

## 外部キーの制約、制約の追加、その他のオプション

1日目

2日目

3日目

4日目

5日目

6日目

7日目

8日目

9日目

10日目

11日目

12日目

13日目

14日目

15日目

16日目

17日目

18日目

19日目

20日目

21日目

# 外部キーの制約

# 参照整合性の崩壊

他のテーブルと紐づけをするキー(外部キー)が存在する場合、紐づけ先が存在しないといけない。この紐づけ先が存在する状態を参照整合性という。この状態が維持できなくなることを参照整合性の崩壊という

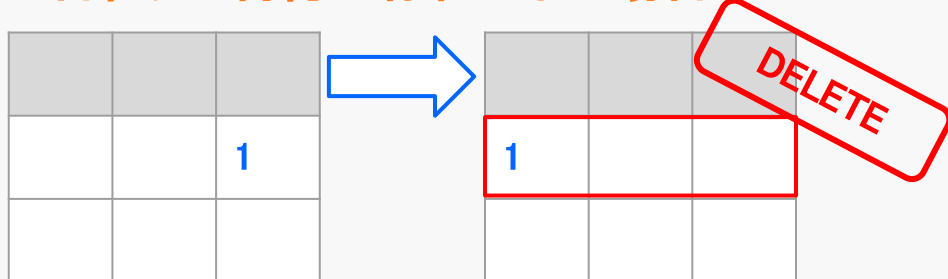
## 参照整合性が崩壊する4パターン

- ① 参照先のテーブルの行がDELETEされる
- ② 参照先のテーブルの行が別の値にUPDATEされる
- ③ 存在しない行を参照する行を、INSERTする
- ④ 存在しない行を参照する値に、UPDATEする

# 外部キー制約

外部キーを追加すると、参照整合性の崩壊が起きた場合にエラーが発生し、参照整合性を維持できるようになる

## 外部キー制約の存在しない場合



参照性制約がない場合、そのまま削除できる

## 外部キー制約の存在する場合



外部キーが存在する場合、参照先のレコードがDELETEされるとエラーが発生する

# 外部キーを付与する

## 外部キーを付与するSQL文

```
[CONSTRAINT constraint_name]  
FOREIGN KEY (column_name, ...)  
REFERENCES parent_table(column_name,...)  
[ON DELETE reference_option]  
[ON UPDATE reference_option]
```

**CONSTRAINT:** 制約の名前を設定する(オプション)

**FOREIGN KEY:** 外部キーの対象となるカラム

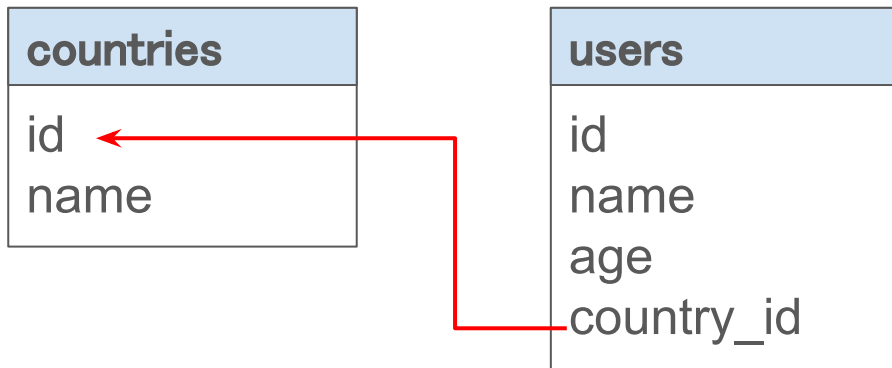
**REFERENCES:** 参照先のテーブルとカラム

**ON DELETE:** 参照先のレコードが削除された場合の挙動を設定する(後述)

**ON UPDATE:** 参照先のレコードが更新された場合の挙動を設定する(後述)

## 外部キーをcountry\_idカラムから、countriesのidカラムに設定する

```
CREATE TABLE users(  
  id INT PRIMARY KEY,  
  name VARCHAR(255),  
  country_id INT,  
  FOREIGN KEY(country_id)  
  REFERENCES countries(id)  
);
```



