



ICE4016 데이터베이스 설계

<제목>

MYSQL

보고서 작성 서약서

1. 나는 타학생의 보고서를 베끼거나 여러 보고서의 내용을 짜집기하지 않겠습니다.
2. 나는 보고서의 주요 내용을 인터넷사이트 등을 통해 얻지 않겠습니다.
3. 나는 보고서의 내용을 조작하지 않겠습니다.
4. 나는 보고서 작성에 참고한 문헌의 출처를 밝히겠습니다.
5. 나는 나의 보고서를 제출 전에 타학생에게 보여주지 않겠습니다.

나는 보고서 작성시 윤리에 어긋난 행동을 하지 않고 정보통신공학인으로서 나의 명예를 지킬 것을 맹세합니다.

2022년 9 월 25일

학부 정보통신공학

학년 3

성명 김민겸

학번 12201863

1. 개요

- STEP 1 : 아래의 요구사항을 만족하는 Inha 데이터베이스를 만들 것.
 - Inha 데이터베이스에는 Building, Room, Department, Class, Student 테이블이 존재
 - Page 21를 참고하여 본인이 생각하는 각 테이블의 참조관계를 추가할 것

Table Name	Attribute	Attribute	Attribute	Attribute	Attribute	Attribute	Attribute
Building	<u>Id</u>	Name					
Room	<u>Id</u>	Name	Capacity				
Department	<u>Id</u>	Name	Email	Phone number			
Class	<u>Id</u>	Name	Professor	Number of participants			
Student	<u>Id</u>	Name	Email	Phone number	Major	Student Id	

- STEP 2 : Inha 데이터베이스를 workbench 상에서 그리고 화면 캡처할 것.
- STEP 3 : Inha 데이터베이스의 각 테이블에 5개 이상의 값을 insert 할 것.
- STEP 4: 각 테이블의 설명(desc 명령어) 및 조회(select) 결과를 캡처할 것

❖ 보고서에 반드시 포함될 내용 (hwp/doc화일)

- STEP2, 4의 캡처 화면
- Inha 데이터베이스의 script파일(.sql)
 - Create table 구문, insert 구문 등이 포함
 - Script 를 실행했을 때 보고서와 동일한 테이블 및 값들이 만들어져야 함
- 제출마감
 - 9월 25일 23:59:59까지

2. 상세 설계 내용

1. Inha 데이터베이스의 각 테이블 설계 및 테이블 간 관계도

먼저 Building과 Room의 관계를 생각해봤을 때, 하나의 Building 안에 여러 개의 Room이 있는 것을 일반적으로 생각할 수 있다. 따라서 1:N의 관계임을 염두에 두었고, Room의 입장에서 Building이 없으면 Room은 존재할 수 없으므로 식별 관계로 매핑하였다.

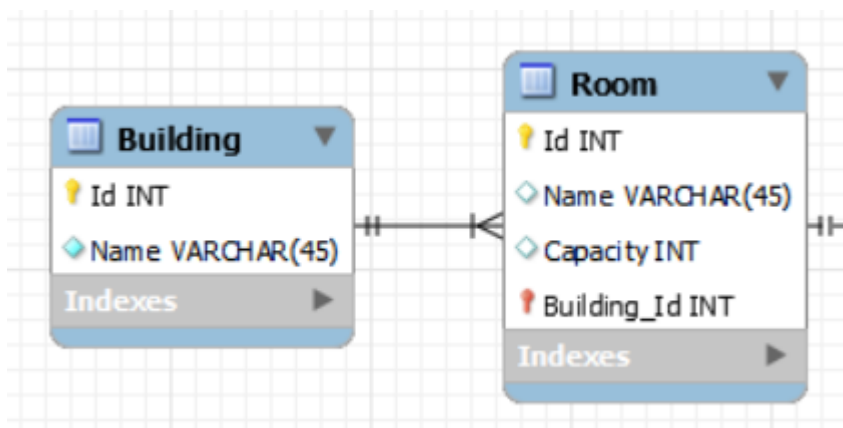
Building의 경우 아래와 같이 Id와 Name을 필드로 뒀고, Id는 PK, not null로 설정하였다. Building의 Id는 사용자가 마음대로 지정하여 사용할 수 있게 하기 위해 Auto Increment를 두지 않았다. Building 이름도 사용자 입장에서 부르기 편하게 꼭 존재해야 하는 것이므로 Not null로 설정하였다.

