制約

# 人為的ミスに備える

データベースの本来の役割を考えると、テーブルに異常な値が格納されてしまうことは絶対に避けなければなりません。そこで、予期しない中断や同時実行など、システム的な理由でデータが異常にならないようにトランザクションによる制御を学習しました。

しかし、データベースの利用者が、文法は正しいもののシステムの意図とは異なったSQL文を実行することは防げません。**誤ったSQL文が指示通りに正しく実行**されます。

ミスをしない人間はいないので、DBMSには人為的ミスによる不正なデータ登録を防ぐしくみを備えています。例えば、前回学習した「データ型」もその安全機構の１つです。

テーブルの各列にデータ型を指定することで、その列に格納できるデータの種類を制限しています。さらに、多くのDBMSには**制約(constraint)**という仕組みがあり、制約を使えば「従業員表の生年月日はNULLになってはいけない」といったルールを決めて制限をかけることが出来ます。まずは、基本的な３つの制約から学習していきます。

|  |  |  |
| --- | --- | --- |
| EMP\_NO | ENAME | BIRTHDAY |
| 00001  00002  00003 | 小林友治  小林圭輔  友保隼平 | 1972-07-04  1986-03-06  ← 制約を設定するとNULLの  格納を許さないことが可能になる  ~~NULL~~ |

# 基本的な３つの制約

制約はテーブルに対してつける為、テーブル作成を作成するCREATE TABLE文やテーブル定義を変更するALTER TABLE文で指定することが出来ます。

・表定義と共に制約を設定する方法 **※基本的にこちらが主流**

CREATE TABLE 表名 (

列名 データ型 制約の指定 ,

…

制約の指定 ( 列名, 列名)

) ;

・表作成後に後から制約を追加する方法

ALTER TABLE 表名 MODIFY 列名 データ型 制約の指定 ;

ALTER TABLE ADD 制約の指定(列名) ;

・NOT NULL制約

NOT NULL制約とはその名のとおり、指定した列に対してNULLの格納を不可します。

テーブルにデータを登録する際に必須の情報を格納する列にはNOT NULL制約を付けましょう。例えば、商品を管理する場合で考えると「カテゴリ、販売価格」はデータを管理する上で必ず必要になります。**※商品番号に関しては主キー制約で説明します。**

ハンズオン　NOT NULL制約を付ける　**※必須**

１．商品表のカテゴリ、販売価格列にNOT NULL制約を定義する。

ALTER TABLE PRODUCT MODIFY CATEGORY VARCHAR(4) NOT NULL;

ALTER TABLE PRODUCT MODIFY PRICE INT(5) NOT NULL ;

SHOW COLUMNS FROM PRODUCT ;

２．カテゴリのデータをNULLに更新してみる。

UPDATE PRODUCT SET CATEGORY = NULL ;

・UNIQUE制約

UNIQUE制約とは、指定した列に対して同じ値の格納を不可にします。内容が重複してはいけない列に制約を付けます。例えば、従業員名は同姓同名の可能性があるので同じ値が格納されても問題はありませんが、商品名は同じ名前の登録を許すと紛らわしくなるのでユニークにしておくほうが良いでしょう。なお、UNIQUE制約がかかっていてもNULLが格納されたデータは複数存在することは許されます。

ハンズオン　UNIQUE制約を付ける **※必須**

１．商品表の商品名にUNIQUE制約を定義する。

ALTER TABLE PRODUCT ADD UNIQUE( PNAME ) ;

SHOW COLUMNS FROM PRODUCT ;

２．すでに存在する商品名「コーラ」の登録をしてみる。

INSERT INTO PRODUCT VALUES ( '9999', 'コーラ', 'ドリンク', 160 ) ;

※UNIQUE制約の削除は以下の構文になります。

ALTER TABLE 表名 DROP INDEX UNIQE制約の列名 ;

・CHECK制約

CHECK制約とは、指定した列に格納される値が妥当であるか判定を行い、条件を満たさない値の格納を不可にします。商品表で考えると「販売価格」が０以下の値はあり得ないので制約を付けることでマイナス値が格納されることを防ぎます。

ハンズオン　CHECK制約を付ける **※必須**

１．商品表の販売価格に０以上のみ格納可能なCHECK制約を定義する。

ALTER TABLE PRODUCT ADD CHECK( PRICE >= 0 ) ;

SHOW CREATE TABLE PRODUCT ;

２．販売価格に0以下の値を登録してみる。

INSERT INTO PRODUCT

VALUES('9999', 'サイダー', 'ドリンク', -100);

# 主キー制約

主キーとは「その列の値を指定すれば、データが完全に特定できる(１行になる)」役割を持つ列のことになります。商品表でいえば、商品番号が主キー列にあたります。

そして、主キーがその役割を果たすためには、「他の行と重複してはならない」、「必ず値が格納されていなければならない」この2つの性質を満たす必要があります。UNIQUE制約とNOT NULL制約にあたりますが主キー列に関しては個別で設定せずに、**主キー制約(PRIMARY KEY)**が用意されているのでこちらの制約をつけます。**主キー制約は表につき1つしかつけることが出来ません。※一つの列ではないので注意すること。**

主キー制約を付けることで、主キー列に重複データやNULLデータの登録を防ぐことが出来ます。また、PRIMARY KEY制約を定義すると、主キー列に対してインデックスが自動的に作成されます。

・主キー制約 列レベル構文

CREATE TABLE 表名 (

列名 データ型 PRIMARY KEY ,

…

) ;

ALTER TABLE 表名 MODIFY 列名 PRIMARY KEY ;

・主キー制約表 レベル構文（複合キーの場合はこちらを使用します。）

CREATE TABEL 表名 (

列名1 データ型 ,

列名2 データ型 ,

…

PRIMARY KEY(列名1, 列名2)

) ;

ALTER TABLE 表名 ADD PRIMARY KEY(列名1, 列名2) ;

・主キー制約の削除

ALTER TABLE 表名 DROP PRIMARY KEY ;

ハンズオン　主キー制約を設定する **※必須**

１．商品表の商品番号に対して、主キー制約を定義する。

ALTER TABLE PRODUCT ADD PRIMARY KEY( PRODUCT\_NO ) ;

SHOW COLUMNS FROM PRODUCT ;

２．同じ商品番号の商品を登録してみる。

INSERT INTO PRODUCT VALUES ( '0001', 'おためしピザ', 'ピザ', 1000 ) ;

※ALTER TABLE文はカンマ区切りで１つの問合せにまとめることが出来ます。

ALTER TABLE 表名 ADD 列名 データ型 制約の指定 ,

ADD 列名 データ型 制約の指定 ,

MODIFY 列名 データ型 制約の指定 ;