

응용주제: 설문조사 서비스

설문조사 서비스 분야에서 필요한 대용량 DB를 설계를 위한 제약조건 분석, E-R 다이어그램 생성, 관계 DB 스키마 설계, 정규화 과정에 대한 프로젝트를 진행하였다.

I. 응용분야 제약조건

- 설문조사 시스템의 유저는 아이디, 비밀번호, 이름, 이메일, 성별, 나이 정보를 입력해야 한다. 고유식별자는 아이디와 이메일이다.
- 설문조사 시스템의 유저는 새로운 설문조사를 개설할 수 있다.
- 설문조사는 id, 설문조사제목, 설명, 개설일, 마감일 정보로 이루어져 있다.
- 설문조사는 누구나 참여할 수 있다.
- 유저가 아닌 설문조사 참여자는 이메일, 이름, 성별, 나이 정보를 입력해야 한다.
- 설문조사에 여러 명이 참여할 수 있고, 한 명도 참여하지 않을 수도 있다.
- 각 설문조사는 질문(question)을 포함한다.
- 질문은 id, 질문번호, 질문내용으로 구성된다.
- 질문타입은 id와 타입이름으로 구성된다.
- 각 질문에는 가능한 답변목록이 있다(1. 매우만족, 2.만족.... 이런형태이다).
- 가능한 답변목록은 id와 목록에 대한 설명으로 구성된다.
- 설문조사 참여자는 각 질문에 대해 대답을 제출한다.
- 대답은 id, 대답시작시간, 대답끝난시간으로 구성된다.
- 모든 대답은 객관식 문제의 선택(예시 1~5 택) 또는 단답 또는 서술형식의 답변으로 이루어져있다.

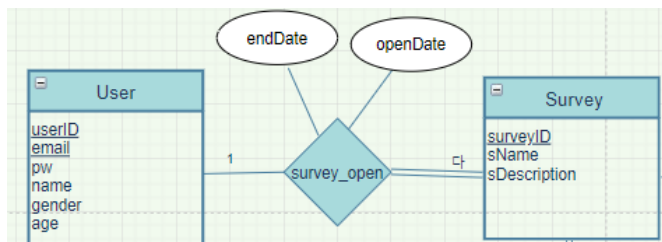
II. 개념적 설계를 통한 E-R 다이어그램 생성

1. 개체(Entity) 및 속성(attribute) 추출

개체	속성
User	<u>userID</u> , email, pw, name, gender, age
Survey	<u>surveyID</u> , sName, sDescription
Respondent	<u>respEmail</u> , rName, rGender, rAge
Question	<u>questionID</u> , qNum, qDescription
Question_Type	<u>typeID</u> , typeName
Possible_Answer_List	<u>answerlistID</u> , aDescription
Answer	<u>respAnswerID</u> , startTime, endTime
Multiple_choice Answer	choiceNum
Text_Answer	text

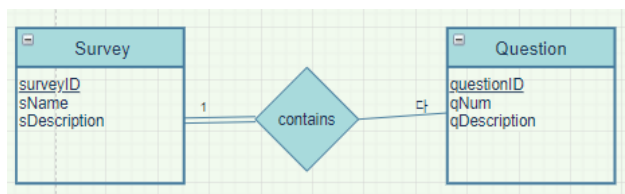
2. 개체 간의 관계(relation)추출

2.1 survey_open 관계(User, Survey)



- 한 유저는 여러 설문조사를 개설할 수 있다.
- 한 설문조사는 한 명의 유저가 개설할 수 있다.
- 설문조사는 반드시 개설한 사람이 있으므로 total participation 이고, 설문조사를 개설하지 않은 유저가 존재할 수 있으므로 partial participation 이다.
- 1 : 다 관계이다.

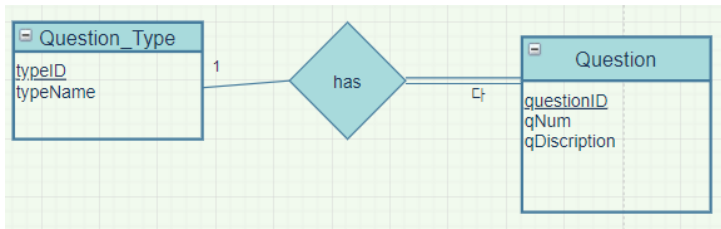
2.2 contains 관계(Survey, Question)



- 한 개의 설문조사는 질문들을(1~N 개) 포함한다.
- 모든 설문조사는 질문을 가진다. Total participation 이다.

