**一些最重要的 SQL 命令**

* **SELECT** - 从数据库中提取数据
* **UPDATE** - 更新数据库中的数据
* **DELETE** - 从数据库中删除数据
* **INSERT INTO** - 向数据库中插入新数据
* **CREATE DATABASE** - 创建新数据库
* **ALTER DATABASE** - 修改数据库
* **CREATE TABLE** - 创建新表
* **ALTER TABLE** - 变更（改变）数据库表
* **DROP TABLE** - 删除表
* **CREATE INDEX** - 创建索引（搜索键）
* **DROP INDEX** - 删除索引



* **示例：**

SELECT语句

句法：

SELECT column\_name(s) FROM table\_name

SELECT语句和WHERE子句

句法：

SELECT [\*] FROM [TableName] WHERE [condition1]

SELECT语句与WHERE和/或子句

句法：

SELECT [\*] FROM [TableName] WHERE [condition1] [AND [OR]] [condition2]...

SELECT语句与ORDER BY

句法：

SELECT column\_name()

FROM table\_name

ORDER BY column\_name() ASC or DESC

* asc是指定列按升序排列,desc则是指定列按降序排列

**1.select \* 选择所有**

**2. SELECT DISTINCT（选择不同）可用于消除重复数据**

SELECT DISTINCT语句用于仅返回不同的（different）值。

SELECT DISTINCT column1, column2, ...

FROM table\_name;

**3. WHERE Clause（查询子句）**

WHERE 子句用于过滤记录

WHERE子句用于提取满足指定标准的记录

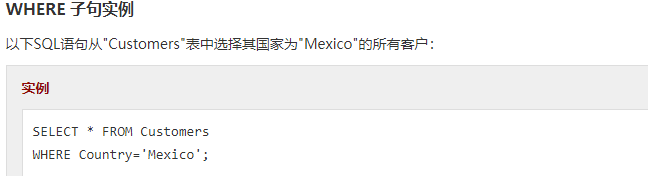
SQL WHERE 语法

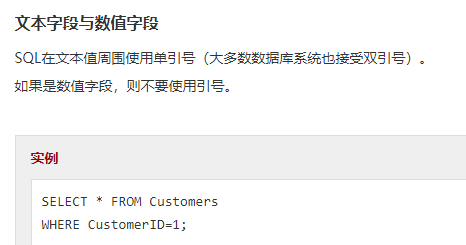
SELECT column1, column2, ...

FROM table\_name

WHERE condition;

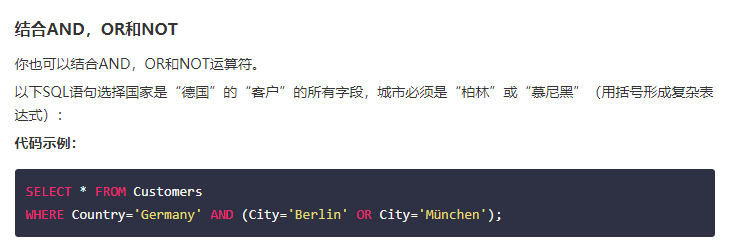
* 注意： WHERE子句不仅用于SELECT语法，还用于UPDATE，DELETE语法等！







