

4. 좋은 테이블 설계 정규화는 왜 하죠?



목차

1. 두개의 테이블
2. 원자적 데이터
3. 정규화
4. 기본키 규칙
5. 테이블을 보여주세요.
6. 테이블 생성
7. 테이블에 기본키 추가 하기
8. SQL 도구 상자



1. 두개의 테이블

1

2

3

4

5

6

7

8

마크 테이블

잭 테이블

common	species	location	weight
bass, largemouth	M. salmoides	Montgomery Lake, GA	22 lb 4 oz
walleye	S. vitreus	Old Hickory Lake, TN	25 lb 0 oz
trout, cutthroat	O. Clarki	Pyramid Lake, NV	41 lb 0 oz
perch, yellow	P. Flavescens	Bordentown, NJ	4 lb 3 oz
bluegill	L. Macrochirus	Ketona Lake, AL	4 lb 12 oz
gar, longnose	L. Osseus	Trinity River, TX	50 lb 5 oz
crappie, white	P. annularis	Enid Dam, MS	5 lb 3 oz
pickerel, grass	E. americanus	Dewart Lake, IN	1 lb 0 oz
goldfish	C. auratus	Lake Hodges, CA	6 lb 10 oz
salmon, chinook	O. Tshawytscha	Kenai River, AK	97 lb 4 oz

first_name	last_name	common	location	state	weight	date
George	Perry	bass, largemouth	Montgomery Lake	GA	22 lb 4 oz	6/2/1932
Mabry	Harper	walleye	Old Hickory Lake	TN	25 lb 0 oz	8/2/1960
John	Skimmerhorn	trout, cutthroat	Pyramid Lake	NV	41 lb 0 oz	12/1/1925
C.C.	Abbot	perch, yellow	Bordentown	NJ	4 lb 3 oz	5/1/1865
Ts	Hudson	bluegill	Ketona Lake	AL	4 lb 12 oz	4/9/1950
Fred	Bright	crappie, white	Trinity River	TX	50 lb 5 oz	7/30/1954
Mike	Berg	pickerel, grass	Enid Dam	MS	5 lb 3 oz	7/31/1957
Florentino	Abena	goldfish	Dewart Lake	IN	1 lb 0 oz	6/9/1990
Les	Anderson	salmon, chinook	Lake Hodges	CA	6 lb 10 oz	4/17/1996
Les	Anderson	salmon, chinook	Kenai River	AK	97 lb 4 oz	5/17/1985

데이터를 어떻게 사용할 것인가??

충분히 쪼개어졌다. = 충분히 원자적이다.

※ 테이블 생성 시 고려할 점

1. 테이블로 표현하려는 것을 선택 (무슨 테이블 - 물고기 리스트)
2. 테이블로부터 얻어야 하는 정보들의 리스트 작성 (데이터 - 물고기 정보)
3. 테이블을 만들 정보들을 조각으로 나눔 (열 - 종류, 지역, 무게 등 ...)



2. 원자적 데이터

1

2

3

4

5

6

7

8

※ 원자 : 쪼갤 수 없는 가장 작은 조각

※ 고려해야 할 점

1. 테이블이 표현하는 것이 무엇인가?
2. 그것을 얻기 위해 어떻게 사용할 것인가?
3. 각각의 열들은 쿼리를 짧게 할 수 있는 원자적 데이터가 있나?

→ 필요한 한도 내에서 가능하면 작은 조각으로 나눠라!

※ 원자적 데이터 규칙

규칙1 : 열은 그 열에 같은 타입의 데이터를 여러 개 가질 수 없다.

위반

음식	재료
빵	밀가루, 우유, 요거트

규칙2 : 같은 타입의 데이터를 여러 열에 가질 수 없다.

위반

선생님	학생1	학생2	학생3
김선생	모씨	이씨	한씨



3. 정규화

1

2

3

4

5

6

7

8

※ 정규화란

관계형 데이터베이스의 설계에서 중복을 최소화하게 데이터를 구조화하는 프로세스

※ 정규 테이블의 장점

데이터베이스의 크기를 줄여줍니다.
데이터가 적어 쿼리가 더 빨라집니다.

