第４章　データの更新

４－１　データの登録

INSERTとは・・・CREATE TABLEでつくったデータを詰めるため（挿入）の文法。

INSERT文は基本的に１回で１行を挿入する。

使い方は、INSERT INTO ＜テーブル名＞ VALUES　（値１、値2）;

例

INSERT INTO ShohinIns VALUES ('0002', '穴あけパンチ', '事務用品', 500, 320, '2009-09-11');

これを打ち込むことでShohinInsというテーブルの中に穴あけパンチの行が追加された。

　　※最初に設定した列数と合わない値の数だとエラーになる。

VARUES句　１つ以上の行のセットをテーブルとして返すもの。

これは一つの文でまとめて行を作ることができる。（複数行インサート）

例

INSERT INTO ShohinIns VALUES ('0002', '穴あけパンチ', '事務用品', 500, 320, '2009-09-11');

INSERT INTO ShohinIns VALUES ('0003', 'カッターシャツ', '衣服', 4000, 2800, NULL);

INSERT INTO ShohinIns VALUES ('0004', '包丁', 'キッチン用品', 3000, 2800, '2009-09-20');

これで一気に追加できるがこれだと面倒くさいので最初の文章以降は省略ができる。

例

INSERT INTO ShohinIns VALUES ('1002', '穴あけパンチ', '事務用品', 500, 320, '2009-09-11'),

('1003', 'カッターシャツ', '衣服', 4000, 2800, NULL),

('1004', '包丁', 'キッチン用品', 3000, 2800, '2009-09-20');

これでも上のものと同じ意味を持つ。

　　　　デフォルト値を設定する

・・・テーブルにはデフォルト値（初期値）を設定できる。

方法

　CREATE　TABLE shohin←テーブル名

(shohin\_id 　　　　CHAR(4)←4文字のID 　NOT NULL,←空白はいらないよって意味

　　　 shohin\_mei VARCHAR(100) 　　　　 NOT NULL,

shohin\_bunrui VARCHAR(32) 　　　　DEFAULT 0,←デフォルトを０に設定

hanbai\_tanka INTEGER 　　 ,←数字なので空白つかないのでNOT　NULLいらない

shire\_tanka INTEGER ,

torokubi DATE ,

PRIMARY KEY (shohin\_id));←主キー

INSERT　ELECT文・・・データをコピーできるコマンド。AのデータをBにコピーすることができる。

例

INSERT INTO A (shohin\_id, shohin\_mei, shohin\_bunrui, hanbai\_tanka, shiire\_tanka, torokubi)

SELECT shohin\_id, shohin\_mei, shohin\_bunrui, hanbai\_tanka, shiire\_tanka, torokubi

FROM B;

これでBからセレクトした列がAテーブルにコピーされたということになる。

４－２　データの削除

DROP TABLE・・・テーブルそのものの削除をする。

１行だけ消したいのにすべてを消すのはとてももったいないですよね？

そこで出るのがDELEAT文です。

DELEAT FROM　＜テーブル名＞

・・・その中のテーブル内のすべての行を消すコマンド。

　　　　探索型デリート・・・削除対象を制限したDELETE文のこと。WHEREで条件を定義する。

例

　 DELETE FROM shohin

WHERE HANBAI\_TANKA >= 4000;

　　これで商品テーブルの中の販売単価が４０００円以上のものを削除ということになる。

※TRUNCATE文というものがあって、TRUNCATE＜テーブル名＞で

　　そのテーブルのすべての行の削除ができる。

４－３　データの更新

DATE文・・・そのテーブルのデータを変更ことが可能になる文。

　UPDATE ＜テ―ブル名＞

|  |  |  |
| --- | --- | --- |
| Shohin\_ID | Shohin\_mei | torokubi |
| 0001 | Tシャツ | 2009-10-10 |
| 0002 | 穴あけパンチ | 2009-10-10 |

　SET ＜列名＞　＝　＜式＞；

これで更新ができる。

例

UPDATE shohin 　　　　↑

SET torokubi = '2009-10-10'; 　　　ここ

これで商品テーブルの中の登録日列がすべて2009-10-10に変更される。

　　　　SET句・・・UPDATE文などで使うどの部分を更新するのかなどを表示するときに使うコマンド

探索型UPDATE・・・更新対象を制限したUPDATE文のこと。WHEREで条件を定義する。

UPDATE　＜テーブル名＞

　　　SET　＜列名＞　＝　＜式＞

WHERE　　＜条件＞；

これでそのテーブルの列にある条件のものが式のものに更新される。

例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0004 | 包丁 | キッチン用品 | 30000 | 2800 | 2009-10-10 |

UPDATE shohin

　　SET hanbai\_tanka = hanbai\_tanka \*10 ↑ 　　↑

　　 WHERE shohin\_bunrui ='キッチン用品'; 　　　　　WHERE分の指定　　更新

となる。

　　　　　NULLでの更新・・NULLクリアとも呼ばれそこが空っぽになる。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0008 | ボールペン | 事務用品 | 100 |  |  |

例

UPDATE shohin

　　 SET torokubi = NULL ↑ ↑

WHERE shohin\_id ='0008'　　　WHERE条件 　　　　NULLに更新

商品テーブルのIDが0008のところの登録日がNULL（空欄）になるというコマンド

　　複数列の更新

例

UPDATE shohin これでキッチン用品の販売単価が10倍。

　SET thanbai\_tanka = hanbaitanka\_tanka\*10 仕入単価が半分になるコマンドができた。

　WHERE shohin\_bunrui ='キッチン用品'; しかし同じような分を作るのは大変

　UPDATE shohin それでできたコマンドが次のコマンド。

　SET shiire\_tanka = shiire\_tanka / 2

　WHERE shohin\_bunrui ='キッチン用品';

　　　　　　　　↓

　UPDATE shohin このようにUPDATEとWHEREを省略

　SET hanbai\_tanka = hanbai\_tanka \* 10, して複数の更新をするとき簡単なコマ

shiire\_tanka = shiire\_tanka / 2 ンドにすることができる。

WHERE shohin\_bunrui = 'キッチン用品';

４－４　トランザクション

トランザクション・・・データベースに対する１つ以上の更新をまとめて呼ぶときの名称

　　１連の処理をひとまとめにしたもの。

例

「A銀行がaを購入したのでお金を送金するためA銀行からお金をおろしてB銀行にお金を送金する、それをB銀行で 確認するためにB銀行で確認するとしっかり振り込まれていた」

・A銀行がB銀行に送金したときA銀行口座は減っていてB銀行が増えているというのをまとめたもの

START　TRANSACTION文　・・・複数のコマンドを一度にする場合に使うコマンド。

下書きのようなものかな、

ROLLVACK・・・ START　TRANSACTION中に失敗したときにやり直しするときに使うコマンド。

COMMIT・・・START　TRANSACTION中のものを確定させるときに使うコマンド。

　　※イメージはトランザクションで打ち込んだものをロールバックは保存せずに終了、

コミットは上書き保存。

　　　ACID特性・・・標準規格によって定められた守るべき４つの大事な約束事のこと。

1. 原始性・・・トランザクションが終わったときにそこに含まれた処理はすべて実行されるか

　　すべて実行されないかの２択という性質。

1. 一貫性・・・トランザクションに含まれる処理はデータベースにあらかじめ設定された制約

　　、たとえば主キーやNOT　NULL制約を満たす、という性質

1. 独立性・・・トランザクション同士が互いに干渉を受けないことを保証する性質。
2. 永続性・・・耐久性のようなもの、トランザクションが終了したら、その時点でのデータ

　　の状態を保存することを保証する性質。