SQL概述

更新时间：2020-04-15 16:52:00

[**本页目录**](javascript:void(0))

* [SQL概述](https://help.aliyun.com/document_detail/27860.html?spm=a2c4g.11186623.6.690.18e7559bBS79XW#title-t49-jfd-z70)
* [关键字](https://help.aliyun.com/document_detail/27860.html?spm=a2c4g.11186623.6.690.18e7559bBS79XW#title-0kl-wv0-dgw)
* [类型转换说明](https://help.aliyun.com/document_detail/27860.html?spm=a2c4g.11186623.6.690.18e7559bBS79XW#title-l39-11h-6h8)
* [分区表](https://help.aliyun.com/document_detail/27860.html?spm=a2c4g.11186623.6.690.18e7559bBS79XW#title-q1g-d5c-lxa)
* [UNION ALL](https://help.aliyun.com/document_detail/27860.html?spm=a2c4g.11186623.6.690.18e7559bBS79XW#title-453-vyu-i7u)
* [Select Transform](https://help.aliyun.com/document_detail/27860.html?spm=a2c4g.11186623.6.690.18e7559bBS79XW#title-qcw-mz2-0pl)
* [使用限制](https://help.aliyun.com/document_detail/27860.html?spm=a2c4g.11186623.6.690.18e7559bBS79XW#title-xxz-wc9-q15)

本文为您介绍MaxCompute SQL的关键字、类型转换说明、分区表、UNION ALL运算及使用限制。

**SQL概述**

MaxCompute SQL适用于海量数据（GB、TB、EB级别）、离线批量计算的场合。MaxCompute作业提交后会有几十秒到数分钟不等的排队调度，所以适合处理运行批作业：一次作业批量处理海量数据，不适合直接对接需要每秒处理几千至数万笔事务的前台业务系统。

MaxCompute SQL采用的是类似于SQL的语法。它的语法是标准语法ANSI SQL92的一个子集，并有自己的扩展。但不能因此简单地把MaxCompute等价成一个数据库，它在很多方面并不具备数据库的特征，如事务、主键约束、索引等，更多差异请参见[与其他SQL语法的差异](https://help.aliyun.com/document_detail/54051.html#concept-fsf-mvf-vdb)。目前在MaxCompute中允许的最大SQL长度是2MB。

**关键字**

MaxCompute将SQL语句的关键字作为保留字。在对表、列或是分区命名时如若使用关键字，需给关键字加``符号进行转义，否则会报错。保留字不区分大小写。下面只给出常用的保留字列表，完整的保留字列表请参见[MaxCompute SQL保留字](https://help.aliyun.com/document_detail/27872.html" \l "concept-zxk-v5f-vdb" \o "本文为您介绍MaxCompute SQL中的所有保留字与关键字。)。

% & && ( ) \* +

- . / ; < <= <>

= > >= ? **ADD** **ALL** **ALTER**

**AND** **AS** **ASC** **BETWEEN** **BIGINT** **BOOLEAN** **BY**

**CASE** **CAST** **COLUMN** **COMMENT** **CREATE** **DESC** **DISTINCT**

DISTRIBUTE **DOUBLE** **DROP** **ELSE** **FALSE** **FROM** **FULL**

**GROUP** **IF** **IN** **INSERT** **INTO** **IS** **JOIN**

LEFT LIFECYCLE **LIKE** **LIMIT** MAPJOIN **NOT** **NULL**

**ON** **OR** **ORDER** **OUTER** OVERWRITE **PARTITION** **RENAME**

REPLACE RIGHT RLIKE **SELECT** SORT STRING **TABLE**

**THEN** TOUCH **TRUE** **UNION** **VIEW** **WHEN** **WHERE**

**类型转换说明**

MaxCompute SQL允许数据类型之间的转换，类型转换方式包括显式类型转换和隐式类型转换。更多详情请参见[类型转换](https://help.aliyun.com/document_detail/57792.html" \l "concept-wyb-sgl-vdb" \o "MaxCompute SQL允许数据类型之间的转换，类型转换方式包括显式类型转换和隐式类型转换。)。

* 显式类型转换：是指用**cast**函数将一种数据类型的值转换为另一种类型的值的行为。
* 隐式类型转换：是指在运行时，由MaxCompute依据上下文使用环境及类型转换规则自动进行的类型转换。隐式转换作用域包括各种运算符、内建函数等。

**分区表**

MaxCompute SQL支持分区表。指定分区表会对您带来诸多便利，例如提高SQL运行效率、减少计费等。关于分区的详情请参见[基本概念>分区](https://help.aliyun.com/document_detail/27820.html" \l "concept-qlm-lz1-5db" \o "分区表是指拥有分区空间的表，即在创建表时指定表内的一个或者某几个字段作为分区列。分区表实际就是对应分布式文件系统上的独立的文件夹，一个分区对应一个文件夹，文件夹下是对应分区所有的数据文件。)。

