

# SQL 必知必会 2 - 查询数据

---

## SQL 必知必会 2 - 查询数据

### 2.1 SELECT 语句

#### 2.2 查询单个列

#### 2.3 查询多个列

#### 2.4 查询所有列

#### 2.5 查询不同的值

#### 2.6 限制结果

#### 2.7 使用注释

#### 2.8 小结

---

## 2.1 SELECT 语句

**SELECT** 用途：从一个或多个表中查询信息。

关键字 **keyword**：作为 **SQL** 组成部分的保留字。关键字不能用做表或者列的名字。

为了使用 **SELECT** 查询数据表，必须至少给出两条信息：

- 想选择什么
  - 以及从什么地方选择
- 

## 2.2 查询单个列

```
SELECT prod_name
FROM Products;
```

结束 **SQL** 语句要使用分号 **;**。  
有些软件不加也可以，但是最好加上。

**SQL** 语句的关键字不区分大小写。  
但是开发人员喜欢对 **SQL** 关键字大写，对列名和表名小写，这样容易阅读。

处理 **SQL** 语句的时候，空格被忽略。  
所以可以写在一行、两行和多行，以下三种写法相同，但是通常建议分成容易阅读的多行：

```
SELECT prod_name  
FROM Products;
```

```
SELECT prod_name FROM Products;
```

```
SELECT  
prod_name  
FROM  
Products;
```

---

## 2.3 查询多个列

跟查询一个列一样，只不过 **SELECT** 后面给出多个列名，中间用逗号隔开。

```
SELECT prod_id, prod_name, prod_price  
FROM Products;
```

**SQL** 语句返回原始的、无格式的数据。  
数据的格式化是表示问题，而不是查询问题。  
通常很少直接使用实际查询出的数据。

---

## 2.4 查询所有列

在实际列名的位置使用星号 `*` 通配符可以做到这一点。

```
SELECT *  
FROM Products;
```

通配符可以用来查询未知列的列名。

但是如果不是确实需要，尽量少用，因为会降低查询和应用程序的性能。

---

## 2.5 查询不同的值

```
SELECT DISTINCT vend_id  
FROM Products;
```

- `DISTINCT` 必须直接放在列名的前面，返回的结果具有唯一性。
- `DISTINCT` 关键字作用于所有的列，不仅仅是跟在它后面的那一列。

---

## 2.6 限制结果

各种数据库中的这一 `SQL` 语句的实现不同。

`SQL Server` 和 `Access` 使用 `TOP` 关键字：

```
SELECT TOP 5 prod_name  
FROM Products;
```

`DB2` :

```
SELECT prod_name  
FROM Products  
FETCH FIRST 5 ROW ONLY;
```

`Oracle` 要使用行计数器 `ROWNUM` :

```
SELECT prod_name
FROM Products
WHERE ROWNUM <= 5;
```

MySQL 、 MariaDB 、 PostgreSQL 、 SQLite 要使用 LIMIT 子句：

```
SELECT prod_name
FROM Products
LIMIT 5;
```

返回从第 5 行开始的 5 条数据：

```
SELECT prod_name
FROM Products
LIMIT 5 OFFSET 5;
```

注意第一个被查询的是第 0 行，行数从 0 开始。

另外 MySQL 和 MariaDB 以及 SQLite 支持简化版的 LIMIT 4 OFFSET 3 ：  
LIMIT 3, 4 从第 3 条数据开始的 4 条数据  
逗号之前对应 OFFSET ，逗号之后对应 LIMIT 。

---

## 2.7 使用注释

使用两个连字符 --

```
SELECT prod_name    -- 这是一条注释
FROM Products;
```

或者使用 # ，但是很少得到支持

```
# 这是一条注释
SELECT prod_name
```

**FROM** Products;

多行注释使用 */\* ... \*/* :

```
/* SELECT prod_name, vend_id  
FROM Products; */  
SELECT prod_name  
FROM Products;
```

---

## 2.8 小结