SQL WHERE 子句

* [SQL distinct](http://www.w3school.com.cn/sql/sql_distinct.asp)
* [SQL AND & OR](http://www.w3school.com.cn/sql/sql_and_or.asp)

**WHERE 子句用于规定选择的标准。**

**WHERE 子句**

如需有条件地从表中选取数据，可将 WHERE 子句添加到 SELECT 语句。

**语法**

SELECT 列名称 FROM 表名称 WHERE 列 运算符 值

下面的运算符可在 WHERE 子句中使用：

|  |  |
| --- | --- |
| **操作符** | **描述** |
| = | 等于 |
| <> | 不等于 |
| > | 大于 |
| < | 小于 |
| >= | 大于等于 |
| <= | 小于等于 |
| BETWEEN | 在某个范围内 |
| LIKE | 搜索某种模式 |

**注释：**在某些版本的 SQL 中，操作符 <> 可以写为 !=。

**使用 WHERE 子句**

如果只希望选取居住在城市 "Beijing" 中的人，我们需要向 SELECT 语句添加 WHERE 子句：

SELECT \* FROM Persons WHERE City='Beijing'

**"Persons" 表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **LastName** | **FirstName** | **Address** | **City** | **Year** |
| Adams | John | Oxford Street | London | 1970 |
| Bush | George | Fifth Avenue | New York | 1975 |
| Carter | Thomas | Changan Street | Beijing | 1980 |
| Gates | Bill | Xuanwumen 10 | Beijing | 1985 |

**结果：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **LastName** | **FirstName** | **Address** | **City** | **Year** |
| Carter | Thomas | Changan Street | Beijing | 1980 |
| Gates | Bill | Xuanwumen 10 | Beijing | 1985 |

**引号的使用**

请注意，我们在例子中的条件值周围使用的是单引号。

SQL 使用单引号来环绕**文本值**（大部分数据库系统也接受双引号）。如果是**数值**，请不要使用引号。

**文本值：**

这是正确的：

SELECT \* FROM Persons WHERE FirstName='Bush'

这是错误的：

SELECT \* FROM Persons WHERE FirstName=Bush

**数值：**

这是正确的：

SELECT \* FROM Persons WHERE Year>1965

这是错误的：

SELECT \* FROM Persons WHERE Year>'1965'

**SQL SELECT 语句**

SELECT 语句用于从表中选取数据。

结果被存储在一个结果表中（称为结果集）。

**SQL SELECT 语法**

SELECT 列名称 FROM 表名称

以及：

SELECT \* FROM 表名称

**注释：**SQL 语句对大小写不敏感。SELECT 等效于 select。

**SQL SELECT 实例**

如需获取名为 "LastName" 和 "FirstName" 的列的内容（从名为 "Persons" 的数据库表），请使用类似这样的 SELECT 语句：

SELECT LastName,FirstName FROM Persons

**"Persons" 表:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id** | **LastName** | **FirstName** | **Address** | **City** |
| 1 | Adams | John | Oxford Street | London |
| 2 | Bush | George | Fifth Avenue | New York |
| 3 | Carter | Thomas | Changan Street | Beijing |

**结果：**

|  |  |
| --- | --- |
| **LastName** | **FirstName** |
| Adams | John |
| Bush | George |
| Carter | Thomas |

**SQL SELECT \* 实例**

现在我们希望从 "Persons" 表中选取所有的列。

请使用符号 \* 取代列的名称，就像这样：

SELECT \* FROM Persons

**提示：**星号（\*）是选取所有列的快捷方式。

**结果：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id** | **LastName** | **FirstName** | **Address** | **City** |
| 1 | Adams | John | Oxford Street | London |
| 2 | Bush | George | Fifth Avenue | New York |
| 3 | Carter | Thomas | Changan Street | Beijing |

**在结果集（result-set）中导航**

由 SQL 查询程序获得的结果被存放在一个结果集中。大多数数据库软件系统都允许使用编程函数在结果集中进行导航，比如：Move-To-First-Record、Get-Record-Content、Move-To-Next-Record 等等。

SQL AND & OR 运算符

* [SQL where](http://www.w3school.com.cn/sql/sql_where.asp)
* [SQL Order By](http://www.w3school.com.cn/sql/sql_orderby.asp)

**AND 和 OR 运算符用于基于一个以上的条件对记录进行过滤。**

**AND 和 OR 运算符**

AND 和 OR 可在 WHERE 子语句中把两个或多个条件结合起来。

如果第一个条件和第二个条件都成立，则 AND 运算符显示一条记录。

如果第一个条件和第二个条件中只要有一个成立，则 OR 运算符显示一条记录。

**原始的表 (用在例子中的)：**

|  |  |  |  |
| --- | --- | --- | --- |
| **LastName** | **FirstName** | **Address** | **City** |
| Adams | John | Oxford Street | London |
| Bush | George | Fifth Avenue | New York |
| Carter | Thomas | Changan Street | Beijing |
| Carter | William | Xuanwumen 10 | Beijing |

**AND 运算符实例**

使用 AND 来显示所有姓为 "Carter" 并且名为 "Thomas" 的人：

SELECT \* FROM Persons WHERE FirstName='Thomas' AND LastName='Carter'

**结果：**

|  |  |  |  |
| --- | --- | --- | --- |
| **LastName** | **FirstName** | **Address** | **City** |
| Carter | Thomas | Changan Street | Beijing |

**OR 运算符实例**

使用 OR 来显示所有姓为 "Carter" 或者名为 "Thomas" 的人：

SELECT \* FROM Persons WHERE firstname='Thomas' OR lastname='Carter'

**结果：**

|  |  |  |  |
| --- | --- | --- | --- |
| **LastName** | **FirstName** | **Address** | **City** |
| Carter | Thomas | Changan Street | Beijing |
| Carter | William | Xuanwumen 10 | Beijing |

**结合 AND 和 OR 运算符**

我们也可以把 AND 和 OR 结合起来（使用圆括号来组成复杂的表达式）:

SELECT \* FROM Persons WHERE (FirstName='Thomas' OR FirstName='William')

AND LastName='Carter'

**结果：**

|  |  |  |  |
| --- | --- | --- | --- |
| **LastName** | **FirstName** | **Address** | **City** |
| Carter | Thomas | Changan Street | Beijing |
| Carter | William | Xuanwumen 10 | Beijing |