

## 第 10 章 系统开发——VB+Oracle 9i

本章介绍如何利用中文版 Visual Basic 6.0（简称 VB）为前端开发工具，后台数据库为 Oracle 9i 进行应用系统的开发。

### 10.1 如何构建开发和运行的环境

#### 10.1.1 如何选择前台开发工具

#### 10.1.2 如何构建开发和运行环境

一个 VB+Oracle 9i 的数据库应用从逻辑上看有 3 个组成部分。如图 10.1 所示。

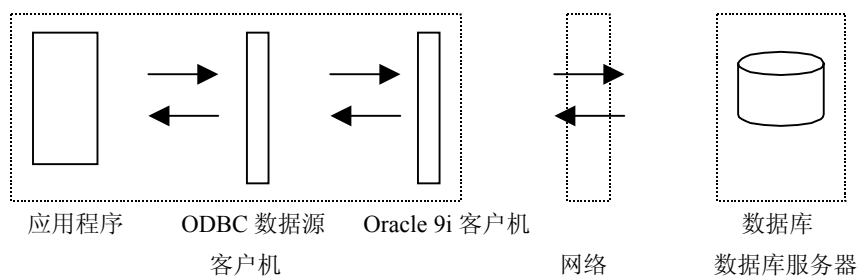


图 10.1 VB 数据库应用的结构

#### 1. 构建开发和运行客户机

在开发客户机上进行以下工作。

- (1) 安装 VB。
- (2) 安装 Oracle 9i 客户机，最好是管理员，这样可以及时管理后台数据库服务器上的数据。

(3) 测试 Oracle 9i 的客户机是否能够正常连通数据库服务器。

在运行客户机上进行如下工作。

- (1) 安装用 VB 开发好的应用程序。
- (2) 安装 Oracle 9i 客户机，最好是连接，这样终端用户不能直接操作数据库。
- (3) 测试 Oracle 9i 的客户机是否能够正常连通数据库服务器。

