Primary Key约束指定了该列的值在表中必须唯一。

不同于其他约束用在特定用途，主键约束在每个表都必须使用，因为它为表的数据提供了内在的结构。这不仅让表更容易被理解，也提升了查询性能。

主键只能在创建表时指定，且后续无法通过alter table来修改。不过可以创建一个包含新主键的表，然后再迁移数据。

细节

* 每个表只能有一个主键。
* 来确保表中每一行都有唯一的标识，Primary Key约束组合了Unique和Not Null约束的属性。这两个约束的属性对于确保每行的主键列包含不同的值是必要的。
* Unique约束的属性确保每个值与所有其他值不同。
* 不过，由于NULL值和其他NULL值永远不相等，所以只有Unique约束不够（如果其中一个值为NULL，则两行可以显示相同）。为了避免出现重复的值，主键约束强制使用Not Null 约束。
* 主键约束中的列用于创建其主索引，CockroachDB默认使用该索引来访问表的数据。

此索引不会占用额外的磁盘空间（与二级索引不同），因为CockroachDB使用主索引来构建kv层中的表数据。更多信息查看 [SQL in CockroachDB: Mapping Table Data to Key-Value Storage](https://www.cockroachlabs.com/blog/sql-in-cockroachdb-mapping-table-data-to-key-value-storage/).

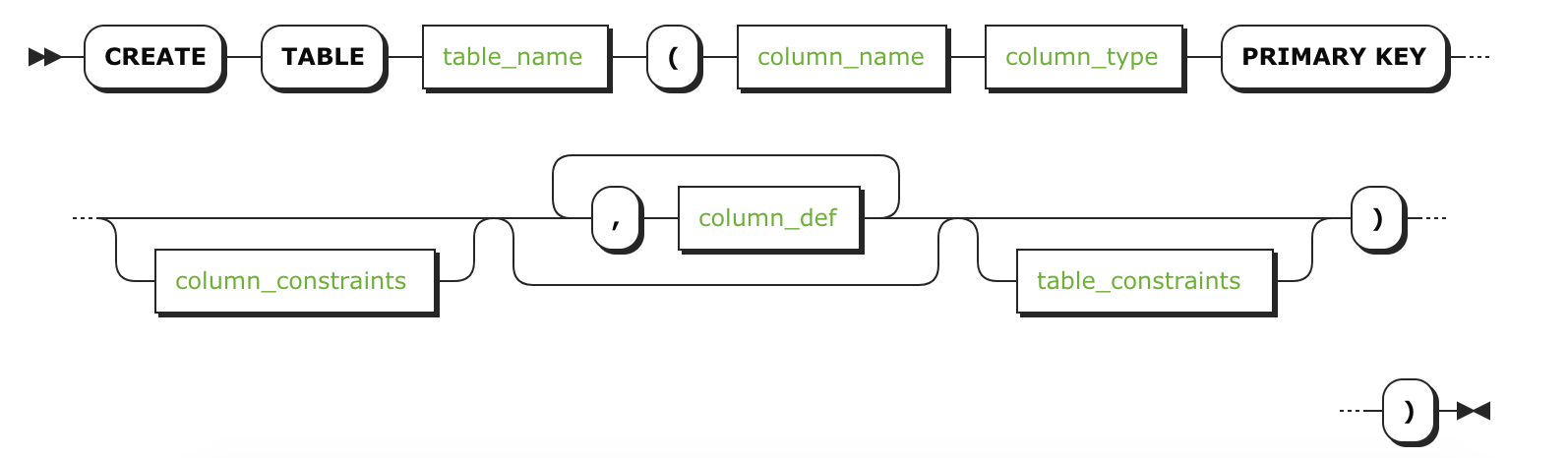
* 为了最佳性能，我们推荐为每个表定义主键。

如果创建一个表而没有定义主键，CockroachDB会为每一行使用唯一标识符，然后将其用作主索引。因为你无法有意义地使用此唯一行标识符列来过滤表数据，所以它不提供任何性能优化。这意味着通过定义表的主键，你将获得更高的性能，更多信息查看：[Index Selection in CockroachDB](https://www.cockroachlabs.com/blog/index-selection-cockroachdb-2/).

