

데이터베이스 프로그래밍

데이터베이스(학습모듈 p.33~)

■ ERD를 작성하고 정규화 수행하기

수행 순서

1. ERD를 작성한다.

1). ERD 작성을 위한 요소를 추출한다.

- ✓ 업무의 흐름으로부터 개체, 관계, 속성을 구분해 내야 한다. 다음 시나리오에서 세 가지 요소를 추출한 뒤 <표 2-6>의 결과와 비교한다.

시나리오

- (1) 보험사에서 사용하는 데이터베이스를 설계한다.
- (2) 고객이 있고 고객이 사용하는 계좌가 있다.
- (3) 고객에 대한 정보로는 이름, 주소, 휴대 전화번호, 주민 등록 번호가 있다.
- (4) 고객 계좌에 대한 정보로는 계좌 번호, 계좌 종류, 잔고, 개설 일자, 인출 한도가 있다.
- (5) 주소 정보에 대한 정보로는 시군구, 동, 상세 주소, 우편 번호가 있다.

데이터베이스(학습모듈 p.33~)

■ ERD를 작성하고 정규화 수행하기

<표 2-6> 업무로부터 개체, 관계, 속성 추출

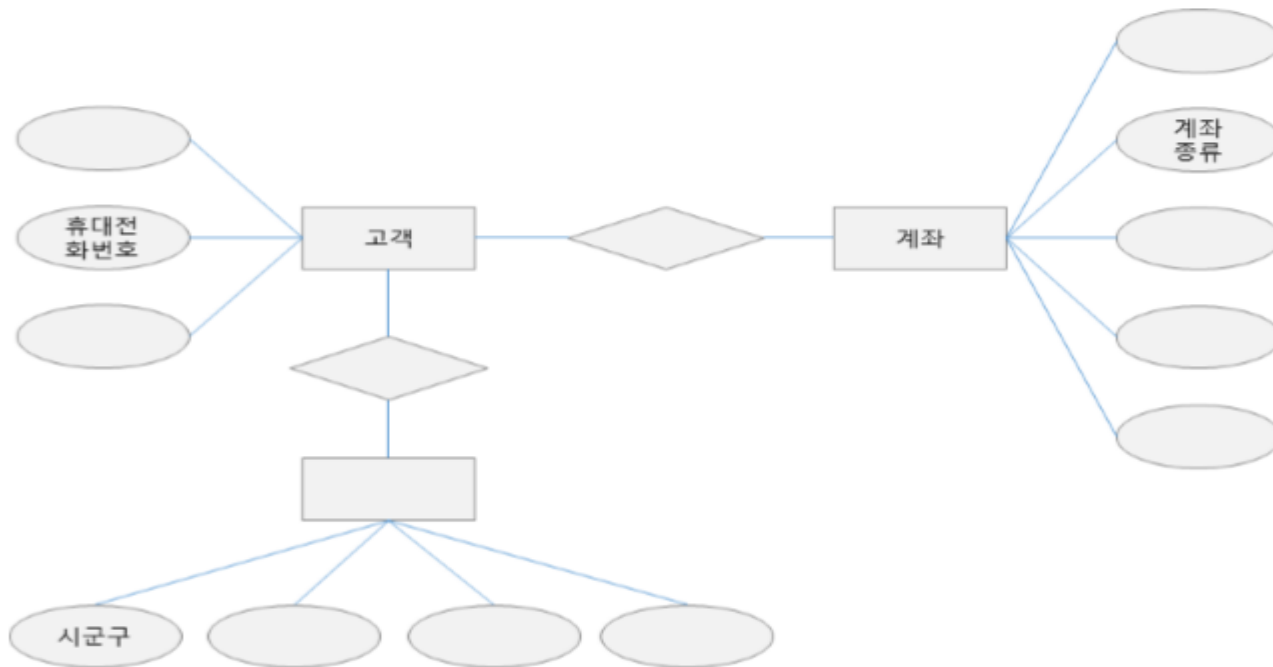
구분	추출 요소
개체	고객, 계좌, 전화 번호
속성	이름, 휴대 전화번호, 주민 등록 번호, 계좌 번호, 계좌 종류, 잔고, 개설 일자, 인출 한도, 시군구, 동, 상세 주소, 우편 번호
관계	고객이 계좌 번호를 소유한다. 고객이 주소를 소유한다.

