**第8章 设计web数据库**

**1. 关系数据库的概念**

**1.1 表格，即关系**

**Book-O-Rama书店客户信息表：**



**一行对应一个客户**

**1.2 列**

**CustomerID列是一个整型数据，而其他3列是字符串类型。列，也叫做域或属性。**

**1.3 行**

**每一行具有相同的格式，也具有相同的属性。行，也叫做记录。**

**1.4 值**

**每一行有对应于每一列的单个值组成，每个值，必须与该列定义的数据类型相同。**

**1.5 键**

**我们必须有一个能识别每一个特定客户的方法。用名字，不好，会有同名的。**

**为每一个客户分配一个唯一的CustomerID。类似于银行账号、会员号。**

**标识列称为键或主键。一个键，可以由几列组成。**

**键，还可以作为表格之间的引用。**





**ORDERS表格存储了客户的订单，每一行表示一个订单，由一个客户所预订。通过CustomerID，再查看CUSTOMERS表，就知道客户姓名。**

**这种关系，就是外键。CustomerID是CUSTOMERS表的主键，在ORDERS表中，它是外键。**

**1.6 模式**

**数据库整套表格的完整设计，称为数据库的模式。一个模式，应该显示表格即表格的列、每个表的主键和外键。**

CUSTOMERS(**CustomerID**,Name,Address,City)

ORDERS(**OrderID**,*CustomerID*,Amount,Date)

1.7 关系

外键 表示两个表格数据的关系。

关系类型有3种：一对一、一对多、多对多。

例如，如果将address放入一个独立的表中，则该表与CUSTOMERS表就是一对一的关系。

例如，一个客户可能有许多订单，CUSTOMERS表与ORDERS表，就是一对多的关系。包含多行的表对应与包含一行的表，应该有一个外键。

例如，BOOKS 和AUTHORS表，一本书可能有两个作者，这两个作者有可能独自著书，或者与其他人合著。通常，这种关系类型，各自都要有一个表，因此，可能需要BOOKS AUTHORS BOOKS\_AUTHORS三个表，第三个表包含其他两个表中的键，将其作为外键对，用来显示哪些作者写了哪些书。

**2. 设计web数据库**

2.1 通过分析，**Book-O-Rama书店应该记录：**

**客户信息、订单信息、图书信息**



**现在，我们还无法知道 哪本书在哪个订单中被订购了。**

**2.2 避免保存冗余数据**

**为什么不能将客户的地址保存在ORDERS表中？**

**1，空间浪费，每次订购，都得写入地址**

**2，容易导致数据更新不一致，比如，地址变动，则所有记录都需要修改地址**

**3，假设订单已经交货，需要删除，那么连同地址一起删除了**

**2.3 使用原子列值**

**对每一行的每个属性，只存储一个数据。例如，订单与图书，是多对多的关系。我们需要知道某一本图书有多少订单，就需要在表ORDERS和BOOKS之间建立一个关联。创建一个新表：ORDER\_ITEMS**



**2.4 选择有意义的键**

**应该确保所选择的键，是唯一的。**

**客户表CustomerID 订单表OrderID 图书表ISBN 在ORDER\_ITEMS表中，OrderID 和 ISBN这两个属性的组合可以是唯一的。**

**2.5 考虑需要询问数据库的问题**

**例如，Book-O-Rama书店，哪些图书卖得最好？**

**从而确定数据库中包含所有需要的数据，并且在表之间有适当的关联。**

**2.6 避免多个空属性的设计**

**希望在数据库中添加一些图书评论，可以**

**（1）在BOOKS表中，增加一列 Review**

**（2） 增加一个新表 BOOKS REVIEWS**



**第一种方法，如果图书很多，评论员无法评论所有的图书，则Review列空值很多，会浪费大量空间，统计列总量或其他数值时，可能出错。**

**选择第二种方法，就避免了这些问题。这里，假设一本图书只有一个评论。**

**2.7 表格类型的总结**

**数据库由两种类型的表组成：**

**1，描述现实世界对象的简单表。它们之间有一对一或一对多的关系。**

**2，描述两个现实世界对象的多对多关系的关联表。通常，是与现实世界某种事务处理相联系的。**

**3. web数据库架构**



**web浏览器与web服务器的 C/S 关系**

**这种架构，适合服务器发布静态页面。**



**web数据库的基本结构，包括web浏览器、web服务器、脚本引擎、数据库服务器**

**以Book-O-Rama书店为例。步骤如下：**

**1）用户的web浏览器发出HTTP请求，请求特定的web页面。例如，该用户可能以HTML表单形式搜索Book-O-Rama书店所有Laura编写的图书，搜索结果页为result.php**

**2）web服务器收到result.php的请求，获取该文件，并将它传到php引擎，要求它处理**

**3）PHP引擎开始解析脚本。脚本中有一条连接数据库的命令，还有执行一个查询的命令。PHP打开通向MYSQL数据库的连接，发送适当的查询**

**4）mysql服务器接受数据库查询并处理。将结果返回到PHP引擎。**

**5）PHP引擎完成脚本运行，将查询结果格式化成HTML格式，将输出的HTML返回到web服务器。**

**6）web服务器将HTML发送到浏览器。 这样用户就可以看到所搜索的图书了。**