

## ■ データベースの制約

作成: 2013.3.7 yoshi

データベースの制約(**constraint**)は、

「データ整合性」と呼ばれるデータの正確性、及びデータ間の関係性が正しく保たれます。  
データベースのテーブルに無効なデータを入力されないようにする為、  
あらかじめデータに対するビジネス・ルールを定義し、データベースへの  
データ挿入、更新、または削除するときにその定義されたルールに従い、  
ルールに従わないデータが入力されないようにします。そのルールのことは制約といいます。

### 【制約の種類】

ヌル(必須)	NOT NULL	NOT NULL制約は、単一の列に対して定義されるルールで、この列のNULL値(値がない)の入力を禁止するとの制約です。即ち、テーブルのすべての行で必ずその列にNULL以外の値が格納されるように制限します。
ユニーク	UNIQUE	ユニーク制約は、列、または複数の列の組み合わせに対して定義されるルールで、それらの列に値が一意であることのデータに限りデータの挿入・更新を許可する。
主キー	PRIMARY KEY	主キー制約は、列、または複数の列の組み合わせに対して定義されるルールで、表の中の行を一意に特定できます。ユニーク・キーと違って、主キーは1つの表に1つしか設定出来ません。また、主キー制約で参照される列(または列の組み合わせ)は、NOT NULL制約及びユニーク制約が自動的に付加されます。
外部キー	FOREIGN KEY～ REFERENCES～ON	外部キー制約は、表と表の間を表内の列(または列の組み合わせ)で参照という関連付けを定義するルールといいます。外部キー制約は表同士の関連付けを行い、子表は親表を参照し、子表の参照列を外部キーを呼びます。親表と子表の間でデータの整合性を保ちます。子表に入力される値は必ず親表に存在しなければなりません。
表チェック	CHECK	表チェック制約は、入力条件を定義した制約で、列に条件を満たす値のみ入力が入力が許可されます。 <b>但し、MySQLではcheck制約が効かない。</b>

### 【制約の基本構文】

```
CREATE TABLE <表名> (
  列名 データ型 [CONSTRAINT 制約名] 制約の種類
  ...
  [CONSTRAINT 制約名] 制約の種類
  ...
)
```

#### 凡例:

<>: 必須構文。  
[: オプション構文。  
| : OR

又は

```
CREATE TABLE <表名> (
  列名 データ型
  ...
  [, CONSTRAINT 制約名] 制約の種類
  ...
)
```

### 【制約構文例】

#### ◆NOT NULL制約

```
CREATE TABLE <表名> (
  列名 データ型 [CONSTRAINT 制約名] NOT NULL
  ...
)
```

#### ◆ユニーク制約

```
CREATE TABLE <表名> (
  列名 データ型 [CONSTRAINT 制約名] UNIQUE
  ...
)
```

又は

```
CREATE TABLE <表名> (
  列名1 データ型,
  列名2 データ型,
  ...
  [CONSTRAINT 制約名] UNIQUE(列名1, 列名2)
  ...
)
```

#### ◆主キー制約

```
CREATE TABLE <表名> (
  列名 データ型 [CONSTRAINT 制約名] PRIMARY KEY
  ...
)
```

又は

```
CREATE TABLE <表名> (
  列名1 データ型,
  列名2 データ型,
  ...
  [CONSTRAINT 制約名] PRIMARY KEY(列名1, 列名2)
  ...
```

## ◆外部キー制約

```
CREATE TABLE <表名> (
  列名 データ型 [CONSTRAINT 制約名] REFERENCES 親表名 (親表の列名)
  ...
```

又は

```
CREATE TABLE <表名> (
  列名 データ型,
  ...
  [CONSTRAINT 制約名] FOREIGN KEY (外部キー列名 [, 外部キー列名] ...)
  REFERENCES 親表名 (親表の列名 [, 親表の列名] ...)
```

## ◆チェック制約

```
CREATE TABLE <表名> (
  列名 データ型 [CONSTRAINT 制約名] CHECK (条件)
  ...
```

又は

```
CREATE TABLE <表名> (
  列名 データ型,
  ...
  [CONSTRAINT 制約名] CHECK (条件)
  ...
```

## 【制約の変更・削除・無効／有効】

## ◆制約の変更

一度制約を削除してから、新たに制約を追加する。

```
ALTER TABLE <表名> MODIFY 列名 [NOT NULL]
```

又は

```
ALTER TABLE <表名> ADD|ALTER 表制約構文
ALTER TABLE <表名> ADD <表制約構文>;
ALTER TABLE <表名> ALTER [COLUMN] 列名 SET DEFAULT デフォルト値 ;
```

## ◆制約の削除

```
ALTER TABLE <表名> DROP [CONSTRAINT | PRIMARY KEY | INDEX 列名] 制約名 [RESTRICT | CASCADE]
```

- ・CASCADEオプションを指定すると、依存するその他制約も削除されます。
- ・UNIQUEの削除は DROP INDEX 列名で

UNIQUE制約の追加: 

```
alter table TESTTABLE ADD UNIQUE TESTUNIQUE_KEY
```

UNIQUE制約の削除: 

```
alter table TESTTABLE DROP INDEX TESTUNIQUE_KEY
```

## ◆制約の有効／無効

ALTER TABLE ~ [ENABLE/DISABLE] CONSTRAINT句を使用

- ・制約の有効化

```
ALTER TABLE <表名> ENABLE [CONSTRAINT | PRIMARY KEY | UNIQUE(列名)] 制約名
[CASCADE]
```

- ・制約の有効化

```
ALTER TABLE <表名> ENABLE [CONSTRAINT | PRIMARY KEY | UNIQUE(列名)] 制約名
[CASCADE]
```

データが制約を満たされない場合、制約の有効化になりません。

●参照先: <http://www.syboos.jp/database/doc/20071108152755969.html>