Relationship Building - Table										
Table Name: Building		Schema: Inha								
Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
Id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Name	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Room의 경우 아래와 같이 Id, Name, Capacity, Building Id 총 네 가지 필드로 설계하였다. Id의 Primary key이며 Not null이다. Room의 경우 데이터베이스에 등록할 때마다 하나씩 증가하도록 하는 것이 관리자 입장에서 편할 것이라 판단하여 Auto increment로 두었다. Name의 경우 이름이 없는 방이 있을 수 있고, Capacity도 방에 들어갈 수 있는 총 인원 수가 미파악된 교실이 있을 것이라 생각해서 Null이 가능한 상태로 두었다.

Table Name: Room										
Table Name: Room		Schema: Inha								
Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
Id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Capacity	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Building_Id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

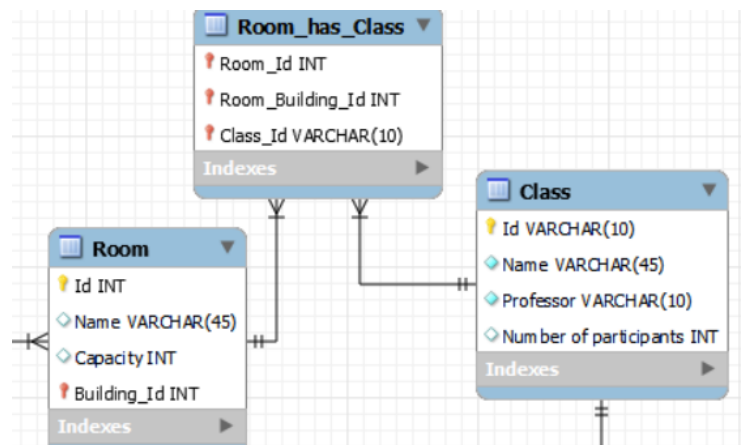
위에서 언급했듯이, Building과 Room은 하나의 Building에 여러 개의 Room이 있는 1:N 관계를 가지고 있기 때문에 아래와 같이 식별 1:N 선을 이용해서 테이블을 설계하였습니다.



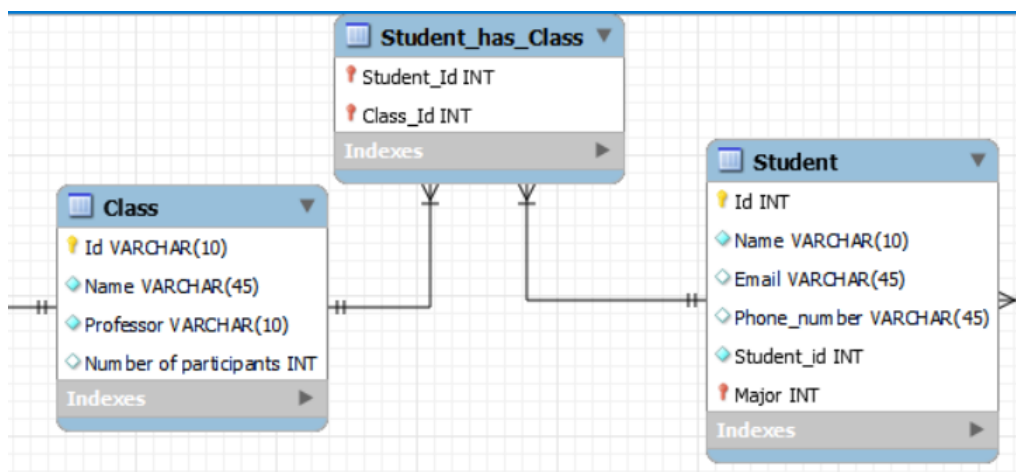
다음은 Class 테이블입니다. Class 테이블의 PK의 경우 우리 학교의 학수번호로 판단하여 Varchar(10)으로 Data Type을 지정하였다. 이에 따라 Auto Increment 속성은 제외하였다. Name은 수업의 이름이고, 이름이 없는 수업은 없으므로 Not null로 두었으며 Professor도 마찬가지로 이유이다. Number of participants의 경우 수업마다 아직 참여 인원이 조사되지 않은 경우도 있으며 수강신청 전에는 참여 인원을 알 수 없지만 class가 데이터베이스에 등록될 수 있으므로 null이 가능한 상태로 두었다.

