

데이터베이스 정규화 과정

디자인 다듬기

- ⑤ 오류에 대한 디자인을 분석을 한다.
- 테이블을 만들고 샘플 데이터를 입력한 다음, <삭제>, <추가>, <수정> 등의 작업을 실행해서 데이터에 이상이 없는지 문제점이 발생하지 않는지를 확인한다.

1.3 데이터 정규화

- 설계한 테이블의 구조가 올바른지 확인하려면 데이터 정규화 규칙을 적용한다.
- 정규화(Normalization)
 - 관계형 데이터베이스를 디자인할 때 데이터의 중복을 최소화하도록 데이터를 구조화하는 처리 절차.
- 정규화의 기본 목표
 - 테이블 간에 관계를 재구성하여 작고 잘 조직된 관계를 생성.
- 정규화 된 테이블의 데이터
 - 추가, 삭제, 수정이 실행될 때 데이터 간에 불일치하거나 이상이 발생하거나 실행 상에 문제가 없게 된다.

1.3 데이터 정규화-필요성

강좌코드(PK)	강좌명	강사명	연락처	주소
E1	영어회화	홍철수	01023456789	서울 서초구 반포로10
E2	영어독해	홍철수	01023456789	서울 서초구 반포로10
J1	일본어	박영희	01044445555	안양시 동안구 23번지
C1	중국어회화	박길수	01089896767	세종시 도움로 15

DELETE

강좌 J1 폐강으로 데이터 삭제->강사에 대한 정보도 없어짐

UPDATE

홍철수의 주소 변경->
모두 변경하지 않으면 불일치 한 데이터 남아있음

INSERT

아직 강좌를 배정받지 못한 강사에 대해서는 추가 불가

1.3 데이터 정규화

- 정규화
 - 크고, 제대로 조직화되지 않은 테이블들과 관계들을 작고 잘 조직된 테이블과 관계들로 나누는 것을 포함.
- 정규화 과정
 - 제 1 정규화(1NF)부터 제 6 정규화(6NF) 까지 정의되었는데, 일반적으로 데이터베이스 테이블이 제 3 정규(3NF)화가 되었으면 정규화 되었다라고 한다.

제 1 정규형

- 테이블의 모든 행에서 컬럼(열)의 값은 1개만 있어야 한다.
- 테이블의 모든 컬럼(열)이 원자 값(Atomic Value)을 가질 때 제 1 정규 형 이라고 한다.
- 정규화 결과로 행의 수가 증가 될 수 있다.

제 1 정규형

수강생정보 테이블

번호	이름	수강과목	연락처	주소
S1	김철수	DB, JAVA	01023456789	서울 서초구 반포로10
S2	박영희	SQL	01056782345	인천시 남동구 110
S3	강기동	JSP, DB	01044445555	안양시 동안구 23번지
S4	이철희	SQL	01089896767	세종시 도움로 15



번호	이름	수강과목	연락처	주소
S1	김철수	DB	01023456789	서울 서초구 반포로10
S1	김철수	JAVA	01023456789	서울 서초구 반포로10
S2	박영희	SQL	01056782345	인천시 남동구 110
S3	강기동	JSP	01044445555	안양시 동안구 23번지
S3	강기동	DB	01044445555	안양시 동안구 23번지
S4	이철희	SQL	01089896767	세종시 도움로 15

제 2 정규 형

- 테이블의 모든 컬럼(열)이 완전 함수적 종속을 만족.
- 어떤 열의 값을 조건으로 또 다른 열을 검색할 때 한 개 값만 검색 된다면 완전 함수적 종속관계를 만족한다고 할 수 있다.

제 2 정규 형

학번	이름	과목코드	과목명	학점
201	김철수	C1	DB	A
201	김철수	C3	JAVA	B
212	박영희	C2	SQL	B
221	강기동	C1	DB	C
221	강기동	C2	SQL	B
231	이철희	C3	JAVA	B

- <학번>을 알고 있고 이 값을 조건으로 검색했을 때 1개 값을 반환하는 열은 <이름>이다. 따라서 <학번>과 <이름>은 완전 함수적 종속관계에 있다고 할 수 있다.
- 함수적 종속관계에서 <학번>을 **결정자** (Determinant)라고 하며 <이름>을 **종속자** (Dependent)라고 한다.

