

# 11 제약조건

---



# Part 1.

---

테이블 생성



# 01 DB/Table 생성

## 기본 DB 생성

```
CREATE DATABASE testDB;  
use testDB;
```

## 기본 Table 생성

```
create table 테이블명(열이름 자료형 옵션1 옵션2 , 열이름 자료형 옵션1 옵션 2..);
```

ex)

```
create table testTbl(num int auto_increment primary key, name char(10) ..);
```

# 01 DB/Table 생성

## 관계형 테이블 만들기(부모테이블 - 기본키)

```
CREATE TABLE 테이블명(  
    열이름 자료형 PRIMARY KEY,  
    ...  
);
```

## 관계형 테이블 만들기(자식테이블 - 외래키)

```
CREATE TABLE buyTbl(  
    열이름 자료형 FOREIGN KEY REFERENCES userTbl(userID),  
    ...  
);
```

# 01 DB/Table 생성

## userTbl - 본 열이름 지정

```
create database testdb;
use testdb;
CREATE TABLE userTbl(
  userID char(8) NOT NULL PRIMARY KEY,
  name varchar(10)NOT NULL,
  birthYear int NOT NULL,
  addr char(2)NOT NULL,
  mobile1 char(3) NULL,
  mobile2 char(8) NULL,
  height smallint NULL,
  mDate date NULL
);
```

## buyTbl-기본 열이름 지정

```
CREATE TABLE buyTbl(
  num int auto_increment PRIMARY KEY,
  userid char(8) ,
  prodName char(6)NOT NULL,
  groupName char(4) NULL,
  price int NULL,
  amount smallint NOT NULL
,FOREIGN KEY (userid) REFERENCES usertbl(userid)
);
```

## 04 값 삽입 01

### userTbl 에 값 삽입

```
INSERT INTO userTbl VALUES('LSG',N'이승기',1987,N'서울','011','1111111',182,'2008-8-8');  
INSERT INTO userTbl VALUES('KBS',N'김범수',1979,N'경남','011','2222222',173,'2012-4-4');  
INSERT INTO userTbl VALUES('KHH',N'김경호',1971,N'전남','019','3333333',177,'2007-7-7');
```

# 00 중간문제

[문제]  
다음 테이블을 SQL문을 통해 생성해 보세요  
testDb에서 진행합니다

[고객주문테이블(customerTbl)]

| num | userid | name | addr | grade | phone        | order(FK) | amount |
|-----|--------|------|------|-------|--------------|-----------|--------|
| 1   | KHC    | 김희철  | 서울   | 일반    | 010-111-1111 | 신발        | 2      |
| 2   | LYJ    | 이영자  | 부산   | 실버    | 010-222-2222 | 가방        | 5      |
| 3   | PHS    | 박호신  | 대구   | 골드    | 010-333-3333 | 모자        | 1      |
| 4   | IU     | 아이유  | 속초   | 다이아   | 010-444-4444 | 신발        | 1      |
| 5   | SNE    | 손나은  | 울산   | 일반    | 010-555-5555 | 아우터       | 3      |

[제품테이블(productTbl)]

| code | prodName(PK) | price  | count |
|------|--------------|--------|-------|
| 111  | 신발           | 20000  | 100   |
| 222  | 가방           | 100000 | 50    |
| 333  | 모자           | 15000  | 30    |
| 444  | 티셔츠          | 10000  | 20    |
| 555  | 아우터          | 200000 | 155   |

## 04 답

### [제품테이블(productTbl) - Create table

use testDB;

```
CREATE TABLE productTbl(  
    code INT NOT NULL,  
    prodName CHAR(10) PRIMARY KEY,  
    price INT NOT NULL,  
    count INT NOT NULL  
);
```



## 04 답

### [고객주문테이블(customerTbl) - Create table

```
CREATE TABLE customerTbl(  
    num INT auto_increment primary key,  
    userid CHAR(3) NOT NULL,  
    name VARCHAR(20) NOT NULL,  
    addr CHAR(4) NOT NULL,  
    grade CHAR(4) NOT NULL,  
    phone VARCHAR(30) NOT NULL,  
    orderN CHAR(10) NOT NULL,  
    amount INT NOT NULL,  
    FOREIGN KEY(orderN) REFERENCES productTbl(prodName)  
);
```

## 04 답

### [제품테이블(productTbl) - INSERT

```
INSERT INTO productTbl VALUES(111,'신발',20000,100);  
INSERT INTO productTbl VALUES(222,'가방',100000,50);  
INSERT INTO productTbl VALUES(333,'모자',15000,30);  
INSERT INTO productTbl VALUES(444,'티셔츠',10000,20);  
INSERT INTO productTbl VALUES(555,'아우터',200000,155);
```

