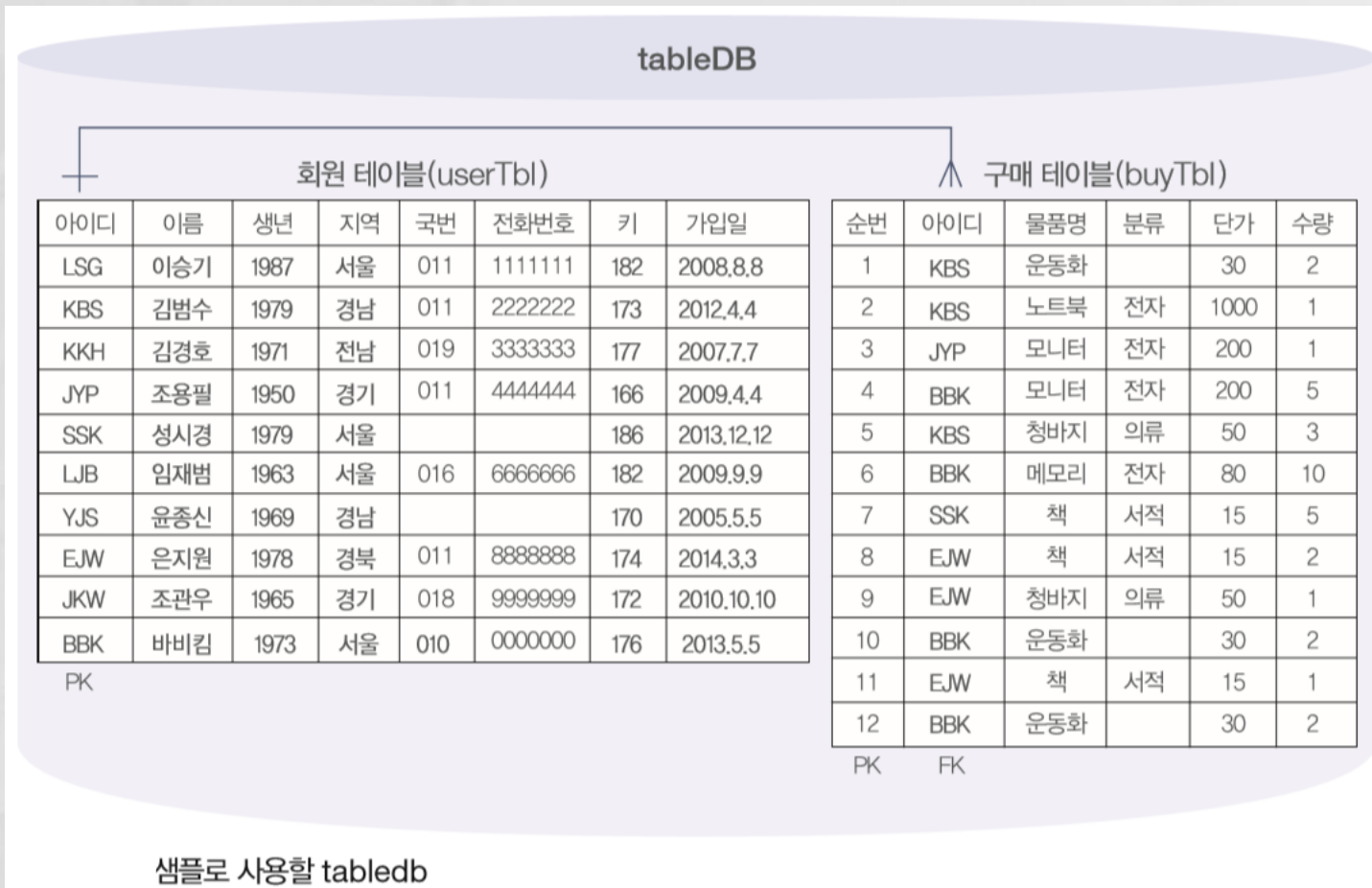


DBMS와 SQL

테이블

테이블 만들기

○ 테이블 만들기

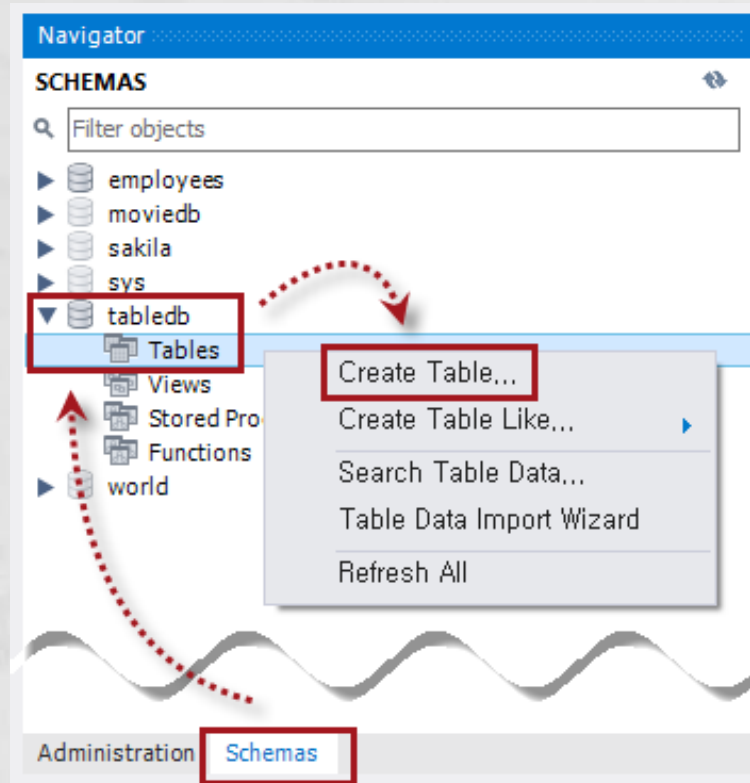


테이블

테이블 만들기

- MySQL Workbench에서 테이블 생성

- Navigator – [Schemas] 클릭 – 'tabledb' 확장 – 'Tables' 마우스 오른쪽 버튼 – [Create Table] 선택



테이블

테이블 만들기

- MySQL Workbench에서 테이블 생성
 - usertbl 생성
 - userID열을 기본 키(Primary Key)로 설정

usertbl - Table

Table Name: **usertbl** Schema: **tabledb**

Charset/Collation: Default Charset Default Collation Engine: InnoDB

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
userID	CHAR(8)	<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(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"/>	
birthYear	INT	<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"/>	
addr	CHAR(2)	<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"/>	
mobile1	CHAR(3)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
mobile2	CHAR(8)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
height	SMALLINT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
mDate	DATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

