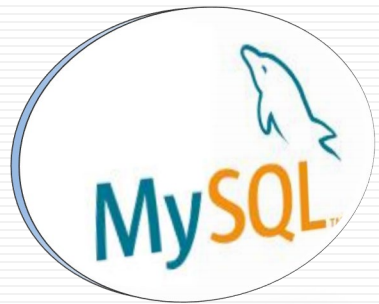




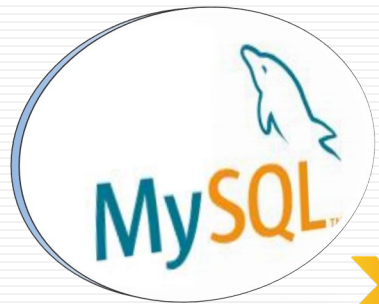
数据库设计与应用

第4章 数据查询、视图



4.3 连接查询

- 理解连接查询
- 内连接查询 (**INNER JOIN**)
- 外连接查询 (**OUTER JOIN**)
- 复合连接条件查询 (连接查询+**WHERE**)
- 课后思考题



理解连接查询

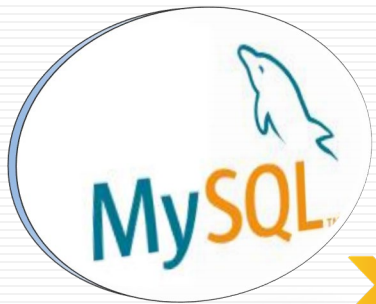
连接是关系数据模型的主要特点，连接查询是关系数据库中最主要的查询，包括内连接、外连接等。

当两个或多个表中存在**相同意义的字段**时，便可以通过这些字段对不同的表进行连接查询，得到存放在多个表中的记录数据。

所谓表中**相同意义的字段**，是指在多个表中名字不一定相同，但取值的含义相同的字段，这是表之间实现连接查询的前提。

如：

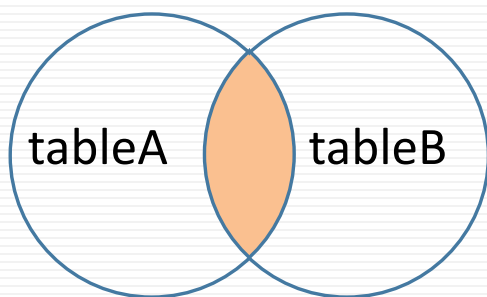
department.**Department_id** <==> teacher.**Department_id**



内连接查询 (INNER JOIN)

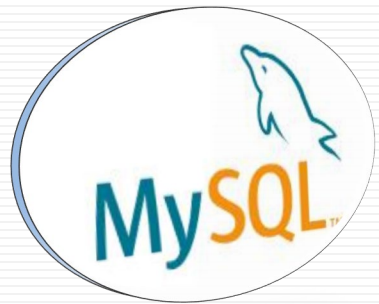
INNER JOIN使用比较运算符(=)根据每个表共有列的列值匹配两个表中的行，其查询结果仅包含符合查询条件和连接条件的行。即查询结果为两个表的交集。

注意：多表查询时，为避免混淆，在查询字段前应添加表名称作为前缀！



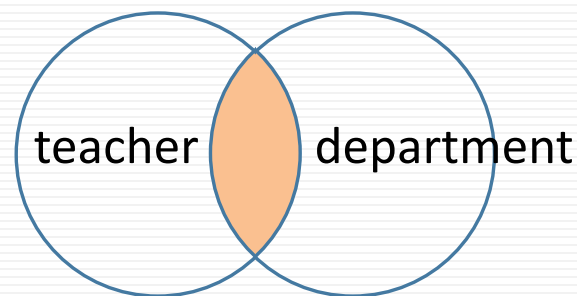
```
SELECT tableA.select_list, tableB.select_list  
FROM tableA INNER JOIN tableB  
ON tableA.Key = tableB.Key
```