### [고객주문테이블(customerTbl) - INSERT

```
INSERT INTO customerTbl VALUES(null,'KHC','김희철','서울','일반','010-111-1111','신발',2);  
INSERT INTO customerTbl VALUES(null,'LYJ','이영자','부산','일반','010-222-2222','가방',5);  
INSERT INTO customerTbl VALUES(null,'PHS','박호신','대구','일반','010-333-3333','모자',1);  
INSERT INTO customerTbl VALUES(null,'IU','아이유','속초','일반','010-444-4444','신발',1);  
INSERT INTO customerTbl VALUES(null,'SNE','손나은','울산','일반','010-555-5555','아우터',3);
```

# Part 2.

---

계약 조건



# 01 제약조건

## 제약조건

특정 데이터를 입력할 때 무조건적으로 저장되는 것이 아닌, 어떠한 조건을 만족했을 때에 입력되도록 제약하는 옵션  
데이터의 무결성을 지키기 위한 제한된 조건

## 종류

Primary Key 제약 조건(관계형성관련 제약)  
Foreign Key 제약 조건(관계형성관련 제약)  
UNIQUE 제약 조건  
CHECK 제약 조건(MySQL 8.0.16부터 정식 지원)  
DEFAULT 정의  
NULL 값 허용

## 02 기본키 제약조건

### 기본키 제약 조건

행의 데이터를 구분할 수 있는 식별자열  
값은 중복될수 없다  
NULL을 허용하지 않는다  
관계 테이블 형성시 사용

### 사용

```
create database testdb2;  
use testdb2;  
create table usertbl(  
  userid char(10) primary key,  
  name char(10) not null,  
  birthyear int not null  
);  
desc usertbl;
```

## 02 기본키 제약조건

### 기본키 이름 지정

```
drop table usertbl;  
create database testdb2;  
use testdb2;  
create table usertbl(  
  userid char(10),  
  name char(10) not null,  
  birthyear int not null,  
  constraint primary key PK_userTbl_userID (userID)  
);  
desc usertbl;
```

## 02 기본키 제약조건

### 테이블 생성 후 기본키지정

```
use testdb2;
create table usertbl2(
userid char(10) ,
name char(10) not null,
birthyear int not null
);
desc usertbl2;

alter table usertbl2 add constraint PK_userid primary key (userid);
desc usertbl2;

-- 키 확인
show keys from usertbl2;
```

## 02 기본키 제약조건

### 복수열을 기본키로 지정

```
use testdb2;  
create table usertbl3(  
  userid char(10) ,  
  name char(10) not null,  
  birthyear int not null  
);  
desc usertbl3;
```

```
alter table usertbl2 add constraint PK_userid primary key (userid,birthyear);  
desc usertbl3;
```

```
-- 키 확인  
show keys from usertbl3;
```



## 02 기본키 제약조건

[문제] 제품코드 + 제품일련번호를 합쳐서 유일한 값으로 만든 뒤 PK로 설정하기

| 제품코드 | 제품 일련 번호 | 제조 일자      | 현 상태 |
|------|----------|------------|------|
| AAA  | 0001     | 2019.10.10 | 판매완료 |
| BBB  | 0002     | 2019.10.11 | 매장진열 |
| BBB  | 0001     | 2019.10.12 | 재고창고 |
| CCC  | 0001     | 2019.10.13 | 판매완료 |

### 여러 키열을 기본키로 설정

```
create table prodtbl
(
  prodCode char(3) not null,
  prodId char(4) not null,
  prodCur char(10) null
);
```

```
alter table prodTbl
add constraint pk_prodTbl_proCode_prodId
primary key (prodCode,ProdId);
```

```
desc prodtbl;
show keys from prodtbl;
```

## 03 외래키 제약조건

### 외래키 제약 조건

기본키열을 참조하는 키열  
테이블간의 관계 형성에 사용  
외래키열은 중복허용  
기본키열의 테이블에 없는 값은 외래키열에 삽입할 수 없다

### 사용

```
drop table if exists buytbl;  
create table buytbl  
(  
    num int auto_increment not null primary key,  
    userid char(8) not null,  
    prodname char(6) not null,  
    foreign key(userid) references usertbl(userid)  
);
```

## 03 외래키 제약조건

### 외래키 이름지정

```
drop table if exists buytbl;  
create table buytbl  
(  
    num int auto_increment not null primary key,  
    userid char(8) not null,  
    prodname char(6) not null,  
    constraint FK_userTbl_buyTbl foreign key(userid) references usertbl(userid)  
);
```

```
show keys from buytbl;
```

## 03 외래키 제약조건

### 외래키 옵션 지정

-- 기존 외래키열 제거

```
alter table buytbl
```

```
DROP FOREIGN KEY FK_usertbl_buytbl;
```

