# ****1.数据库的内连接、左外连接、右外连接****

## ****1.内连接****

**内连就是抓取两张表共同满足on条件后的数据。**

关键字：inner join on

语句：select \* from a\_table a inner join b\_table bon a.a\_id = b.b\_id;

说明：组合两个表中的记录，返回关联字段相符的记录，也就是返回两个表的交集部分。

## ****2.左连接****

**左连就是以左表数据为主，再查出满足on条件后的数据。**

关键字：left join on / left outer join on

语句：select \* from a\_table a left join b\_table bon a.a\_id = b.b\_id;

说明：

left join 是left outer join的简写，它的全称是左外连接，是外连接中的一种。

左(外)连接，左表(a\_table)的满足条件记录将会全部表示出来，而右表(b\_table)只会显示符合搜索条件的记录。右表记录不足的地方均为NULL。3.右连接

**右连就是以右表数据为主，再查出满足on条件后的数据。**

关键字：right join on / right outer join on

语句：select \* from a\_table a right outer join b\_table b on a.a\_id = b.b\_id;

说明：

right join是right outer join的简写，它的全称是右外连接，是外连接中的一种。

与左(外)连接相反，右(外)连接，左表(a\_table)只会显示符合搜索条件的记录，而右表(b\_table)的记录将会全部表示出来。左表记录不足的地方均为NULL。

**4.A表左连B表等同于B表右连A表，A表右连B表等同于B表左连A表。**

**5.** MySQL不支持全外连接，所以只能采取关键字UNION来联合左、右连接的方法。

查询语句：SELECT s.\*,subject,score FROM student s LEFT JOIN mark m ON s.id=m.id   
UNION   
SELECT s.\*,subject,score FROM student s RIGHT JOIN mark m ON s.id=m.id;  
查询结果：

结论：返回左右表的所有行。哪个表中没有的就用null填充。

# 2.数据库的索引

索引我们分为四类来讲：单列索引(普通索引，唯一索引，主键索引)、组合索引、全文索引、空间索引。

### 单列索引

普通索引：也叫常规索引。MySQL中基本索引类型，没有什么限制，允许在定义索引的列中插入重复值和空值，纯粹为了查询数据更快一点。它可以常规地提高查询效率。一张数据表中可以有多个常规索引。常规索引是使用最普遍的索引类型，如果没有明确指明索引的类型，我们所说的索引都是指常规索引。

唯一索引：可以提高查询效率，并提供唯一性约束。一张表中可以有多个唯一索引。

外键索引：简称外键，它可以提高查询效率，外键会自动和对应的其他表的主键关联。外键的主要作用是保证记录的一致性和完整性。只有InnoDB存储引擎的表才支持外键。

主键索引：是一种特殊的唯一索引，不允许有空值。

### 联合索引

在表中的多个字段组合上创建的索引，只有在查询条件中使用了这些字段的左边字段时，索引才会被使用，使用组合索引时遵循最左前缀集合。

联合索引又叫复合索引。对于复合索引，Mysql从左到右的使用索引中的字段，一个查询可以只使用索引中的一部分，但只能是最左侧部分。例如索引是key index (a,b,c). 可以支持**a** | **a,b**| **a,b,c** 3种组合进行查找，但不支持 b,c进行查找 .当最左侧字段是常量引用时，索引就十分有效。

### 全文索引

全文索引，只有在MyISAM引擎上才能使用，只能在CHAR,VARCHAR,TEXT类型字段上使用全文索引。

### 空间索引

空间索引是对空间数据类型的字段建立的索引，MySQL中的空间数据类型有四种，GEOMETRY、POINT、LINESTRING、POLYGON。在创建空间索引时，使用SPATIAL关键字。要求，引擎为MyISAM，创建空间索引的列，必须将其声明为NOT NULL。

### 创建索引

在执行CREATE TABLE语句时可以创建索引，也可以单独用CREATE INDEX或ALTER TABLE来为表增加索引。

1．ALTER TABLE

ALTER TABLE用来创建普通索引、UNIQUE索引或PRIMARY KEY索引。

ALTER TABLE table\_name ADD INDEX index\_name (column\_list)

ALTER TABLE table\_name ADD UNIQUE (column\_list)

ALTER TABLE table\_name ADD PRIMARY KEY (column\_list)

2．CREATE INDEX

CREATE INDEX可对表增加普通索引或UNIQUE索引。

CREATE INDEX index\_name ON table\_name (column\_list)

CREATE UNIQUE INDEX index\_name ON table\_name (column\_list)

### 删除索引

可利用ALTER TABLE或DROP INDEX语句来删除索引。类似于CREATE INDEX语句，DROP INDEX可以在ALTER TABLE内部作为一条语句处理，语法如下。

DROP INDEX index\_name ON talbe\_name

ALTER TABLE table\_name DROP INDEX index\_name

ALTER TABLE table\_name DROP PRIMARY KEY