CREATE TABLE ... AS语句从选择查询中创建一个新表。

预期用途

使用CREATE TABLE ... AS创建的表旨在保留查询的结果，以便以后重用。

当满足以下两个条件时，它比视图更有效：

* 查询结果会多次使用。
* 随着时间的推移，副本不需要与原始表保持同步。

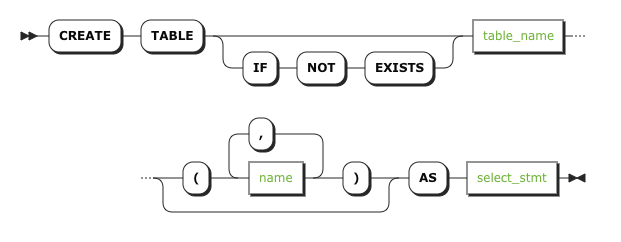
如果在较大的查询中多次重复使用查询结果，则建议使用视图。 查询优化器可以“探视”到视图中，并使用视图查询中提到的表的主键和索引来优化周围的查询。

当结果必须是最新的时，也建议采用视图; 视图始终从视图查询提到的表中检索当前数据。

所需权限

用户必须具有父数据库的CREATE权限。

概要



参数

| **Parameter** | **Description** |
| --- | --- |
| IF NOT EXISTS | 仅当数据库中不存在同名表时才创建新表; 如果存在，不返回错误。  请注意，IF NOT EXISTS仅检查表名; 它不检查现有表是否具有新表的相同列，索引，约束等。 |
| table\_name | 要创建的表的名称，在其数据库中必须是唯一的，并遵循标识符规则。 如果未设置默认父数据库，则必须将名称格式设置为database.name。  UPSERT和INSERT ON CONFLICT语句使用名为excluded的临时表来处理执行期间的唯一性冲突。 因此，建议不要使用表名excluded。 |
| name | 要使用的列的名称，而不是select\_stmt中列的名称。 |
| select\_stmt | 用于提供数据的选择查询。 |

限制

使用CREATE TABLE ... AS创建的表的主键不是从查询结果派生的；与其他表一样，创建后无法添加或更改主键； 而且这些表不与其他表交错；列族的默认规则适用。

举个例子:

> **CREATE** **TABLE** logoff (

user\_id INT **PRIMARY** **KEY**,

user\_email **STRING** **UNIQUE**,

logoff\_date DATE **NOT** NULL,

);

> **CREATE** **TABLE** logoff\_copy **AS** **TABLE** logoff;

> **SHOW** **CREATE** **TABLE** logoff\_copy;

+-------------+-----------------------------------------------------------------+

| Table | CreateTable |

+-------------+-----------------------------------------------------------------+

| logoff\_copy | CREATE TABLE logoff\_copy (␤ |

| | user\_id INT NULL,␤ |

| | user\_email STRING NULL,␤ |

| | logoff\_date DATE NULL,␤ |

| | FAMILY "primary" (user\_id, user\_email, logoff\_date, rowid)␤ |

| | ) |

+-------------+-----------------------------------------------------------------+

(1 row)

该示例说明主键，唯一和“非空”约束不会传播到新的表。

不过可以在CREATE TABLE ... AS之后创建二级索引。

举个例子:

> **CREATE** **INDEX** logoff\_copy\_id\_idx **ON** logoff\_copy(user\_id);

> **SHOW** **CREATE** **TABLE** logoff\_copy;

+-------------+-----------------------------------------------------------------+

| Table | CreateTable |

+-------------+-----------------------------------------------------------------+

| logoff\_copy | CREATE TABLE logoff\_copy (␤ |

| | user\_id INT NULL,␤ |

| | user\_email STRING NULL,␤ |

| | logoff\_date DATE NULL,␤ |

| | INDEX logoff\_copy\_id\_idx (user\_id ASC),␤ |

| | FAMILY "primary" (user\_id, user\_email, logoff\_date, rowid)␤ |

| | ) |

+-------------+-----------------------------------------------------------------+

(1 row)

为了最大化优化数据存储，请考虑单独使用CREATE后使用INSERT INTO ...来使用查询结果填充表。

示例

从SELECT查询创建一个表

> **SELECT** \* **FROM** customers **WHERE** state = 'NY';

+----+---------+-------+

| id | name | state |

+----+---------+-------+

| 6 | Dorotea | NY |

| 15 | Thales | NY |

+----+---------+-------+

> **CREATE** **TABLE** customers\_ny **AS** **SELECT** \* **FROM** customers **WHERE** state = 'NY';

> **SELECT** \* **FROM** customers\_ny;

+----+---------+-------+

| id | name | state |

+----+---------+-------+

| 6 | Dorotea | NY |

| 15 | Thales | NY |

+----+---------+-------+

更改列名

此语句创建现有表的副本，但具有更改后的的列名。

> **CREATE** **TABLE** customers\_ny (id, first\_name) **AS** **SELECT** id, name **FROM** customers **WHERE** state = 'NY';

> **SELECT** \* **FROM** customers\_ny;

+----+------------+

| id | first\_name |

+----+------------+

| 6 | Dorotea |

| 15 | Thales |

+----+------------+

从VALUES子句创建一个表

> **CREATE** **TABLE** tech\_states **AS** **VALUES** ('CA'), ('NY'), ('WA');

> **SELECT** \* **FROM** tech\_states;

+---------+

| column1 |

+---------+

| CA |

| NY |

| WA |

+---------+

(3 rows)

创建现有表的副本

> **CREATE** **TABLE** customers\_ny\_copy **AS** **TABLE** customers\_ny;

> **SELECT** \* **FROM** customers\_ny\_copy;

+----+------------+

| id | first\_name |

+----+------------+

| 6 | Dorotea |

| 15 | Thales |

+----+------------+

以这种方式创建表副本时，副本不会与原始表上的任何主键，二级索引或约束相关联。

See Also

* [Selection Queries](http://doc.cockroachchina.baidu.com/#develop/sql-syntax/selection-queries/)
* [Simple SELECT Clause](http://doc.cockroachchina.baidu.com/#develop/sql-statements/SELECT/)
* [CREATE TABLE](http://doc.cockroachchina.baidu.com/#develop/sql-statements/CREATE-TABLE/)
* [CREATE VIEW](http://doc.cockroachchina.baidu.com/#develop/sql-statements/CREATE-VIEW/)
* [INSERT](http://doc.cockroachchina.baidu.com/#develop/sql-statements/INSERT/)
* [DROP TABLE](http://doc.cockroachchina.baidu.com/#develop/sql-statements/DROP-TABLE/)
* [Other SQL Statements](http://doc.cockroachchina.baidu.com/#develop/sql-statements/overview/)