

华侨大学

信息科学与工程学院

课程设计报告

(2014~2015 学年 第 一 学期)

课程设计题目: 旅行社管理信息系统

课程名称: 数据库课程设计

专业年级: 自动化 2011 级

班 级: _____

学 号: _____

姓 名: _____

日 期: 2014 年 12—2015 年 1 月

指导教师: _____

目 录

引言	3
第一章 概述	4
1.1 数据库设计背景	4
1.2 数据库设计意义	4
1.3 数据库设计内容	4
第二章 旅行社管理信息系统设计	5
2.1 需求分析	5
2.2 概念模式设计	6
2.3 逻辑模式设计	9
2.4 数据库的实施与维护	10
总结	17
参考文献	18

引言

《数据库课程设计》是自动化专业的实践课程。在较为系统地学完《数据库技术及应用》后，通过此课程设计，一方面可以增强学生对数据库的理解；另一方面可以培养学生查找资料、分析问题和解决问题的能力。

关系数据库设计过程分为以下 5 个阶段：需求分析、概念模式设计、逻辑模式设计、数据库实施、数据库运行维护。也就是说，数据库的设计过程是先进行认真细致的需求分析，在清楚用户的要求后抽象出实体和实体之间的关系，用建模方法 E—R 表示出来，然后根据具体实施的 DBMS 平台将其转换为相应平台所支持的数据库逻辑模式并进行规范化处理，最后进行建库建标等物理模式的设计，从而完成整个数据库模式的设计过程。

本次课程设计是对旅行社管理信息系统的设计，设计过程中也严格遵循以上的 5 个阶段。在设计完成的数据库中，应该具备有两大功能：1、对于用户，数据库的设计应该满足团员个人信息的查询、团队信息的查询和旅行路线的查询；2、对于管理员，数据库的设计应该满足团员信息、团队信息、旅行路线的添加、删除、更新、取消等操作。也就是，数据库设计应该具备基本的增删改查功能。

在第一部分，我们将介绍旅行社管理信息系统的设计背景、设计目的、和设计内容。通过介绍对大数据时代的旅行社数据管理，旅行社信息服务有一个初步的认识。

第二部分是本报告的重点，基于旅行社管理信息系统，对数据库设计的 5 个阶段，即需求分析、概念模式设计、逻辑模式设计、数据库实施、数据库运行与维护。

数据库的设计对于旅行社有事半功倍的效果，如何提高办事效率，一直是数据库工作者的研究课题！

第一章 概述

1.1 数据库设计背景

随着科技发展和社会进步,尤其是计算机大范围的普及,计算机应用逐渐由大规模科学计算的海量数据处理转向大规模的事务处理和对工作流的管理,以数据库管理系统为开发环境的管理信息系统在大规模的事务处理和对工作流的管理等方面的应用,特别是在旅行社信息管理之中的应用日益受到人们的关注。

经济的发展,人民生活水平的提高,越来越多的人选择了在节假日旅游作为放松的一种方式。旅行社是我国旅游业的龙头产业,具有显著的市场导向作用和强大的产业关联作用。近年来我国信息事业发展迅速,手工管理方式在旅行社信息管理等需要大量事务处理的应用中已显得不相适应,采用信息技术提高服务质量和管理水平势在必行。目前,对外开放必然趋势使信息行业直面外国同行单位的直接挑战,因此,信息行业必须提高其工作效率,改善其工作环境。这样,旅行社信息管理的信息化势在必行。

1.2 数据库设计意义

作好旅行社的经营管理与对游客的服务工作是旅行社工作管理的一项重要内容。首先为了方便从事旅行社工作人员的管理工作,提高工作效率,更重要的是有助于提高旅行社的信誉。

旅行社的业务是一项复杂的组织工作。旅行社信息管理以游客、旅行团队、旅行路线的信息管理为核心,在此过程中又需要经过若干道手续,因为整个过程都需要手工操作,效率十分低下,且由于他们之间关联复杂,统计和查询的方式各不相同;且会出现信息的重复传递问题。这种复杂性不仅仅指旅游活动程序安排上环节众多,更突出地表现在旅行社服务对象(即旅游者)的多样性。所以开发一个实用、高效的旅行社管理信息系统是很有必要的。

1.3 数据库设计内容

运用所学的数据库技术理论知识,,对旅行社信息管理系统进行数据库设计。设计应包含需求分析、概念模式设计、逻辑模式设计、数据库实施、数据库运行维护五个阶段。数据库设计完成后,应该包含以下几个功能:

- 1、旅游团队、团队团员及旅游路线相关信息的维护(修改、浏览、删除和撤销);
- 2、旅游团队管理信息的查询(如按团队编号);
- 3、团队团员基本情况的查询(可选多种方式);
- 4、旅游路线相关信息的查询(如按线路编号)。

第二章 旅行社管理信息系统设计

（重点篇幅，请根据自己的设计具体阐述！之前的仪器仪表课设报告，很多同学出现自己设计部分的内容比如软件设计内容写得非常少，参考资料内容偏多的状况，这不符合我们的设计要求！）

本次对于旅行社信息管理系统的设计，应该包含以下 5 个阶段：需求分析、概念模式设计、逻辑模式设计、数据库实施、数据库运行维护。如下图所示：

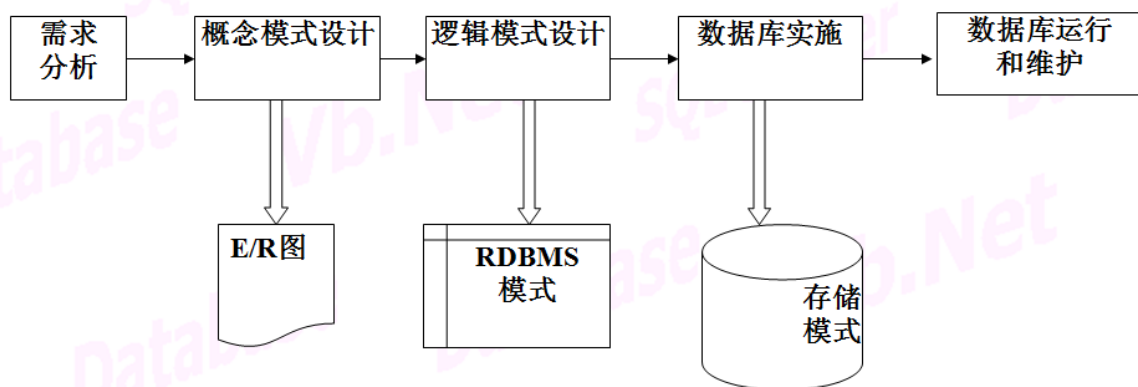


图 2.1

2.1 需求分析

在此次《数据库课程设计》中，根据课程内容安排，所选题目为：旅行社管理信息系统。指导老师根据设计内容，已经给出了设计表的基本结构：

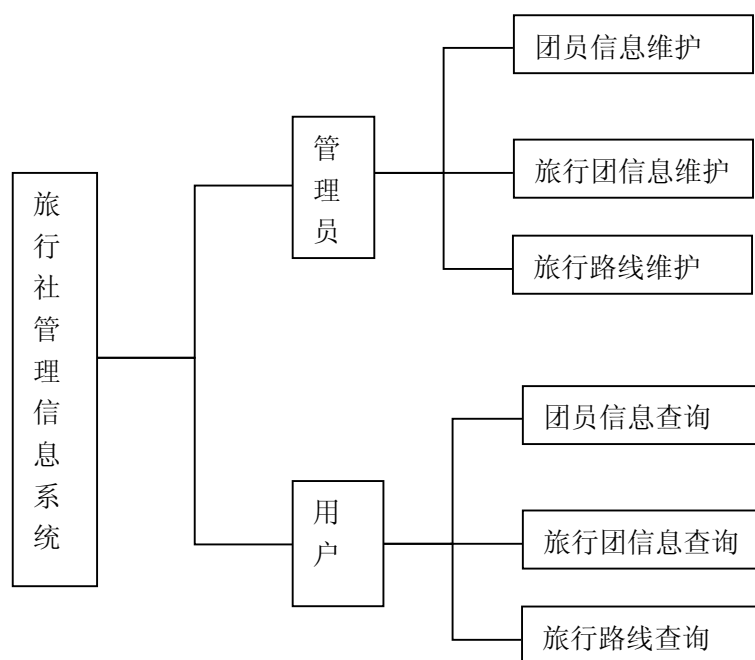
- 1、团员信息表（路线编号，团队编号，团员编号，姓名，性别，电话，通信地址，身份证号码，团费交否，备注）
- 2、线路信息表（路线名称，团费，简介，图形，路线编号）
- 3、团队信息表（团队编号，路线编号，团员人数，出发日期，返程日期）
- 4、旅游团队信息表（团队编号，团队负责人，团员人数，建团时间，是否出发，团费，盈亏）
- 5、密码信息（操作员，密码）