테이블

테이블 만들기

- MySQL Workbench에서 테이블 생성
 - buytbl 생성
 - num열을 기본 키(Primary Key)로 설정

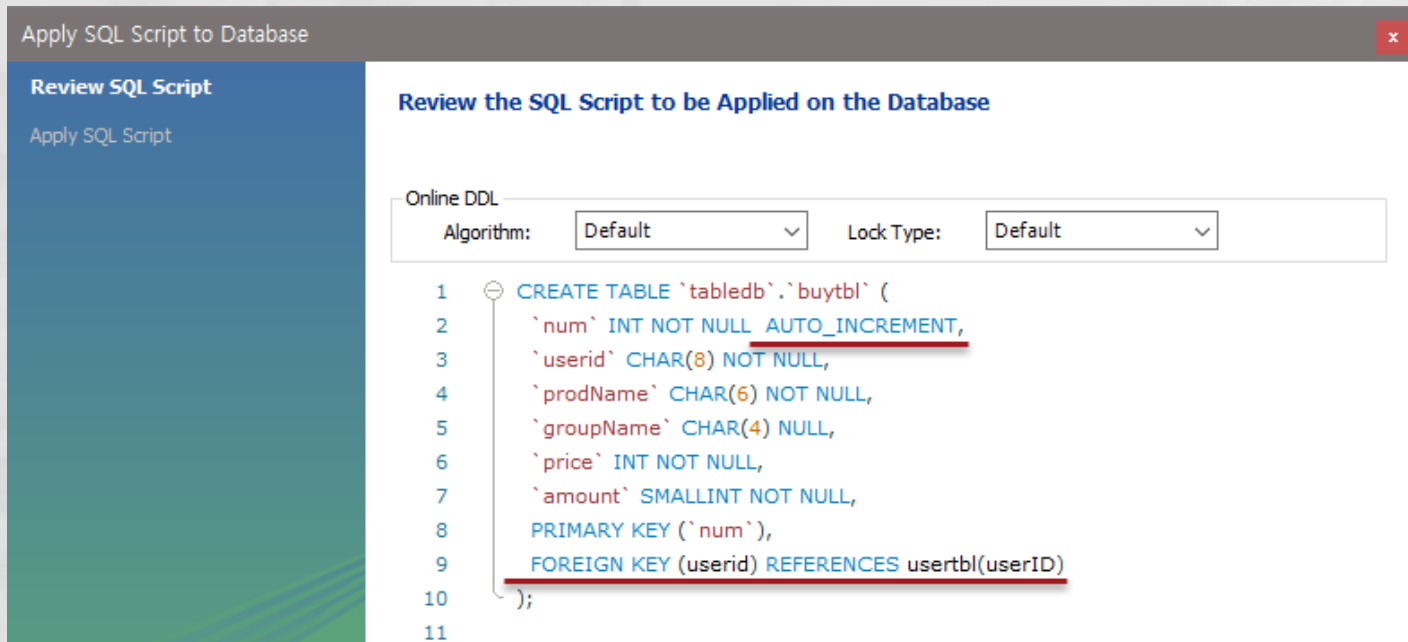
The screenshot shows the MySQL Workbench Table Designer interface for a table named 'buytbl'. The table is located in the 'tabledb' schema and uses the 'InnoDB' engine. The 'num' column is highlighted with a red box and a red arrow pointing to the 'PK' checkbox, indicating it is the primary key. The 'num' column is of type 'INT' and is not nullable. The other columns are 'userid' (CHAR(8)), 'prodName' (CHAR(6)), 'groupName' (CHAR(4)), 'price' (INT), and 'amount' (SMALLINT), all of which are nullable.

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
num	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"/>	
userid	CHAR(8)	<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"/>	
prodName	CHAR(6)	<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"/>	
groupName	CHAR(4)	<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"/>	
price	INT	<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"/>	
amount	SMALLINT	<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"/>	

테이블

테이블 만들기

- MySQL Workbench에서 테이블 생성
 - buytbl 생성
 - num열에 AUTO_INCREMENT, FOREIGN KEY 추가



테이블

테이블 만들기

MySQL Workbench에서 데이터 입력

- Navigator에서 usertbl 선택 – 마우스 오른쪽 버튼 클릭 – [Select Rows – Limit 1000] 선택
- <Insert new row> 아이콘 클릭한 후, 3개 행 입력 - <Apply> 클릭 - <Finish> 클릭

Result Grid

	userID	name	birthYear	addr	mobile1	mobile2	height	mDate
	KBS	김범수	1979	경남	011	2222222	173	2012-04-04
	KKH	김경호	1971	전남	019	3333333	177	2007-07-07
	LSG	이승기	1987	서울	011	1111111	182	2008-08-08
▶w	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

usertbl 1 x

Apply Revert

테이블

테이블 만들기

MySQL Workbench에서 데이터 입력

- buytbl 선택 – 마우스 오른쪽 버튼 클릭 – [Select Rows – Limit 1000] 선택
- <Insert new row> 아이콘 클릭한 후, 3개 행 입력
 - num열은 자동 입력되니 NULL 값은 그대로 둬 - <Apply> 클릭

