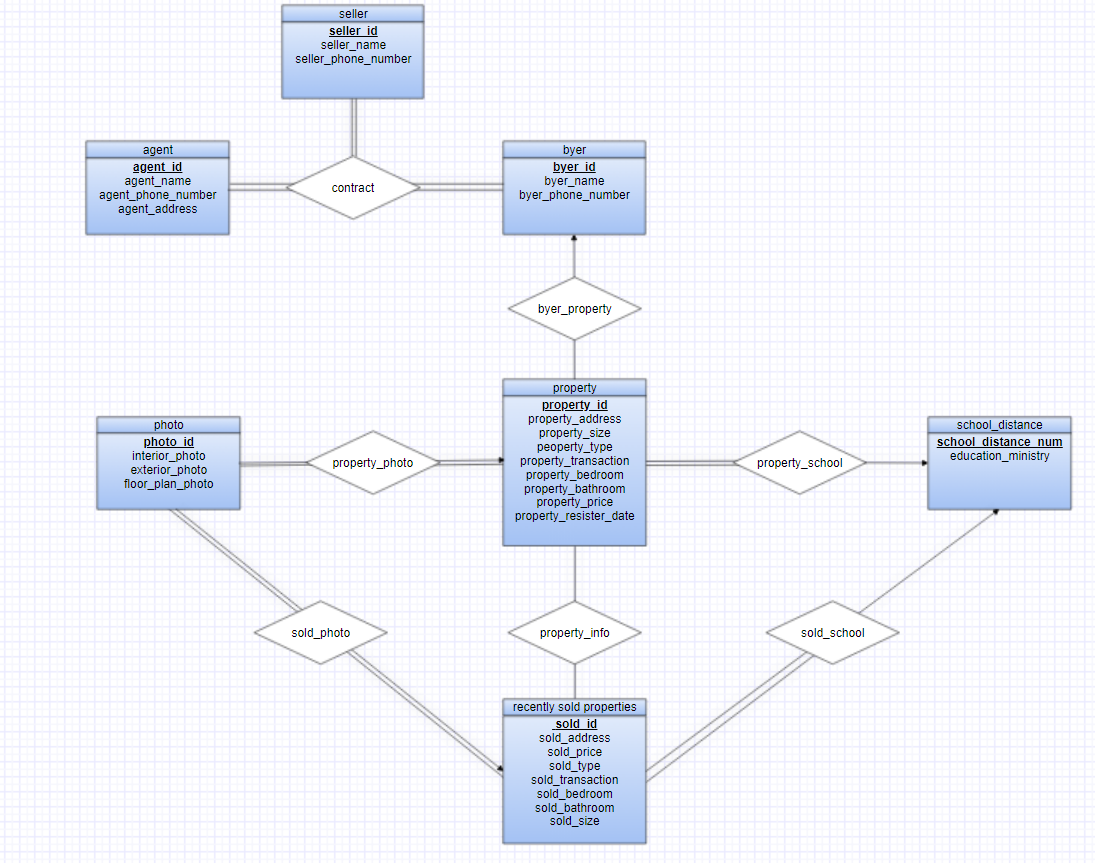
**데이터베이스시스템**

**(CSE4110-02)**

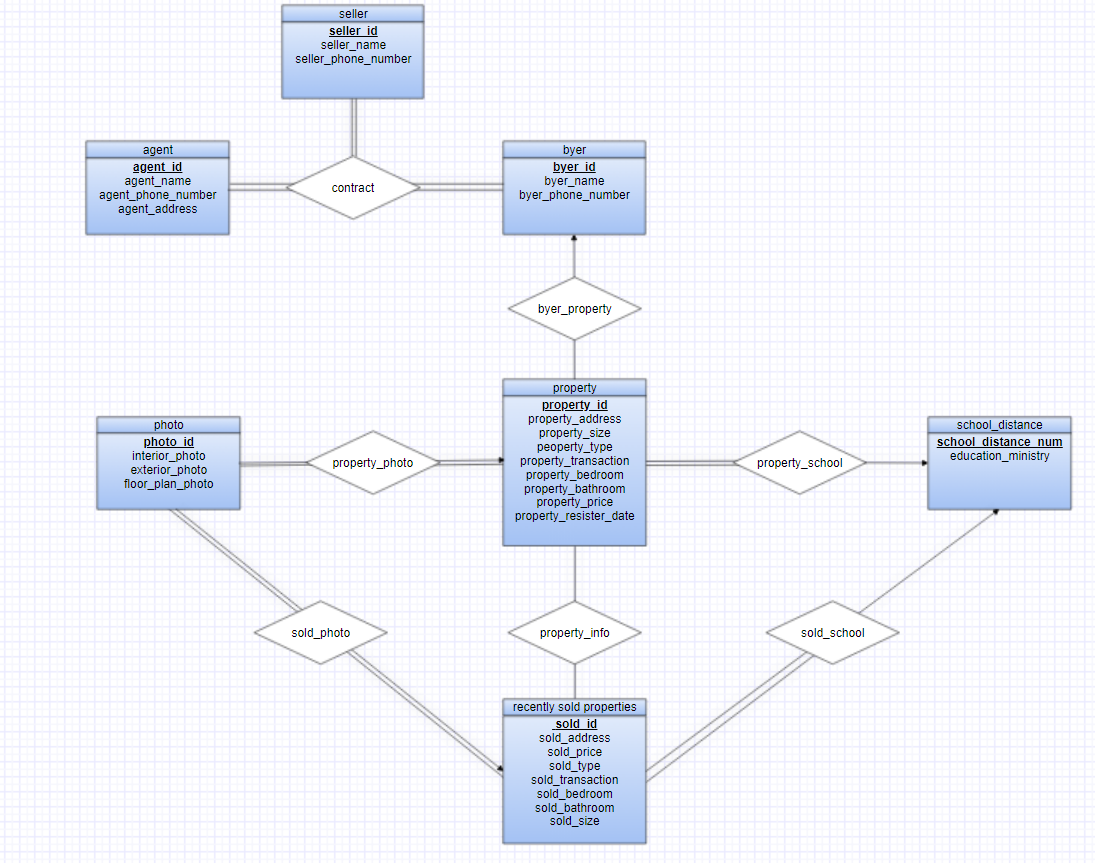
**project1**

**학번: 20211188**

**이름: 방수윤**



**1. E-R diagram**



**1-1. Entity set and Attribute**

**(1) Property**

중개사에서 관리하는 부동산 매물의 정보를 저장하고 있는 set이다.

-property\_id: 매물별로 부여 받은 고유의 id를 의미한다.

-property\_address: 매물의 위치를 나타내는 주소를 의미한다. 시, 도와 함께 상세주소에 관한 value를 갖는다.

-property\_size: 매물의 크기를 나타낸다. 평수를 value로 갖는다.

-property\_type: 매물의 종류를 나타낸다. 오피스텔, 빌라, 아파트 등의 종류 중 어떤 type인지를 저장한다.

-property\_transaction: 매물의 전세/매매/월세와 같은 부동산 거래 유형을 value로 갖는다.

-property\_bedroom: 매물의 침실 개수를 갖는다.

-property\_bathroom: 매물의 화장실 개수를 가진다.

-property\_price: 매물의 가격을 value로 갖는다.

-property\_resister\_date: 매물 등록 년도, 월, 일, 시간의 형태식의 value를 갖는다.

property에서의 primary key는 property\_id이다.

**(2) agent**

부동산 매물을 거래하는 중개사에 관한 정보를 담고 있는 set이다.

-agent\_id: 중개사의 가입 아이디로 고유번호를 나타낸다.

-agent\_name: 중개사의 이름을 나타낸다.