2. 保证网络畅通

确保网络 TCP/IP 协议能够畅通。

3. 构建数据库服务器

2.1 节已经详细做了介绍。

10.1.3 环境构建实例

1.检查数据库服务器工作情况

2.安装 Oracle 9i 客户机

图 10.2 所示为 Oracle 9i 客户机安装步骤中的【安装类型】界面。

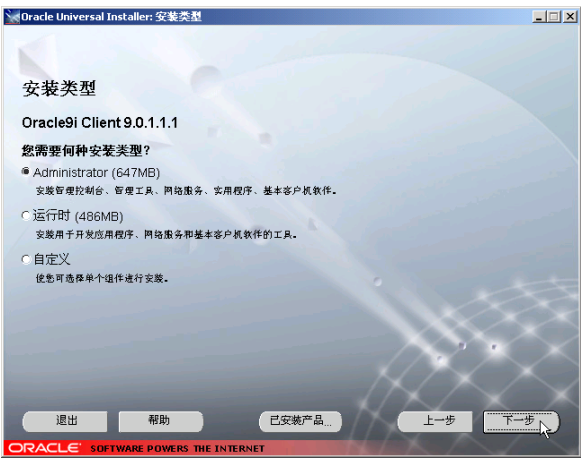


图 10.2 【安装类型】界面

3.安装 VB

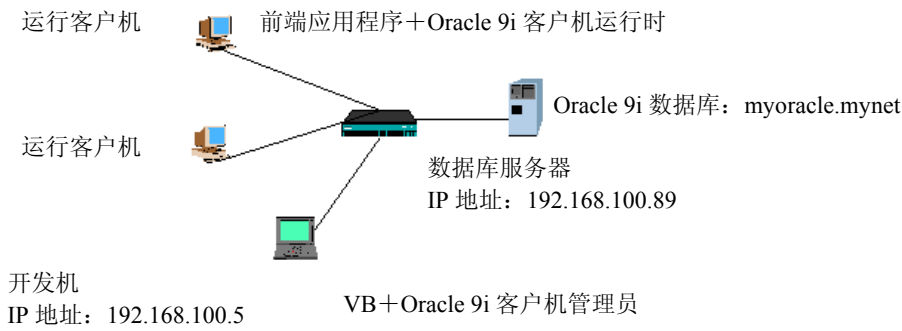


图 10.3 VB+Oracle 9i 数据库应用开发环境

## 10.2 VB 程序如何连接数据库

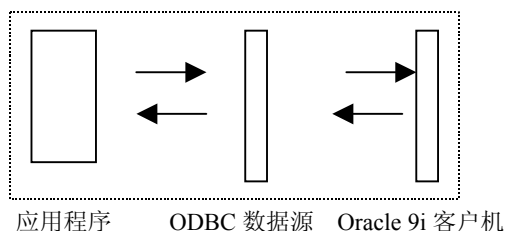


图 10.4 运行客户机的结构

### 10.2.1 ODBC 数据源

#### 1. 什么是 ODBC

ODBC (Open Database Connectivity, 开放式数据库互联), 是目前国际上通用的数据库访问标准。

#### 2. 什么是 ODBC 数据源

【ODBC 数据源】就是计算机上的 ODBC 配置和管理的工具, 利用这个工具, 用户就可以定制使用 ODBC 来连接数据库。

### 10.2.2 Oracle 9i 客户机

### 10.2.3 VB 程序连接数据库的过程分析

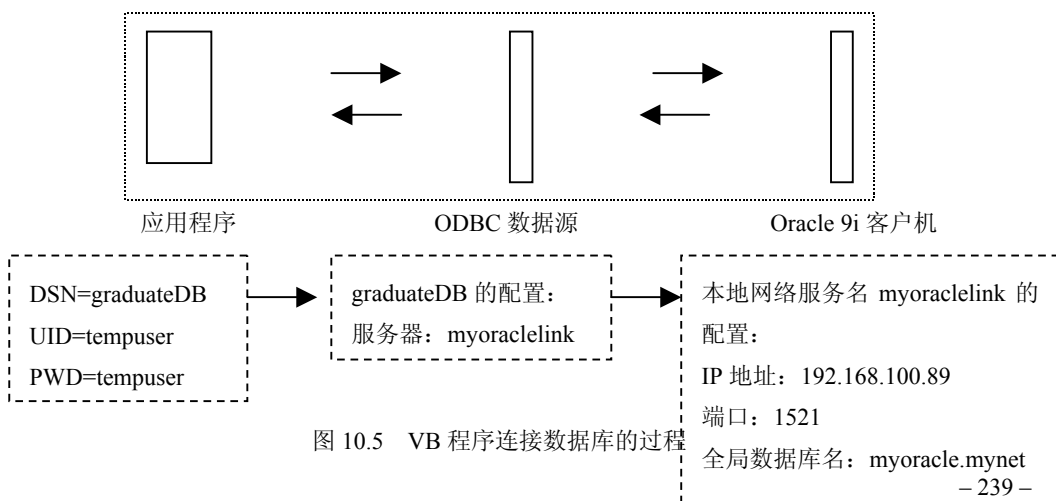


图 10.5 VB 程序连接数据库的过程

1. VB 程序指定 DSN 名称
2. 【ODBC 数据源】解析 DSN 名称
3. Oracle 9i 客户机解析数据库服务器名称
4. 数据库服务器验证 DSN 中的用户名和密码

## 10.2.4 如何建立 ODBC 数据源

- (1) 如图 10.6 所示的 ODBC 数据源管理器的【用户 DSN】选项卡。
- (2) 切换到如图 10.7 所示的 ODBC 数据源管理器的【系统 DSN】选项卡。