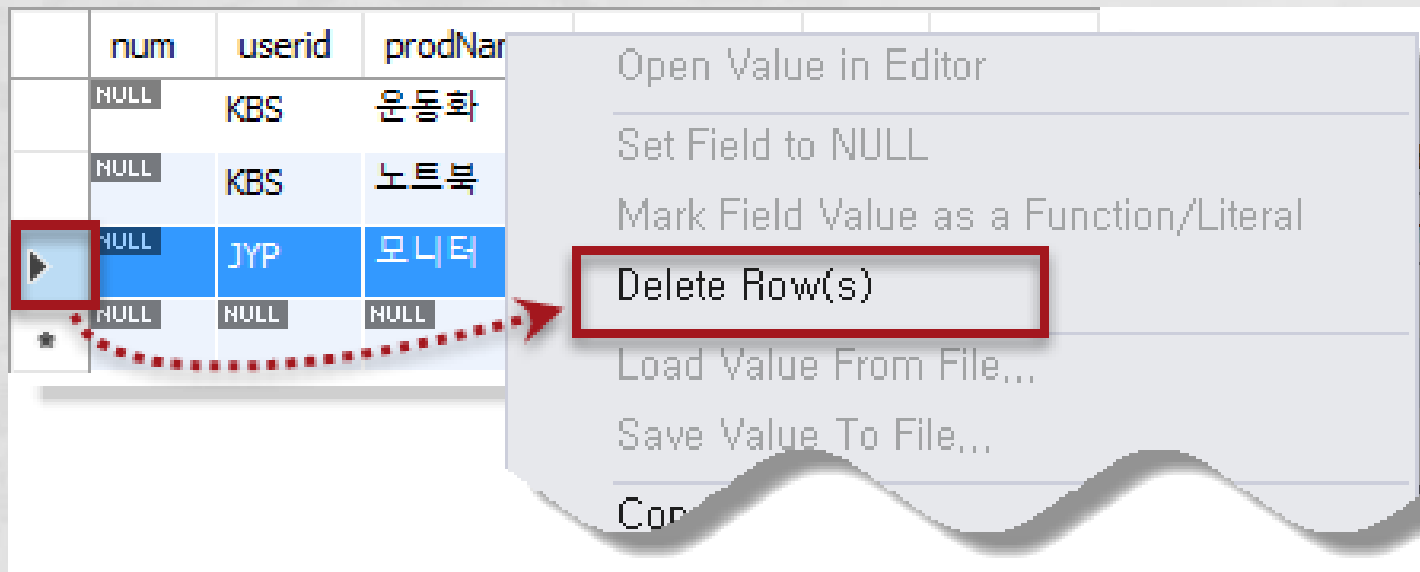
	num	userid	prodName	groupName	price	amount
	NULL	KBS	운동화	NULL	30	2
	NULL	KBS	노트북	전자	1000	1
▶	NULL	JYP	모니터	전자	200	1
✱	NULL	NULL	NULL	NULL	NULL	NULL

테이블

테이블 만들기

MySQL Workbench에서 데이터 입력

- JYP 열 선택 - 마우스 오른쪽 버튼 - [Delete Row(s)] 선택
- <Apply> 클릭 - <Finish> 클릭
 - 문제없이 입력 됨



테이블

테이블 만들기

- SQL로 테이블 생성

- usertbl 생성

```
CREATE TABLE buytbl
(  num INT AUTO_INCREMENT NOT NULL PRIMARY KEY,
   userid CHAR(8) NOT NULL ,
   prodName CHAR(6) NOT NULL,
   groupName CHAR(4) NULL ,
   price      INT NOT NULL,
   amount     SMALLINT NOT NULL
);
```