两个表中意义相同的字段



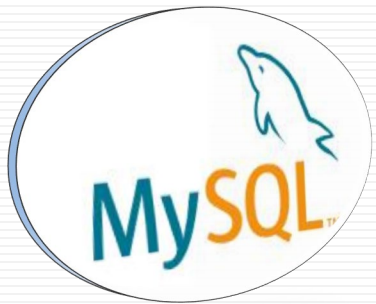
teacher表和department表通过共有列Department_id实现连接，通过内连接查询（INNER JOIN）可以得到两个表中Department_id值相同的所有行。

```
SELECT  
    teacher.* ,  
    department.*  
FROM teacher INNER JOIN department  
    ON teacher.Department_id = department.Department_id;
```



实例

① 查询教师所在学院

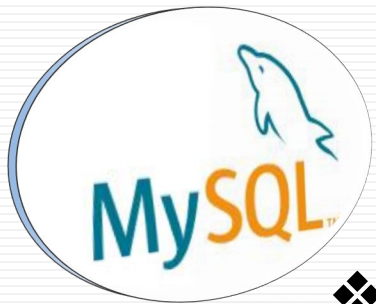


❖ 学生表student和班级表classes的内连接查询

```
SELECT ... FROM student  
INNER JOIN classes ON student.Class_id = classes.Class_id;
```

❖ 选课表choose和课程表course的内连接查询

```
SELECT ... FROM choose  
INNER JOIN course ON choose.Course_id = course.Course_id;
```



❖ student **A**、choose **B**、course **C** 三个表的内连接查询

SELECT

A.Student_id, **A**.Student_name, **C**.Course_name, **C**.Term, **B**.Score
FROM *student A*
INNER JOIN *choose B* **ON** **A**.Student_id = **B**.Student_id
INNER JOIN *course C* **ON** **B**.Course_id = **C**.Course_id;

实例

- ② 查询学生所在班级、年级和学院。
- ③ 查询学生的选课信息。

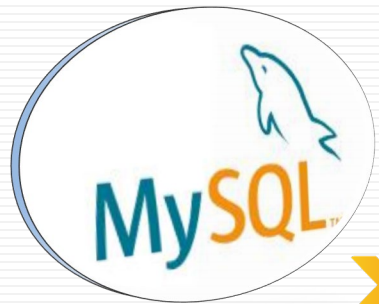


在设计内连接查询时，强调几点：

- ❶ 表之间一定要有连接字段。
- ❷ 查询结果的多个列来源于不同的表，需要在列名称前加表名称作为前缀。
- ❸ 为了简化书写，可以给表指定别名，格式为：

SELECT ... FROM table **A**

别名**A**和表名**table**之间至少加一个空格



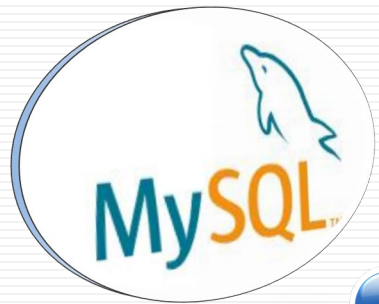
外连接查询 (OUTER JOIN)

内连接的查询结果仅包含符合连接条件的行。如果需要查询结果不仅包含符合连接条件的行，而且还包括左表、右表或两个连接表中的所有数据行，则应该使用**外连接**。

MySQL支持的外连接有两种类型：

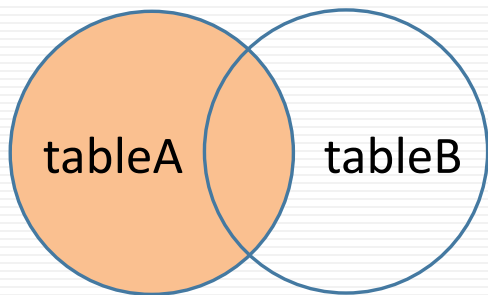
- ❖ 左外连接（左连接 LEFT **[OUTER]** JOIN）
- ❖ 右外连接（右连接 RIGHT **[OUTER]** JOIN）

MySQL不能直接支持FULL JOIN，要实现FULL JOIN，应该使用**LEFT JOIN UNION RIGHT JOIN**的方式。



左连接 (LEFT OUTER JOIN)

