

SQL是管理关系数据库管理系统，SQL的范围包括数据插入、查询、更新和删除，数据库的模式创建和修改，以及数据访问控制。

(1)

```
mysql> use RUNOOB; # 命令用于选择数据库
Database changed # 命令用于设置使用的字符集

mysql> set names utf8;
Query OK, 0 rows affected (0.00 sec) # 读取数据表的信息

mysql> SELECT * FROM Websites;
+----+-----+-----+-----+-----+
| id | name      | url                      | alexa | country |
+----+-----+-----+-----+-----+
| 1  | Google    | https://www.google.cm/   | 1     | USA     |
| 2  | 淘宝      | https://www.taobao.com/  | 13    | CN      |
| 3  | 菜鸟教程  | http://www.runoob.com/   | 4689  | CN      |
| 4  | 微博      | http://weibo.com/        | 20    | CN      |
| 5  | Facebook  | https://www.facebook.com/| 3     | USA     |
+----+-----+-----+-----+-----+
5 rows in set (0.01 sec)
```

SQL对大小写不敏感，因此SELECT与select对于SQL来说是相同的。

在每条SQL语句的末端应该跟上分号。

命令	作用
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

SQL SELECT 语句

SELECT 语句用于从数据库中选取数据。结果被存储在一个结果表中，称为结果集。
SQL SELECT 语法:

```
SELECT column_name,column_name
FROM table_name;
```

与

```
SELECT * FROM table_name;
```

SQL SELECT DISTINCT 语句

SELECT DISTINCT 语句用于返回唯一不同的值。
SQL SELECT DISTINCT 语句
在表中，一个列可能会包含多个重复值，有时您也许希望仅仅列出不同（distinct）的值。DISTINCT 关键词用于返回唯一不同的值。

SQL WHERE 子句

WHERE 子句用于提取那些满足指定条件的记录。
SQL WHERE 语法

```
SELECT column_name,column_name
FROM table_name
WHERE column_name operator value;
```

下面的运算符可以在 WHERE 子句中使用：

运算符	描述
=	等于
<>	不等于。注释：在 SQL 的一些版本中，该操作符可被写成 !=
>	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
IN	指定针对某个列的多个可能值

此处还可以使用逻辑运算
逻辑运算
And:与 同时满足两个条件的值。

```
select * from emp where sal > 2000 and sal < 3000;
```

查询 EMP 表中 SAL 列中大于 2000 小于 3000 的值。

Or:或 满足其中一个条件的值

```
select * from emp where sal > 2000 or comm > 500;
```

查询 emp 表中 SAL 大于 2000 或 COMM 大于500的值。

Not:非 满足不包含该条件的值。

```
select * from emp where not sal > 1500;
```

查询EMP表中 sal 小于等于 1500 的值。

逻辑运算的优先级：

() not and or

特殊条件

1.空值判断： is null

```
select * from emp where comm is null;
```

查询 emp 表中 comm 列中的空值。

2.between and (在 之间的值)

```
select * from emp where sal between 1500 and 3000;
```

查询 emp 表中 SAL 列中大于 1500 的小于 3000 的值。

注意：大于等于 1500 且小于等于 3000， 1500 为下限，3000 为上限，下限在前，上限在后，查询的范围包涵有上下限的值。

3.In

```
select * from emp where sal in (5000,3000,1500);
```

查询 EMP 表 SAL 列中等于 5000, 3000, 1500 的值。

4.like

Like模糊查询

```
select * from emp where ename like 'M%';
```

查询 EMP 表中 Ename 列中有 M 的值，M 为要查询内容中的模糊信息。

% 表示多个字值，_ 下划线表示一个字符；
M% ： 为能配符，正则表达式，表示的意思为模糊查询信息为 M 开头的。
%M% ： 表示查询包含M的所有内容。
%M_ ： 表示查询以M在倒数第二位的所有内容。

SQL AND & OR 运算符

AND & OR 运算符用于基于一个以上的条件对记录进行过滤。

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。

SQL ORDER BY 关键字

ORDER BY 关键字用于对结果集按照一个列或者多个列进行排序。

ORDER BY 关键字默认按照升序对记录进行排序。如果需要按照降序对记录进行排序，您可以使用 DESC 关键字。

SQL ORDER BY 语法

```
SELECT column_name,column_name
FROM table_name
ORDER BY column_name,column_name ASC|DESC;
```