-agent\_phone\_number: 중개사의 전화번호를 나타낸다. 중개사는 하나의 전화번호를 가진다.

-agent\_address: 중개소의 위치를 나타낸다.

agent의 primary key는 agent\_id이다.

**(3) byer**

부동산 매물을 구매하는 구매자에 관한 정보를 담고 있는 set이다.

-byer\_id: 구매자의 고유 id를 나타낸다.

-byer\_name: 구매자의 이름을 의미한다.

-byer\_phone\_number: 구매자의 전화번호를 의미한다. 구매자는 하나의 전화번호만을 가진다.

byer의 primary key는 byer\_id이다.

**(4) seller**

부동산 매물을 판매하는 판매자에 관한 정보를 저장하는 set이다.

-seller\_id: 판매자의 고유한 id를 나타낸다.

-seller\_name: 판매자의 이름을 나타낸다.

-seller\_phone\_number: 판매자의 전화번호를 의미한다. 판매자는 하나의 전화번호만을 가진다.

seller의 primary key는 seller\_id이다.

**(5) Photo**

매물의 사진 정보를 저장하는 set이다.

-photo\_id: 매물 사진의 고유한 id를 나타낸다.

-interior\_photo: 매물의 외부사진을 나타낸다.

-exterior\_photo: 매물의 내부사진을 나타낸다.

-floor\_plan\_photo: 매물의 바닥 도면 사진을 나타낸다.

photo의 primary key는 photo\_id이다.

**(6) school\_distance**

매물과 가까운 학교 정보를 저장하는 set이다.

-school\_distance\_num: 서울에서 1~11학군 중 해당 매물과 가장 가까운 학군 number을 의미한다.

-education\_ministry: 해당 학군의 교육청 이름을 의미한다.

school\_distance의 primary key는 school\_distance\_num이다.

**(7) recently sold properties**

같은 중개인이 최근에 판매했던 매물의 정보를 저장하는 set이다.

-sold\_id: 최근 판매된 매물의 고유한 id이다.

-sold\_address: 최근 판매된 매물의 주소를 나타낸다.

-sold\_price: 최근 판매된 매물의 가격을 나타낸다.

-sold\_type: 최근 판매된 매물의 종류를 나타낸다. 오피스텔, 빌라, 아파트 등의 종류 중 어떤 type인지를 저장한다.

-sold\_transaction: 최근 판매된 매물의 전세/매매/월세와 같은 부동산 거래 유형을 value로 갖는다.

-sold\_\_bedroom: 최근 판매된 매물의 침실 개수를 갖는다.

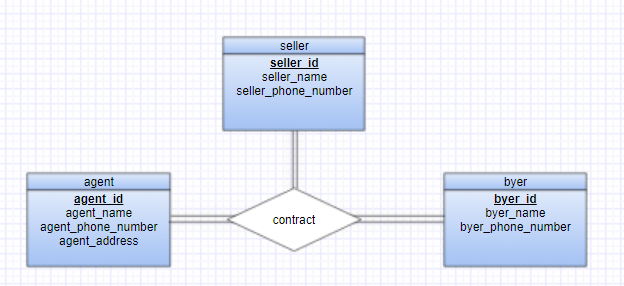
-sold\_bathroom: 최근 판매된 매물의 화장실 개수를 갖는다.

-sold\_size: 최근 판매된 매물의 평수를 나타낸다.

recently sold properties의 primary key는 sold\_id이다.