**4. AND, OR and NOT（与，或不是运算符）**



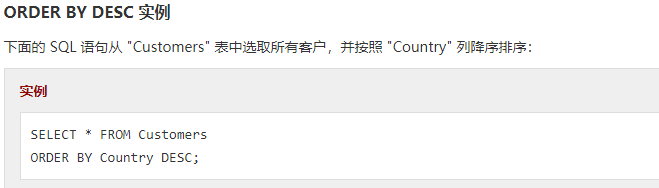
**5. ORDER BY Keyword（按关键字排序）**

**ORDER BY 关键字**

**ORDER BY 关键字用于按升序或降序对结果集进行排序。**

**ORDER BY 关键字默认情况下按升序排序记录。**

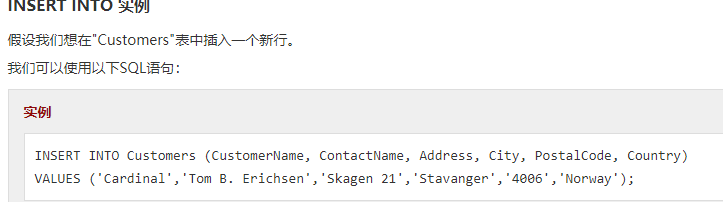
**如果需要按降序对记录进行排序，可以使用DESC关键字**





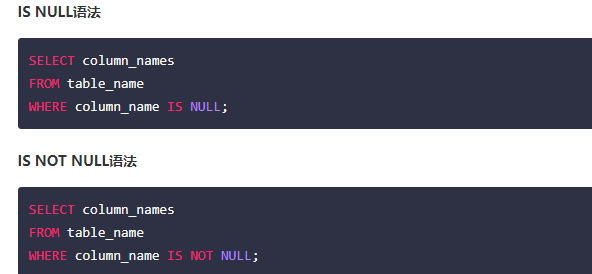
**6.** **INSERT INTO 语句（在表中插入）**

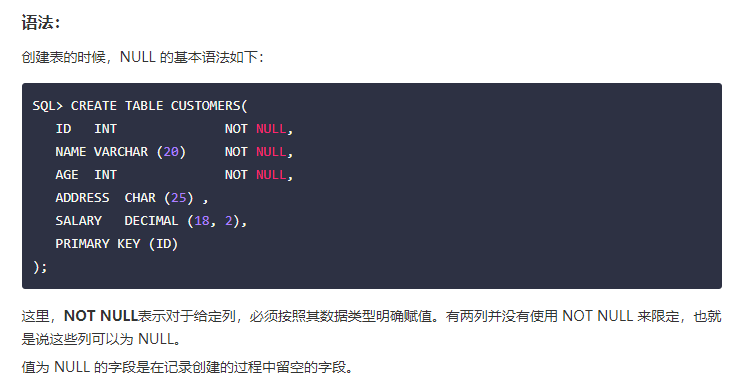
**INSERT INTO 语句用于向表中插入新记录。**



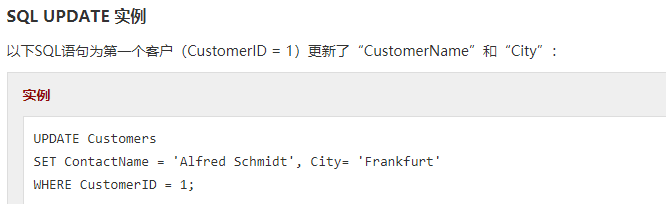


**7. NULL Values（空值）**





**8. UPDATE 语句（更新表中的记录）**



**9.** **Delete 语句（删除表中的记录）**



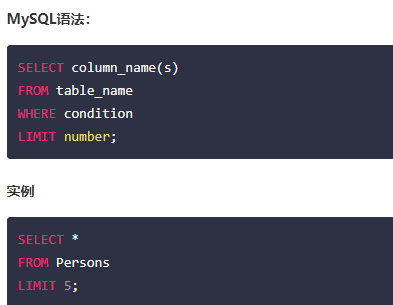
**10. SELECT TOP, LIMIT, ROWNUM**

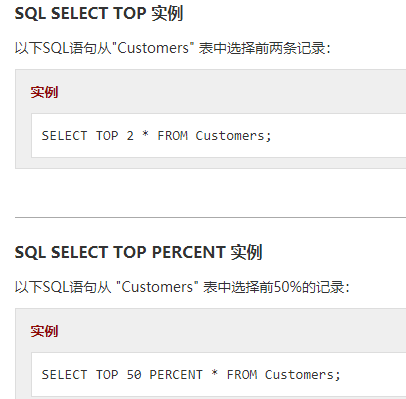
**SELECT TOP 子句**

**SELECT TOP 子句用于指定要返回的记录数量。**

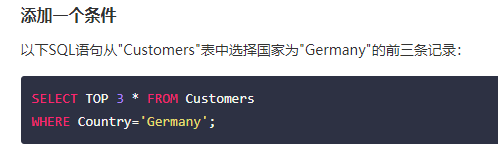
**SELECT TOP子句在包含数千条记录的大型表上很有用。返回大量记录会影响性能。**