SQL INSERT INTO 语句

INSERT INTO 语句用于向表中插入新记录。

SQL INSERT INTO 语法

INSERT INTO 语句可以有两种编写形式。

第一种形式无需指定要插入数据的列名，只需提供被插入的值即可：

```
INSERT INTO table_name
VALUES (value1,value2,value3,...);
```

第二种形式需要指定列名及被插入的值：

```
INSERT INTO table_name (column1,column2,column3,...)
VALUES (value1,value2,value3,...);
```

insert into select 和select into from 的区别

insert into scorebak select * from socre where neza='neza' --插入一行,要求表scorebak 必须存在
select * into scorebak from score where neza='neza' --也是插入一行,要求表scorebak 不存在

SQL UPDATE 语句

UPDATE 语句用于更新表中已存在的记录。

SQL UPDATE 语法

```
UPDATE table_name
SET column1=value1,column2=value2,...
WHERE some_column=some_value;
```

SQL DELETE 语句

DELETE 语句用于删除表中的行。

SQL DELETE 语法

```
DELETE FROM table_name
WHERE some_column=some_value;
```

SQL SELECT TOP 子句

SELECT TOP 子句用于规定要返回的记录数目。

SELECT TOP 子句对于拥有数千条记录的大型表来说，是非常有用的。

注意:并非所有的数据库系统都支持 SELECT TOP 语句。MySQL 支持 LIMIT 语句来选取指定的条数数据，Oracle 可以使用 ROWNUM 来选取。

SQL Server / MS Access 语法

```
SELECT TOP number|percent column_name(s)
FROM table_name;
```

MySQL 语法

```
SELECT column_name(s)
FROM table_name
LIMIT number;
```

SQL LIKE 操作符

LIKE 操作符用于在 WHERE 子句中搜索列中的指定模式。

SQL LIKE 语法

```
SELECT column_name(s)
FROM table_name
WHERE column_name LIKE pattern;
```

通配符	描述
%	替代 0 个或多个字符
_	替代一个字符
[charlist]	字符列中的任何单一字符
[^charlist]或[!charlist]	不在字符列中的任何单一字符

SQL JOIN

SQL 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。

JOIN模式	结合方式
	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

SQL UNION 操作符

UNION 操作符用于合并两个或多个 SELECT 语句的结果集。

请注意，UNION 内部的每个 SELECT 语句必须拥有相同数量的列。列也必须拥有相似的数据类型。同时，每个 SELECT 语句中的列的顺序必须相同。

SQL UNION 语法

```
SELECT column_name(s) FROM table1
UNION
SELECT column_name(s) FROM table2;
```

注释：默认地，UNION 操作符选取不同的值。如果允许重复的值，请使用 UNION ALL。

SQL UNION ALL 语法

```
SELECT column_name(s) FROM table1
UNION ALL
SELECT column_name(s) FROM table2;
```

注释：UNION 结果集中的列名总是等于 UNION 中第一个 SELECT 语句中的列名。

SQL INSERT INTO SELECT 语句

INSERT INTO SELECT 语句从一个表复制数据，然后把数据插入到一个已存在的表中。目标表中任何已存在的行都不会受影响。

SQL INSERT INTO SELECT 语法

我们可以从一个表中复制所有的列插入到另一个已存在的表中：

```
INSERT INTO table2
SELECT * FROM table1;
```

或者我们可以只复制希望的列插入到另一个已存在的表中：

```
INSERT INTO table2
(column_name(s))
SELECT column_name(s)
FROM table1;
```

ALTER TABLE 语句

ALTER TABLE 语句用于在已有的表中添加、删除或修改列。

SQL ALTER TABLE 语法

如需在表中添加列，请使用下面的语法：

```
ALTER TABLE table_name  
ADD column_name datatype
```

如需删除表中的列，请使用下面的语法（请注意，某些数据库系统不允许这种在数据库表中删除列的方式）：

```
ALTER TABLE table_name  
DROP COLUMN column_name
```

要改变表中列的数据类型，请使用下面的语法：

```
SQL Server / MS Access:  
ALTER TABLE table_name  
ALTER COLUMN column_name datatype
```

My SQL / Oracle:

```
ALTER TABLE table_name  
MODIFY COLUMN column_name datatype
```