

# SQL 必知必会 12 - 联结表

## SQL 必知必会 12 - 联结表

### 12.1 联结

#### 12.1.1 关系表

#### 12.1.2 为什么使用联结

### 12.2 创建联结

#### 12.2.1 WHERE 子句的重要性

#### 12.2.2 内联结

#### 12.2.3 联结多个表

### 12.3 小结

## 12.1 联结

SQL 最强大的功能之一就是能在数据查询的执行中联结 ( `join` ) 表。

### 12.1.1 关系表

相同的数据出现多次决不是一件好事，这是关系数据库设计的基础。

关系表的设计就是要把信息分解成多个表，一类数据一个表。

各表通过某些共同的值互相关联（所以才叫关系数据库）。

唯一的标识称为主键 `primary key`。

#### 可伸缩 `scale`

能够适应不断增加的工作量而不失败

设计良好的数据库或应用程序称为可伸缩性好 `scale well`。

### 12.1.2 为什么使用联结

在一条 `SELECT` 语句中关联表，因此称为联结。

联结在实际的数据库表中并不存在，`DBMS` 会根据需要建立联结，它在查询执行期间一直存在。

## 使用交互式的 DBMS 工具

可指示数据库只允许在 `Products` 表的供应商 `ID` 列中出现合法值（即出现在 `Vendors` 表中的供应商）。

引用完整性标识 `DBMS` 强制实施数据完整性规则。

这些规则一般由提供了界面的 `DBMS` 管理。

# 12.2 创建联结

指定要联结的所有表以及关联方式即可：

```
SELECT vend_name, prod_name, prod_price
FROM Vendors, Products
WHERE Vendors.vend_id = Products.vend_id;
```

- `FROM` 子句列出了两个要联结的表
- `WHERE` 子句给出了联表条件，用到了完全限定列名（用一个句点 `.` 分隔表名和列名）

## 12.2.1 WHERE 子句的重要性

在联结两个表时，实际要做的是将第一个表中的每一行与第二个表中的每一行配对。

`WHERE` 子句作为过滤条件，只包含那些匹配给定条件（这里是联结条件）的行。

没有 `WHERE` 子句，第一个表中的每一行将和第二个表中的每一行配对，而不管逻辑关系。

### 笛卡尔积 `cartesian product`

由没有联结条件的表关系返回的结果为笛卡尔积。

检索出的行的数目是将第一个表中的行数乘以第二个表中的行数。

返回笛卡尔积的联结，也称叉联结 `cross join`。

## 12.2.2 内联结

目前为止使用的联结称为等值联结 `equijoin`。

它基于两个表之间的相等测试。

这种联结也称为内联结 `inner join`。

另一种语法：

```
SELECT vend_name, prod_name, prod_price
FROM Vendors INNER JOIN Products
```

```
ON Vendors.vend_id = Products.vend_id;
```

ANSI SQL 规范首选 INNER JOIN 语法。

其实，SQL 语言纯正论者是用鄙视的眼光看待简单语法的（即用 WHERE 子句）简单格式和标准格式，看自己选择。

## 12.2.3 联结多个表

```
SELECT prod_name, vend_name, prod_price, quantity
FROM OrderItems, Products, Vendors
WHERE Products.vend_id = Vendors.vend_id
AND OrderItems.prod_id = Products.prod_id
AND order_num = 20007;
```

### 性能考虑

不要联结不需要的表，联结的表越多，性能下降越厉害。

### 联结中表的最大数目

虽然 SQL 本身不限制每个联结约束中表的数目，但实际上许多 DBMS 都有限制。

### 用联结代替子查询

子查询写法：

```
SELECT cust_name, cust_contact
FROM Customers
WHERE cust_id IN (SELECT cust_id
                  FROM Orders
                  WHERE order_num IN (SELECT order_num
                                     FROM OrderItems
                                     WHERE prod_id = 'RGAN01'));
```

联表写法：

```
SELECT cust_name, cust_contact
FROM Customers, Orders, OrderItems
WHERE Customers.cust_id = Orders.cust_id
AND OrderItems.order_num = Orders.order_num
AND prod_id = 'RGAN01';
```

## 12.3 小结