**注：并不是所有的数据库系统都支持SELECT TOP子句。MySQL支持LIMIT子句来选择有限数量的记录，而Oracle使用ROWNUM。**









**11.** **LIKE 运算符**

**在WHERE子句中使用LIKE运算符来搜索列中的指定模式。**

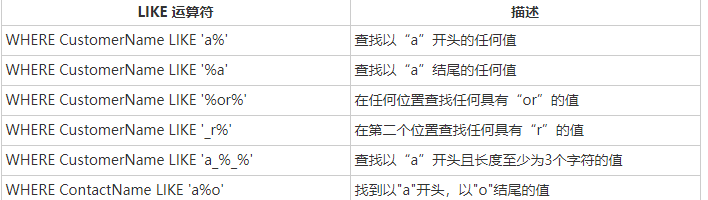
**有两个通配符与LIKE运算符一起使用：**

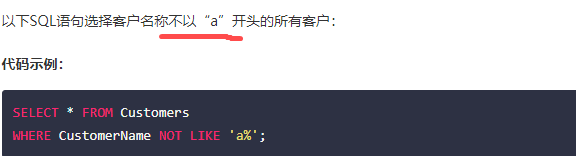
**％ - 百分号表示零个，一个或多个字符**

**\_ - 下划线表示单个字符**

**注意： MS Access使用问号（?）而不是下划线（\_）。**

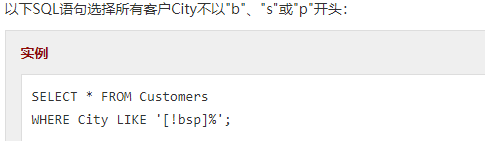
**百分号和下划线也可以组合使用！**





**补充通配符 [ ]**







**11.** **IN 运算符**

**IN运算符允许您在WHERE子句中指定多个值。**

**IN运算符是多个OR条件的简写**

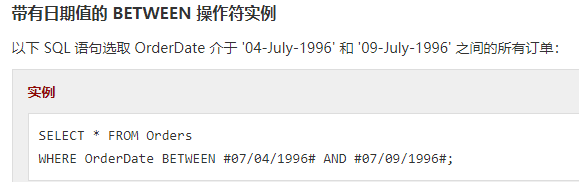


**12.** **BETWEEN运算符**

**BETWEEN运算符选择给定范围内的值。值可以是数字，文本或日期。**

**BETWEEN运算符是包含性的：包括开始和结束值。**





**12.** **AS (alias) 可以为表名称或列名称指定别名**

**SQL 别名用于为表或表中的列提供临时名称。**

**SQL 别名通常用于使列名更具可读性。**

**SQL 一个别名只存在于查询期间。**

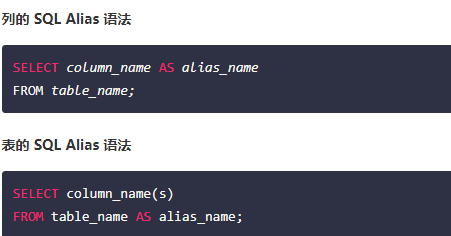
**SELECT column\_name AS column\_alias**