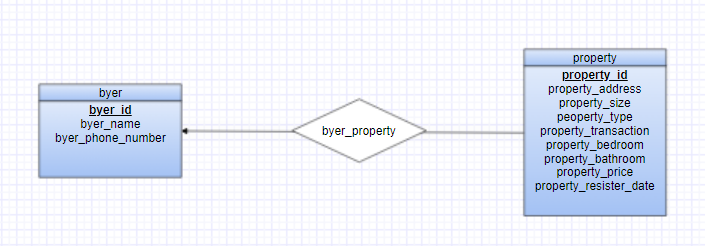
**1-2. Relationship set and mapping cardinality**

**(1) contract**

****

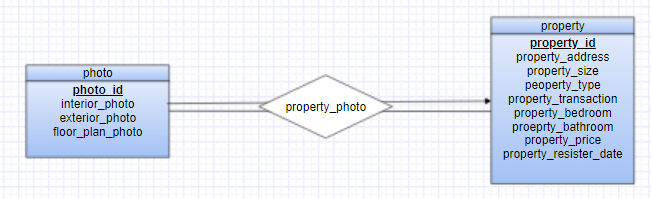
판매자인 seller와 이를 중계하는 agent, 매물을 살 byer와의 계약 관계를 나타내는 ternary relationship set 이다. 하나의 agent는 다른 여러명의 byer와, 여러명의 seller와 계약할 수 있고, 하나의 byer 또한 여러 agent와, 여러 seller와 계약을 할 수 있다. 하나의 seller는 여러 agent와 여러 byer와 계약을 할 수 있기 때문에 각각 many-to-many 관계를 가진다. 또한 직방 앱에서는 모든 seller가 최소 한 곳의 agent, 최소 한 명의 byer와 contract되기에 total이다. 모든 byer는 최소한 한곳의 agent, 최소 한 명의 seller와 contract되기에 total이다. 모든 agent는 적어도 한명의 seller, 적어도 한명의 byer와 함께 contract되기에 total이다.

**(2) byer\_property**

****

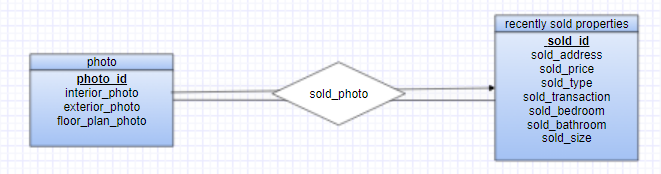
agent와 계약한 byer와 매물의 property간의 관계를 나타내는 relationship set이다. 여러개의 매물을 한명의 byer는 구입할 수 있지만, 하나의 매물은 여러명이 구매할 수 없으므로 one-many이다.

**(3) property\_photo**

****

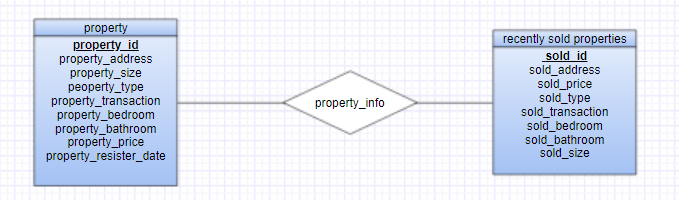
매물의 property와 매물의 photo간의 정보를 연결하여 만든 relationship set이다. 하나의 매물은 여러개의 사진을 가질 수 있고, 하나의 사진은 여러개의 매물을 가질 수 없으므로 many to one이다. 모든 photo는 반드시 최소 한개의 매물 property에 대한 정보이며, 모든 매물 property는 최소한 한개의 photo를 가지기 때문에 total 이다.

**(4) sold\_photo**

****

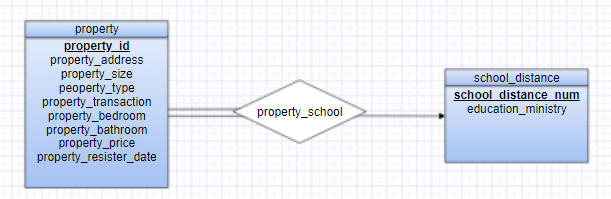
최근 판매된 정보와 photo와의 정보를 연결하여 만든 relationship set이다. 하나의 photo는 여러개의 recently sold properties 를 담을 수 없고 오직 하나만 담을 수 있다. 하나의 recently sold property는 여러개의 photo를 담을 수 있으므로 many-to-one이다. 모든 photo는 적어도 하나의 recently sold propertiy를 포함하기 때문에 total이다. 또한, 모든 recently sold properties는 적어도 하나의 photo를 포함하기 때문에 total이다.