-- 외래키 cascade 설정

```
alter table buytbl
```

```
add constraint FK_usertbl_buytbl
```

```
foreign key (userid)
```

```
references usertbl(userid)
```

```
on update cascade
```

```
on delete cascade;
```

```
show keys from buytbl;
```

-- 기본적용 옵션

```
ON UPDATE NO ACTION / ON DELETE NO ACTION
```

- 부모테이블의 수정(UPDATE) 와 삭제(DELETE) 와 무관

## 03 외래키 제약조건

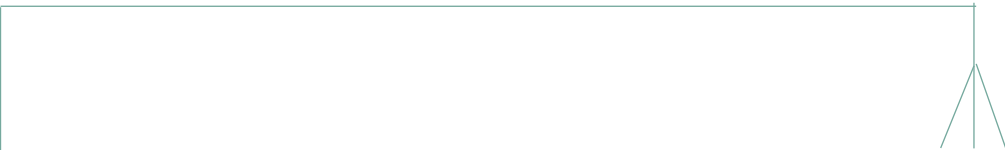
### 외래키 옵션 종류

1. CASCADE : 참조되는 테이블에서 데이터를 삭제하거나 수정하면, 참조하는 테이블에서도 삭제와 수정이 같이 이루어짐
2. SET NULL : 참조되는 테이블에서 데이터를 삭제하거나 수정하면, 참조하는 테이블의 데이터는 NULL로 변경
3. NO ACTION : 참조되는 테이블에서 데이터를 삭제하거나 수정해도, 참조하는 테이블의 데이터는 변경되지 않음
4. SET DEFAULT : 참조되는 테이블에서 데이터를 삭제하거나 수정하면, 참조하는 테이블의 데이터는 필드의 기본값으로 설정
5. RESTRICT : 참조하는 테이블에 데이터가 남아 있으면, 참조되는 테이블의 데이터를 삭제하거나 수정할 수 없음.

## 02 문제

[문제] SQL 문으로 테이블 관계를 포함 생성하세요

- db 는 testdb에서 진행한다
- 외래키 옵션은 update cascade, delete cascade 로 설정한다



| 이름(PK) | 지역 |
|--------|----|
| 김범수    | 경남 |
| 성시경    | 서울 |
| 조용필    | 경기 |
| 은지원    | 경북 |
| 바비킴    | 서울 |

학생테이블(stdTbl)



| 일련번호 | 이름(FK) | 동아리명(FK) |
|------|--------|----------|
| 1    | 김범수    | 바둑       |
| 2    | 김범수    | 축구       |
| 3    | 조용필    | 축구       |
| 4    | 은지원    | 축구       |
| 4    | 은지원    | 봉사       |
| 5    | 바비킴    | 봉사       |

학생\_동아리 테이블  
(stdclubTbl)

| 동아리명(PK) | 동아리방 |
|----------|------|
| 수영       | 101호 |
| 바둑       | 102호 |
| 축구       | 103호 |
| 봉사       | 104호 |

동아리 테이블(clubTbl)

## 04 UNIQUE 제약 조건

### UNIQUE 제약 조건

중복되지 않는 유일한 값

NULL 값 허용

보통 email 주소를 unique로 설정하는 경우가 많음

## 04 UNIQUE 제약 조건

### 사용

```
use testdb;
drop table if exists buytbl,usertbl;
create table uertbl
(
    userid char(8) not null primary key,
    name varchar(10) not null,
    birthYear int not null,
    email char(30) null unique
);
drop table if exists usertbl;
create table usertbl
(
    userid char(8) not null primary key,
    name varchar(10) not null,
    birthyear int not null,
    email char(30) null,
    constraint ak_email unique (email)
);
```



## 00 중간문제

- 다음 code 열을 UNIQUE 설정하세요 (UNIQUE명:AK\_CODE)

[제품테이블(productTbl)]

| code | prodName(PK) | price  | count |
|------|--------------|--------|-------|
| 111  | 신발           | 20000  | 100   |
| 222  | 가방           | 100000 | 50    |
| 333  | 모자           | 15000  | 30    |
| 444  | 티셔츠          | 10000  | 20    |
| 555  | 아우터          | 200000 | 155   |

[답]

```
ALTER TABLE productTbl ADD CONSTRAINT AK_CODE  
UNIQUE(code);
```

```
select * from  
information_schema.table_constraints  
where table_name='productTbl' ;
```

## 05 CHECK 제약 조건

### CHECK 제약조건

입력되는 데이터를 점검하는 기능

### 확인

-- 출생년도가 1900 년 이후 그리고 2023년 이전, 이름은 반드시 넣어야 함

```
drop table if exists usertbl;
```

```
create table usertbl
```

```
(
```

```
    userid char(8) primary key,
```

```
    name varchar(10),
```

```
    birthyear int check (birthYear >= 1900 and birthYear <=2023),
```

```
    mobile1 char(3) null,
```

```
    constraint CK_name CHECK (name is NOT NULL)
```

```
);
```

