo에 대응할 수 있으므로 일 대일 대응관계다. 또한 propertyid가 외래키가 되며 non identifying 관계다.
- 4) Onsale(Property-SaleProperty) : 하나의 property에 대해 saleproperty가 존재하지 않을 수도, 여러 개 존재할 수도 있다. 따라서 일 대 일 대응 관계다.(같은 건물이 여러 번 사고 팔며 여러 번 매물로 나오는 경우), 또한 모든 SaleProperty는 관계에 참여해야 한다.
- 5) Represent(Agent-SaleProperty) : 이 관계는 identifying 관계다. 부모 entity의 기본 키인 AgentID를 자식 entity가 자신의 기본키인 SellerAgentID로 사용하게 된다. (agent없는 saleproperty는 존재 불가능하다.) 하나의 agent에 대해 zero 혹은 여러 개의 SaleProperty가 대응된다.
- 6) Conduct(agent-transaction) : 하나의 agent에 대해 transaction이 존재하지 않을 수도 있고 여러 개가 대응될 수 있다. 일 대 다 관계다. 모든 transaction은 최소 하나의 agent와 연결되어야한다. Agent와 연결될 때 BuyingAgentID를 agent로부터 가져와 transaction이 외래 키로 사용하게 된다.
- 7) Soldout(SaleProperty- Transaction) : 하나의 SaleProperty는 하나 또는 0개의 Transaction과 관계된다.(SaleProperty는 아직 팔리지 않았다면 거래를 하지 않은 것이므로 0개가 관계될 수 있다, 반대로 모든 Transaction은 매물로 먼저 나와야 하므로 무조건 관계에 참여해야 한다). 또한 이 관계는 일 대일 관계다. 하나의 매물은 하나의 거래만 진행가능하기 때문이다. 이 관계를 통해 SaleProperty의 주요 키들이 Transaction의 외래키로 작용하게 된다.

3. Query

- a) 마포구에 있는 매물 중 10억에서 15억 사이인 집들의 주소 찾기

매물에 대한 정보를 담고 있는 Property 테이블과 매매 정보가 있는 SaleProperty 테이블을 결합하고, District가 'Mapo'이며 ListingPrice가 10억과 15억 사이인 레코드의 address 필드를 조회한다.

- 이를 위해 district 속성을 추가했고 그냥 property entity와 매물로 나온 saleproperty entity를 나눴다.

b) 8학군에 있는 매물 중 4개 이상의 침실과 2개의 욕실이 있는 집들의 주소 찾기

Property 테이블에서 SchoolDistrict가 '8'이고, NumberOfBedrooms가 4 이상이며, NumberOfBathrooms가 2인 레코드의 address 필드를 조회한다.

- 이를 위해 bedroom과 bathroom 개수를 속성으로 넣었으며 schooldistrict 속성도 추가했다.

c) 2022년에 가장 많은 금액의 매물을 판 중개인의 이름 찾기

Transaction 테이블에서 2022년에 판매한 모든 거래를 찾아, SalePrice를 합산하고, 가장 높은 금액을 기록한 SellerAgentID를 가진 중개인의 정보를 Agent 테이블에서 조회한다.

- 이를 위해 transaction에 date 속성과 Saleprice를 추가했고 또한 transaction이 selleragentID를 외래키로 받게 만들었다.

d) 각 중개인별로 2022년에 판매한 매물의 평균 판매 가격과 시장에 나와 있던 평균 기간 계산하기

Transaction 테이블에서 2022년에 판매된 모든 매물에 대한 정보를 가져와, SellerAgentID 별로 그룹화한 뒤, 평균 SalePrice와 시장에 나와 있던 평균 일수(판매 날짜에서 등록 날짜를 빼서 계산)를 구한다.

- 이를 위해 transaction와 saleProperty을 구분하며 DateSale과 DateSold 속성을 구분해 판매된 매물에 대해 시장에 나와 있던 일수를 계산할 수 있도록 했다.

e) 가장 비싼 스튜디오, 원룸, 다가구 주택, 단독주택의 사진 표시하기

Photo 테이블과 SaleProperty 테이블을 결합하고, 각 주택 유형(Type)에 따라 가장 높은 ListingPrice를 가진 매물의 사진을 조회한다.

- 이를 위해 Property에서 주택 유형을 구분하는 Type 속성을 추가하고 photo entity를 추가했다.

f) 매물의 판매 기록하기

판매된 매물의 정보를 포함하여 Transaction 테이블에 새로운 레코드를 추가합니다. 이 레코드에는 판매 가격, 구매자(BuyerID), 판매 중개인(SellerAgentID), 구매자의 중개인(BuyerAgentID - 있을 경우), 판매 날짜(DateSold)가 포함된다.

- 이를 위해 transaction 테이블에 BuyerAgentID, SellerAgentID, DateSold, 등을 속성으로 추가했다. BuyerID의 경우, Buyer 테이블에 BuyerAgentID와 함께 저장한다.

g) 새 중개인을 데이터베이스에 추가하기

Agent 테이블에 새 중개인의 정보를 입력하여 새 레코드를 추가한다. 이때 중개인의 이름, 연락처 등의 정보가 포함되어야 한다.

- 이를 위해 agent와 다른 entity와의 관계에서 entity는 다른 관계의 entity가 없어도 존재가능하게 했다.