학번이 "201"이면 이름은 "김철수"
 학번이 "221"이면 이름은 "강기동"

- <학번> 값 조건으로 검색했을 때 <과목코드>는 여러 값이 반환될 수 있기 때문에 <학번>과 <과목코드>와는 완전 종속관계에 있지 않다.
 학번이 "231"이면 과목코드는 "C3"
 학번이 "201"이면 과목코드는 "C1"과 "C3"
- <과목명>은 <학번, 과목코드> 2가지 값 조건으로 검색하면 1개 값을 반환하지만 <학번, 과목코드>에서 <과목코드>만을 조건으로 검색해도 1개 값을 반환하므로 이렇게 부분적으로 종속 관계가 만족할 경우에는 <학번>과 <과목코드>는 분리해야 한다.
 학번과 과목코드가 "201, C1"이면 과목코드는 "DB"
 과목코드가 "C1"이면 과목명은 "DB"
- 그리고 <학번, 과목코드> 2가지 값 조건으로 검색했을 때 <학점>은 1개 값을 반환하므로, <학번, 과목코드>와 <학점>은 완전 종속관계를 만족한다.
 학번과 과목코드가 "201, C1"이면 학점은 "A"
 학번과 과목코드가 "231, C3"이면 학점은 "B"

제 2 정규 형

- 먼저, <학번>에 완전히 종속적인 컬럼(열)을 찾아보면 <이름>이 해당된다. 따라서 <학번>과 <이름>으로 구성된 테이블 1개를 추가한다.

<학번>, <이름>

- 그리고, <학번, 과목코드>를 분리해야 하므로 <과목코드>와 <과목코드>에 완전히 종속적인 <과목명>을 포함하는 테이블 1개를 만든다.

<과목코드>, <과목명>

- 마지막으로, <학번, 과목코드>에 종속적인 <학점>을 포함하는 테이블을 만든다.

<학번>, <과목코드>, <학점>

- 이 정규화 과정을 통하면 1개의 <성적> 테이블이 3개의 테이블로 분할된다.

학번	이름	과목코드	과목명	학점
201	김철수	C1	DB	A
201	김철수	C3	JAVA	B
212	박영희	C2	SQL	B
221	강기동	C1	DB	C
221	강기동	C2	SQL	B
231	이철희	C3	JAVA	B

제 2 정규 형

성적 테이블

학번	이름	과목코드	과목명	학점
201	김철수	C1	DB	A
201	김철수	C3	JAVA	B
212	박영희	C2	SQL	B
221	강기동	C1	DB	C
221	강기동	C2	SQL	B
231	이철희	C3	JAVA	B

<>을 알면 -> <>을 알 수 있다.

<학번> -> <이름>

<학번, 과목코드> -> <과목명>

<과목코드> -> <과목명>

<학번, 과목코드> -> <학점>



학생 테이블

학번	이름
201	김철수
201	김철수
212	박영희
221	강기동
221	강기동
231	이철희

과목 테이블

과목코드	과목명
C1	DB
C3	JAVA
C2	SQL
C1	DB
C2	SQL
C3	JAVA

성적 테이블

학번	과목코드	학점
201	C1	A
201	C3	B
212	C2	B
221	C1	C
221	C2	B
231	C3	B

제 3 정규 형

- 관계형 데이터 모델에서 기본 키 이외의 컬럼(열)들 간에 함수적 종속성 (Dependency)을 가지지 않도록 정리하는 절차.
- 기본 키로 정의된 컬럼(열)이외의 컬럼(열)들 사이에 함수적 종속성이 있는지 확인해서 제거한다.

사번(PK)	이름	팀명	팀내선번호
912	김철수	개발팀	912
921	박길동	자재팀	913
913	심영희	인사팀	915
914	최영수	인사팀	915
985	강철희	기획팀	917
986	이길수	개발팀	912

제 3 정규 형

- <사번>이 기본 키(Primary Key)이고 <팀명>은 <사번>에 종속되어 있다.
- 그러나 <팀내선번호>는 기본 키가 아닌 <팀명>에 완전한 함수적 종속관계를 가지고 있는 것을 볼 수 있다.

사번(PK)	이름	팀명	팀내선번호
912	김철수	개발팀	912
921	박길동	자재팀	913
913	심영희	인사팀	915
914	최영수	인사팀	915
985	강철희	기획팀	917
986	이길수	개발팀	912

사번이 "912" -> 팀명은 "개발팀"

팀명이 "개발팀" -> 팀내선번호는 "912"

- 따라서, 기본 키가 아닌 <팀명>과 <팀명>에 완전한 함수적 종속관계를 가지고 있는 <팀내선번호>를 포함하는 새로운 테이블로 만든다.

<팀명>, <팀내선번호>

제 3 정규 형

사원 테이블

사번(PK)	이름	팀명	팀내선번호
912	김철수	개발팀	912
921	박길동	자재팀	913
913	심영희	인사팀	915
914	최영수	인사팀	915
985	강철희	기획팀	917
986	이길수	개발팀	912

<>을 알면 -> <>을 알 수 있다.

<사번> -> <팀명>

<팀명> -> <팀내선번호>

사원 테이블

사번(PK)	이름	팀명
912	김철수	개발팀
921	박길동	자재팀
913	심영희	인사팀
914	최영수	인사팀
985	강철희	기획팀
986	이길수	개발팀

팀명	팀내선번호
개발팀	912
자재팀	913
인사팀	915
인사팀	915
기획팀	917
개발팀	912

팀정보 테이블

팀명	팀내선번호
개발팀	912
자재팀	913
인사팀	915
기획팀	917