图 10.6 【用户 DSN】选项卡

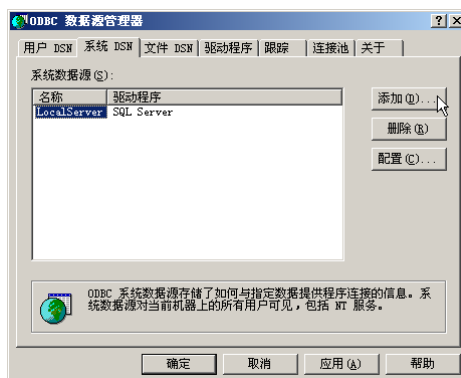


图 10.7 【系统 DSN】选项卡

- (3) 出现如图 10.8 所示的【创建新数据源】界面。



图 10.8 【创建新数据源】界面

- (4) 出现如图 10.9 所示的【Microsoft ODBC for Oracle Setup】界面。



图 10.9 【Microsoft ODBC for Oracle Setup】界面

(5) 成功建立系统 DSN 名称后的【系统 DSN】选项卡如图 10.10 所示。

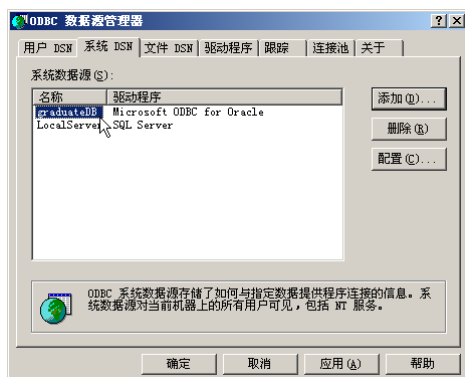


图 10.10 成功建立的 ODBC 数据源

## 10.3 VB 程序如何访问数据库

### 10.3.1 通过【Data】控件访问数据库

- (1) 启动 VB。
- (2) 出现如图 10.12 所示的新建工程的【新建】选项卡。

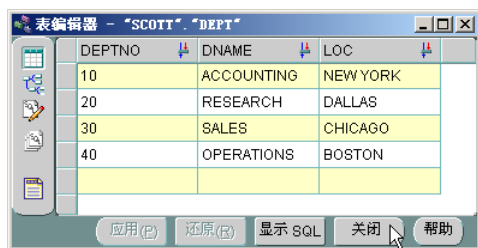


图 10.11 scott.dept 数据表的数据



图 10.12 新建工程的【新建】选项卡

(3) 出现如图 10.13 所示的 VB 主界面。

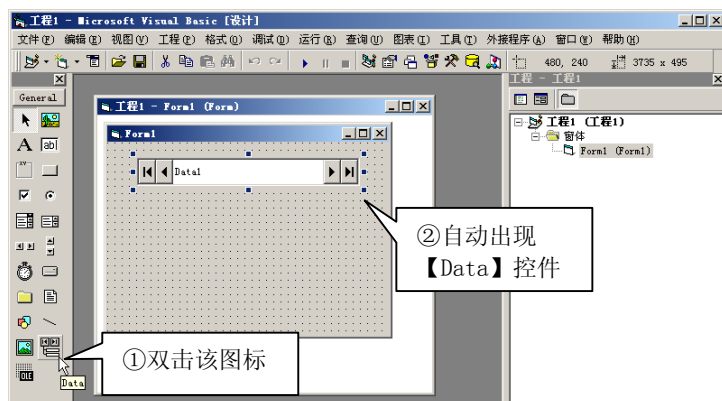


图 10.13 VB 主界面

(4) 如图 10.14 所示。

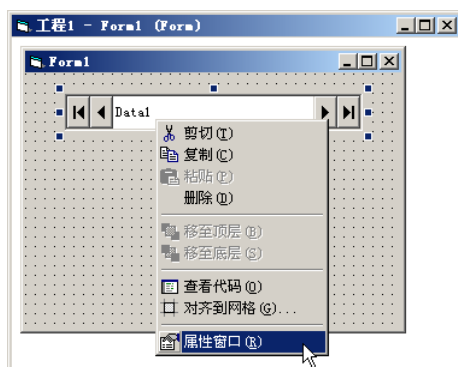


图 10.14 选择查看 data1 控件的属性窗口

(5) 出现如图 10.15 所示的【属性窗口】界面。

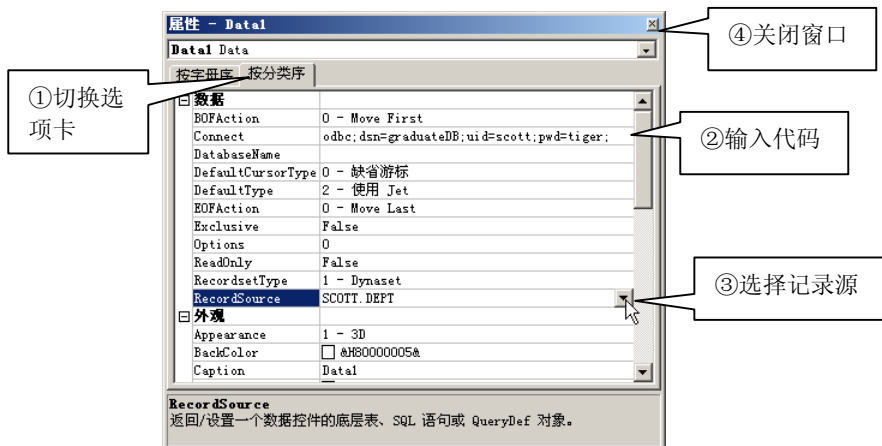


图 10.15 【Data】控件的【属性窗口】界面

odbc;dsn=graduateDB;uid=scott;pwd=tiger;