**(5) property\_info**

****

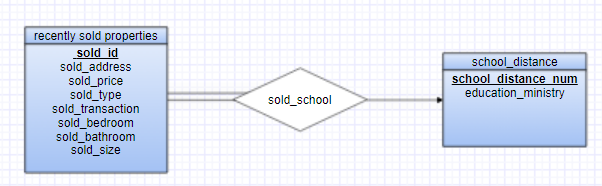
해당 property를 올린 agent가 최근 판매할 매물의 property인 recently sold properties와의 관계를 연결한 relationship set이다. 하나의 property에는 여러개의 recently sold properties가 있을 수 있고 하나의 recently sold properties에 이와 연결되는 여러개의 property가 존재할 수 있으므로 many-to-many 이다. 최근 판매된 매물 이후에 더이상 매물이 올라오지 않을 경우, 처음으로 매물을 올려서 판매된 매물이 없을 경우를 고려해 total이 아니다.

**(6) property\_school**

****

부동산 매물정보인 property와 해당 매물의 학군정보를 담고있는 school distance를 연결한 relationship set이다. 하나의 property는 가장 가까운 하나의 school distance 정보만이 속하고 하나의 school\_distance에는 다양한 property가 연결될 수 있기 때문에 many-to-one이다. 또한 모든 매물 property는 학군정보를 담고 있기 때문에 total이다.

