

101集約関数(1)(2)(3)(4)

集約関数(1)(2)(3)(4)

count

今まで、selectの出力をそのまま表示してきたが、例えば、「横浜市在住の従業員は何人いるか？」という問にはどう答えたらいいであろうか？ こうした時、SQLでは、次の例1のように、関数count()を使用することが出来る。

例 1: 住所に「横浜市」という言葉を含む従業員の数

```
select count(住所)
from 従業員
where 住所 like '%横浜市%'
```

Expr1000
3

ここで関数count()は、count()無しの検索

```
select 住所
from 従業員
where 住所 like '%横浜市%'
```

が検索した行の数（すなわち、横浜市在住の従業員の数）を返すことになる。

住所
横浜市
横浜市
横浜市

次の例では、count()の形で、テーブルの全行数がカウントされている。

例 2: テーブル「仕入」の行数を数える

```
select count(*)
from 仕入
```

Expr1000
159

この例では、count(distinct 項目名) という形で、項目の相互にあい異なる値の数が 数えられる。
selectの結果に同じ値の出力が含まれていても、distinct が項目名につ いていれば、重複した出力は省かれるのと同じである。

例 3: 単価100以下の納入を行った業者の数

```
select count( distinct 業者名 )
from 業者,仕入
where 単価 < 100
and 仕入.業者番号 = 業者.業者番号
```

----- 3 件ヒットする！ これはcount(distinct 業者名)を*に変えて実行すると検証できる

もう少し複雑な例も見よう。次の例は、「仕入」で業者番号 1 の納入回数をカウントするものである。ここではwhere句で業者番号=1の指定を行っている。

例 4: 「仕入」で業者番号 1 の納入回数

```
SELECT Count(業者.業者番号) AS 納入回数のカウント
FROM 業者 INNER JOIN 仕入 ON 業者.業者番号 = 仕入.業者番号
WHERE (((業者.業者番号)=1))
```

納入回数のカウント
37

その他の集約関数

selectは、通常は複数の行を返すが、これらの一群の行の上で定義される関数を、集約（aggregate）関数と呼ぶ。先にみたcountは、代表的な集約関数の一つである。SQLで は、countのほかにも何種類かの集約関数が用意されている。以下、それを見てみよう。

sum(¥[distinct¥] 式)	選ばれた行中の式の（異なった）値の総和
avg(¥[distinct¥] 式)	選ばれた行中の式の（ // ） 値の平均
count(¥[distinct¥] 式)	選ばれた行中の式の（ // ） 非null値の数

count(罫*)

選ばれた行数

max(式)

選ばれた行中の式の値の最大値

min(式)

選ばれた行中の式の値の最小値