(6) 如图 10.16 所示。

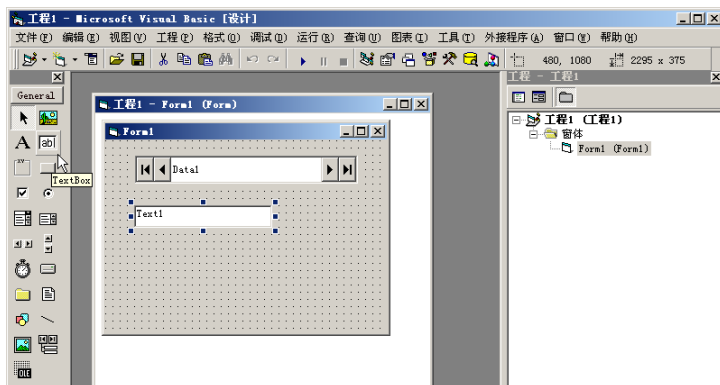


图 10.16 向窗体中添加文本框控件

(7) 选择【Form1】窗体上出现名为 Text1 的【TextBox】控件。

(8) 出现如图 10.17 所示的【TextBox】控件的属性窗口。

(9) 按照和 Text1 同样的步骤添加名为 Text2 的【TextBox】控件，其主要的设置为。

(10) 按照和 Text2 同样的步骤添加名为 Text3 的【TextBox】控件，其主要的设置为。

(11) 设计好的窗体 Form1 如图 10.18 所示。

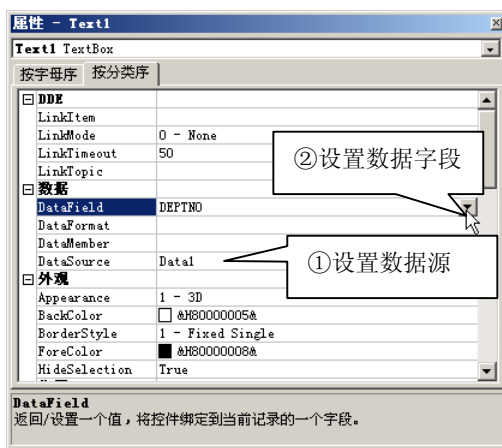


图 10.17 【TextBox】控件的【属性窗口】界面

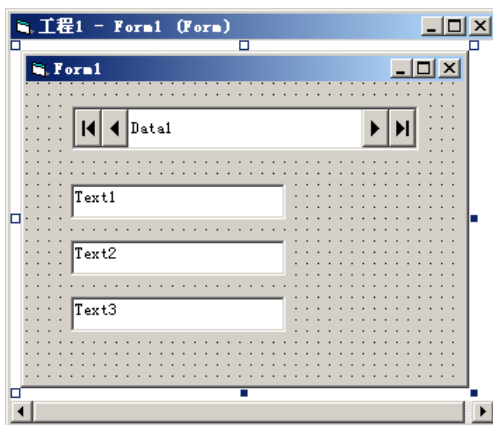



图 10.18 设计好的 Form1 窗体

(12) 在 VB 主界面的【快捷按钮区】单击运行快捷按钮, 如图 10.19 所示。

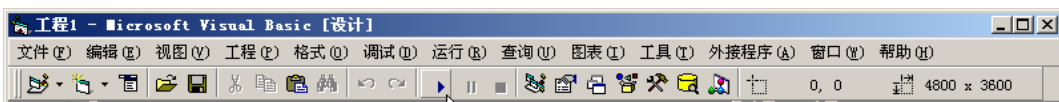


图 10.19 选择运行 Form1 窗体

(13) 运行结果如图 10.20 所示。

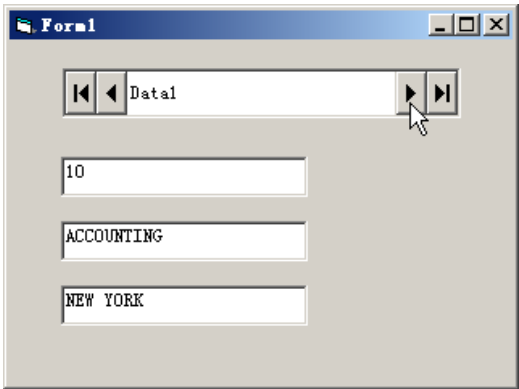


图 10.20 Form1 的运行结果

(14) 读者可以从本书的配套光盘上找到实例的配套程序。

【参见光盘文件】: 第 10 章\使用 Data 控件访问数据库.vbp, 窗体为 Form1.frm。

(15) 利用表格来表示上述过程中窗体的设置, 如表 10.1 所示。

表 10.1 Form1 窗体的设计

控件名称	类型	主要属性	属性值
Form1	【窗体】	名称	Form1
Data1	【Data】	Caption	Data1
		Connect	odbc;dsn=graduateDB;uid=scott;pwd=tiger;
		RecordSource	Scott.dept
Text1	【TextBox】	DataSource	Data1
		DataField	DEPTNO
		名称	Text1
Text2	【TextBox】	DataSource	Data1
		DataField	DNAME
		名称	Text2
Text3	【TextBox】	DataSource	Data1
		DataField	LOC
		名称	Text3