테이블

테이블 만들기

- SQL로 테이블 생성
 - buytbl 생성

```
CREATE TABLE buytbl
(  num INT AUTO_INCREMENT NOT NULL PRIMARY KEY,
   userid CHAR(8) NOT NULL ,
   prodName CHAR(6) NOT NULL,
   groupName CHAR(4) NULL ,
   price      INT  NOT NULL,
   amount     SMALLINT  NOT NULL
   , FOREIGN KEY(userid) REFERENCES usertbl(userID)
);
```

테이블

테이블 만들기

- SQL로 테이블 생성

- 회원 테이블 데이터 입력

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

- 구매 테이블 데이터 입력

- INSERT INTO buytbl VALUES(NULL, 'KBS', '운동화', NULL, 30, 2);
 - INSERT INTO buytbl VALUES(NULL, 'KBS', '노트북', '전자', 1000, 1);
 - INSERT INTO buytbl VALUES(NULL, 'JYP', '모니터', '전자', 200, 1);

- 구매 테이블 데이터 입력시 3번째 행은 앞과 같이 에러 발생하므로 삭제하고 입력

테이블

제약 조건

- 제약 조건(Constraint)이란?
 - 데이터의 무결성을 지키기 위한 제한된 조건 의미
 - 특정 데이터를 입력 시 어떠한 조건을 만족했을 때에 입력되도록 제약
 - ex) 동일한 아이디로 다시 회원 가입이 안되는 것
- 데이터 무결성을 위한 제약조건
 - PRIMARY KEY 제약 조건
 - FOREIGN KEY 제약 조건
 - UNIQUE 제약 조건
 - CHECK 제약 조건(MySQL 8.0.16부터 지원)
 - DEFAULT 정의
 - NULL 값 허용

테이블

데이터 무결성을 위한 제약 조건

◦ 기본 키(Primary Key) 제약 조건

- 기본 키(Primary Key) 란?
 - 테이블에 존재하는 많은 행의 데이터를 구분할 수 있는 식별자
 - 중복이나 NULL값이 입력될 수 없음
 - ex) 회원 테이블의 회원 아이디, 학생 테이블이 학번
- 기본 키로 생성한 것은 자동으로 클러스터형 인덱스 생성
- 테이블에서는 기본 키를 하나 이상 열에 설정 가능
- 기본 키 생성 방법

```
DROP TABLE IF EXISTS userTBL;  
CREATE TABLE userTBL  
( userID CHAR(8) NOT NULL,  
  name VARCHAR(10) NOT NULL,  
  birthYear INT NOT NULL,  
  CONSTRAINT PRIMARY KEY PK_userTBL_userID (userID)  
);
```

테이블

데이터 무결성을 위한 제약 조건

- 기본 키(Primary Key) 제약 조건

- 제품 테이블

```
DROP TABLE IF EXISTS prodTbl;  
CREATE TABLE prodTbl  
( prodCode CHAR(3) NOT NULL,  
  prodID   CHAR(4) NOT NULL,  
  prodDate DATETIME NOT NULL,  
  prodCur  CHAR(10) NULL,  
  CONSTRAINT PK_prodTbl_proCode_prodID  
    PRIMARY KEY (prodCode, prodID)  
);
```

- SHOW INDEX FROM prodTbl;

	Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment
▶	prodtbl	0	PRIMARY	1	prodCode	A	0	NULL	NULL		BTREE		
	prodtbl	0	PRIMARY	2	prodID	A	0	NULL	NULL		BTREE		

테이블

데이터 무결성을 위한 제약 조건

◦ 외래 키(Foreign Key) 제약 조건

- 두 테이블 사이의 관계 선언하여 데이터의 무결성 보장해주는 역할
- 외래 키 관계를 설정하면 하나의 테이블이 다른 테이블에 의존
- 외래 키 테이블이 참조하는 기준 테이블의 열은 반드시 Primary Key이거나 Unique 제약 조건이 설정되어 있어야 함
- 외래 키의 옵션 중 ON DELETE CASCADE 또는 ON UPDATE CASCADE
- 기준 테이블의 데이터가 변경되었을 때 외래 키 테이블도 자동으로 적용되도록 설정