**UNION ALL**

参与UNION ALL运算的所有列的数据类型、列个数、列名称必须完全一致，否则会报异常。

**Select Transform**

Select Transform功能明显简化了对脚本代码的引用，支持Java、Python、Shell、Perl等语言，且编写过程简单，适合Adhoc功能的实现。详情请参见[SELECT TRANSFORM语法](https://help.aliyun.com/document_detail/73719.html" \l "concept-a45-b1c-wdb" \o "SELECT TRANSFORM语法允许您启动一个指定的子进程，将输入数据按照一定的格式通过标准输入至子进程，并且通过解析子进程的标准输出获取输出数据。SELECT TRANSFORM让您无需编写UDF，便可以实现MaxCompute SQL对其它脚本语言的支持。)。

目前MaxCompute的SELECT TRANSFORM完全兼容了Hive的语法、功能和行为，包括input/output row format以及reader/writer。Hive上的脚本，大部分可以直接运行，部分脚本只需要经过稍微的改动即可运行。

本文将MaxCompute SQL与Hive、MySQL、Oracle、SQL Server的SQL语法进行对比，为您介绍MaxCompute不支持的语法。

## MaxComputeDDL语法差异对比

| **语法** | **MaxCompute** | **Hive** | **MySQL** | **Oracle** | **SQL Server** |
| --- | --- | --- | --- | --- | --- |
| CREATE TABLE—PRIMARY KEY | N | N | Y | Y | Y |
| CREATE TABLE—NOT NULL | Y | N | Y | Y | Y |
| CREATE TABLE—CLUSTER BY | Y | Y | N | Y | Y |
| CREATE TABLE—EXTERNAL TABLE | Y (OSS, OTS, TDDL) | Y | N | Y | N |
| CREATE TABLE—TEMPORARY TABLE | N | Y | Y | Y | Y （with #prefix） |
| INDEX—CREATE INDEX | N | Y | Y | Y | Y |
| VIRTUAL COLUMN | N | N | N | Y | Y |

## MaxComputeDML语法差异对比

| **语法** | **MaxCompute** | **Hive** | **MySQL** | **Oracle** | **SQL Server** |
| --- | --- | --- | --- | --- | --- |
| CTE | Y | Y | Y | Y | Y |
| SELECT—recursive CTE | N | N | N | Y | Y |
| SELECT—GROUP BY ROLL UP | Y | Y | Y | Y | Y |
| SELECT—GROUP BY CUBE | Y | Y | N | Y | Y |
| SELECT—GROUPING SET | Y | Y | N | Y | Y |
| SELECT—IMPLICT JOIN | Y | Y | N | Y | Y |
| SELECT—PIVOT | N | N | N | Y | Y |
| SEMI JOIN | Y | Y | Y | N | N |
| SELEC TRANSFROM | Y | Y | N | N | N |
| SELECT—corelated subquery | Y | Y | Y | Y | Y |
| ORDER BY NULLS FIRST/LAST | N | Y | Y | Y | Y |
| LATERAL VIEW | Y | Y | N | Y | Y (CROSS APPLY keyword) |
| SET OPERATOR—UNION (disintct) | Y | Y | Y | Y | Y |
| SET OPERATOR—INTERSECT | Y | N | N | Y | Y |
| SET OPERATOR—MINUS/EXCEPT | Y | N | N | Y | Y（keyword EXCEPT） |
| INSERT INTO ... VALUES | Y | Y | Y | Y | Y |
| INSERT INTO (ColumnList) | Y | Y | Y | Y | Y |
| UPDATE … WHERE | N | Y | Y | Y | Y |
| UPDATE … ORDER BY LIMIT | N | N | Y | N | Y |
| DELETE … WHERE | N | Y | Y | Y | Y |
| DELETE … ORDER BY LIMIT | N | N | Y | N | N |
| ANALYTIC—reusable WINDOWING CLUSUE | Y | Y | N | N | N |
| ANALYTIC—CURRENT ROW | Y | Y | N | Y | Y |
| ANALYTIC—UNBOUNDED | Y | N | Y | Y | Y |
| ANALYTIC—RANGE … | N | Y | N | Y | Y |
| WHILE DO | N | N | Y | Y | Y |
| [VIEW WITH PARAMETERS](https://help.aliyun.com/document_detail/123145.html#concept-827322) | Y | N | N | N  **说明** SQL Function/Storage Procedure一定程度上能满足MaxCompute参数化VIEW的功能。 | N |

## MaxCompute SCRIPTING语法差异对比

| **语法** | **MaxCompute** | **Hive** | **MySQL** | **Oracle** | **SQL Server** |
| --- | --- | --- | --- | --- | --- |
| TABLE VARIABLE | Y | Y | Y | Y | Y |
| SCALER VARIABLE | Y | Y | Y | Y | Y |
| ERROR HANDLING—RAISE ERROR | N | N | Y | Y | Y |
| ERROR HANDLING—TRY CATCH | N | N | N | Y | Y |
| FLOW CONTROL—LOOP | N | N | Y | Y | Y |
| CURSOR | N | N | Y | Y | Y |