

SQL 必知必会 10 - 分组数据

SQL 必知必会 10 - 分组数据

10.1 数据分组

10.2 创建分组

10.3 过滤分组

10.4 分组和排序

10.5 SELECT 子句顺序

10.6 小结

10.1 数据分组

使用分组可以将数据分为多个逻辑组，对每个组进行聚合计算。

10.2 创建分组

分组是使用 `SELECT` 语句的 `GROUP BY` 子句建立的。

```
SELECT vend_id, COUNT(*) AS num_prods
FROM Products
GROUP BY vend_id;
```

注意事项：

- `GROUP BY` 子句可以包含任意数目的列，因而可以对分组进行嵌套，更细致地进行数据分组。
- 如果在 `GROUP BY` 子句中嵌套了分组，数据将在最后指定的分组上进行汇总。换句话说，在建立分组时，指定的所有列都一起计算（所以不能从个别的列取回数据）

个人理解是：假如有红绿蓝三种球，每种有大中小三种规格。

那么 `SELECT count(*) AS number FROM Balls GROUP BY color, size` 会返回红大、红中、红小、绿大、绿中、绿小、蓝大、蓝中、蓝小，而不能单独得出红色多少个、或者中等大小的多少个。

- **GROUP BY** 子句中列出的每一列都必须是检索列或有效的表达式（但不能是聚合函数）。如果在 **SELECT** 中使用表达式，则必须在 **GROUP BY** 子句中指定相同的表达式。不能使用别名。
- 大多数 **SQL** 实现不允许 **GROUP BY** 列带有长度可变的数据类型
- 除聚合计算语句外，**SELECT** 语句中的每一列都必须在 **GROUP BY** 子句中给出。
- 如果分组列中包含具有 **NULL** 值的行，则 **NULL** 将作为一个分组返回。如果列中有多行 **NULL** 值，它们将分为一组。
- **GROUP BY** 子句必须出现在 **WHERE** 子句之后，**ORDER BY** 子句之前。

Microsoft SQL Server 等有些 **SQL** 实现在 **GROUP BY** 中支持可选的 **ALL** 子句。这个子句可用来返回所有分组，即使是没有匹配行的分组也返回（在此情况下，聚合将返回 **NULL**）。

有的 **SQL** 实现允许根据 **SELECT** 列表中的位置指定 **GROUP BY** 的列。
例如，**GROUP BY 2, 1** 可表示按选择的第二个列分组，然后再按第一个列分组。

10.3 过滤分组

SQL 还允许过滤分组，规定包括哪些分组，排除哪些分组。
但是，**WHERE** 不能完成任务，因为 **WHERE** 过滤指定的是行而不是分组。
WHERE 没有分组的概念。

要使用 **HAVING** 子句过滤分组。
目前为止所学过的所有类型的 **WHERE** 子句都可以用 **HAVING** 来替代。
唯一的差别是，**WHERE** 过滤行，而 **HAVING** 过滤分组。

```
SELECT cust_id, COUNT(*) AS orders
FROM Orders
GROUP BY cust_id
HAVING COUNT(*) >= 2;
```

另外一种从语句顺序上的理解

WHERE 在数据分组前进行过滤，**HAVING** 在数据分组后进行过滤。

具有两个以上产品且其价格大于等于 4 的供应商

```
SELECT vend_id, COUNT(*) AS num_prods
FROM Products
WHERE prod_price >= 4
GROUP BY vend_id
HAVING COUNT(*) >= 2;
```

如果不指定 `GROUP BY`，则大多数 `DBMS` 会同等对待它们。
不过，你自己要能区分这一点。
使用 `HAVING` 时应该结合 `GROUP BY` 子句
而 `WHERE` 子句用于标准的行级过滤。

10.4 分组和排序

表10-1 `ORDER BY`与`GROUP BY`

ORDER BY	GROUP BY
对产生的输出排序	对行分组，但输出可能不是分组的顺序
任意列都可以使用（甚至非选择的列也可以使用）	只可能使用选择列或表达式列，而且必须使用每个选择列表达式
不一定需要	如果与聚集函数一起使用列（或表达式），则必须使用

一般在使用 `GROUP BY` 子句时，应该也给出 `ORDER BY` 子句。这是保证数据正确排序的唯一方法。

查询包含三个或更多物品的订单号，和订购数量：

```
SELECT order_num, COUNT(*) AS items
FROM OrderItems
GROUP BY order_num
HAVING COUNT(*) >= 3
ORDER BY items, order_num;
```

`Microsoft Access` 不允许按别名排序，因此这个例子在 `Access` 中将失败。
解决方法是用实际的计算或字段位置替换。

10.5 `SELECT` 子句顺序

表10-2 SELECT子句及其顺序

子 句	说 明	是否必须使用
SELECT	要返回的列或表达式	是
FROM	从中检索数据的表	仅在从表选择数据时使用
WHERE	行级过滤	否
GROUP BY	分组说明	仅在按组计算聚集时使用
HAVING	组级过滤	否
ORDER BY	输出排序顺序	否

10.6 小结