Table Name: Class		Schema: Inha								
Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
Id	VARCHAR(10)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Name	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Professor	VARCHAR(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Number of participants	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

이어서 Room과 Class는 다대다 매핑을 하였는데, 하나의 강의실에 여러 강의가 당연히 있을 수 있고, 데이터베이스 설계와 같이 하나의 수업에 여러 Room을 사용하는 경우가 있을 수 있다. 따라서 N:N 매핑을 수행하였으며 이에 따라 매핑 테이블이 생성되었다.



Room과 Class를 연결해주는 매핑 테이블인 Room_has_Class는 Room의 PK와 FK(Building_Id), 그리고 Class의 PK를 담고 있다. 매핑 테이블의 Room의 Id가 포함되는 이유는 Room과 Building이 식별관계이기 때문에 매핑 테이블에서 포함되어 함께 제시되는 것이다. 이 테이블에 Room과 Class의 매핑 정보를 저장하여, 조회 시 하나의 Room에 들어 있는 여러 Class를, 하나의 Class에 들어있는 여러 Room을 조회할 수 있는 것이다.

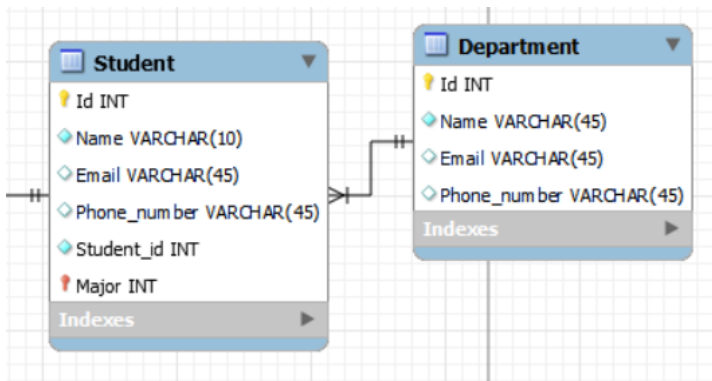


다음은 Student 테이블에 대한 설명이다.

먼저 필드들은 아래와 같이 정의하였다. Id의 경우 단순히 student를 조회하기 위해 사용하는 필드이므로 Data를 추가할 때마다 증가할 수 있도록 Auto Increment설정을 하였다. 또한 Name도 필수적인 필드이므로 Not null 설정을 하였다. Email과 Phone_number은 학생이 자유롭게 입력하는 정보이지만 중복될 수 없으므로 Unique를 설정하였다. Student_id라는 필드는 12201863과 같이 학교에서 고유하게 제공하는 번호로 판단하여 Not null, unique설정을 하였다. Major은 아래에서 자세하게 설명할 예정이지만, Department와 연관 관계를 맺어 FK로 지정하였다. 이 때 Major가 없는 학생은 없으므로 Not null로 설정하였으며 Major은 주전공 하나만 저장하는 것으로 가정하였다.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
Id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Name	VARCHAR(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Email	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Phone_number	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Student_id	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Major	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Student와 Department의 관계도는 아래와 같다. 먼저 Department는 Student와 1:N 관계를 갖는다. 하나의 학과에 여러 학생이 있기 때문이다. 물론 한 학생이 여러 학과를 전공할 수 있지만, 대체로 학생의 정보를 조회할 때 주전공 하나를 기준으로 조회하기 때문에 Major은 주전공으로 간주하여 Department와 연결하였다. 그리고 학생은 전공을 선택하지 않고는 학교에 입학할 수 없기 때문에 학과와 식별 관계를 가진다. 즉 Student에 하나의 데이터를 삽입할 때 존재하지 않는 Department를 Major로 두고 삽입할 수 없다는 것이다. 따라서 Major로 Primary key로 지정해서 식별 관계로 1:N 매핑을 하였다.



마지막으로 Department 테이블의 컬럼을 보면 아래와 같이 Id는 Auto Increment로 단순히 학과를 조회하기 위한 용도로 사용하였고, Name은 학과의 이름을 의미하므로 Unique로 설정하였다. 또한 학과 대표 메일과 전화번호도 타 학과의 것과 겹칠 수 없으므로 Unique로 설정하였으며 신설 학과의 경우 설정된 대표 메일이나 전화번호가 없을 수 있어 null이 가능한 상태로 두었다.