데이터베이스(학습모듈 p.33~)

■ ERD를 작성하고 정규화 수행하기

2). ERD를 작성한다.

- ✓ 추출된 요소를 가지고 ERD를 작성한다. 일부 작성된 ERD에 추출 요소들을 입력한다



[그림 2-4] ERD 작성 예제

데이터베이스(학습모듈 p.33~)

■ ERD를 작성하고 정규화 수행하기

2.ERD 관계형 스키마를 작성한다.

- ✓ 작성된 ERD에 다음의 추가적인 제약 조건을 고려하여 논리 관계형 스키마를 작성한다.

제약 조건

- (1) 고객은 다수의 집을 보유할 수 있다.
- (2) 고객은 다수의 계좌를 보유할 수 있다.

데이터베이스(학습모듈 p.33~)

■ ERD를 작성하고 정규화 수행하기

2.ERD 관계형 스키마를 작성한다.

1). 관계형 스키마 작성을 위해 테이블 형태로 표시한다.

- ✓ 테이블의 가장 위 칸은 유일하게 필드를 구분할 수 있는 구분자(유일키)가 되어야 한다.
- ✓ 유일키를 만드는 방법은 여러 가지가 있으며 일련번호를 부여하기도 한다.

데이터베이스(학습모듈 p.33~)

■ ERD를 작성하고 정규화 수행하기

2.ERD 관계형 스키마를 작성한다.

1). 관계형 스키마 작성을 위해 테이블 형태로 표시한다.

✓ 유일키 아래 칸에는 속성들을 기입한다.

주소
일련번호
시군구 동 상세주소 우편번호

고객
고객번호
이름 주소 휴대전화번호 주민등록번호

계좌번호
일련번호
계좌번호 계좌종류 잔고 개설일자 인출한도

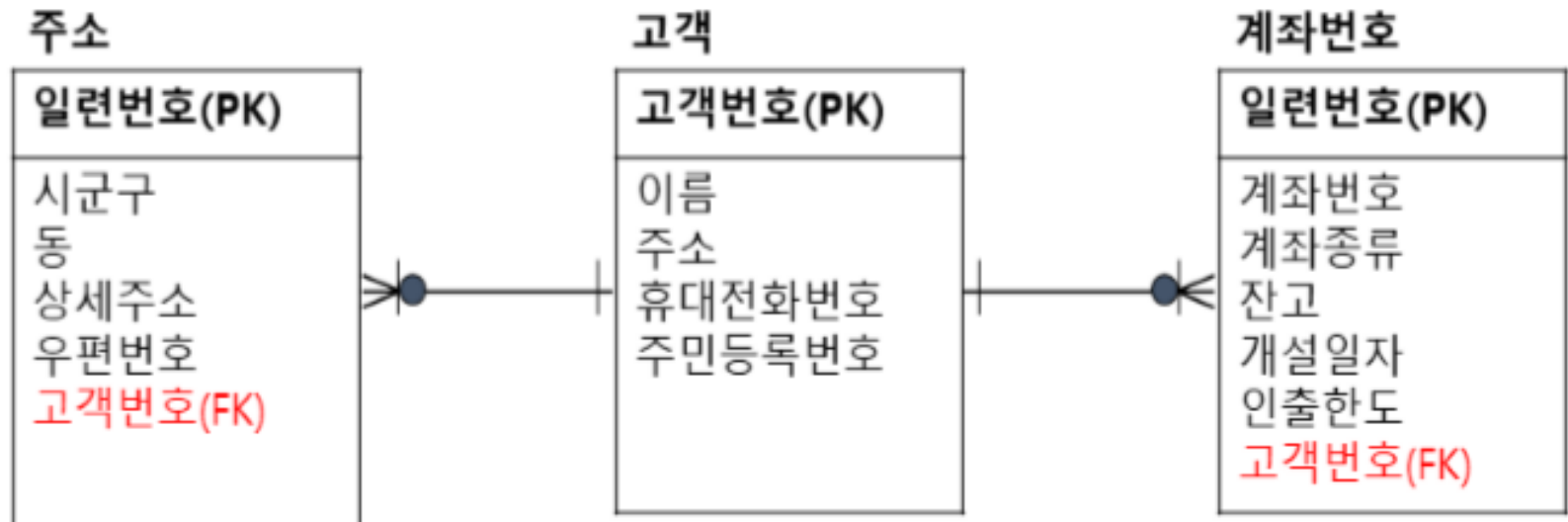
데이터베이스(학습모듈 p.33~)

