

sql中的group by 和 having 用法解析 - 夕诺 - 博客园

--sql中的group by 用法解析:

-- Group By语句从英文的字面意义上理解就是“根据(by)一定的规则进行分组(Group)”。

--它的作用是通过一定的规则将一个数据集划分成若干个小的区域,然后针对若干个小区域进行数据处理。

--注意:group by 是先排序后分组;

--举个例子说明:如果要用到group by 一般用到的就是“每这个字” 例如说明现在有一个这样的表:每个部门有多少人就要用到分组的技术

```
select DepartmentID as '部门名称',
```

```
COUNT(*) as '个数' from BasicDepartment group by DepartmentID
```

--这个就是使用了group by +字段进行了分组,其中我们就可以理解为我们按照了部门的名称ID

--DepartmentID将数据集进行了分组;然后再进行各个组的统计数据分别有多少;

--如果不用count(*) 而用类似下面的语法

```
select DepartmentID,DepartmentName from BasicDepartment group by DepartmentID
```

--将会出现错误

--消息 8120, 级别 16, 状态 1, 第 1 行

--选择列表中的列 'BasicDepartment.DepartmentName' 无效,因为该列没有包含在聚合函数或 GROUP BY 子句中。

--这就是我们需要注意的一点,如果在返回集字段中,这些字段要么就要包含在Group By语句的后面,

--作为分组的依据;要么就要被包含在聚合函数中。

--出现的错误详解:咱们看看group by 的执行的过程,先执行select 的操作返回一个程序集,

--然后去执行分组的操作,这时候他将根据group by 后面的字段

--进行分组,并且将相同的字段并称一列数据,如果group by 后面没有这个字段的话就要分成好多的数据。

--但是分组就只能将相同的数据分成两列数据,而一列中又只能放入一个字段,所以那些没有进行分组的

--数据系统不知道将数据放入哪里,所以就出现此错误

--目前一种分组情况只有一条记录,一个数据格是无法放入多个数值的,

--所以这里就需要通过一定的处理将这些多值的列转化成单值,然后将其放在对应的

--数据格中,那么完成这个步骤的就是聚合函数。这就是为什么这些函数叫聚合函数(aggregate functions)了

--group by all语法解析:

--如果使用 ALL 关键字,那么查询结果将包括由 GROUP BY 子句产生的所有组,即使某些组没有符合搜索条件的行。

--没有 ALL 关键字,包含 GROUP BY 子句的 SELECT 语句将不显示没有符合条件的行的组。

```
select DepartmentID,DepartmentName as '部门名称',
```

```
COUNT(*) as '个数' from BasicDepartment group by all DepartmentID,DepartmentName
```

--group by 和having 解释:前提必须了解sql语言中一种特殊的函数:聚合函数,

--例如SUM, COUNT, MAX, AVG等。这些函数和其它函数的根本区别就是它们一般作用在多条记录上。

--WHERE关键字在使用集合函数时不能使用,所以在集合函数中加上了HAVING来起到测试查询结果是否符合条件的作用。

```
create TABLE Table1
```

```
(
```

```
    ID int identity(1,1) primary key NOT NULL,
```

```
    classid int,
```

```
    sex varchar(10),
```

```
    age int,
```

```
)
```

--添加测试数据

```
Insert into Table1 values(1,'男',20)
```

```
Insert into Table1 values(2,'女',22)
```

```
Insert into Table1 values(3,'男',23)
Insert into Table1 values(4,'男',22)
Insert into Table1 values(1,'男',24)
Insert into Table1 values(2,'女',19)
Insert into Table1 values(4,'男',26)
Insert into Table1 values(1,'男',24)
Insert into Table1 values(1,'男',20)
Insert into Table1 values(2,'女',22)
Insert into Table1 values(3,'男',23)
Insert into Table1 values(4,'男',22)
Insert into Table1 values(1,'男',24)
Insert into Table1 values(2,'女',19)
```

--举例子说明：查询table表查询每一个班级中年龄大于20，性别为男的人数

```
select COUNT(*)as '>20岁人数',classid from Table1 where sex='男' group by classid,age having age>20
```

--**需要注意说明**：当同时含有where子句、group by 子句 、having子句及聚集函数时，执行顺序如下：

--执行where子句查找符合条件的数据；

--使用group by 子句对数据进行分组；对group by 子句形成的组运行聚集函数计算每一组的值；最后用having 子句去掉不符合条件的组。

--having 子句中的每一个元素也必须出现在select列表中。有些数据库例外，如oracle。

--having子句和where子句都可以用来设定限制条件以使查询结果满足一定的条件限制。

--having子句限制的是组，而不是行。where子句中不能使用聚集函数，而having子句中可以。