左连接的查询结果为左表的所有记录以及右表中连接字段相等的记录。如果左表的某行在右表中没有匹配行，则在相关联的结果行中，右表的所有选择列均为空值(NULL)。

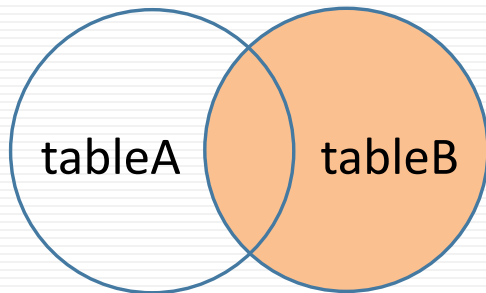


```
SELECT tableA.select_list, tableB.select_list  
FROM tableA  
LEFT [OUTER] JOIN tableB  
ON tableA.Key = tableB.Key
```



右连接 (RIGHT OUTER JOIN)

右连接是左连接的反向连接，查询结果为右表中的所有记录以及左表中连接字段相等的记录。如果右表的某行在左表中没有匹配行，左表的选择列将返回空值(NULL)。

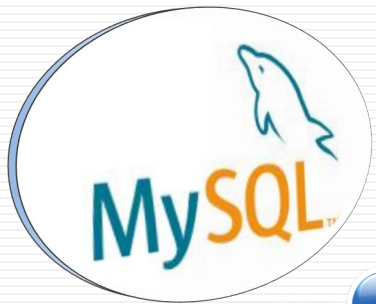


```
SELECT tableA.select_list, tableB.select_list  
FROM tableA  
    RIGHT [OUTER] JOIN tableB  
    ON tableA.Key = tableB.Key
```

实例

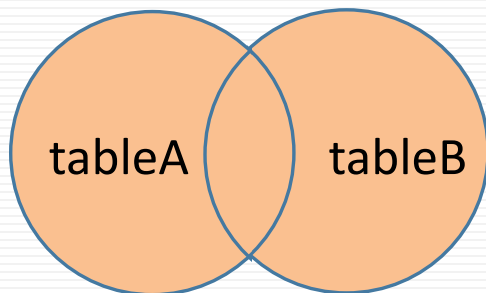
4

通过外连接查询查看哪些学院还没有教师。



完全连接 (FULL JOIN)

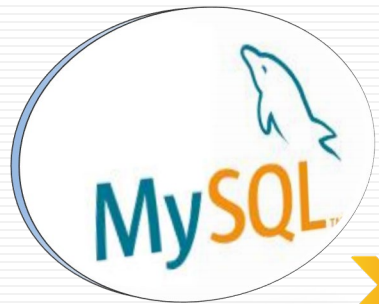
完全连接返回左表和右表中的所有记录，包括连接字段相等的记录和不相等的记录。如果左表的某行在右表中没有匹配行，将返回空值(NULL)，反之亦然。



```
SELECT tableA.select_list, tableB.select_list  
FROM tableA LEFT [OUTER] JOIN tableB  
ON tableA.Key = tableB.Key
```

UNION

```
SELECT tableA.select_list, tableB.select_list  
FROM tableA RIGHT [OUTER] JOIN tableB  
ON tableA.Key = tableB.Key
```

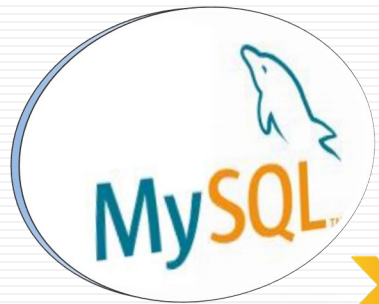


复合连接条件查询

复合条件连接查询是在连接查询的过程中，通过添加过滤条件，限制查询的结果，也就是JOIN和WHERE条件组合，使查询的结果更加准确。

实例

- ⑤ 查询第2学期成绩在90分以上的学生、课程和成绩信息。
- ⑥ 查询还没有教师的学院信息。



课后思考题



- 1 查询"第1学期"开课, 学分为5分的课程信息!
显示课程号, 课程名, 开课学期和学分四个字段!
- 2 查询2017级, 课程号为1001, 成绩在80分到90分之间的同学成绩!