테이블

데이터 무결성을 위한 제약 조건

- 외래 키(Foreign Key) 제약 조건

- 외래 키 생성 방법

- CREATE TABLE 끝에 FOREIGN KEY 키워드로 설정

```
DROP TABLE IF EXISTS buyTBL, userTBL;
CREATE TABLE userTBL
( userID CHAR(8) NOT NULL PRIMARY KEY,
  name VARCHAR(10) NOT NULL,
  birthYear INT NOT NULL
);
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)
);
```

테이블

데이터 무결성을 위한 제약 조건

- 외래 키(Foreign Key) 제약 조건

- 외래 키 생성 방법

- ALTER TABLE 구문 이용

```
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
);
ALTER TABLE buyTBL
  ADD CONSTRAINT FK_userTBL_buyTBL
  FOREIGN KEY (userID)
  REFERENCES userTBL(userID);
```

테이블

데이터 무결성을 위한 제약 조건

◦ UNIQUE 제약 조건

- '중복되지 않는 유일한 값'을 입력해야 하는 조건
- PRIMARY KEY와 비슷하나 UNIQUE는 NULL 값 허용
- NULL은 여러 개가 입력되어도 상관 없음
- ex) 회원 테이블 Email 주소 Unique로 설정

```
USE tableDB;  
DROP TABLE IF EXISTS buyTBL, userTBL;  
CREATE TABLE userTBL  
( userID CHAR(8) NOT NULL PRIMARY KEY,  
  name VARCHAR(10) NOT NULL,  
  birthYear INT NOT NULL,  
  email CHAR(30) NULL UNIQUE  
);
```

테이블

데이터 무결성을 위한 제약 조건

◦ CHECK 제약 조건

- 입력되는 데이터를 점검하는 기능
 - ex) 키(Height) 제한 - 마이너스 값이 들어올수 없도록,
 - 출생년도 제한 - 1900년 이후이고 현재시점 이전
- ALTER TABLE문으로 제약 조건 추가 가능

```
-- 출생년도가 1900년 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)  
);
```

테이블

데이터 무결성을 위한 제약 조건

◦ DEFAULT 정의

- 값 입력하지 않았을 때 자동으로 입력되는 기본 값 정의하는 방법
- ALTER TABLE 사용 시에 열에 DEFAULT를 지정하기 위해서 ALTER COLUMN문

사용

```
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
);
```

테이블

데이터 무결성을 위한 제약 조건

◦ DEFAULT 정의

- 디폴트 설정된 열에는 다음과 같은 방법으로 데이터 입력

```
-- default문은 DEFAULT로 설정된 값을 자동 입력한다.  
INSERT INTO usertbl VALUES ('LHL', '이혜리', default, default, '011', '1234567',  
    default, '2023.12.12');  
  
-- 열 이름이 명시되지 않으면 DEFAULT로 설정된 값을 자동 입력한다.  
INSERT INTO usertbl(userID, name) VALUES('KAY', '김아영');  
  
-- 값이 직접 명기되면 DEFAULT로 설정된 값은 무시된다.  
INSERT INTO usertbl VALUES ('WB', '원빈', 1982, '대전', '019', '9876543', 176,  
    '2020.5.5');  
  
SELECT * FROM usertbl;
```

	userID	name	birthYear	addr	mobile1	mobile2	height	mDate
▶	KAY	김아영	-1	서울	NULL	NULL	170	NULL
	LHL	이혜리	-1	서울	011	1234567	170	2023-12-12
	WB	원빈	1982	대전	019	9876543	176	2020-05-05

테이블

데이터 무결성을 위한 제약 조건

- Null 값 허용

- NULL 값을 허용하려면 NULL을, 허용하지 않으려면 NOT NULL을 사용
- PRIMARY KEY가 설정된 열에는 생략하면 자동으로 NOT NULL
- NULL 값은 '아무 것도 없다'라는 의미, 공백(' ') 이나 0과 다름

정리

정리

- 테이블