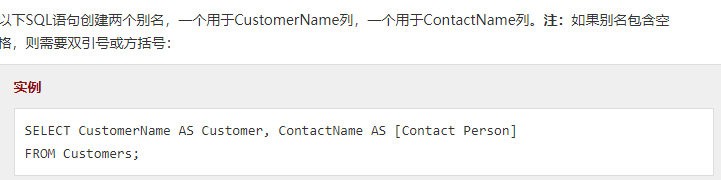
**FROM table\_name**

**or**

**SELECT column\_name**

**FROM table\_name AS table\_alias**







**13.** **join 用于把来自两个或多个表的行结合起来。**

**JOIN 子句用于把来自两个或多个表的行结合起来，基于这些表之间的共同字段。**

**最常见的 JOIN 类型：SQL INNER JOIN（简单的 JOIN）。 SQL INNER JOIN 从多个表中返回满足 JOIN 条件的所有行。**

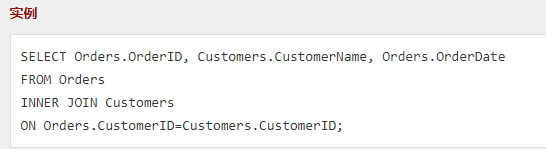
**INNER JOIN：如果表中有至少一个匹配，则返回行**

**LEFT JOIN：即使右表中没有匹配，也从左表返回所有的行**

**RIGHT JOIN：即使左表中没有匹配，也从右表返回所有的行**

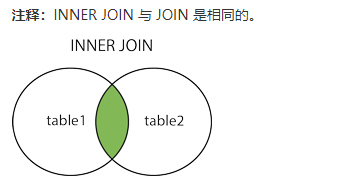
**FULL JOIN：只要其中一个表中存在匹配，则返回行**

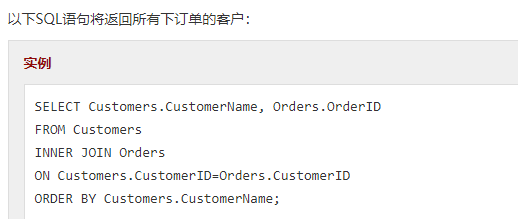
https://www.w3cschool.cn/sql/6d5ftfp4.html

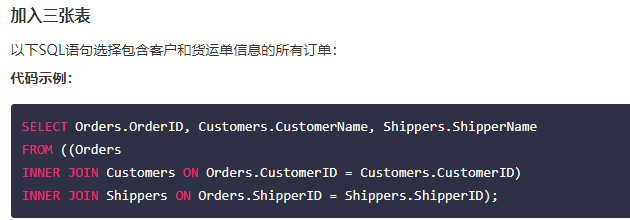


* **INNER JOIN 关键字（内部连接）**

内部链接INNER JOIN关键字选择两个表中具有匹配值的记录。

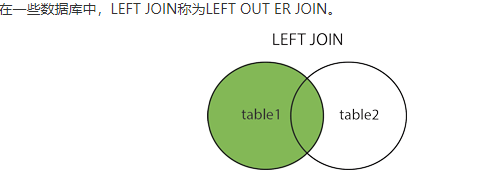


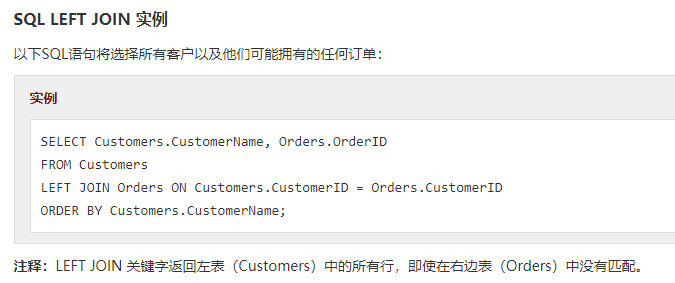




* **LEFT JOIN 关键字**

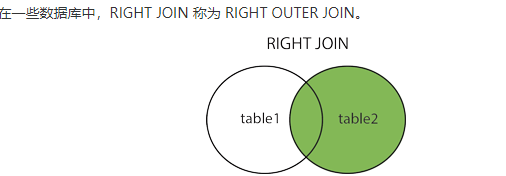
**SQL左链接LEFT JOIN关键字返回左表（表1）中的所有行，即使在右表（表2）中没有匹配。如果在正确的表中没有匹配，结果是NULL。**