根据以上 5 个表的属性，可以总结出，设计出来的数据库应该面向管理员和团员。对于团员，应该具备团员信息、旅行团信息、旅行路线、出游信息的查询。对于管理员，则需要包含有团员信息、旅行团信息、旅行路线、出游信息的维护，即包含基本的增加、删除、更新、浏览等。

从上面表的结构关系较为混乱，如果没有进行必要的修改，将导致：（1）数据冗

余；(2) 修改异常；(3) 删除异常；(4) 插入异常等状况。所以接下来将根据需求分析所需要的功能做后续的分析 and 设计。

下图为简单的数据流程图：



2.1.1 数据流程图

2.2 概念模式设计

数据库概念模式与计算机实现无关。它是对现实世界的第 1 层抽象。

直接把现实世界事物及其联系抽象为数据世界的数据库模型过于复杂，因此一般是先将现实世界的事物及其联系抽象为信息世界实体及其联系（概念模型），然后再将其转换为计算机世界的数据库模型（关系数据库模式）。

实体-联系模型（E-R 模型）是 1976 年美籍华人陈平山提出的。这个模型直接将现实世界中的事物及其之间的联系抽象为实体类型和实体间联系，然后用实体联系图表示数据模型。

E-R 模型是用 E-R 图表示的。E-R 图中有下面四个基本成分：①矩形框：表示实体类型；②菱形框：表示联系；③椭圆形框：表示实体或联系类型的属性；④直线：联系类型与其所涉及的实体之间用直线连接，实体与实体之间用直线连接。

E-R 模型的基本元素由以下几个：(1)实体：客观存在并可相互区别的事物称为实体。(2)属性：实体所具有的某一特征称为实体的属性。(3)键：也称关键字或码。唯一标识实体的最小的属性集称为实体的键。(4)联系：现实世界的事物彼此是有联系的，反映在信息世界就是实体之间的联系。

实体间的相互关系称为联系。联系在数据库中的反映是实体集之间存在着这样和那样的联系，这种联系实际上表示了实体集之间的某种函数映射关系。可分为以下三类：

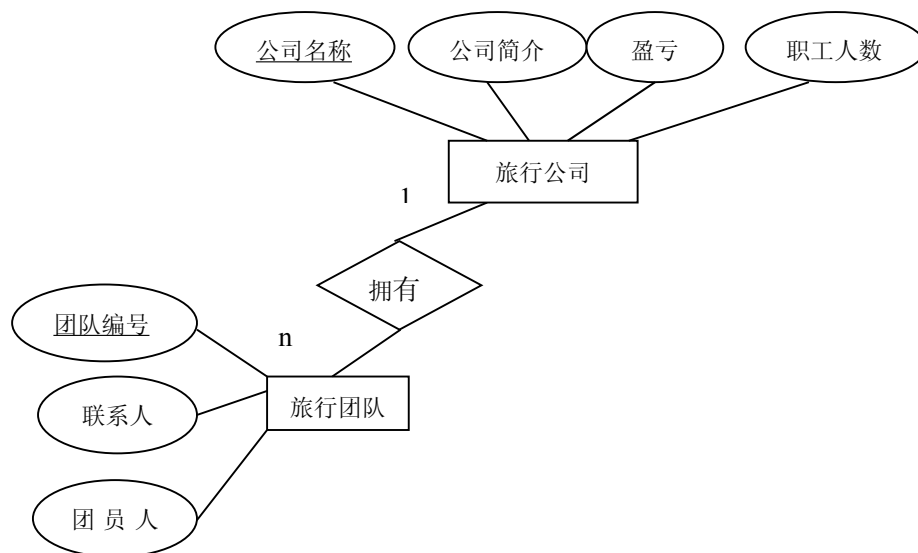
①1:1 联系:已知实体集 A 和 B，若其中每个实体集中任一实体至多与另一实体集中的一个实体有联系，则称 A 和 B 的联系为“1 对 1 联系”，简记为 1:1 联系。

②1:n 联系: 已知实体集 A 和 B，若 A 中每个实体可与 B 中任意个实体有联系，而 B 中每个实体至多与 A 中的一个实体有联系，则称 A 和 B 的联系为“1 对多联系”，简记为 1:n 联系。

