

范式

数据一对多 多对多

每张表都有自己的功能

如何存储数据量更小一点 提升性能？

范式:数据库的一个规范 帮助我们更好地去建表 达到高效 简洁

id	name	mobile	zip	province	city	district	<u>deptNo</u>	<u>deptName</u>
101	张三	13910000001 13910000002	100001	北京	北京	海淀区	D1	部门1
101	张三	13910000001 13910000002	100001	北京	北京	海淀区	D2	部门2
102	李四	13910000003	200001	上海	上海	静安区	D3	部门3
103	王五	13910000004	510001	广东省	广州	白云区	D4	部门4
103	王五	13910000004	510001	广东省	广州	白云区	D5	部门 5

不规范！！

第一范式：

- 表中的列只能含有原子性(不可再分的值)

id	name	mobile	zip	province	city	district	<u>deptNo</u>	<u>deptName</u>
101	张三	13910000001	100001	北京	北京	海淀区	D1	部门1
101	张三	13910000002	100001	北京	北京	海淀区	D1	部门1
101	张三	13910000001	100001	北京	北京	海淀区	D2	部门2
101	张三	13910000002	100001	北京	北京	海淀区	D2	部门2
102	李四	13910000003	200001	上海	上海	静安区	D3	部门3
103	王五	13910000004	510001	广东省	广州	白云区	D4	部门4
103	王五	13910000004	510001	广东省	广州	白云区	D5	部门 5

第二范式：

- 满足第一范式
- 没有部分依赖

知道deptNo其实能得出deptName

员工表 employee

id	name	zip	province	city	district
101	张三	100001	北京	北京	海淀区
102	李四	200001	上海	上海	静安区
103	王五	510001	广东省	广州	白云区

部门表 dept

<u>deptNo</u>	<u>deptName</u>
D1	部门1
D2	部门2
D3	部门3
D4	部门4
D5	部门5

****员工部门关系表 employee_dept****

id	<u>deptNo</u>
101	D1
101	D2
102	D3
103	D4

第三范式：

- 满足前两个范式
 - 没有传递性依赖

employee

id	name	zip
101	张三	100001
102	李四	200001
103	王五	510001

地区表area

zip	province	city	district
100001	北京	北京	海淀区
200001	上海	上海	静安区
510001	广东省	广州	白云区

总结：一张表只做一件事情
一对一 一对多 多对多

teacherid	stduentid
1	1
1	2
2	1
2	1
3	3

外键 (会消耗性能哦)
保证数据的有效性 可通过外键来约束数据来进行有效性验证
建立约束 只有另外一个表有数据 这张表才能插入

```
1 alter table scores add constraint stu_sco foreign key(stuid) references students(id)
```

此时插入或者修改数据,如果stuid的值在students表中就不会报错
创建表的时候可以直接创建约束

```
1 create table scores(  
2     id int primary key auto_increment,  
3     stuid int,  
4     subid int,  
5     score decimal(5,2),  
6     foreign key(stuid) references students(id),
```

```

7     foreign key(subid) references subjects(id),
8 );

```

外键的创建

设计表 -> 外键 (老师和学生表都设主键)

字段	参考数据库
	...
名	
<input checked="" type="checkbox"/> teacherid	
<input type="checkbox"/> stduentid	

名	字段	参考数据库	参考表	参考字段	删除时	更新时
	teacherid	book	teacher	id		▼