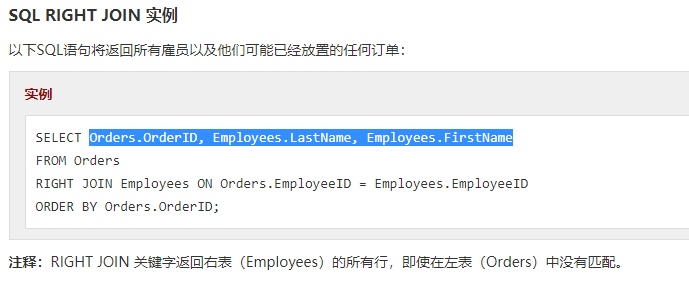




* **RIGHT JOIN 关键字**

**右链接 RIGHT JOIN 关键字返回右表（table2）的所有行，即使在左表（table1）上没有匹配。如果左表没有匹配，则结果为NULL。**

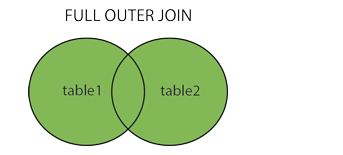




**指的是select中选取的会呈现一些空值NULL**

* **FULL OUTER JOIN 关键字（完整外部连接）**

**当左（表1）或右（表2）表记录匹配时，FULL OUTER JOIN关键字将返回所有记录**



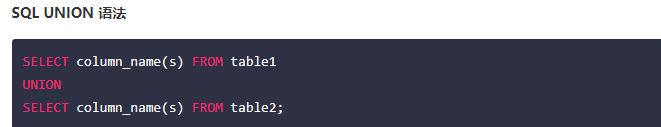
**14.** **UNION 运算符**

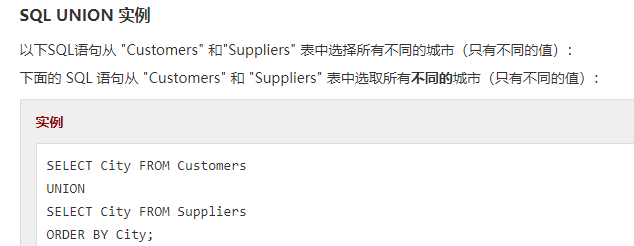
**UNION运算符用于组合两个或更多SELECT语句的结果集。**

**UNION中的每个SELECT语句必须具有相同的列数，这些列也必须具有相似的数据类型**

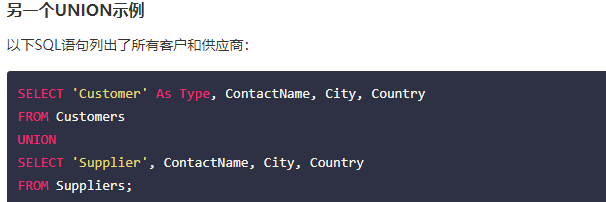
**每个SELECT语句中的列也必须以相同的顺序排列**

即：在使用 UNION 的时候，每个 SELECT 语句必须有相同数量的选中列、相同数量的列表达式、相同的数据类型，并且它们出现的次序要一致，不过长度不一定要相同。