### 10.3.2 通过【ADODC】控件访问数据库

(1) 启动 VB, 建立新的工程。

(2) 如图 10.21 所示的 VB 主界面的【菜单栏】选择【工程】/【部件】选项。



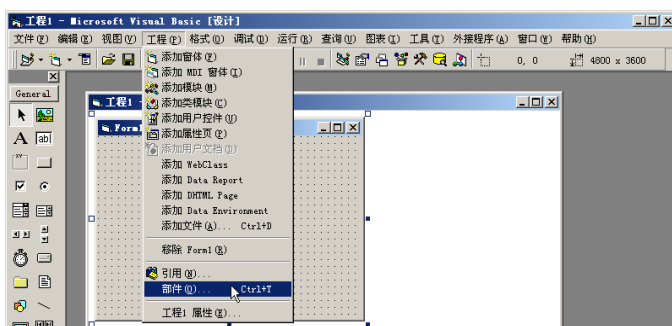


图 10.21 选择加载【ADODC】控件

(3) 出现如图 10.22 所示的部件的【控件】选项卡。

(4) 如图 10.23 所示。

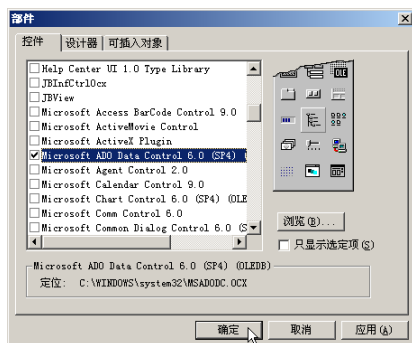


图 10.22 部件的【控件】选项卡

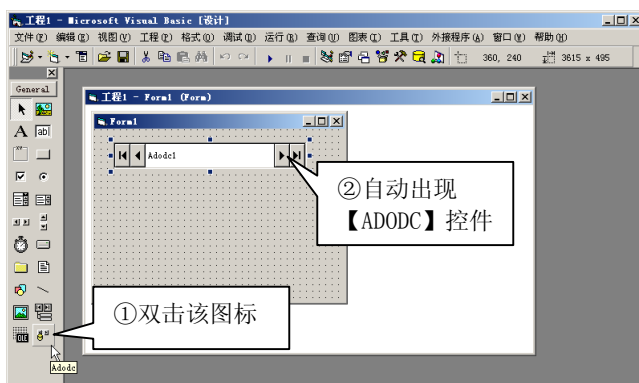


图 10.23 向窗体中添加【ADODC】控件

(5) 如图 10.24 所示。

(6) 出现如图 10.25 所示的属性页的【通用】选项卡。



图 10.24 选择查看 adodc1 的属性窗口

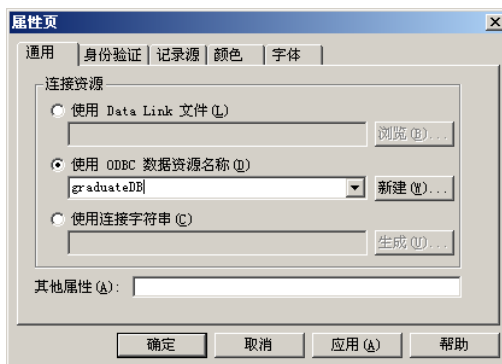


图 10.25 属性页的【通用】选项卡

(7) 切换到如图 10.26 所示的属性页的【身份验证】选项卡。

(8) 切换到如图 10.27 所示的属性页的【记录源】选项卡。

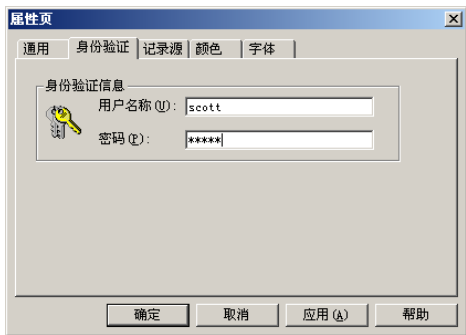


图 10.26 属性页的【身份验证】选项卡

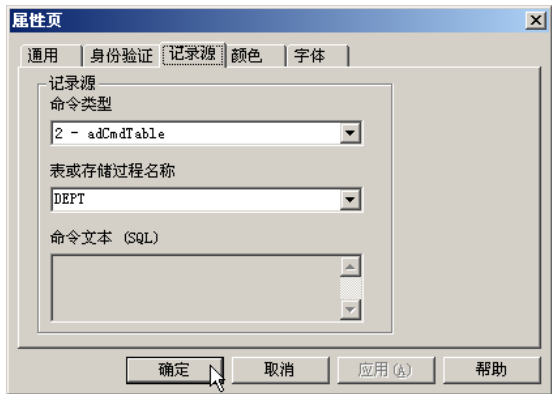


图 10.27 属性页的【记录源】选项卡

(9) 主要的属性设置如图 10.28 所示。