Table Name: <input type="text" value="Department"/>		Schema:								
Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
Id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Name	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Email	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Phone_number	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

3. 실행 화면

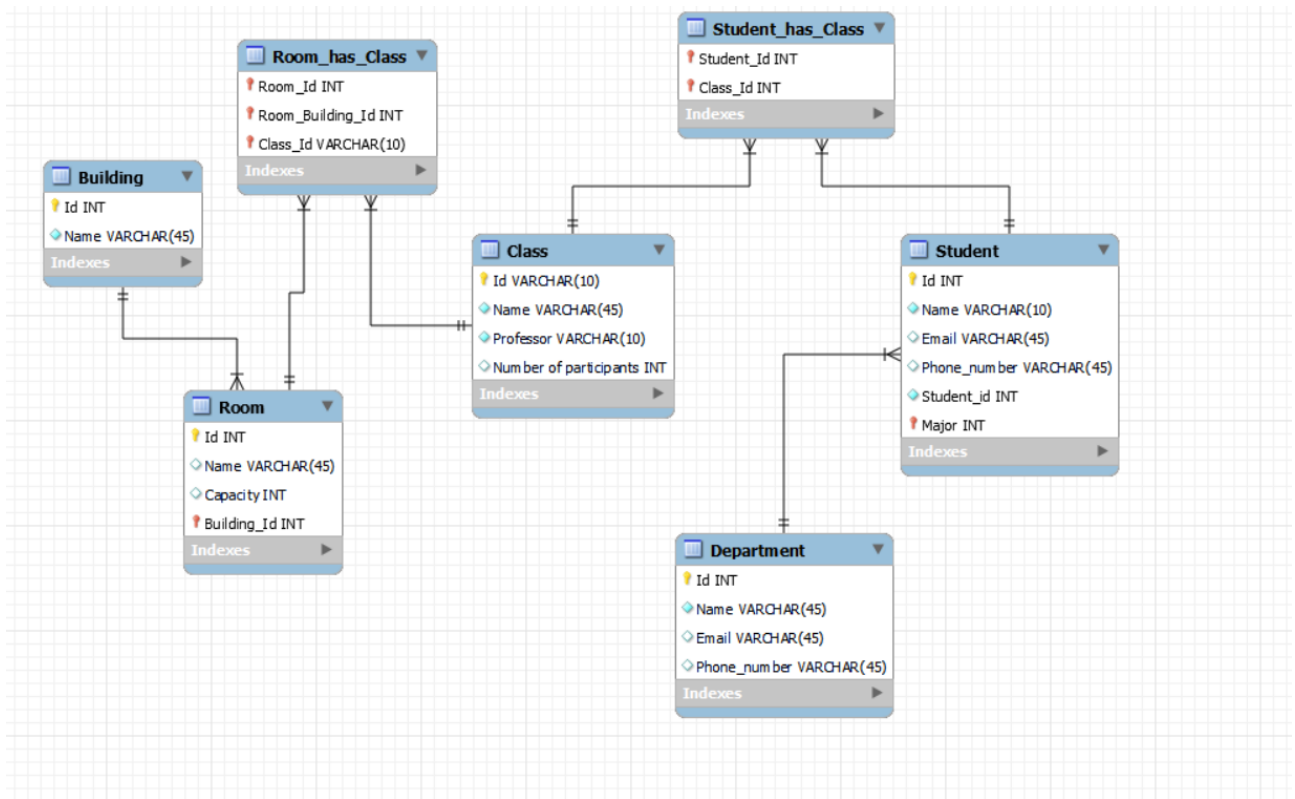


그림 1 데이터 베이스 전체 테이블

```

mysql> desc building;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Id     | int    | NO   | PRI | NULL    |       |
| Name   | varchar(45) | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.01 sec)
  
```

그림 2 Building 테이블 Description

```
mysql> select * from building;
```

Id	Name
1	본관
2	2호관
4	서호관
9	9호관
10	하이테크관

```
5 rows in set (0.00 sec)
```

그림 3 Building 테이블 Select

```
mysql> desc room
-> ;
```

Field	Type	Null	Key	Default	Extra
Id	int	NO	PRI	NULL	auto_increment
Name	varchar(45)	YES		NULL	
Capacity	int	YES		NULL	
Building_Id	int	NO	PRI	NULL	

```
4 rows in set (0.00 sec)
```