■ ERD를 작성하고 정규화 수행하기

2. ERD 관계형 스키마를 작성한다.

2). 테이블 간 관계를 설정한다.

- ✓ 테이블 간 관계를 표시한다. 표기 시에는 1:1, 1:N, N:1, N:N의 관계를 고려해야 한다.
- ✓ 테이블 간 연결할 수 있는 속성을 부여하여 하나의 테이블에서 다른 테이블의 값을 찾아갈 수 있어야 한다.



[그림 2-6] ERD 관계 설정

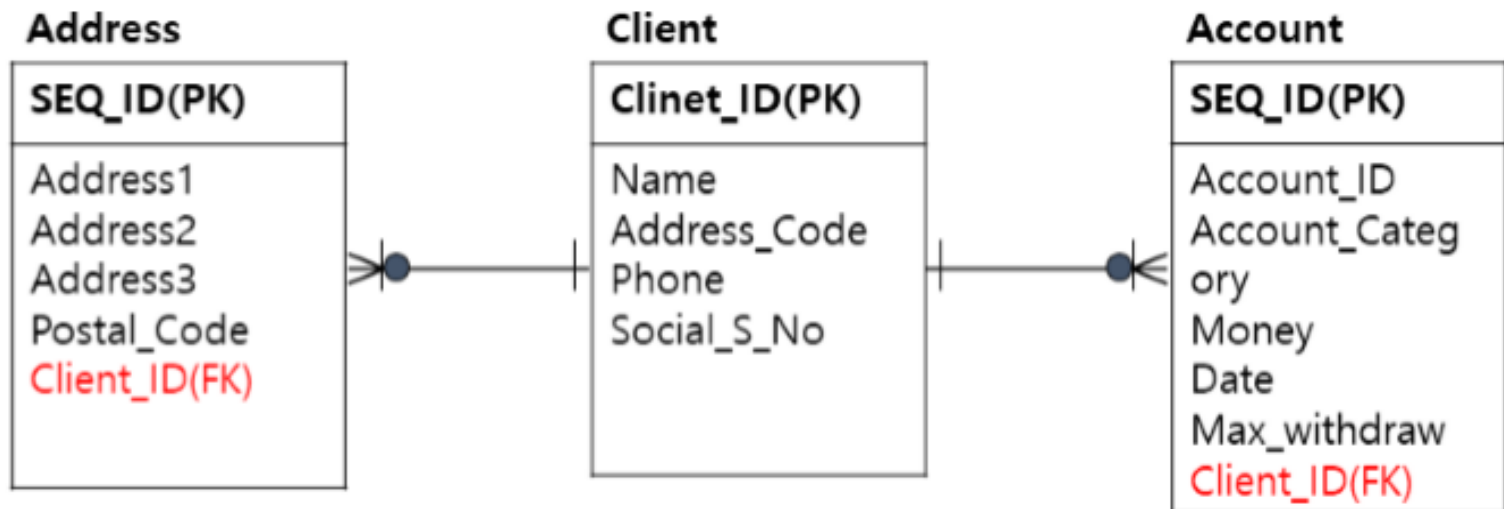
데이터베이스(학습모듈 p.33~)

■ ERD를 작성하고 정규화 수행하기

3. 관계형 데이터베이스 테이블을 생성한다.