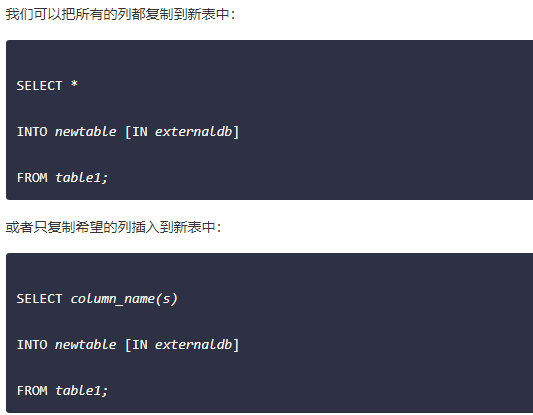


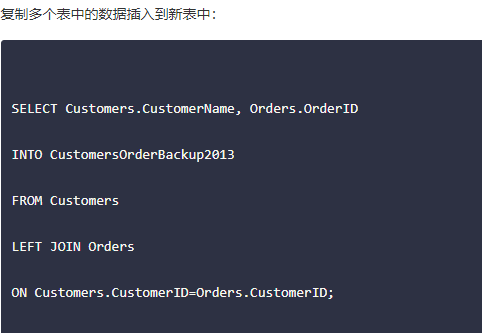




**15.** **SELECT INTO 语句**

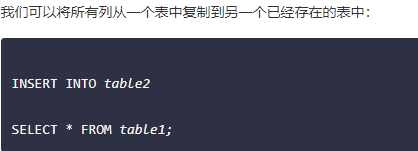
**SELECT INTO 语句从一个表复制数据，然后把数据插入到另一个新表中。**

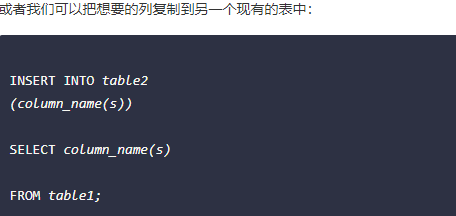




**16.** **INSERT INTO SELECT 语句**

**语句从表中复制数据，并将数据插入现有的表中。目标表中的任何现有行都不会受到影响。**







**17.** **SQL约束用于指定表中数据的规则。**

**SQL 约束**

**约束是作用于数据表中列上的规则，用于限制表中数据的类型。约束的存在保证了数据库中数据的精确性和可靠性。**

**约束有列级和表级之分，列级约束作用于单一的列，而表级约束作用于整张数据表。**

**下面是 SQL 中常用的约束，这些约束虽然已经在关系型数据库管理系统一章中讨论过了，但是仍然值得在这里回顾一遍。**

**NOT NULL 约束：保证列中数据不能有 NULL 值**

**DEFAULT 约束：提供该列数据未指定时所采用的默认值**

**UNIQUE 约束：保证列中的所有数据各不相同**

**主键约束：唯一标识数据表中的行/记录**

**外键约束：唯一标识其他表中的一条行/记录**

**CHECK 约束：此约束保证列中的所有值满足某一条件**

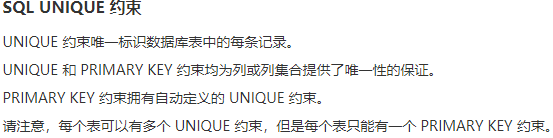
**索引：用于在数据库中快速创建或检索数据**

**约束可以在创建表时规定（通过 CREATE TABLE 语句），或者在表创建之后规定（通过 ALTER TABLE 语句）**

**完整性约束**

**完整性约束用于保证关系型数据库中数据的精确性和一致性。对于关系型数据库来说，数据完整性由参照完整性（referential integrity，RI）来保证。**

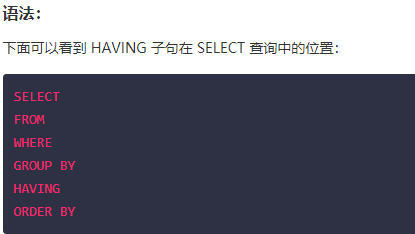
**有很多种约束可以起到参照完整性的作用，这些约束包括主键约束（Primary Key）、外键约束（Foreign Key）、唯一性约束（Unique Constraint）以及上面提到的其他约束。**