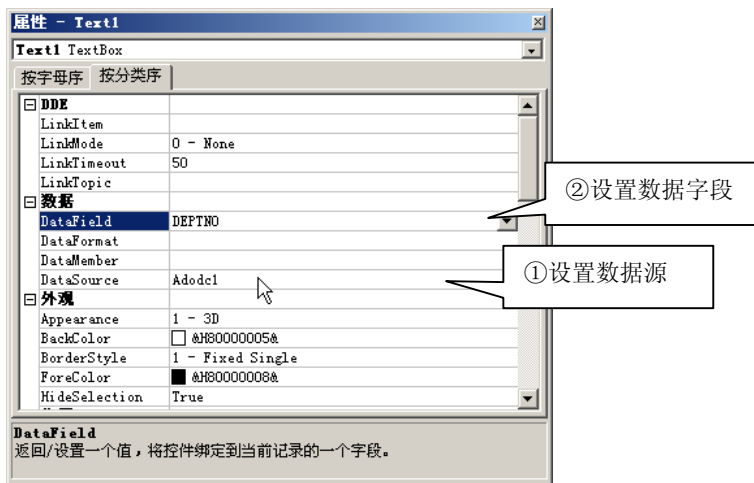


图 10.28 【TextBox】控件的【属性窗口】界面

(10) 向 Form1 窗体中添加一个名为 Text2 的【TextBox】控件。

(11) 向 Form1 窗体中添加一个名为 Text3 的【TextBox】控件。

(12) 读者可以从本书的配套光盘上找到实例的配套程序。

【参见光盘文件】：第 10 章\使用 ADODC 控件访问数据库.vbp，窗体为 Form2.frm。

(13) 利用表格来表示上述过程中窗体的设置，如表 10.2 所示。

表 10.2 Form2 窗体的设计

控件名称	类型	主要属性	属性值
Form2	【窗体】	名称	Form2
		Caption	Form2
Adodc1	【ADODC】	【通用】选项卡	使用 ODBC 数据源 graduateDB
		【身份验证】选项卡	【用户名】scott，【口令】tiger

		【记录源】选项卡	【命令类型】2-adCmdTable, 【表名】dept
Text1	【TextBox】	DataSource	Adodc1
		DataField	DEPTNO
		名称	Text1
Text2	【TextBox】	DataSource	Adodc1
		DataField	DNAME
		名称	Text2
Text3	【TextBox】	DataSource	Adodc1
		DataField	LOC
		名称	Text3

(14) 程序运行结果如图 10.29 所示。

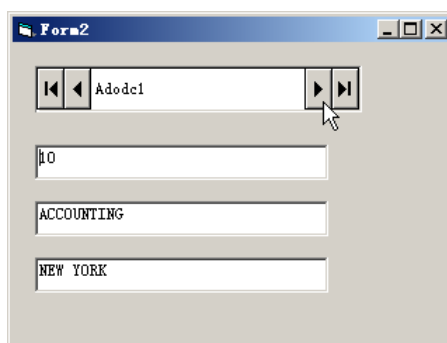