③m:n 联系: 已知实体集 A 和 B，若其中每个实体集中任一实体可与另一实体集中的多个实体有联系，则称 A 和 B 的联系为“多对多联系”，简记为 m:n 联系。

在下面的图形中我们首先将进行局部 E—R 图的分析设计，接着将进行全局 E-R 图设计。

1、局部 E-R 图设计



2.2.1 旅行公司和旅行团队的局部 E-R 图

分析：从以上的旅行公司和旅行团队的局部 E-R 图中，我们可以看到：一个旅行公司可以拥有许多旅行团队，一个旅行团队只属于一个旅行公司，所以是 1:n 的关系。其中旅行公司主键：公司名称；旅行团队主键：团队编号。

不提供参考

2.2.2 旅行公司和旅行路线的局部 E-R 图

分析:

不提供参考

2.2.3 旅行团队和旅行路线的局部 E-R 图

分析:

不提供参考

2.2.4 旅行团队和团员的局部 E-R 图

分析:

2、全局 E-R 图设计

将局部的 E-R 图进行组合将变成全局的 E-R 图,通过全局的 E—R 图可以清晰的看出各个实体之间的关系。

不提供参考

2.2.5 旅行社管理信息系统全局 E—R 图

2.3 逻辑模式设计

数据库概念模式设计完成后,数据库的设计进入到逻辑模式设计阶段。这时的数据库设计与具体的机器世界是相关联的。逻辑模式设计阶段有两大任务:1、按一定的规

则将 E-R 图转换成关系模式；2、关系模式的规范化处理。所谓关系模式的规范化处理，就是对关系模式进行某种处理，使它满足一定的规范化要求。规范化的主要目的就是要尽可能地消除关系操作过程中的异常情况。

根据 E-R 模型向关系模型转换规则，对上述的 E-R 图进行转换，转换结果如下：

1、旅行公司（公司名称，公司简介，盈亏，职工人数）

。。。。。。部分关系不提供参考

分析：主键和外键设置的原因。

对以上的关系模式进行规范化处理：

我们设计的最终目的是使关系模式规范化，最低应该达到第三范式。

首先，上述的关系中的每一个分量都是不可再分的数据，所以符合第一范式要求。

其次，符合第二范式的关系必须要属于第一范式，且每个非主属性都完全依赖于主键。在上述的关系中，由于旅行路线表、团队信息表、旅行路线表和团员信息表的主键只有一个，所以肯定不存在部分依赖关系，属于第二范式。在关系“出游”中有两个主键，但是非主属性“返程日期”、“出发日期”必须要由两个主键“团队编号”、“路线编号”共同决定。所以上述关系属于第二范式。

最后，要符合第三范式，必须要达到第一范式的要求，且每一个非主属性都不传递依赖于主键。在“旅行公司”、“团队信息”、“出游”、“旅行路线”、“团队信息表”中不存在有传递函数依赖于主键，所以上述关系达到第三范式。

2.4 数据库的实施与维护

2.4.1 界面开发

旅行社管理信息系统选择在 VB.NET 环境下进行开发。

设计出来的数据库应该面向管理员和团员。对于团员，应该具备团员信息、旅行团信息、旅行路线、出游信息的查询。对于管理员，则需要包含有团员信息、旅行团信息、旅行路线、出游信息的维护，即包含基本的增加、删除、更新、浏览等。

对于用户来讲，能对旅行社管理的基本信息进行查询：



图 2.4.1 团员信息的查询

对于操作员来讲，需要对旅行社的基本信息进行维护：



图 2.4.2 团员信息的维护

2.4.2 源代码的讲解及运行效果截图

以下将对 VB 访问数据库技术的代码进行注释讲解，由于用户的查询程序和管理员的维护程序具有代表性，这里将做重点介绍。至于其他程序，多有雷同，限于篇幅，就不再赘述。

