

SQLite Group By

SQLite 的 **GROUP BY** 子句用于与 SELECT 语句一起使用，来对相同的数据进行分组。
在 SELECT 语句中，GROUP BY 子句放在 WHERE 子句之后，放在 ORDER BY 子句之前。

语法

下面给出了 GROUP BY 子句的基本语法。GROUP BY 子句必须放在 WHERE 子句中的条件之后，必须放在 ORDER BY 子句之前。

```
SELECT column-list
FROM table_name
WHERE [ conditions ]
GROUP BY column1, column2....columnN
ORDER BY column1, column2....columnN
```

您可以在 GROUP BY 子句中使用多个列。确保您使用的分组列在列清单中。

实例

假设 COMPANY 表有以下记录：

| ID | NAME | AGE | ADDRESS | SALARY |
|-------|-------|-------|------------|---------|
| ----- | ----- | ----- | ----- | ----- |
| 1 | Paul | 32 | California | 20000.0 |
| 2 | Allen | 25 | Texas | 15000.0 |
| 3 | Teddy | 23 | Norway | 20000.0 |
| 4 | Mark | 25 | Rich-Mond | 65000.0 |
| 5 | David | 27 | Texas | 85000.0 |
| 6 | Kim | 22 | South-Hall | 45000.0 |
| 7 | James | 24 | Houston | 10000.0 |

如果您想了解每个客户的工资总额，则可使用 GROUP BY 查询，如下所示：

```
sqlite> SELECT NAME, SUM(SALARY) FROM COMPANY GROUP BY NAME;
```

这将产生以下结果：

| NAME | SUM(SALARY) |
|-------|-------------|
| ----- | ----- |
| Allen | 15000.0 |

| | |
|-------|---------|
| David | 85000.0 |
| James | 10000.0 |
| Kim | 45000.0 |
| Mark | 65000.0 |
| Paul | 20000.0 |
| Teddy | 20000.0 |

现在，让我们使用下面的 INSERT 语句在 COMPANY 表中另外创建三个记录：

```
INSERT INTO COMPANY VALUES (8, 'Paul', 24, 'Houston', 20000.00 );
INSERT INTO COMPANY VALUES (9, 'James', 44, 'Norway', 5000.00 );
INSERT INTO COMPANY VALUES (10, 'James', 45, 'Texas', 5000.00 );
```

现在，我们的表具有重复名称的记录，如下所示：

| ID | NAME | AGE | ADDRESS | SALARY |
|-------|-------|-------|------------|---------|
| ----- | ----- | ----- | ----- | ----- |
| 1 | Paul | 32 | California | 20000.0 |
| 2 | Allen | 25 | Texas | 15000.0 |
| 3 | Teddy | 23 | Norway | 20000.0 |
| 4 | Mark | 25 | Rich-Mond | 65000.0 |
| 5 | David | 27 | Texas | 85000.0 |
| 6 | Kim | 22 | South-Hall | 45000.0 |
| 7 | James | 24 | Houston | 10000.0 |
| 8 | Paul | 24 | Houston | 20000.0 |
| 9 | James | 44 | Norway | 5000.0 |
| 10 | James | 45 | Texas | 5000.0 |

让我们用同样的 GROUP BY 语句来对所有记录按 NAME 列进行分组，如下所示：

```
sqlite> SELECT NAME, SUM(SALARY) FROM COMPANY GROUP BY NAME ORDER BY NAME;
```

这将产生以下结果：

| NAME | SUM(SALARY) |
|-------|-------------|
| ----- | ----- |
| Allen | 15000 |
| David | 85000 |
| James | 20000 |
| Kim | 45000 |
| Mark | 65000 |
| Paul | 40000 |
| Teddy | 20000 |

让我们把 ORDER BY 子句与 GROUP BY 子句一起使用，如下所示：

```
sqlite> SELECT NAME, SUM(SALARY)
        FROM COMPANY GROUP BY NAME ORDER BY NAME DESC;
```

这将产生以下结果：

| NAME | SUM(SALARY) |
|-------|-------------|
| ----- | ----- |
| Teddy | 20000 |
| Paul | 40000 |
| Mark | 65000 |
| Kim | 45000 |
| James | 20000 |
| David | 85000 |
| Allen | 15000 |

[← SQLite Order By](#)[SQLite Having 子句 →](#)[✍ 点我分享笔记](#)