

## SQLite 视图 ( View )

视图 ( View ) 只不过是通过相关的名称存储在数据库中的一个 SQLite 语句。视图 ( View ) 实际上是一个以预定义的 SQLite 查询形式存在的表的组合。

视图 ( View ) 可以包含一个表的所有行或从一个或多个表选定行。视图 ( View ) 可以从一个或多个表创建，这取决于要创建视图的 SQLite 查询。

视图 ( View ) 是一种虚表，允许用户实现以下几点：

- 用户或用户组查找结构数据的方式更自然或直观。
- 限制数据访问，用户只能看到有限的数据库，而不是完整的表。
- 汇总各种表中的数据，用于生成报告。

SQLite 视图是只读的，因此可能无法在视图上执行 DELETE、INSERT 或 UPDATE 语句。但是可以在视图上创建一个触发器，当尝试 DELETE、INSERT 或 UPDATE 视图时触发，需要做的动作在触发器内容中定义。

### 创建视图

SQLite 的视图是使用 **CREATE VIEW** 语句创建的。SQLite 视图可以从一个单一的表、多个表或其他视图创建。

CREATE VIEW 的基本语法如下：

```
CREATE [TEMP | TEMPORARY] VIEW view_name AS
SELECT column1, column2.....
FROM table_name
WHERE [condition];
```

您可以在 SELECT 语句中包含多个表，这与在正常的 SQL SELECT 查询中的方式非常相似。如果使用了可选的 TEMP 或 TEMPORARY 关键字，则将在临时数据库中创建视图。

### 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

现在，下面是一个从 COMPANY 表创建视图的实例。视图只从 COMPANY 表中选取几列：

```
sqlite> CREATE VIEW COMPANY_VIEW AS
SELECT ID, NAME, AGE
FROM COMPANY;
```

现在，可以查询 COMPANY\_VIEW，与查询实际表的方式类似。下面是实例：

```
sqlite> SELECT * FROM COMPANY_VIEW;
```

这将产生以下结果：

ID	NAME	AGE
-----	-----	-----
1	Paul	32
2	Allen	25
3	Teddy	23
4	Mark	25
5	David	27
6	Kim	22
7	James	24

## 删除视图

要删除视图，只需使用带有 **view\_name** 的 DROP VIEW 语句。DROP VIEW 的基本语法如下：

```
sqlite> DROP VIEW view_name;
```

下面的命令将删除我们在前面创建的 COMPANY\_VIEW 视图：

```
sqlite> DROP VIEW COMPANY_VIEW;
```

← SQLite Truncate Table

SQLite 事务 →

 点我分享笔记