下面的程序是对团员信息进行的模糊查找：

```
' 清空数据集ds，因为ds是全局变量，在控件下调用时必须清空更为安全  
ds.Clear()
```

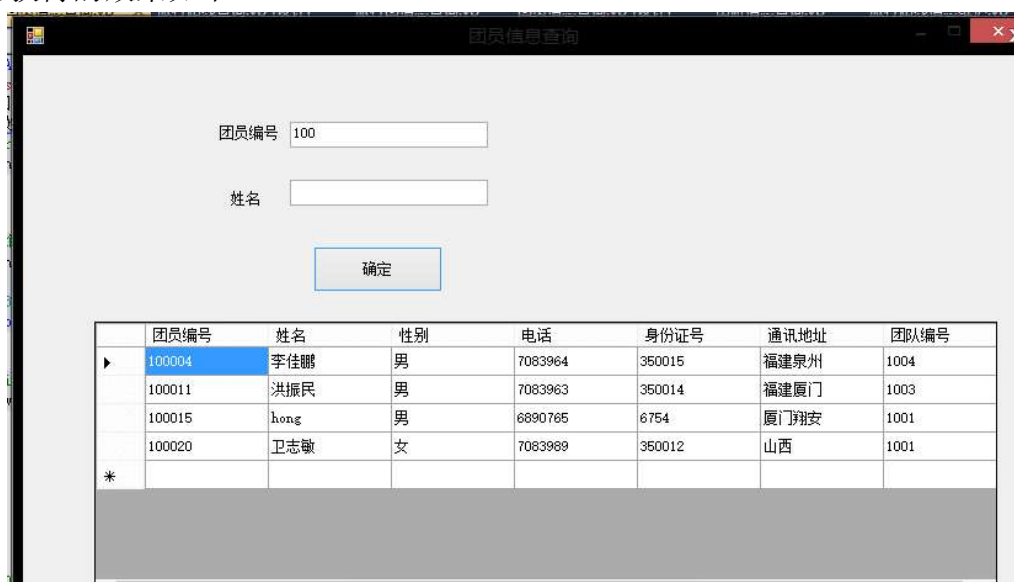
```

' 连接数据库
Dim strConnection As String =
"Server=.....;Database=.....;Integrated Security=....."
Dim myConnection As SqlConnection = New SqlConnection(strConnection)
' 定义存放sql语句的字符串
Dim sqlstr As String
' 对团员信息表的团员编号和姓名进行模糊查找
sqlstr = " select * From 团员信息表 Where 团员编号 like '" & _
    TextBox团员编号.Text & "%'" & " and 姓名 like '" & _
    TextBox姓名.Text & "%'" & " 'sql语句
' 创建adapter实例，并指定要执行的sql语句文本
Dim myCommand As SqlDataAdapter = New SqlDataAdapter(sqlstr,
myConnection)

Try
    ' 根据查询条件填写数据集ds
    myCommand.Fill( ..... )
Catch ex As Exception
    MessageBox.Show(ex.Message)
Exit Sub
End Try
' 设置DataGridView1的数据源
DataGridView1.DataSource = ds.Tables( ..... )

