**数据表设计入门**

**前言**

  目的：这篇文章旨在带领新进的同学对基本的数据库知识、数据表逻辑设计、mysql数据库有一个基本了解，以便快速上手实际业务

**数据表逻辑设计**

  经过上面的介绍，我们知道在存储数据之前，需要设计好表的结构，比如存储什么类型的信息等等，那么如何设计呢？开发中我们需要遵守一些约定。  
  往下走之前，需要先有基本的认知：比如下面的用户信息表格：

| **用户名** | **密码** | **邮箱** | **手机号** |
| --- | --- | --- | --- |
| 张三 | 1234123 | 810938372@qq.com | 15757821111 |
| 李四 | 1312 | 813617368@qq.com | 15757841231 |

  这张表格中，每一行都对应一个用户的信息，然后这个表格有四个字段（列名），每个字段都表示用户的一个属性。这样能理解就可以了。

  现在有一个场景：有几对夫妻，每对都是夫妻二人共同购买了一间房子，也共同购买了两辆以上的车。现在要存储每个人的信息，房子信息，车信息，如何存储呢？  
首先建立人的信息表，具体如下：

| **身份证号** | **姓名** | **性别** | **手机号** | **年龄** |
| --- | --- | --- | --- | --- |

然后建立房子的信息表：

| **地址** | **使用面积** | **建筑面积** |
| --- | --- | --- |

最后建立车的信息表：

| **车牌** | **车龄** | **车型** |
| --- | --- | --- |

  好了，第一步就建立好了，可以看出每张表都是一个独立的表格，人的信息表只存储人的数据，车的信息表只存储车的数据。假设我们已经往里面填充了一些数据，那么现在有个问题：我现在拥有C这个人的身份证号信息，那么我如何查找C拥有的车的信息呢？  
  答：无法查找。  
  这答案显而易见，上面三个表格之间根本没有任何关联关系，彼此都是相互独立的，那么我怎么可能根据人的某项信息来找到他下面的车呢？看来我们的表的存储结构设计的有待完善。  
  既然人、车、房之间在现实中本来就有关联关系，那么我们设计的表格中也应该有关联关系才对。经过分析，我们可以发现，每个人都只有一间房子，而一间房子反过来又和多个人（这里是夫妻二人）有关系；每个人都有多辆车，而每辆车又属于多个人（这里是同属于夫妻二人）。这些就是他们之间的关系。  
  所以我们的改进如下：

* 房子和人之间是一对多的关系：一个人只有一套房子，一个房子对应夫妻两个人。所以我们在人的数据表中加入一列，表格如下：

| **身份证号** | **姓名** | **性别** | **手机号** | **年龄** | **房子地址** |
| --- | --- | --- | --- | --- | --- |

* 人和车之间是多对多的关系：一个人拥有好几辆车，每辆车又会对应两个人（夫妻二人）。面对多对多的这种关系，我们提取出一张中间表，而不再对人的表和车的表进行修改：

| **身份证号** | **车牌号** |
| --- | --- |

可以看出，这张表就通过身份证号和车牌号建立了人和车之间的关联关系。

* 车和房子之间没有直接的联系，所以不再进行处理

好了，现在我们根据一个人的身份证号来查询这个人下面的房子信息怎么查询呢？

直接根据身份证号在人的信息表中找到一行（也即这个人），然后就可以看到这行数据中的房子地址了，我们拿到房子地址后就可以根据地址去房子的数据表中查找该房子的具体信息了。

那么，如何根据身份证号查询车的信息呢？

我们去人-车关联表中查找到这个身份证号对应的几个车牌号，然后就可以根据这几个车牌号去车的数据表中查询有关的数据信息了。

**数据表设计规则总结**

数据库设计是有几大范式的，但是他们都比较高深莫测，实际开发中我们遵循下面的规则即可：

1. 每张表只存储一类数据，比如用户表就存储用户信息；（做关联用的列不算，比如上面例子中人的表中还存储了房子的地址，但是地址是做关联用的）
2. 设计表之前首先要划分清楚各模块是什么，比如用户模块、车、房子，然后理清楚各模块间的关系，两张表之间如果存在关联关系，则该关系可以分为三种：一对一，一对多，多对多。
3. 分别先独立的设计各个表的结构，这样设计完后是一张张相互独立没有关联的表
4. 然后按照两表之间的关系来进行完善即可。

那么两表之间的关系如何完善呢？

* A和B之间是1对1关系，则将一张表的唯一字段（唯一能够确定一行数据的）放入另外一张表中：比如用户表和手机表是1对1的关系，那么我们可以将用户身份证号放入手机表中，作为手机表的列名（字段）之一。
* A和B之间是1对多 关系，则将“一”这一端的唯一字段放入“多“的一端中：比如用户表和衣服表，一个用户有多件衣服，一件衣服只属于一个用户，那么就将身份证号放入衣服表中即可
* A和B之间是 多对多 关系，则建立一张中间表，把AB两表中的唯一键放入中间表即可：比如手机表和APP表，一个手机可以安装多个APP，一个APP可以被多个手机安装。那么我们将APP的ID号和手机的手机号一同放入一张中间表（也叫关联表）中即可。

遵照上面的规则去设计数据库中的数据表就没有什么问题了。