图 10.29 Form2 窗体的运行结果

### 10.3.3 通过【ADODB】编程对象访问数据库

#### 1. 引用【ADODB】对象

要使用【ADODB】对象，必须首先进行引用。

- (1) 启动 VB，新建工程。
- (2) 如图 10.30 所示。
- (3) 出现如图 10.31 所示的【引用】界面。

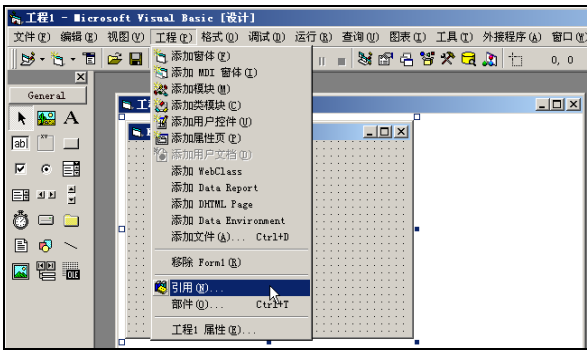


图 10.30 选择引用【ADODB】对象

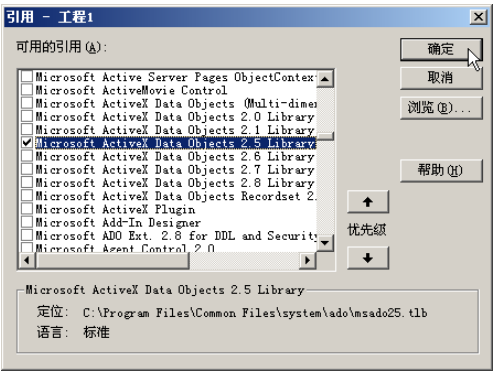


图 10.31 【引用】界面

## 2. 使用【ADODB】对象编程

(1) 设计如图 10.32 所示的窗体 Form3。



图 10.32 Form3 窗口的设计

利用表格来表示上述过程中窗体的设置，如表 10.3 所示。

表 10.3 Form3 窗体的设计

控件名称	类型	主要属性	属性值
Form3	【窗体】	名称	Form3
		Caption	使用 ADODB 访问数据库
Text1	【TextBox】	名称	Text1
Text2	【TextBox】	名称	Text2
Text3	【TextBox】	名称	Text3
CmdFirst	【CommandButton】	名称	CmdFirst
		Caption	首记录
CmdPrevious	【CommandButton】	名称	CmdPrevious
		Caption	上记录
CmdNext	【CommandButton】	名称	CmdNext