```

程序运行后获得的效果如下：



2.4.3 团员信息查找截图

下面是对团员信息的维护（添加、删除、更新、确认、取消）：

在对程序讲解的前面，我们应该重点介绍数据库的一种访问方法，那就是：基于绑定的数据访问。

用户希望能使用用户界面控件来浏览数据和修改数据，并把对数据的修改保存到数据源中。数据库的“绑定”方法允许自动用底层数据源中的数据填充窗体控件，通过绑

定对象的位置指针 (position)，使窗体各个控件能同步再现数据源的信息，还提供了一种机制，可以更新底层的数据源，以响应用户做出的修改。

以为源代码：

'' 导入命名空间

```
Imports System.Data.SqlClient
```

```
Public Class 团员信息维护
```

```
    Inherits System.Windows.Forms.Form
```

```
    ' 建立类中可访问的数据集对象(), 名为ds, 且为全局变量
```

```
    Dim ds As DataSet = New DataSet()
```

```
    Public mybind As BindingManagerBase
```

```
    Private Sub 团员信息维护_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
```

```
        ' 建立一个连接字符串
```

```
        Dim strConnection As String = "Server=localhost;Database=旅行社信息管理系统;Integrated Security=SSPI"
```

```
        ' 建立一个查询命令字符串
```

```
        Dim strSql As String = "SELECT 团员编号, 姓名, 性别, 电话, 身份证号, 通讯地址, 团队编号 FROM 团员信息表"
```

```
        ' 建立一个数据连接
```

```
        Dim myConnection As SqlConnection = New SqlConnection(strConnection)
```

```
        ' 建立一个数据适配器以便针对数据源执行SELECT 语句来提取出要填入数据集的数据记录
```

```
        Dim myCommand As SqlDataAdapter = New SqlDataAdapter(strSql, myConnection)
```

```
        ' 将数据填入数据集
```

```
        myCommand.Fill(..... )
```

```
        ' 指定绑定的数据源
```

```
        mybind = Me.BindingContext(.....)
```

```
        ' 将TextBox 控件的Text 属性绑定至数据集ds 内的学生数据表的学号、姓名等字段
```

```
        TextBox团员编号.DataBindings.Add("Text", ds, "团员信息表. 团员编号")
```

```
        .....
```

```
    End Sub
```

```
    ' 按下 第一条 按钮
```

```
    Private Sub ButtonFirst_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ButtonFirst.Click
```

```
        ' 将 Position 属性设定成 0
```

```
        mybind.Position = 0
```

```
    End Sub
```

```
    ' 按下 上一条 按钮
```

```
    Private Sub ButtonUp_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ButtonUp.Click
```

```

    If mybind.Position > 0 Then
        ' 将 Position 属性递减 1
        mybind.Position -= 1
    End If
End Sub
' 按下 下一条 按钮
Private Sub ButtonDown_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles ButtonDown.Click
    If mybind.Position < Me.mybind.Count - 1 Then
        ' 将 Position 属性递增 1
        mybind.Position += 1
    End If
End Sub

Private Sub ButtonLast_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles ButtonLast.Click
    ' 按下 最后一条 按钮
    mybind.Position = mybind.Count - 1
End Sub

Private Sub Button1_Click_1(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    ' 添加操作
    mybind.AddNew()
End Sub

Private Sub Button3_Click_1(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
    ' 删除操作
    mybind.RemoveAt(mybind.Position)
End Sub

Private Sub Button5_Click_1(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
    ' 更新操作
    mybind.EndCurrentEdit()
End Sub

Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
    mybind.EndCurrentEdit()
    ' 建立一个连接字符串
    Dim strConnection As String = "Server=localhost;Database=旅行社信息
管理系统;Integrated Security=SSPI"
    ' 建立一个查询命令字符串
    Dim strSql As String = "SELECT 团员编号, 姓名, 性别, 电话, 身份证号, 通讯

```

地址, 团队编号 FROM 团员信息表”

```

' 建立一个数据连接
Dim myConnection As SqlConnection = New SqlConnection(strConnection)
' 建立一个数据适配器以便针对数据源执行SELECT 语句来提取出要填入数据
集的数据记录
Dim myCommand As SqlDataAdapter = New SqlDataAdapter(strSql,
myConnection)
Dim SB As New SqlCommandBuilder(.....)
' 将数据填入数据集
myCommand.Update(.....)

End Sub

```

```

Private Sub Button2_Click_1(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
' 取消操作
mybind.CancelCurrentEdit()
End Sub
End Class

```

程序运行后获得的效果如下：

图2.4.4 团员信息表的维护

1、当按下添加按钮后，所有行将清0，填入所需添加数据，按下确认按钮，在源程序将找到所添加行。

2、当按下删除按钮时，界面上的数据将被删除，当按下确认按钮时，源程序将把那条数据删除掉。

3、修改完数据后，按下更新再按确认键或者直接按确认键后，修改的数据将保存到源程序当中。

4、修改了一点数据，发现不想修改了，按下取消，修改的数据将重新回到未修改前。

总 结

通过这次课程设计，我对数据库有了更深刻的体会。

1、在做数据库设计时，必须包含需求分析、概念模式设计、逻辑模式设计、数据库实施、数据库运行与维护 5 个阶段，每一个都不可偏废。在开始做数据库时，请教过郑老师，他叮嘱我一定要做好需求分析、ER 图设计，前面几个阶段做好了，后面 VB 与数据库连接是水到渠成的事情。

2、在数据库与 VB 连接方法上刚开始一头雾水，在郑老师的指导下，运用绑定对象集的方法，程序较为简单，理解起来也容易

另外，这次数据库的课程设计，让我懂得如何高效率的查找资料。查找资料有针对性，事情往往能事半功倍。

同时，我很感谢老师的教导，在我一窍不通的时候，他们总能不吝赐教。感谢同学的帮助，让我的学习更有方向。

参考文献

- [1] 龚沛曾, 袁科萍, 杨志强 . 数据库原理及应用[M].北京: 高等教育出版社
- [2] 张丽萍, 汤涌涛 . Visual Basic 2010 从入门到精通[M].北京: 清华大学出版社