1). 관계형 테이블 생성을 위한 물리 스키마를 작성한다.

- ✓ SQL 표준 언어는 한글 입력이 허용되지 않는다. 위 논리 스키마와 매칭되는 물리 스키마를 작성한다.



[그림 2-7] ERD 물리 모델 작성

데이터베이스(학습모듈 p.33~)

■ ERD를 작성하고 정규화 수행하기

3. 관계형 데이터베이스 테이블을 생성한다.

2). 테이블 생성을 위한 SQL 명령어를 작성한다.

✓ 작성 시 필요한 datatype은 정수형, 실수형, 날짜형 등 다양하다. SQL 표준을 참조하여 필요로 하는 데이터 형을 입력하도록 한다.

<표 2-7> 관계형 테이블 생성 예제

SQL 명령어	예제
<pre>CREATE TABLE table_name(column1 datatype, column2 datatype, );</pre>	<pre>CREATE TABLE Address(Address1 varchar(255), Address2 varchar(255), Address3 varchar(255), Postal_Code varchar(255), Client_ID varchar(255));</pre>

데이터베이스(학습모듈 p.33~)

■ ERD를 작성하고 정규화 수행하기

4. ERD 작성을 위한 자동화 툴을 사용하여 생산성을 높인다.
- ✓ ERD를 종이나 파워포인트 같은 툴로 그리는 것은 한계가 있다.
무료로 배포되는 ERD 작성 툴을 활용하여 ERD를 작성해 본다.
DbDesigner와 같은 무료 소프트웨어를 사용하면 생산성을 높일 수 있는데, 무료로 배포되고 있는 소프트웨어가 많으므로 자신에게 적합한 툴을 선택하여 사용한다.

2-3. 데이터베이스 관리 (학습모듈 p.37~)

1. 데이터베이스 기본 연산

- ✓ CRUD는 데이터베이스가 가지는 기본적인 데이터 처리 기능인 Create(생성), Read(읽기), Update(갱신), Delete(삭제)를 말한다.

<표 2-8> CRUD 기본 처리 SQL 명령어

기본 처리	SQL	설명
Create	Insert	테이블 내 칼럼에 데이터를 추가한다.
Read	Select	테이블 내 칼럼에 저장된 데이터를 불러온다.
Update	Update	테이블 내 칼럼에 저장된 데이터를 수정한다.
Delete	Delete	테이블 내 칼럼에 저장된 데이터를 삭제한다.

2-3. 데이터베이스 관리 (학습모듈 p.37~)

1. 데이터베이스 관리하기

수행 순서

1. 데이터베이스 기본 연산을 수행한다.

1). SQL 표준을 준수하여 저장되어 있는 데이터에 대하여 Insert(삽입), Read(읽기), Update(갱신), Delete(삭제) 명령어를 작성한다.

SEQ_ID	Address1	Address2	Address3	Postal_Code	Client_ID
1	서울특별시	강남구 삼성동 44번지	A아파트 1동 101호	50693	321
2	경기도	성남시 분당구 정자동 66번지	B아파트 5동 111호	12301	441
3	서울특별시	강남구 삼성동 64길	C아파트 3동 190호	50693	216
4	경기도	성남시 분당구 정자동 44길	D아파트 5동 111호	12301	435

[그림 2-8] Address 테이블 내 저장되어 있는 데이터 예제

2-3. 데이터베이스 관리 (학습모듈 p.37~)

데이터베이스 관리하기

수행 순서

1. 데이터베이스 기본 연산을 수행한다.

(1) 데이터 삽입 명령어를 작성한다. 테이블에서 데이터를 읽기 위해서는 "Insert" 명령어를 사용한다.