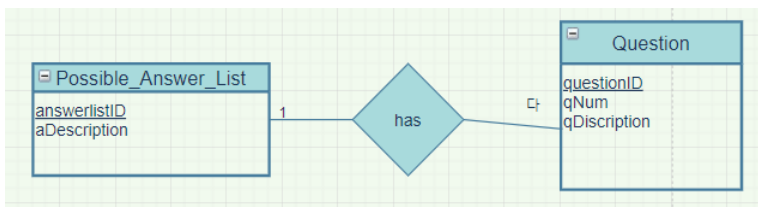
- 각 질문은 한 가지 설문조사와 연동된다.
- 설문조사에 사용되지 않은 질문들이 있을 수 있다. Partial participation 이다.
- 1 대 다 관계이다.

2.3 has 관계(Question, Question_Type)



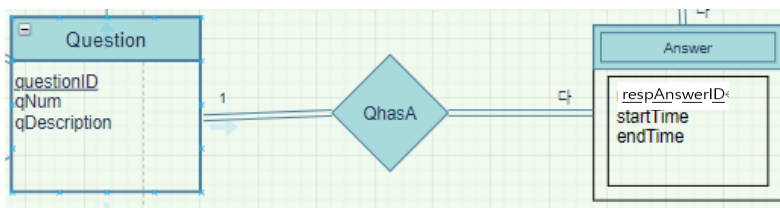
- 하나의 질문은 한 가지의 타입만 가진다(객관식, 단답형 중).
- 모든 질문은 타입을 가진다 Total participation 이다.
- 각 질문타입은 여러 질문에서 사용될 수 있고, 사용되지 않는 질문타입이 있을 수 있다. Partial participation 이다.

2.4 has 관계(Question, Possible_Answer_List)



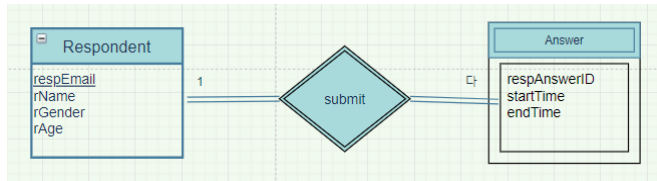
- 하나의 질문은 한 가지의 가능한 답변목록(possible answer list)이 존재한다.
- 단답형 질문의 경우에도 가능한 답변목록이 존재한다. 사용되지 않는 가능한 답변목록이 있을 수 있다. 그래서 양쪽개체 모두 partial participation 이다.
- 단답형 질문의 경우에 가능한 답변목록은 (AAA, 단답형으로 적어주세요) 형태가 된다.

2.5 QhasA 관계(Question, Answer)



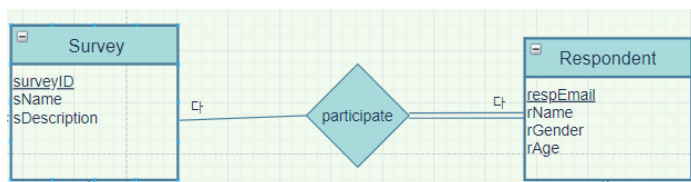
- 하나의 질문은 여러 답변을 가질 수 있다. 설문조사 참여자마다 답변이 다를 수 있기 때문이다.
- 각 답변은 하나의 질문에 대한 것이다.
- 모든 질문은 답변을 가지고, 모든 답변은 매칭되는 질문이 존재한다. 양쪽개체 모두 total participation 이다.