-- 제약조건 확인

```
select * from information_schema.table_constraints
```

```
where table_name='usertbl' ;
```

# 00 중간문제

[고객주문테이블(customerTbl)]

| num | userid | name | addr | grade | phone        | order(FK) | amount |
|-----|--------|------|------|-------|--------------|-----------|--------|
| 1   | KHC    | 김희철  | 서울   | 일반    | 010-111-1111 | 신발        | 2      |
| 2   | LYJ    | 이영자  | 부산   | 실버    | 010-222-2222 | 가방        | 5      |
| 3   | PHS    | 박호신  | 대구   | 골드    | 010-333-3333 | 모자        | 1      |
| 4   | IU     | 아이유  | 속초   | 다이아   | 010-444-4444 | 신발        | 1      |
| 5   | SNE    | 손나은  | 울산   | 일반    | 010-555-5555 | 아우터       | 3      |

1 amount 열에 1-100사이의값으로 제약(CHECK명:ck\_amount)

2 grade에 '일반', '실버', '골드', '다이아'만 입력되도록 제약(CHECK명:ck\_grade)

## 00 중간문제

### 답

```
-- 1 amount 열에 1-100사이의값으로 제약(CHECK명:ck_amount)
```

```
ALTER TABLE customerTbl ADD CONSTRAINT ck_amount CHECK(amount>=1 OR amount <= 100);
```

```
-- 2 grade에 '일반','실버','골드','다이아'만 입력되도록 제약(CHECK명:ck_grade)
```

```
ALTER TABLE customerTbl ADD CONSTRAINT ck_grade CHECK(grade IN('일반','실버','골드','다이아'));
```

```
-- 확인
```

```
select * from information_schema.table_constraints  
where table_name='customerTbl' ;
```

## 06 DEFAULT 제약 조건

### DEFAULT 정의

값을 입력하지 않았을 때 자동으로 입력되는 기본값을 정의

### DEFAULT 설정

```
drop table if exists usertbl;  
create table usertbl  
(  
    userid char(8) not null primary key,  
    name varchar(10) not null,  
    birthYear int not null default -1,  
    addr char(2) not null default '서울',  
    mobile1 char(3) null,  
    mobile2 char(8) null,  
    height smallint null default 170,  
    mDate date null  
);
```

## 06 DEFAULT 제약 조건

### DEFAULT 설정

```
drop table if exists usertbl;
create table usertbl
(
    userid char(8) not null primary key,
    name varchar(10) not null,
    birthYear int not null,
    addr char(2) not null,
    mobile1 char(3) null,
    mobile2 char(8) null,
    height smallint null,
    mDate date null
);
alter table usertbl alter column birthyear set default -1;
alter table usertbl alter column addr set default '서울';
alter table usertbl alter column height set default 170;

desc usertbl;
```

## 06 DEFAULT 제약 조건

### 확인

-- default 입력 시 디폴트값 자동 입력

```
insert into usertbl values ('LHL' , '이혜리', default, default, '011', '1234567', default, '2023.12.12');  
select * from usertbl;
```

-- 열이름이 명시되지 않으면 default로 설정된 값을 자동 입력

```
insert into usertbl(userid,name) values('KEY','김아영');  
select * from usertbl;
```

-- 값이 직접 명기되면 디폴트로 설정된 값은 무시된다

```
insert into usertbl values('wb','원빈',1982,'대전','019','9876543',176,'2020.05.05');  
select * from usertbl;
```

# 00 중간문제

[고객주문테이블(customerTbl)]

| num | userid | name | addr | grade | phone        | order(FK) | amount |
|-----|--------|------|------|-------|--------------|-----------|--------|
| 1   | KHC    | 김희철  | 서울   | 일반    | 010-111-1111 | 신발        | 2      |
| 2   | LYJ    | 이영자  | 부산   | 실버    | 010-222-2222 | 가방        | 5      |
| 3   | PHS    | 박호신  | 대구   | 골드    | 010-333-3333 | 모자        | 1      |
| 4   | IU     | 아이유  | 속초   | 다이아   | 010-444-4444 | 신발        | 1      |
| 5   | SNE    | 손나은  | 울산   | 일반    | 010-555-5555 | 아우터       | 3      |

-- 1 grade에 DEFAULT값을 '일반'으로 설정(DEFAULT명:df\_grade)

ALTER TABLE customerTbl alter column df\_grade set default '일반';

-- 2 addr에 DEFAULT값을 '대구'로 설정(DEFAULT명:df\_addr)

ALTER TABLE customerTbl alter column df\_addr set default '대구';



# END.

---

고생하셨습니다