<표 2-10> 관계형 테이블 데이터 삽입 예제

SQL 명령어	예제
INSERT INTO table_name VALUES(value1, value2, value3, ...);	INSERT INTO Address VALUES('서울특별시', '성동구 XXehd', '대신아파트 1동 101호', '09100', '121');

2-3. 데이터베이스 관리 (학습모듈 p.37~)

데이터베이스 관리하기

수행 순서

1. 데이터베이스 기본 연산을 수행한다.

(2) 데이터 읽기 명령어를 작성한다. 테이블에서 데이터를 읽기 위해서는 "Select" 명령어를 사용한다. Select 다음 * 표기를 사용하면 모든 데이터를 읽어 오고 칼럼명을 지정하면 특정 칼럼만을 읽어 온다.

<표 2-11> 관계형 테이블 생성 예제

SQL 명령어	예제
SELECT column1, column2, ... FROM table_name;	(1) SELECT * FROM Address; (2) SELECT SEQ_ID, address1 FROM Address;

2-3. 데이터베이스 관리 (학습모듈 p.37~)

데이터베이스 관리하기

수행 순서

1. 데이터베이스 기본 연산을 수행한다.

(3) 데이터 갱신 명령어를 작성한다. 테이블에서 데이터를 갱신하기 위해서는 "Update" 명령어를 사용한다. where 조건절을 사용하여 업데이트할 데이터를 특정해야 한다.

<표 2-12> 관계형 테이블 생성 예제

SQL 명령어	예제
UPDATE table_name SET column1 = value1, column2 = value2, ... WHERE condition;	UPDATE Address SET address1 = '제주도', address2 = '서귀포시 색달동', address3 = '청아아파트 111동 11호' WHERE SEQ_ID = 1;

2-3. 데이터베이스 관리 (학습모듈 p.37~)

데이터베이스 관리하기

수행 순서

1. 데이터베이스 기본 연산을 수행한다.

(3) 데이터 갱신 명령어를 작성한다.

✓ 다음 두 문장을 사용했을 때 update에 대한 결과에 어떤 차이점이 있는지 기록한다.

- UPDATE Address SET address1 = '제주도', address2 = '서귀포시 색달동', address3 = '청아아파트 111동 11호' WHERE SEQ_ID = 1;
- UPDATE Address SET address1 = '제주도', address2 = '서귀포시 색달동', address3 = '청아아파트 111동 11호' WHERE address1 = '경기도';

2-3. 데이터베이스 관리 (학습모듈 p.37~)

데이터베이스 관리하기

수행 순서

1. 데이터베이스 기본 연산을 수행한다.

(4) 데이터 삭제 명령어를 작성한다. 테이블에서 데이터를 삭제하기 위해서는 "Delete" 명령어를 사용한다. where 조건절을 사용하여 삭제할 데이터를 특정해야 한다

<표 2-13> 관계형 테이블 생성 예제

SQL 명령어	예제
DELETE FROM table_name WHERE condition;	DELETE FROM Address WHERE Address1='서울특별시';

2-3. 데이터베이스 관리 (학습모듈 p.37~)

데이터베이스 관리하기

수행 순서

1. 데이터베이스 기본 연산을 수행한다.

((5) SQL 기타 명령어 사용

- ALTER DATABASE 명령어를 사용하여 데이터베이스를 수정한다.
- ALTER TABLE 명령어를 사용하여 테이블 구조를 수정한다.
- DROP TABLE 명령어를 사용하여 테이블을 전체 삭제한다.
- CREATE INDEX 명령어를 사용하여 테이블 내 데이터의 검색 속도를 향상시킬 수 있는 인덱스를 생성한다.
- DROP INDEX 명령어를 사용하여 인덱스를 삭제한다.
- JOINT 기능을 사용하여 복수의 테이블로부터 데이터를 조합하여 가져온다.
- 그 외 UNION, GROUP BY 등의 다양한 명령어를 활용한다.