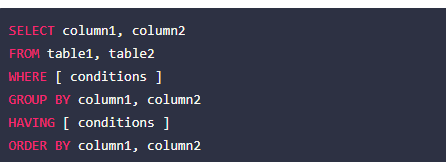


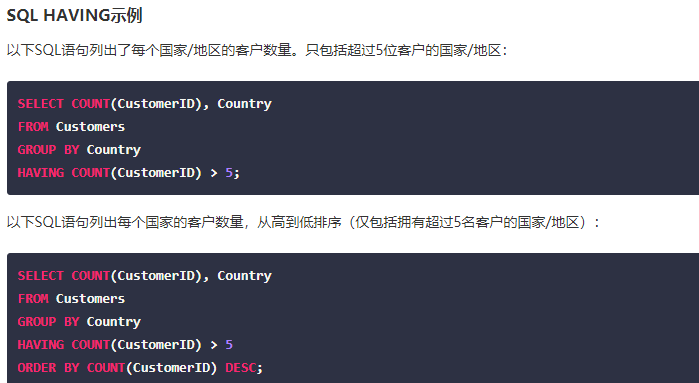
**18.** **HAVING 子句**

**HAVING 子句使你能够指定过滤条件，从而控制查询结果中哪些组可以出现在最终结果里面。**

**WHERE 子句对被选择的列施加条件，而 HAVING 子句则对 GROUP BY 子句所产生的组施加条件。**



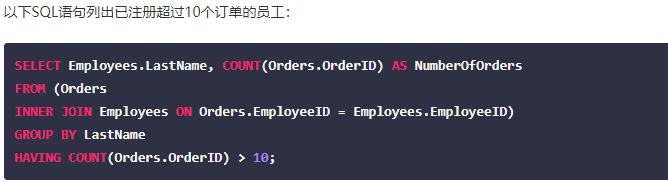


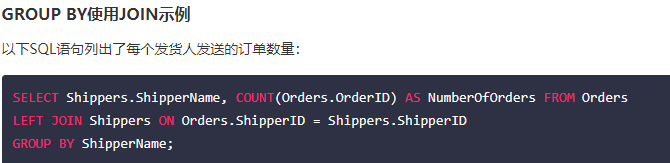


**19.** **GROUP BY语句**

**通常与集合函数（COUNT，MAX，MIN，SUM，AVG）一起使用，以按一个或多个列对结果集进行分组。**







**SQL函数**

**AVG() - 返回平均值**

**COUNT() - 返回行数**

**FIRST() - 返回第一个记录的值**

**LAST() - 返回最后一个记录的值**

**MAX() - 返回最大值**

**MIN() - 返回最小值**

**SUM() - 返回总和**

**有用的 Scalar 函数：**

**UCASE() - 将某个字段转换为大写**

**LCASE() - 将某个字段转换为小写**

**MID() - 从某个文本字段提取字符**

**LEN() - 返回某个文本字段的长度**

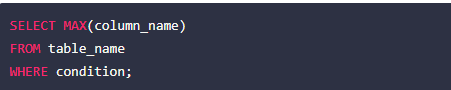
**ROUND() - 对某个数值字段进行指定小数位数的四舍五入**

**NOW() - 返回当前的系统日期和时间**

**FORMAT() - 格式化某个字段的显示方式**

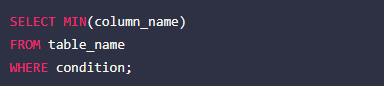
**1.** **MAX() 函数**

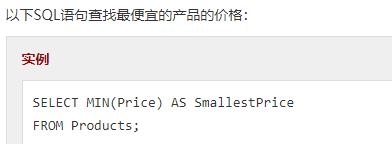
**MAX() 函数返回所选列的最大值。**



**2.** **MIN() 函数**

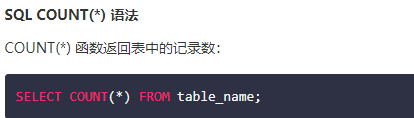
**MIN() 函数返回所选列的最小值。**

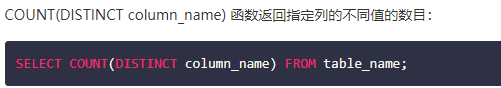




**3.** **COUNT() 函数**

**函数返回符合指定条件的行数。**







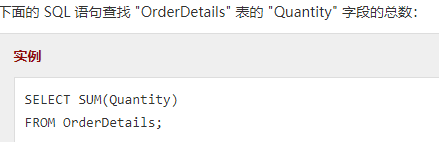