2.6 submit 관계(Respondent, Answer)



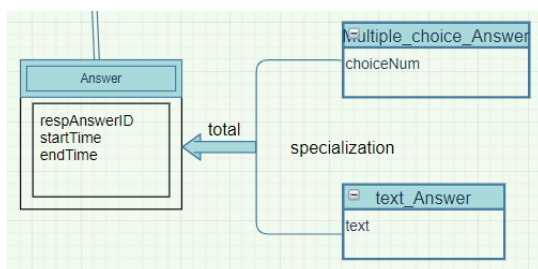
- 답변은 응답자 없이 존재할 수 없다. 즉 답변은 약한개체이고, 응답자는 식별해주는 개체이다.
- 모든 답변은 응답자가 존재한다. 응답자는 모두 답변을 하기 때문에 응답자이다. 그래서 양쪽 개체 모두 total participation 이다.
- 한 명의 응답자는 여러 답변을 제출할 수 있다. 한 설문조사에 여러 개의 질문이 있을 수 있기 때문에 그렇다. 예를 들어서 한 설문조사에 5 개의 질문이 있다면 응답자는 5 개의 답변을 제출한 것이다. 반면 한 개의 답변은 한 명의 응답자가 제출한 것이다 그래서 1:다 관계이다.

2.7 participate 관계(Respondent, Survey)



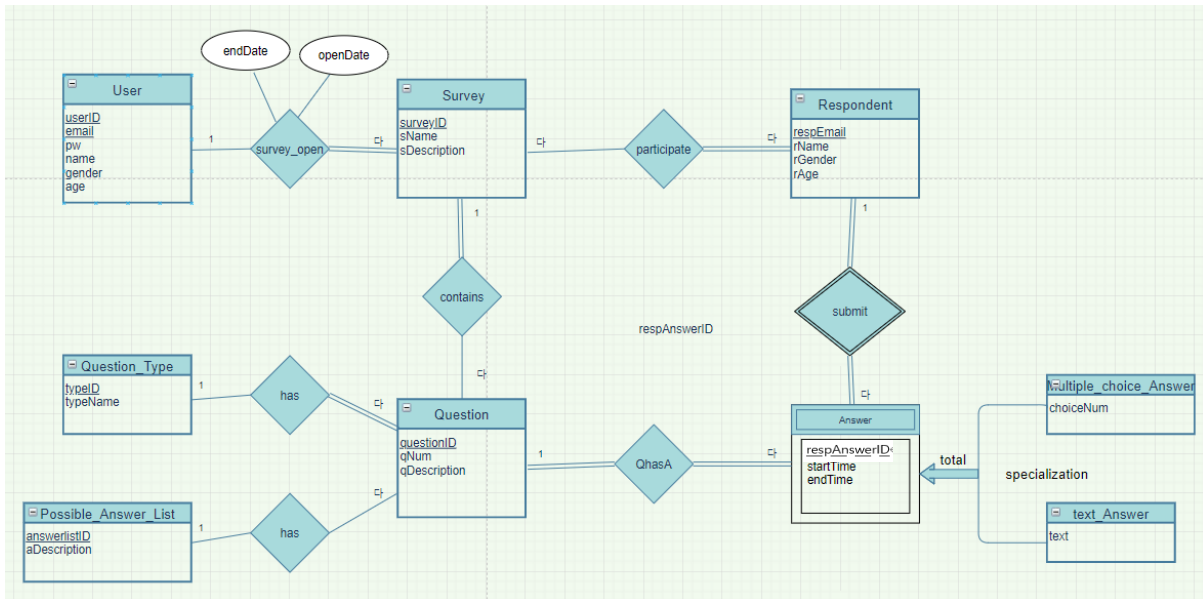
- 모든 응답자는 설문조사에 참여한 사람들이다. Total participation 이다.
- 어떤 설문조사는 참여자가 없을 수도 있다. Partial participation 이다.
- 한 설문조사에 여러 참여자가 있을 수 있고, 한 명의 응답자가 여러 설문조사에 참여할 수 있으므로 다:다 관계이다.

2.8 specialization (Answer 을 multiple_choice_Answer, text_Answer 로 specialize)



- Total specialization 이다. 왜냐하면 한 가지 답변은 객관식 선택, 단답형 작성 모두의 속성을 가질 수 없다. 둘 중 하나에만 속하기 때문이다.
- Answer 개체를 specialize 한 것이 multiple_choice_Answer 과 text_Answer 이다. 상속의 관계에 있다.

3. 분석한 내용으로 ERD 생성



Ⅲ. ERD ⑦ 테이블 스키마 변환 1.

