



链滴

DQL: 查询表中的记录

作者: [chenyuan995](#)

原文链接: <https://ld246.com/article/1587208984581>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



语法:

```
select * from 表名;  
select 字段名 1, 字段名 2, 字段名 3, ... from 表名;  
select 列名 from 表名 [where 条件表达式]
```

- 1.SELECT 命令可以读取一行或者多行记录。
- 2.你可以使用星号 (*) 来代替其他字段, SELECT语句会返回表的所有字段数据
- 3.你可以使用 WHERE 语句来包含任何条件。

使用*表示查询所有的学生:

```
select * from student;
```

查询 student 表中的 name 和 age 列

```
select name,age from student;
```

去除重复: distinct

- 语法:
 - SELECT DISTINCT 字段名 FROM 表名;
- 只对一列操作

```
select distinct country from person
```

country
China
Japan
Korea
(Null)

- 对多列进行操作

`select distinct country, province from person`

country	province
China	ZheJiang
China	JiangSu
Japan	BingKu
Korea	JingJi
(Null)	(Null)

从上例中可以发现，当distinct应用到多个字段的时候，其应用的范围是其后面的所有字段，而不只紧挨着它的一个字段，而且distinct只能放到所有字段的前面，如下语句是错误的：

`SELECT country, distinct province from person; // 该语句是错误的`

计算列

- 语法：

`SELECT 列名 1 + 固定值 FROM 表名;`

`SELECT 列名 1 + 列名 2 FROM 表名;`

注意: 参与运算的必须是数值类型

- 一般可以使用四则运算计算一些列的值。（一般只会进行数值型的计算）
- ifnull(表达式1,表达式2): null参与的运算，计算结果都为null
- 表达式1: 哪个字段需要判断是否为null
- 如果该字段为null后的替换值。

起别名

- 语法：

- 对列指定别名

`SELECT 字段名1 AS 别名, 字段名2 AS 别名... FROM 表名;`

- 对列和表同时指定别名

`SELECT 字段名1 AS 别名, 字段名2 AS 别名... FROM 表名 AS 表别名;`

`SELECT 字段名1 别名, 字段名2 别名... FROM 表名;`

as: as也可以省略

条件查询

- where子句后跟条件

- 运算符

-

、 <、 <=、 >=、 =、 <>

- BETWEEN...AND

- IN(集合)

- LIKE: 模糊查询

- 占位符:

- _:单个任意字符

- %: 多个任意字符

- IS NULL

- and 或 &&

- or 或 ||

- not 或 !

查询年龄大于20岁

```
SELECT * FROM student WHERE age > 20;  
SELECT * FROM student WHERE age >= 20;
```

查询年龄等于20岁

```
SELECT * FROM student WHERE age = 20;
```

查询年龄不等于20岁

```
SELECT * FROM student WHERE age != 20;  
SELECT * FROM student WHERE age <> 20;
```

查询年龄大于等于20 小于等于30

```
SELECT * FROM student WHERE age >= 20 && age <=30;  
SELECT * FROM student WHERE age >= 20 AND age <=30;  
SELECT * FROM student WHERE age BETWEEN 20 AND 30;
```

查询年龄22岁, 18岁, 25岁的信息

```
SELECT * FROM student WHERE age = 22 OR age = 18 OR age = 25  
SELECT * FROM student WHERE age IN (22,18,25);
```

查询英语成绩为null

```
SELECT * FROM student WHERE english = NULL;-- 不对的。null值不能使用 = (!=) 判断  
SELECT * FROM student WHERE english IS NULL;
```

查询英语成绩不为null

```
SELECT * FROM student WHERE english IS NOT NULL;
```

查询姓马的有哪些? like

```
SELECT * FROM student WHERE NAME LIKE '马%';
```

查询姓名第二个字是化的人

```
SELECT * FROM student WHERE NAME LIKE "_化%";
```

查询姓名是3个字的人

```
SELECT * FROM student WHERE NAME LIKE '___';
```

查询姓名中包含德的人

```
SELECT * FROM student WHERE NAME LIKE '%德%';
```