**4.** **AVG() 函数**

**AVG() 函数返回数字列的平均值。**



**5.** **SUM() 函数**

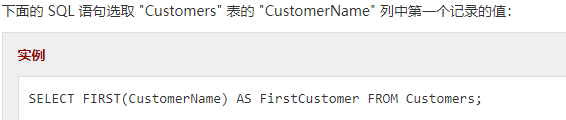
**SUM() 函数返回数字列的总和。**



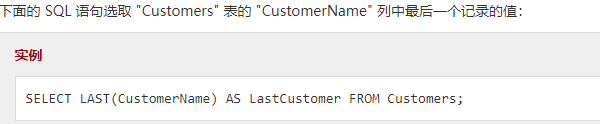
**6.日期函数**

[**https://www.w3cschool.cn/sql/82rg1ozi.html**](https://www.w3cschool.cn/sql/82rg1ozi.html)

**7.** **SQL FIRST() 实例**



**LAST() Example**

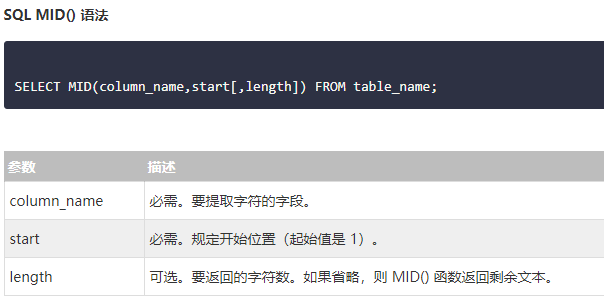


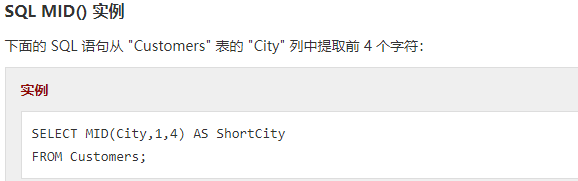
**8.** **SQL 字母大小写转换函数UPPER(s)、UCASE(s)、LOWER(s)和LCASE(s)**

* **LOWER(s)或者LCASE(s)函数可以将字符串s中的字母字符全部转换成小写字母。**
* **UPPER(s)或UCASE(s)函数可以将字符串s中的字母字符全部转换成大写字母。**

**9.** **MID() 函数**

**MID() 函数用于从文本字段中提取字符。**



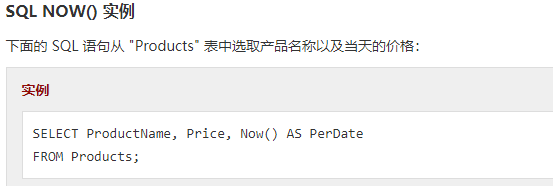


**10.** **LEN() 函数**

**函数返回文本字段中值的长度**

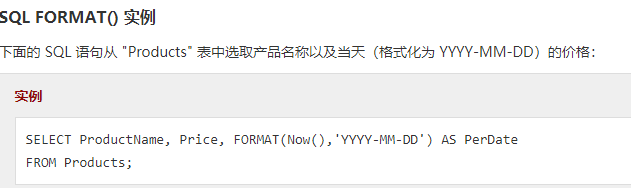
**11.NOW() 函数**

**NOW() 函数返回当前系统的日期和时间。**



**12.** **FORMAT() 函数**

**FORMAT() 函数用于对字段的显示进行格式化**



**13.** **SQRT 函数**

**SQRT 函数用于计算得出任何数值的平方根**

**14.** **RAND 函数**

**SQL 有一个 RAND 函数，用于产生 0 至 1 之间的随机数：**

**15.** **CONCAT 函数**

**CONCAT 函数用于将两个字符串连接为一个字符串**

**16.** **REPLACE()字符串替换函数**

**17.** **TRIM()函数去除字符串头尾空格**

**SQL 中的 TRIM 函数是用来移除掉一个字串中的字头或字尾。最常见的用途是移除字首或字尾的空白。这个函数在不同的资料库中有不同的名称：**

**MySQL: TRIM( ), RTRIM( ), LTRIM( )**

**Oracle: RTRIM( ), LTRIM( )**

**SQL Server: RTRIM( ), LTRIM( )**

