



TOKYO IT SCHOOL

# データの更新と削除

## 目次

1. データの更新	1
2. データの削除	4

## 1. データの更新

### 1 UPDATE 文の基本構文

テーブルに登録したデータを変更する際に使用するのが、UPDATE 文です。UPDATE 文は、INSERT 文や DELETE 文と同じく DML 文に属します。基本的な構文は、次の通りです。

(構文) UPDATE 文

```
UPDATE <テーブル名>  
SET <列名> = <式>;
```

更新の対象の列と、更新後の値は、SET 句に記述します。それでは、以下の表の「emp\_pass」列の値を全て「7777」に変更してみましょう。

Employee テーブル

emp_id	emp_pass	emp_name	gender	address	birthday	dept_id
1	1111	system shared	1	千葉県	2009/09/09	1
2	2222	systemsss	1	松戸市	2009/09/10	1
3	3333	administrator	2	東京都	2009/09/10	2
4	4444	edudbuser	2	埼玉県	2009/09/14	3
5	9999	system555	1	千葉県	2009/09/09	1
6	9999	system666	1	千葉県	2009/09/09	1

全行「7777」に変更

(サンプルコード) UPDATE 文

```
UPDATE Employee SET emp_pass = 7777;  
  
COMMIT;
```

変更後の内容を SELECT 文で確認すると以下ようになります。

(実行結果)

EMP_ID	EMP_PASS	EMP_NAME	GENDER	ADDRESS	BIRTHDAY	DEPT_ID
1	7777	system shared	1	千葉県	09-09-09	1
2	7777	systemsss	1	松戸市	09-09-10	1
3	7777	administrator	2	東京都	09-09-10	2
4	7777	edudbuser	2	埼玉県	09-09-14	3
5	7777	system555	1	千葉県	09-09-09	1
6	7777	system666	1	千葉県	09-09-09	1

6 行選択されました

## 2 条件を指定した UPDATE 文（探索型 UPDATE）

次に、全行を更新するのではなく、更新対象の行を絞り込んでみましょう。行を絞り込む場合は、WHERE 句を使用します。構文は次の通りです。

（構文）一部の行だけを変更する UPDATE 文

```
UPDATE <テーブル名>
  SET <列名> = <式>
 WHERE <条件>;
```

例として、「dept\_id」列の値が「1」の行の「emp\_pass」列の値を「1111」に変更する UPDATE 文は次の通りです。

Employee テーブル

emp_id	emp_pass	emp_name	gender	address	birthday	dept_id
1	7777	system shared	1	千葉県	2009/09/09	1
2	7777	systemsss	1	松戸市	2009/09/10	1
3	7777	administrator	2	東京都	2009/09/10	2
4	7777	edudbuser	2	埼玉県	2009/09/14	3
5	7777	system555	1	千葉県	2009/09/09	1
6	7777	system666	1	千葉県	2009/09/09	1

↑ ↑

「1111」に変更

（サンプルコード）一部の行だけを変更する UPDATE 文

```
UPDATE Employee SET emp_pass = 1111 WHERE dept_id = 1;

COMMIT;
```

変更後の内容を SELECT 文で確認すると以下ようになります。

（実行結果）

EMP_ID	EMP_PASS	EMP_NAME	GENDER	ADDRESS	BIRTHDAY	DEPT_ID
-----						
1	1111	system shared	1	千葉県	09-09-09	1
2	1111	systemsss	1	松戸市	09-09-10	1
3	7777	administrator	2	東京都	09-09-10	2
4	7777	edudbuser	2	埼玉県	09-09-14	3
5	1111	system555	1	千葉県	09-09-09	1
6	1111	system666	1	千葉県	09-09-09	1

6 行選択されました

### 3 NULL で更新するには

UPDATE 文を使うことで、列を NULL で更新することも出来ます。NULL で更新する方法は単純で代入式の右辺にそのまま NULL を記述するだけです。

ただし、NULL クリアが可能なのは、NOT NULL 制約や主キー制約のついていない列に限られます。こうした制約のついている列を NULL に更新しようとしたときは、エラーになります。

### 4 複数列の更新

UPDATE 文の SET 句には、複数の列を更新対象として記述することが可能です。更新対象が複数ある場合は、「<列名> = <式>」をカンマ区切りで記述します。

(構文) 複数列を変更する UPDATE 文

```
UPDATE <テーブル名>  
SET <列名> = <式>, <列名> = <式>, ……;
```

## 2. データの削除

### 1 DELETE 文の基本構文

登録したデータを削除したい場合は、DELETE 文を使用します。DELETE 文の基本構文は、次の通りです。

(構文) DELETE 文

```
DELETE FROM <テーブル名>;
```

また、UPDATE 文同様に WHERE 句を用いて対象行を絞り込むことも可能です。

(構文) WHERE 句を使用した DELETE 文

```
DELETE FROM <テーブル名> WHERE <条件>;
```

例として、「emp\_id」列の値が「6」の行を削除する DELETE 文は次の通りです。

Employee テーブル

emp_id	emp_pass	emp_name	gender	address	birthday	dept_id
1	1111	system shared	1	千葉県	2009/09/09	1
2	1111	systemsss	1	松戸市	2009/09/10	1
3	7777	administrator	2	東京都	2009/09/10	2
4	7777	edudbuser	2	埼玉県	2009/09/14	3
5	1111	system555	1	千葉県	2009/09/09	1
6	1111	system666	1	千葉県	2009/09/09	1

削除

(サンプルコード) WHERE 句を使用した DELETE 文

```
DELETE FROM Employee WHERE emp_id = 6;
```

```
COMMIT;
```

変更後の内容を SELECT 文で確認すると以下ようになります。

(実行結果)

EMP_ID	EMP_PASS	EMP_NAME	GENDER	ADDRESS	BIRTHDAY	DEPT_ID
1	1111	system shared	1	千葉県	09-09-09	1
2	1111	systemsss	1	松戸市	09-09-10	1
3	7777	administrator	2	東京都	09-09-10	2
4	7777	edudbuser	2	埼玉県	09-09-14	3
5	1111	system555	1	千葉県	09-09-09	1

5 行選択されました

## 2 TRUNCATE 文

テーブルからデータを削除する方法として、標準 SQL が用意しているのは DELETE 文だけです。しかし、多くのデータベース製品には、「TRUNCATE 文」というコマンドが用意されています。

TRUNCATE は、必ずテーブルを全行削除します。WHERE 句で条件を指定して、一部の行だけ削除するということは出来ません。従って、細かい制御は出来ないのですが、その代わり、DELETE よりも削除の処理が高速であるというメリットがあります。

ただし、Oracle では、TRUNCATE を DML ではなく DDL として定義しています。すなわち、Oracle では TRUNCATE には ROLLBACK が効きません。TRUNCATE を実行することで、暗黙の COMMIT が発行されます。

(構文) TRUNCATE 文

```
TRUNCATE <テーブル名>;
```

(サンプルコード) TRUNCATE 文

```
TRUNCATE TABLE Employee;
```

変更後の内容を SELECT 文で確認すると以下ようになります。

(実行結果)

行が選択されていません