

108分類(14)(15)(16)

分類(14)(15)(16)

グループ化 group by

selectの出力を、グループに分けて整理したいことがある。つぎのような例を考えて見よう。この例は、部門ごとに部品を仕入れた回数を一覧表示するものである。

例 14: 部門ごとの仕入回数の一覧 1

```
SELECT 仕入.部門番号, 仕入.数量
FROM 部門 INNER JOIN 仕入 ON 部門.部門番号 = 仕入.部門番号
```

```
SELECT 仕入.部門番号, 仕入.数量
FROM 部門 ,仕入 where 部門.部門番号 = 仕入.部門番号
```

部門番号	数量
1	63
1	60
1	29
1	87
1	23
1	67
1	94
1	50
1	34
1	25
1	75
1	79
2	69
2	84
2	70
2	47
2	33

部門番号	数量
2	80
2	16
2	58
2	83
2	71
2	51
2	30
3	93
3	37
3	79
3	36
3	35
3	25
3	10
3	13
3	51
3	72
3	40
3	62

この出力から、部門ごとに仕入数量を加算するのは、結構、手間がかかる。先に見た order by 句で、部門について整列すれば、少しは数え易くはなるのだが、それにしても大変である。仕入れた回数を知るには、普通なら、集約関数のcountを使えばいいのだが、問題は、countは、selectの出力全てを数え上げるので、それぞれの部門毎の数は数えてはくれないことである。

こうした時、group by 句を用いれば、selectの出力は、グループに分けられ、しかも、集約関数は、selectの全出力の上ではなく、このグループの上を走るようになる。

例 15: 部門ごとの仕入回数の一覧 2

```
SELECT 仕入.部門番号, Count(仕入.数量) AS 数量のカウント
FROM 部門 INNER JOIN 仕入 ON 部門.部門番号 = 仕入.部門番号
GROUP BY 仕入.部門番号
```

```
SELECT 仕入.部門番号, Count(仕入.数量) AS 数量のカウント
FROM 部門 ,仕入 where 部門.部門番号 = 仕入.部門番号
GROUP BY 仕入.部門番号
```

部門番号	数量のカウント
1	12
2	12
3	12
4	33
5	12
6	34
7	8
8	7
9	14
10	15

このように、group by 句は、selectの出力をグループに分けて、そのグループ毎に 集約関数を走らせるときに使われる。

上記の例でcount()関数をsum()関数に変更すると、部門ごとの仕入数量の合計一覧を表示することができる。

例 16: 部門ごとの仕入数量の合計一覧

```
SELECT 仕入.部門番号, Sum(仕入.数量) AS 数量の合計
FROM 部門 INNER JOIN 仕入 ON 部門.部門番号 = 仕入.部門番号
GROUP BY 仕入.部門番号;
```

```
SELECT 仕入.部門番号, Sum(仕入.数量) AS 数量の合計
FROM 部門 , 仕入 where 部門.部門番号 = 仕入.部門番号
GROUP BY 仕入.部門番号
```

部門番号	数量の合計
1	686
2	692
3	553

部門番号	数量の合計
4	1681
5	605
6	1949
7	466
8	296
9	764
10	1230