**DELETE FROM countries;** # 削除しようとするとき**参照整合性制約が発生して**エラーになる

# ON DELETEのオプション一覧

外部キー作成時に、ON DELETEオプションを追加すると、参照先が削除された際の挙動を設定できる

```
FOREIGN KEY(country_id)
REFERENCES countries(id)
ON DELETE CASCADE
```

オプション名	説明
CASCADE	参照先が削除されると、外部キーに設定している行は同時に削除される
SET NULL	参照先が削除されると、外部キーに設定している行にはNULLが設定される
RESTRICT	参照先が削除されそうになると、エラーが発生する
SET DEFAULT	参照先が削除されると、デフォルトの値が設定される

# ON UPDATEのオプション一覧

外部キー作成時に、ON UPDATEオプションを追加すると、参照先が更新された際の挙動を設定できる

```
FOREIGN KEY(country_id)
REFERENCES countries(id)
ON UPDATE CASCADE
```

オプション名	説明
CASCADE	参照先が更新されると、外部キーに設定している行は同じ値に更新される
SET NULL	参照先が更新されると、外部キーに設定している行にはNULLが設定される
RESTRICT	参照先が更新されそうになると、エラーが発生する
SET DEFAULT	参照先が更新されると、デフォルトの値が設定される



**ALTER TABLEで制約を後から追加する**

## UNIQUE制約を追加する

```
ALTER TABLE table_name  
ADD [CONSTRAINT constraint_name] UNIQUE (column1, column2, ... column_n);
```

## 制約を削除する

```
ALTER TABLE table_name DROP CONSTRAINT constraint_name;
```

## 特定のテーブルの制約一覧を表示する

```
SELECT column_name, constraint_name, referenced_column_name, referenced_table_name  
FROM information_schema.key_column_usage  
WHERE table_name = 'TableName';
```

## DEFAULTを追加する

ALTER TABLEとSET DEFAULTでデフォルト値を設定する

```
ALTER TABLE table_name
```

```
ALTER column_name SET DEFAULT default_value;
```

## NOT NULLを追加する

ALTER TABLEとMODIFYでNOT NULLを設定する

```
ALTER TABLE products MODIFY stocks INT NOT NULL
```

## CHECK制約を追加する

```
ALTER TABLE users ADD CONSTRAINT check_age CHECK (age >= 0);
```

## 主キーを追加する

```
ALTER TABLE persons  
ADD PRIMARY KEY (id);
```

```
ALTER TABLE persons  
ADD CONSTRAINT pk_person PRIMARY KEY (id,last_name);
```

## 外部キーを追加する

```
ALTER TABLE users  
ADD CONSTRAINT fk_grade_id  
FOREIGN KEY (grade_id) REFERENCES grades(id);
```

その他のオプション

# AUTO\_INCREMENT

AUTO\_INCREMENTを追加すると、値をNULLでINSERTした場合に自動的に1からカウントアップされた値が挿入される(整数型、浮動小数点型につけることができる)

```
CREATE TABLE animals (  
  id INT PRIMARY KEY AUTO_INCREMENT,  
  name CHAR(30) NOT NULL  
);
```

# id: 1, name: Dogが挿入される

```
INSERT INTO animals VALUES(NULL, "Dog");
```

# id: 2, name: Catが挿入される

```
INSERT INTO animals(name) VALUES("Cat");
```

# 自動で挿入される値を100に設定する

```
ALTER TABLE animals AUTO_INCREMENT = 100;
```

# id: 100, name: Birdが挿入される

```
INSERT INTO animals VALUES(NULL, "Bird");
```

# コメントの追加

カラムにどういう意味があるのか、他の利用者もわかるようにコメントを追加します

```
CREATE TABLE animals (  
  id INT PRIMARY KEY AUTO_INCREMENT COMMENT 'これはIDです',  
  name CHAR(30) NOT NULL COMMENT 'nameです NULLはいれない'  
);
```

SHOW FULL COLUMNS FROM animals — コメントの確認