语法

可以在表级定义主键约束，不过，如果希望将约束应用于单个列，也可以在列级别应用该约束。

列级别



| **Parameter** | **Description** |
| --- | --- |
| table\_name | 表名 |
| column\_name | 主键列名 |
| column\_type | 主键列的数据类型 |
| column\_constraints | 要应用于此列的任何其他列级约束。 |
| column\_def | 表中任何其他列的定义。 |
| table\_constraints | 要应用的任何表级约束。 |

**举例**

> **CREATE** **TABLE** orders (

order\_id INT **PRIMARY** **KEY**,

order\_date **TIMESTAMP** **NOT** NULL,

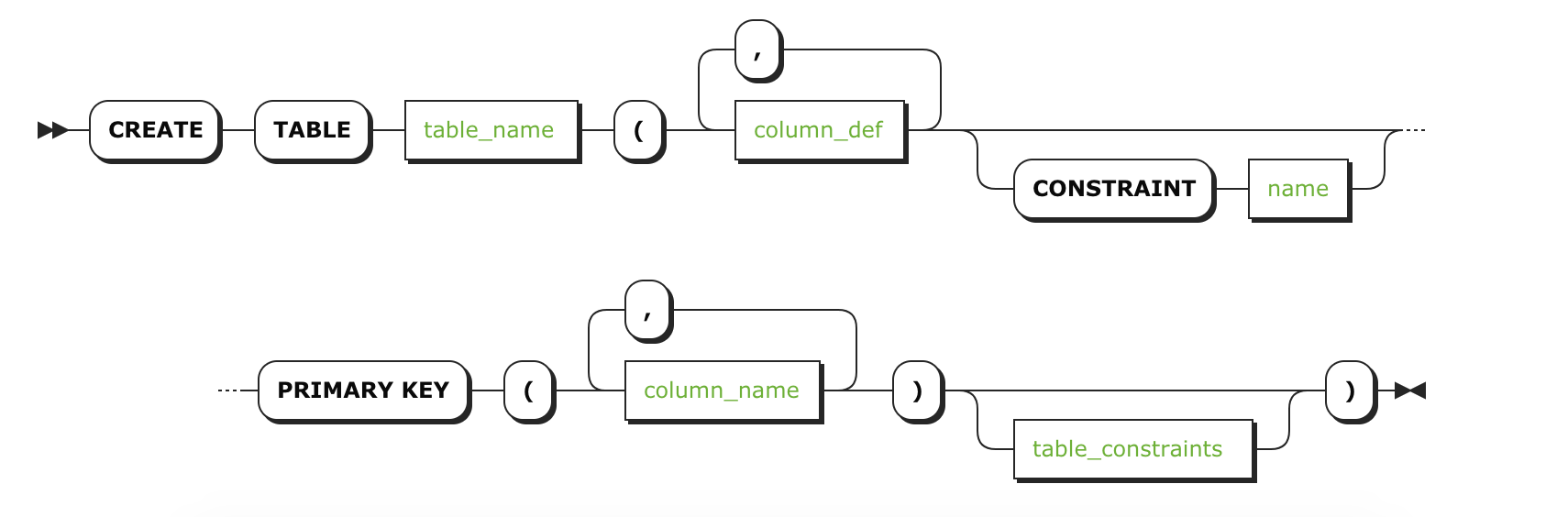
order\_mode **STRING**(8),

customer\_id INT,

order\_status INT

);

表级别



| **Parameter** | **Description** |
| --- | --- |
| table\_name | 表名 |
| column\_def | 表中任何其他列的定义。 |
| name | 约束名，必须在表中为，且满足[identifier rules](http://doc.cockroachchina.baidu.com/#develop/sql-syntax/keywords-and-identifiers/#identifiers). |
| column\_name | 要用作主键的列的名称.  列出的列名的顺序会影响primary索引的结构。 |
| table\_constraints | 要应用的任何表级约束。 |

**举例**

> **CREATE** **TABLE** **IF** **NOT** **EXISTS** inventories (

product\_id INT,

warehouse\_id INT,

quantity\_on\_hand INT **NOT** NULL,

**PRIMARY** **KEY** (product\_id, warehouse\_id)

);

用法示例

> **CREATE** **TABLE** **IF** **NOT** **EXISTS** inventories (

product\_id INT,

warehouse\_id INT,

quantity\_on\_hand INT **NOT** NULL,

**PRIMARY** **KEY** (product\_id, warehouse\_id)

);

> **INSERT** **INTO** inventories **VALUES** (1, 1, 100);

> **INSERT** **INTO** inventories **VALUES** (1, 1, 200);

pq: duplicate key value (product\_id,warehouse\_id)=(1,1) violates unique constraint "primary"

> **INSERT** **INTO** inventories **VALUES** (1, NULL, 100);

pq: null value in column "warehouse\_id" violates not-null constraint

See Also

* [Constraints](http://doc.cockroachchina.baidu.com/#develop/constraints/overview/)
* [Check constraint](http://doc.cockroachchina.baidu.com/#develop/constraints/check/)
* [Default Value constraint](http://doc.cockroachchina.baidu.com/#develop/constraints/default-value/)
* [Foreign Key constraint](http://doc.cockroachchina.baidu.com/#develop/constraints/foreign-key/)
* [Not Null constraint](http://doc.cockroachchina.baidu.com/#develop/constraints/not-null/)
* [Unique constraint](http://doc.cockroachchina.baidu.com/#develop/constraints/unique/)
* [SHOW CONSTRAINTS](http://doc.cockroachchina.baidu.com/#develop/sql-statements/SHOW-CONSTRAINTS/)