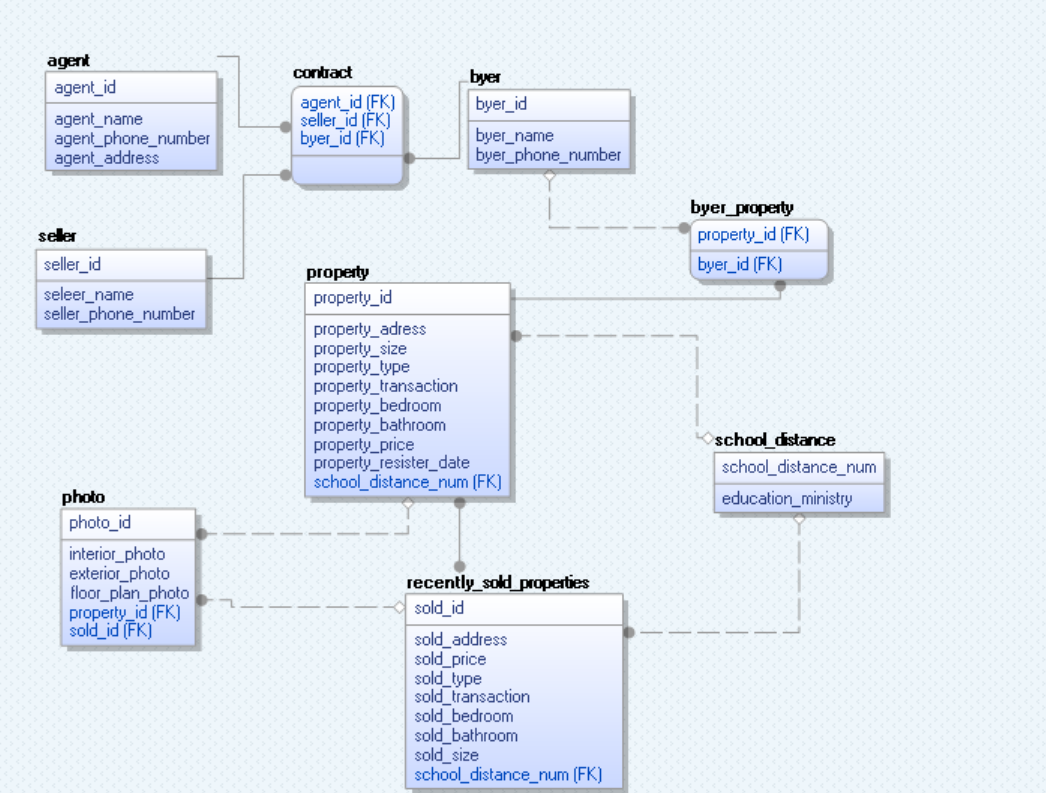
**(7) sold\_school**

****

최근 판매된 매물정보인 recently sold properties와 학군 정보인 school distance와의 정보를 연결한 relationship set이다. 하나의 recently sold property는 해당 위치와 가장 가까운 학군 한개의 정보와만 연결되고, school distance는 여러개의 매물들과 연결될 수 있으므로 many-to-one이다.

또한 모든 recently sold properties는 학군 정보를 담고 있기 때문에 total이다.

**2. Schema diagram**



relation set을 schema로 구현하여 표현을 하는 방법을 기반으로 위의 schema diagram을 설명하겠다.

**(1) contract**

seller, agent, byer 사이의 관계가 모두 total이지만 many-to-many이므로 constract 라는 새로운 schema를 생성하였다. constract는 seller의 primary key인 seller\_id와 agent의 primary key인 agent\_id , byer의 primary key인 byer\_id를 attribute로 갖고 이 attribute들은 모두 primaty key이다. 또한 constract는 seller의 primary key인 seller\_id와 agent의 primary key인 agent\_id , byer의 primary key인 byer\_id는 constract에서 foreign key이다.

**(2) byer\_property**

property와 byer가 one-to-many이지만, many side가 total이 아니므로 byer\_property라는 새로운 schema를 생성했다. byer\_property는 byer의 primary key인 byer\_id와 property의 primary key인 property\_id를 속성으로 갖는다. 그중에서 many side인 property의 primary key인 property\_id가 byer\_property의 primary key가 된다. 그리고 byer\_property는 byer의 primary key인 byer\_id와 property의 primary key인 property\_id가 각각에서 primary key이므로 byer\_property에서 foreign key이다.

**(3) property\_photo**

property와 photo는 one-to-many인데 many-side가 total이므로 별도의 schema를 생성하지 않았다. one-side인 property의 primary key인 primary\_id가 photo의 attribute로 추가된다. 그리고 photo 안에 primary\_id는 property 안에서 primary key 이므로 photo 안에서 foreign key이다.

**(4) sold\_photo**

recently\_sole\_properties와 photo는 one-to-many인데 many-side가 total이므로 별도의 schema를 생성하지 않았다. one-side인 recently\_sole\_properties의 primary key인 sold\_id가 photo의 attribute로 추가된다. 그리고 photo 안에 sold\_id는 recently\_sole\_properties 안에서 primary key 이므로 photo 안에서 foreign key이다.

**(5) property\_info**

property와 recently\_solt\_properties는 many-to-many라서 별도의 schema를 생성해 줄 필요가 없다. 또한 many-to-many이므로 property\_info에서 foreign key는 없다.

**(6) property\_school**

property와 school\_distance는 many-to-one인데 many-side가 total이므로, 별도의 schema를 생성하지 않는다. one-side인 school\_distance의 primary key인 school\_distance\_num이 property의 attribute에 추가되고 이는 property의 primary key이므로 school\_distance에서 foreign key이다.

**(7) sold\_school**

recently\_sole\_properties와 school\_distance는 many-to-one인데 many-side가 total이므로, 별도의 schema를 생성하지 않는다. one-side인 school\_distance의 primary key인 school\_distance\_num이 recently\_sole\_properties의 attribute에 추가되고 이는 recently\_sole\_properties의 primary key이므로 school\_distance에서 foreign key이다.