Strong entity set 의 테이블 변환

- ⇒ User(userID, email, pw, name, gender, age)
- ⇒ Survey(surveyID, sName, sDescription)
- ⇒ Respondent(respEmail, rName, rGender, rAge)
- ⇒ Question (questionID, qNum, qDescription)
- ⇒ Question_Type(typeID, typeName)
- ⇒ Possible_Answer_List(answerlistID, aDescription)

2. Weak entity set 의 테이블 변환

- ⇒ Answer(respEmail, respAnswerID, startTime, endTime)

3. 다 : 다 relationship set 테이블 변환

- ⇒ participate(surveyID, respEmail)

4. 1 : 다 relationship set 테이블 변환

- ⇒ survey_open

- survey_open 는 독립된 테이블로 변환했다.
- '다'쪽이 survey 개체이므로 surveyID 가 기본키이다.
- survey_open(surveyID, userID, openDate, endDate)

↗ contains

- contains 는 외래키 관계로 표현했다. '다'쪽 개체인 Question 개체에 외래키를 설치했다.
- Question (questionID, qNum, qDescription)
 Survey(surveyID, sName, sDescription)



Question (questionID, qNum, qDescription, **surveyID**)
 Survey(surveyID, sName, sDescription)

↗ has(Question, Question_Type)

- has 는 외래키 관계로 표현했다. '다'쪽 개체인 Question 개체에 외래키를 설치했다.
- Question (questionID, qNum, qDescription, surveyID)
 Question_Type(typeID, typeName)



Question (questionID, qNum, qDescription, surveyID, **typeID**)
 Question_Type(typeID, typeName)

↗ has(Question, Possible_Answer_List)

- has 는 외래키 관계로 표현했다. '다'쪽 개체인 Question 개체에 외래키를 설치했다.
- Question (questionID, qNum, qDescription, surveyID, typeID)
 Possible_Answer_List(answerlistID, aDescription)



Question (questionID, qNum, qDescription, surveyID, typeID, **answerlistID**)
 Possible_Answer_List(answerlistID, aDescription)

↗ QhasA

- '다'쪽 개체인 Answer 개체에 외래키를 설치했다.
- Question (questionID, qNum, qDescription, surveyID, typeID, **answerlistID**)

Answer(respEmail, respAnswerID, startTime, endTime)



Question (questionID, qNum, qDiscription, surveyID, typeID, answerlistID)

Answer(respEmail, respAnswerID, startTime, endTime, **questionID**)

⇒ submit

약한개체와 관련된 relationship 이다. 테이블로 만들면 redundancy 가 발생하기 때문에 테이블로 변환하지 않았다.

5. specialization 을 테이블로 변환

위에서 Answer 을 specialize 한 것이 Multiple_choice_Answer 과 text_Answer 이라고 하였다. Answer 의 기본키를 두 entity set 에 포함시켰다. 즉 외래키 참조관계를 통해서 테이블로 표현하였다.

Multiple_choice Answer(choiceNum)

Text_Answer(text)



Multiple_choice Answer(respEmail, respAnswerID, choiceNum)

Text_Answer(respEmail, respAnswerID, text)

6. 변환규칙에 의한 스키마(총 테이블 11 개)

- ⇒ User(userID, email, pw, name, gender, age)
- ⇒ Survey(surveyID, sName, sDescription)
- ⇒ Respondent(respEmail, rName, rGender, rAge)
- ⇒ Question (questionID, qNum, qDescription, **surveyID**, **typeID**, **answerlistID**)
- ⇒ Question_Type(typeID, typeName)
- ⇒ Possible_Answer_List(answerlistID, aDescription)
- ⇒ Answer(**respEmail**, respAnswerID, startTime, endTime, **questionID**)
- ⇒ participate(**surveyID**, **respEmail**)
- ⇒ survey_open(**surveyID**, **userID**, openDate, endDate)
- ⇒ Multiple_choice Answer(**respEmail**, **respAnswerID**, choiceNum)
- ⇒ Text_Answer(**respEmail**, **respAnswerID**, text)

속성들 부연설명

- sName, sDescriptor 의 앞에있는 s 는 survey 의 약자이다.
- respEmail 의 앞에있는 resp, rName, rGender, rAge 앞의 r 은 모두 respondent 의 약자이다
- qNum, qDescription 의 앞에 있는 q 는 question 의 약자이다.

