

FOREIGN KEY

FOREIGN KEY

FOREIGN KEY 제약 조건을 설정한 필드는 외래 키라고 부르며, 한 테이블을 다른 테이블과 연결해주는 역할을 합니다.

외래 키가 설정된 테이블에 레코드를 입력하면, 기준이 되는 테이블의 내용을 참조해서 레코드가 입력됩니다.

즉, FOREIGN KEY 제약 조건은 하나의 테이블을 다른 테이블에 의존하게 만듭니다.

FOREIGN KEY 제약 조건을 설정할 때 참조되는 테이블의 필드는 반드시 UNIQUE나 PRIMARY KEY 제약 조건이 설정되어 있어야 합니다.

CREATE 문으로 FOREIGN KEY설정

CREATE 문에서 테이블을 생성할 때 다음과 같이 FOREIGN KEY제약 조건을 설정할 수 있습니다.

CREATE 문으로 테이블을 생성할 때 해당 필드의 타입 뒤에 FOREIGN KEY를 명시하면, 해당 필드가 외래키로 설정됩니다.

문법

```
CREATE TABLE 테이블이름
(
  필드이름 필드타입,
  ...,
  [CONSTRAINT 제약조건이름]
  FOREIGN KEY(필드이름)
  REFERENCES 테이블이름 (필드이름)
)
```

위의 문법을 사용하면 해당 필드에 FOREIGN KEY 제약 조건을 설정합니다.

이때 참조되는 테이블의 이름은 REFERENCES 키워드 다음에 명시됩니다.

다음 예제는 Test2 테이블의 ParentID 필드에 Test1 테이블의 ID필드를 참조하는 FOREIGN KEY 제약 조건을 설정하는 예제입니다.

예제

```
CREATE TABLE Test1
(p_id int Primary key,
```

```

    p_name varchar(10)
);

CREATE TABLE Test2
(f_id int,
 cname varchar(10),
 p_id int,
 FOREIGN KEY(p_id)
 REFERENCES Test1(p_id) ON DELETE CASCADE
);

```

실행 결과

Field	Type	Null	Key	Default	Extra
ID	int(11)	YES		NULL	
ParentID	int(11)	YES	MUL	NULL	

위의 예제에서 Test2 테이블의 ParentID 필드는 Test1테이블의 ID 필드를 참조하도록 설정됩니다.

따라서 Test1 테이블의 ID 필드가 삭제되면, 같은 값의 Test2 테이블의 ParentID 필드도 같이 삭제 됩니다.

ALTER 문으로 FOREIGN KEY 설정

ALTER 문으로 테이블에 새로운 필드를 추가하거나 수정할 때도 FOREIGN KEY 제약 조건을 설정할 수 있습니다.

테이블에 새로운 필드를 추가할 때 해당 필드를 외래키로 설정하는 문법은 다음과 같습니다.

```

문법
ALTER TABLE 테이블이름
ADD [CONSTRAINT 제약조건이름]
FOREIGN KEY(필드이름)
REFERENCES 테이블이름 (필드이름)

```

다음 예제는 Reservation 테이블의 ID 필드에 Customer 테이블의 ID 필드를 참조하는 FOREIGN KEY 제약 조건을 설정하는 예제입니다.

예제

```
ALTER TABLE Reservation
ADD CONSTRAINT CustomerID
FOREIGN KEY(ID)
REFERENCES Customer (ID);
```

실행 결과

Field	Type	Null	Key	Default	Extra
ID	int(11)	YES	MUL	NULL	
Name	varchar(30)	YES		NULL	
ReserveDate	date	YES		NULL	
RoomNum	int(11)	YES		NULL	

MySQL 수업예제에서 사용되는 Reservation 테이블과 Customer 테이블은 다음과 같습니다.

Reservation 테이블

Customer 테이블

ID	Name	ReserveDate	RoomNum	ID	Name	Age	Address
1	홍길동	2016-01-05	2014	1	홍길동	17	서울
2	임꺽정	2016-02-12	918	2	임꺽정	11	인천
3	장길산	2016-01-16	1208	3	장길산	13	서울
4	홍길동	2016-03-17	504	4	전우치	17	수원

다음과 같은 방법을 사용하여 FOREIGN KEY 제약 조건을 삭제할 수 있습니다.

문법

```
ALTER TABLE 테이블이름
DROP FOREIGN KEY 제약조건이름
```

다음 예제는 Reservation 테이블의 ID 필드에 설정된 FOREIGN KEY 제약 조건을 삭제하는 예제입니다.

예제

```
ALTER TABLE Reservation
DROP FOREIGN KEY CustomerID;
```

ON DELETE, ON UPDATE

FOREIGN KEY 제약 조건에 의해 참조되는 테이블에서 데이터의 수정이나 삭제가 발생하면, 참조하고 있는 테이블의 데이터도 같이 영향을 받습니다.

이때 참조하고 있는 테이블의 동작은 다음 키워드를 사용하여 FOREIGN KEY 제약 조건에서 미리 설정할 수 있습니다.

1. ON DELETE

2. ON UPDATE

참조되는 테이블의 값이 삭제 될 경우의 동작은 ON DELETE 구문으로 설정할 수 있습니다. 또한, 참조되는 테이블의 값이 수정될 경우의 동작은 ON UPDATE 구문으로 설정할 수 있습니다.

이때 설정할 수 있는 동작은 다음과 같습니다.

1. CASCADE : 참조되는 테이블에서 데이터를 삭제하거나 수정하면, 참조하는 테이블에서도 삭제와 수정이 같이 이루어집니다.
2. SET NULL : 참조되는 테이블에서 데이터를 삭제하거나 수정하면, 참조하는 테이블의 데이터는 NULL로 변경됩니다.
3. NO ACTION : 참조되는 테이블에서 데이터를 삭제하거나 수정해도, 참조하는 테이블의 데이터는 변경되지 않습니다.
4. SET DEFAULT : 참조되는 테이블에서 데이터를 삭제하거나 수정하면, 참조하는 테이블의 데이터는 필드의 기본값으로 설정됩니다.
5. RESTRICT : 참조하는 테이블에 데이터가 남아 있으면, 참조되는 테이블의 데이터를 삭제하거나 수정할 수 없습니다.

다음 예제는 Test 테이블의 ID필드를 Customer 테이블의 ID 필드를 참조하는 외래 키로 설정하는 예제입니다.

이때 참조되는 필드의 데이터가 수정될 때는 CASCADE 방식으로, 삭제될 때는 RESTRICT 방식으로 동작합니다.

예제

```
CREATE TABLE Test2
(ID INT,
ParentID INT,
FOREIGN KEY(ParentID)
REFERENCES Test1(ID) ON UPDATE CASCADE ON DELETE RESTRICT
);
```

실행 결과

Field	Type	Null	Key	Default	Extra
ID	int(11)	YES		NULL	
ParentID	int(11)	YES	MUL	NULL	

위의 예제에서 Test2 테이블의 ID 필드가 참조하고 있는 Test1 테이블의 ID 값이 수정되면, Test2테이블의 ID 값도 같이 수정됩니다.

또한, Test2 테이블의 레코드 중에서 Test1 테이블의 레코드가 ID인 필드를 참조하는 레코드는 삭제할 수 없습니다.