그림 4 Room 테이블 description

```
mysql> select * from room;
```

Id	Name	Capacity	Building_Id
1	정보통신공학과 학생회의실	NULL	10
2	002 강의실	40	4
3	하이테크 대강당	230	10
4	공과대학 행정실	NULL	2
5	252 강의실	100	9

```
5 rows in set (0.00 sec)
```

그림 5 Room 테이블 Select

```
mysql> desc class;
```

Field	Type	Null	Key	Default	Extra
Id	varchar(10)	NO	PRI	NULL	
Name	varchar(45)	NO		NULL	
Professor	varchar(10)	NO		NULL	
Number of participants	int	YES		NULL	

```
4 rows in set (0.00 sec)
```

그림 6 Class 테이블 Description

```
mysql> select * from class;
```

Id	Name	Professor	Number of participants
GEE4011	세상을 바꾸는 스타트업 이야기	허원창	320
ICE4008	컴퓨터 네트워크	유상조 교수님	50
ICE4010	이동통신	김덕경 교수님	45
ICE4016	Operating System	김기창 교수님	42
ICE4020	정보 보호론	김기창 교수님	30

```
5 rows in set (0.00 sec)
```

그림 7 Class 테이블 select

```
mysql> desc department;
```

Field	Type	Null	Key	Default	Extra
Id	int	NO	PRI	NULL	auto_increment
Name	varchar(45)	NO	UNI	NULL	
Email	varchar(45)	YES	UNI	NULL	
Phone_number	varchar(45)	YES	UNI	NULL	

```
4 rows in set (0.00 sec)
```

그림 8 Department 테이블 description

```
mysql> select * from department;
```

Id	Name	Email	Phone_number
9	정보통신공학과	ice@inha.ac.kr	032-860-7430
10	전기공학과	electrical@inha.ac.kr	032-860-7390
11	전자공학과	ee@inha.ac.kr	032-860-7410
12	컴퓨터공학과	cse@inha.ac.kr	032-860-7440
13	기계공학과	mech@inha.ac.kr	032-860-7300

```
5 rows in set (0.00 sec)
```

그림 9 Department 테이블 select


```
mysql> desc student;
```

Field	Type	Null	Key	Default	Extra
Id	int	NO	PRI	NULL	auto_increment
Name	varchar(10)	NO		NULL	
Email	varchar(45)	YES	UNI	NULL	
Phone_number	varchar(45)	YES	UNI	NULL	
Student_id	int	NO	UNI	NULL	
Major	int	NO	PRI	NULL	

6 rows in set (0.00 sec)

그림 10 Student 테이블 description

```
mysql> select * from student;
```

Id	Name	Email	Phone_number	Student_id	Major
8	김민경	mingyum119@naver.com	010-9479-0373	12201863	9
9	이슬	leeseul@inha.ac.kr	010-0000-1234	12201943	13
10	조한나	whgkssk@inha.edu	010-0000-0000	12201512	11
11	김다영	ralekdud@naver.com	010-0000-3465	12201212	10
12	최수빈	chltnqls@google.com	010-0000-2333	12201313	9

5 rows in set (0.00 sec)

그림 11 student 테이블 select

4. 결론

매핑 연관 관계의 결정 부분에서 많이 헷갈리고 고민하였던 것 같다. 결국에는 매핑을 할 때 하나의 Entity의 입장이 되어 관계를 생각해보면 답이 나왔던 것 같고, 식별 관계와 비식별 관계를 고민할 때도 이를 고려하였다.

```
145 • ALTER TABLE buliding CONVERT TO CHARSET UTF8;
146 • ALTER TABLE room CONVERT TO CHARSET UTF8;
147 • ALTER TABLE department CONVERT TO CHARSET UTF8;
148 • ALTER TABLE class CONVERT TO CHARSET UTF8;
149 • ALTER TABLE student CONVERT TO CHARSET UTF8;
```

또한 SQL 테스트 시 "Incorrect string value: 'WxEDWx98WxB8WxEAxB4Wx80' for column `inha`.`building`.`Name` at row 1"와 같은 에러 메시지가 떠서, 위의 사진처럼 ALTER TABLE로 UTF8 설정을 해 오류를 해결하였다.