IV. 함수종속 및 BCNF 정규화

BCNF 인지 확인하기 위해서는 non-trivial 한 함수종속만 따져보면 된다.

User(userID, email, pw, name, gender, age)

- (userID, email)이 후보키이므로 (userID, email)이 결정자인 함수종속은 고려하지 않는다. 결정자가 후보키인 함수종속은 BCNF 에 유효한 함수종속이기 때문이다.
- 그 외의 함수종속은 없다.

Survey(surveyID, sName, sDescription)

- surveyID 가 후보키이므로 surveyID 가 결정자인 함수종속은 고려하지 않는다. 결정자가 후보키인 함수종속은 BCNF 에 유효한 함수종속이기 때문이다.
- 그 외의 함수종속은 없다.

Respondent(respEmail, rName, rGender, rAge)

- respEmail 이 후보키이므로 respEmail 이 결정자인 함수종속은 고려하지 않는다. 결정자가 후보키인 함수종속은 BCNF 에 유효한 함수종속이기 때문이다.
- 그 외의 함수종속은 없다.

Question (questionID, qNum, qDiscription, surveyID, typeID, answerlistID)

- questionID 가 후보키이므로 questionID 가 결정자인 함수종속은 고려하지 않는다. 결정자가 후보키인 함수종속은 BCNF 에 유효한 함수종속이기 때문이다.
- 그 외의 함수종속은 없다.

Question_Type(typeID, typeName)

- typeID 가 후보키이므로 typeID 가 결정자인 함수종속은 고려하지 않는다. 결정자가 후보키인 함수종속은 BCNF 에 유효한 함수종속이기 때문이다.
- 그 외의 함수종속은 없다.

Possible_Answer_List(answerlistID, aDescription)

- answerlistID 가 후보키이므로 answerlistID 가 결정자인 함수종속은 고려하지 않는다. 결정자가 후보키인 함수종속은 BCNF 에 유효한 함수종속이기 때문이다.
- 그 외의 함수종속은 없다.

Answer(respEmail, respAnswerID, startTime, endTime, questionID)

- (respEmail, respAnswerID)가 후보키이므로 (respEmail, respAnswerID)가 결정자인 함수종속은 고려하지 않는다. 결정자가 후보키인 함수종속은 BCNF 에 유효한 함수종속이기 때문이다.
- 그 외의 함수종속은 없다.

participate(surveyID, respEmail)

- (surveyID, respEmail)이 후보키이므로 (surveyID, respEmail)이 결정자인 함수종속은 고려하지 않는다. 결정자가 후보키인 함수종속은 BCNF 에 유효한 함수종속이기 때문이다.
- 그 외의 함수종속은 없다.

survey_open(surveyID, userID, openDate, endDate)

- respEmail 이 후보키이므로 respEmail 이 결정자인 함수종속은 고려하지 않는다. 결정자가 후보키인 함수종속은 BCNF 에 유효한 함수종속이기 때문이다.
- 그 외의 함수종속은 없다.

Multiple_choice Answer(respEmail, respAnswerID, choiceNum)

- (respEmail, respAnswerID)가 후보키이므로 (respEmail, respAnswerID)가 결정자인 함수종속은 고려하지 않는다. 결정자가 후보키인 함수종속은 BCNF 에 유효한 함수종속이기 때문이다.
- 그 외의 함수종속은 없다.

Text_Answer(respEmail, respAnswerID, text)

- (respEmail, respAnswerID)가 후보키이므로 (respEmail, respAnswerID)가 결정자인 함수종속은 고려하지 않는다. 결정자가 후보키인 함수종속은 BCNF 에 유효한 함수종속이기 때문이다.
- 그 외의 함수종속은 없다.

즉 모든 테이블이 BCNF 정규화 되어있다..

VI. JDBC/MySQL 프로그래밍

설계 예정 테이블 중에서 survey 테이블을 생성하고 2 개의 튜플을 삽입하였다.