학번	이름	수강과목
1	모씨	수학
2	김씨	과학
3	한씨	영어
3	한씨	수학
3	한씨	과학

학번	이름
1	모씨
2	김씨
3	한씨

수강과목
수학
과학
영어

※ 정규화 첫 단계 : 제1정규화 (1NF 1 normal form)

각 행의 데이터들은 **원자적 값**을 가져야 한다.

각 행은 유일무이한 식별자인 **기본키** (primary key) 를 가지고 있어야 한다.



4. 기본키 규칙

1

2

3

4

5

6

7

8

1. 각 레코드를 다른 레코드와 구분하는 열로 중복 될 수 없다.
2. Null 이 될 수 없다.
3. 기본키는 레코드가 삽입될 때 값이 있어야 한다.
4. 기본키는 간결해야 한다.
5. 기본키는 변경 불가 하다.



5. 테이블을 보여주세요.

1

2

3

4

5

6

7

8

※ SHOW 명령어

Create table 생성 **코드**를 보려고 할 때
SHOW CREATE TABLE my_contacts;

```
mysql> show create table drink_info;
```

```
| Table      | Create Table
```

```
drink_info | CREATE TABLE `drink_info` (  
  `drink_name` varchar(16) DEFAULT NULL,  
  `cost` decimal(2,1) DEFAULT NULL,  
  `carbs` decimal(3,1) DEFAULT NULL,  
  `color` varchar(20) DEFAULT NULL,  
  `ice` varchar(4) DEFAULT NULL,  
  `calories` int(4) DEFAULT NULL,  
  ENGINE=InnoDB DEFAULT CHARSET=latin1 |
```

백틱

지정하지 않은 값에
대해서는 default
값이 지정

기타 SHOW 명령어 이용
모든 열을 표시하고 열의 데이터 타입과 열의 세부사항 표시
SHOW COLUMNS FROM tablename;

데이터가 어떻게 저장되고
문자열은 무엇을
사용할지를 표시

데이터베이스를 생성하는 명령 표시
SHOW CREATE DATABASE databasename;

인덱싱되어 d는 열과 무슨 타입의 인덱스를 가지고 있는지를 표시
SHOW INDEX FROM tablename;

SQL 명령이 경고 메시지를 표시한다면 이것을 사용해서 실제 경고 메시지 확인
SHOW WARNINGS;



6. 테이블 생성

1

2

3

4

5

6

7

8

※ 자동 증가하는 기본키가 있는 테이블 생성 코드

```
CREATE TABLE my_contacts
```

```
(
```

```
Contact_id INT NOT NULL AUTO_INCREMENT,
```

```
Last_name VARCHAR(30),
```

```
....
```

```
Interests VARCHAR(100),
```

```
Seeking VARCHAR(100),
```

```
PRIMARY KEY(contact_id)
```

```
)
```

AUTO_INCREMENT

- 1에서 시작해서 새로운 열을 추가 할 때마다 1씩 자동으로 추가
- null 을 무시하고 값을 넣음
- 임의로 값을 넣는 경우 임의 값으로 덮어 씌움

기본키 지정



7. 테이블에 기본키 추가하기

1

2

3

4

5

6

7

8

※ 기존에 존재 하는 테이블에 기본키 열 추가 코드

```
ALTER TABLE my_contacts  
ADD COLUMN contact_id INT NOT NULL AUTO_INCREMENT FIRST,  
ADD PRIMARY KEY (contact_id);
```

새 열을 열 리스트의
첫 번째 열로 하라는
키워드

기본키 지정

테이블에 새로운 열 추가

ALTER TABLE [테이블 명] ADD COLUMN [컬럼명] [컬럼 데이터형]

※ 테이블 데이터 모두 보는 코드

```
SELECT * FROM my_contacts;
```



8. SQL 도구 상자

1

2

3

4

5

6

7

8



감사합니다!
2015.12.19 권진희