

【カーソルを使用した**現在行**のレコード処理】

カーソルによるカレント・レコードに対するデータ更新及び、削除処理

[UPDATE | DELETE] **WHERE CURRENT OF** カーソル名 ;

(例 1)

DECLARE

p_dept emp_table.dept_no%TYPE ; -- WHERE 文中の値指定用変数定義

~~emp_record emp_table%ROWTYPE ;~~ -- レコード型変数の定義

/* カーソルの定義 */

CURSOR c_emp IS

SELECT * FROM emp_table WHERE

dept_no = p_dept FOR UPDATE ;

-- 更新ロックが掛るまで待機する

BEGIN

p_dept := 4 ;

~~OPEN c_emp ;~~ -- この時点で排他的ロックが実施

FOR r_emp IN c_emp LOOP

/* r_emp がレコード型変数として自動定義され、レコードの値がループごとに
セットされる */

r_emp.price = r_emp.price * 1.08

UPDATE emp_table SET price = r_emp.price

WHERE CURRENT OF カーソル名 ;

/* カーソルの現在行に対しての更新処理 */

~~LOOP~~

~~FETCH c_emp INTO emp_record ;~~ -- フェッチ

~~EXIT WHEN c_emp%NOTFOUND ;~~ -- フェッチ失敗判断

~~/* データが存在した時の処理 */~~

END LOOP ;

~~CLOSE c_emp ;~~

-- カーソル・クローズ

COMMIT;

END ;

使用例)

DECLARE

```
p_dept emp_table.empno%TYPE ;          -- WHERE 文中の値指定用変数定義  
/*   カーソルの定義   */
```

```
CURSOR c_emp IS
```

```
    SELECT * FROM emp_table WHERE empno = p_dept  
    FOR UPDATE ;          --   更新ロックが掛る  
                           まで待機する
```

BEGIN

```
p_dept := 4 ;
```

```
FOR r_emp IN c_emp LOOP
```

```
/*   r_emp がレコード型変数として自動定義され、レコードの値がループごとに  
    セットされる   */
```

```
    UPDATE emp_table SET ename = 'hatsune'
```

```
        WHERE CURRENT OF c_emp ;
```

```
/*   カーソルの現在行に対しての更新処理   */
```

```
END LOOP ;
```

```
COMMIT;
```

```
END ;
```

```
/
```