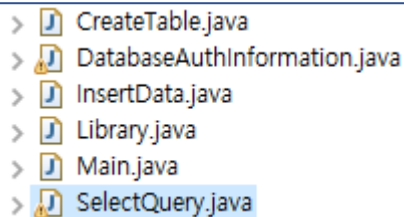
아래는 삽입한 2 개 튜플의 내용이다.

```
AAA111 비대면강의만족도설문조사: 이 설문조사는 2학기 비대면강의에 대한 학생들의 만족도 조사를 위한 것입니다.  
BBB222 제품만족도설문조사: 이 설문조사는 2분기 신제품에 대한 소비자 만족도 조사를 위한 것입니다.
```

아래처럼 surveyID 가 'AAA111'인 튜플을 검색하였고, 결과가 잘 나오는 것을 확인하였다.

```
String query_string = "select sName,sDescription from survey where surveyID='AAA111';";
```

```
비대면강의만족도설문조사: 이 설문조사는 2학기 비대면강의에 대한 학생들의 만족도 조사를 위한 것입니다.
```



위의 그림은 작성한 클래스 목록이다. createTable 로 테이블을 생성하는 기능을 구현하였고, InsertData 로 생성한 테이블에 튜플을 입력하는 기능을 구현했다. selectQuery 는 튜플을 특정조건에서 검색하는 기능이다. 이 기능을 Main 에서 호출해서 사용할 수 있게 하였다.

```
public class Main {  
  
    public static void main(String[] args) {  
        //survey테이블 생성  
        CreateTable.createT();  
  
        //튜플 2개 삽입  
        //InsertData.insertDataToSurvey();  
  
        //삽입한 튜플2개의 내용  
        //SelectQuery.SelectfromSurvey();  
  
        //튜플 검색  
        SelectQuery.SelectQueryfromInput();  
    }  
}
```

메인클래스의 메인메소드이다. 주석처리 한 곳을 풀면 기능을 수행할 수 있다.

Ⅶ. 응용의 복잡도 높은 기능을 지원하기 위한 자연어 질의 리스트 및 해당 SQL 문

1. '대학 커리큘럼 만족도 조사' 설문조사에 응답한 사람의 이름과 이메일을 모두 찾아라.

```
select R.rName, R.respEmail
from Participate as P, Survey as S, Respondent as R
where P.survevID = S.survevID and P.respEmail = R.respEmail and S.sName = '대학 커리큘럼
만족도 조사'
```

- ⑨ 이 질의는 설문조사 참여자들의 목록을 얻고 싶을 때 사용할 수 있다. 설문조사 참여시
경품응모 행사를 한다고 했을 때, 추천대상 목록을 얻을 수 있다.

2. '제품 만족도 조사'에서 3 번 질문에 '매우만족'으로 응답한 사람들의 수를 구하시오.

```
select count(respEmail)
from Respondent
where respEmail in (select respEmail
                    from Answer as A, Question as Q
                    where A.questionID = Q.questionID and Q.qnum=3 and surveyID
                    in (select surveyID
                        from survey
                        where sName = '제품 만족도 조사'))
```

- ⑨ 이 질의는 특정 설문조사의 특정 질문에서 사람들의 응답의 비율을 계산하는 기능을 구
현할 때 사용할 수 있다. 통계를 낼 때 필요하다.

3. '제품 만족도 조사'에 참여한 남성이 총 몇 명인지 구하라.

```
select count(respEmail)
from Respondent as R
where R.rGender = 'M' and respEmail in (select respEmail
                                         from participate as P, survey as S
                                         where P.survevID = S.survevID and S.sName = '제품
만족도 조사')
```

- ⑨ 이 질의는 특정 설문조사에서 남녀의 참여비율을 구하고 싶을 때 사용할 수 있다. 통계를
낼 때 필요하다.

4. '제품 만족도 조사'에 참여한 사람들의 평균나이를 구하라.

```
select avg(rAge)
from Respondent R
where respEmail in (select respEmail
                    from participate as P, survey as S
                    where P.survevID = S.survevID and S.sName = '제품 만족도 조사')
```

- ⑨ 이 질의는 설문조사 참여인원의 평균 연령대를 구하는 통계를 낼 때 사용할 수 있다.