

4-4-7.UPDATE文

UPDATE文

はじめに

UPDATE文は、SQLのなかでもデータ操作言語と呼ばれ、データベース内のデータを変更、更新する命令です。UPDATE文を使用することによって、すでに登録したデータの変更を行うことができます。

Step1: 概念を知る

UPDATE文
テーブル内のデータを変更する際に用います。

```
UPDATE "テーブル名"  
SET "列名" = 値
```

Step2: 使い方を知る

では、さっそくテーブル内のデータを変更してみましょう。

postgres on postgres@PostgreSQL 10

```
1 UPDATE person  
2 SET  
3   fname = 'Gwyneth',  
4   lname = 'Paltrow',  
5   gender = 'F',  
6   birth_date = '1972/9/27'  
7 WHERE  
8   person_id = 1
```

データ出力 EXPLAIN メッセージ クエリの履歴

UPDATE 1

クエリが 75 msec で成功しました

postgres on postgres@PostgreSQL 10

```
1 select *  
2 from person
```

データ出力 EXPLAIN メッセージ クエリの履歴

	person_id integer	fname character varying (20)	lname character varying (20)	gender character (1)	birth_date date
1	2	Jonny	Buckland	M	1977-09-11
2	3	Guy	Berryman	M	1978-04-12
3	4	Will	Champion	M	1978-07-31
4	1	Gwyneth	Paltrow	F	1972-09-27

2. UPDATE文は **SET** の宣言も必要ですのでセットで覚えましょう。
3. 変更したい値が属する、カラム名、値を列挙します。
4. **変更したい行が決まっている場合は WHERE句 で指定します。**

実行して構文にエラーがなければテーブル内のデータが変更されます。
これで、データに変更があってもテーブルを新しくすることができます。

さて、解説の4番目が太字になっていますが、ここが大事です！
UPDATE文は、**WHERE句** を指定しなかった場合、
テーブル内にある行すべてに対して変更命令を出します。
つまりこういうことです。

```
postgres on postgres@PostgreSQL 10
1  UPDATE person
2  SET
3  fname = 'Gwyneth',
4  lname = 'Paltrow',
5  gender = 'F',
6  birth_date = '1972/9/27'
```

データ出力 [EXPLAIN](#) [メッセージ](#) [クエリの履歴](#)

UPDATE 4

クエリが 78 msec で成功しました

```
postgres on postgres@PostgreSQL 10
1  select *
2  FROM person
```

データ出力 [EXPLAIN](#) [メッセージ](#) [クエリの履歴](#)

	person_id integer	fname character varying (20)	lname character varying (20)	gender character (1)	birth_date date
1	2	Gwyneth	Paltrow	F	1972-09-27
2	3	Gwyneth	Paltrow	F	1972-09-27
3	4	Gwyneth	Paltrow	F	1972-09-27
4	1	Gwyneth	Paltrow	F	1972-09-27

すべてのデータが同一に変更されました。
元々すべてのデータを同一に変更したい場合はこれでいいですが、
ケースとしては少ないと思いますので、
WHERE句 を指定するクセをつけましょう。

本番データ（お客さんが実際に使用するデータ）を稼働中に誤って全更新。
なんてあってはいけませんので！

課題

提出課題はありませんので、一通り学習が終わったら次の章に進んで下さい。

最終更新日時: 2022年 09月 10日(土曜日) 07:47