		Caption	下记录
CmdLast	【CommandButton】	名称	CmdLast
		Caption	末记录
CmdExit	【CommandButton】	名称	CmdExit
		Caption	退出

(2) 读者可以从本书的配套光盘上找到实例的配套程序。

【参见光盘文件】：第 10 章\使用 ADODB 对象编程访问数据库.vbp，窗体为 Form3.frm。

(3) 窗体 Form3 内所有的 VB 代码如下。

```

'强制性声明变量后才能使用
Option Explicit
'定义RS为ADODB对象的记录集
Dim Rs As New ADODB.Recordset
'定义conn为ADODB对象的连接
Private conn As ADODB.Connection
'窗体运行时初始化过程
Private Sub Form_Load()
    '定义ConnStr为连接字符串
    Dim ConnStr As String
    '定义Cmd为ADODB的命令字符串
    Dim Cmd As New ADODB.Command
    Set conn = New ADODB.Connection
    ConnStr = "DSN=graduateDB;uid=scott;pwd=tiger;"
    conn.ConnectionString = ConnStr
    '按照ConnStr的内容连接数据库
    conn.Open
    '查询数据表dept的内容
    With Cmd
        .ActiveConnection = conn
        .CommandType = adCmdTable
        .CommandText = "dept"
    End With
    '将数据表dept的内容送记录集RS
    With Rs
        .CursorLocation = adUseClient
        .CursorType = adOpenStatic
        .LockType = adLockPessimistic
        .Open Cmd
    End With

```

```
End With
    '定义3个文本框的数据源为RS
Set Text1.DataSource = Rs
Set Text2.DataSource = Rs
Set Text3.DataSource = Rs
    '定义3个文本框显示的字段
Text1.DataField = "DEPTNO"
Text2.DataField = "DNAME"
Text3.DataField = "LOC"
End Sub

    '单击【首记录】按钮的执行过程
Private Sub CmdFirst_Click()
    Rs.MoveFirst
    Rs.Update
End Sub

    '单击【上记录】按钮的执行过程
Private Sub CmdPrevious_Click()
    Rs.MovePrevious
    If Rs.BOF Then
        Rs.MoveFirst
        Rs.Update
    End If
End Sub

    '单击【下记录】按钮的执行过程
Private Sub CmdNext_Click()
    Rs.MoveNext
    If Rs.EOF Then
        Rs.MoveLast
        Rs.Update
    End If
End Sub

    '单击【末记录】按钮的执行过程
Private Sub CmdLast_Click()
    Rs.MoveLast
    Rs.Update
End Sub

    '单击【退出】按钮的执行过程
Private Sub CmdExit_Click()
    conn.Close
    Set conn = Nothing
```

Unload Me  
End Sub

---

(4) 窗体 Form3 的运行结果如图 10.33 所示。

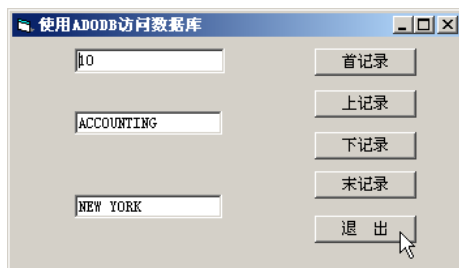


图 10.33 窗体 Form3 的运行结果

## 10.4 研究生管理信息系统的开发流程

### 10.4.1 系统需求分析

- ☐ 研究生个人信息管理
- ☐ 导师信息管理
- ☐ 专业信息管理

### 10.4.2 功能模块设计

针对上述需求，设计的功能模块如图 10.34 所示。

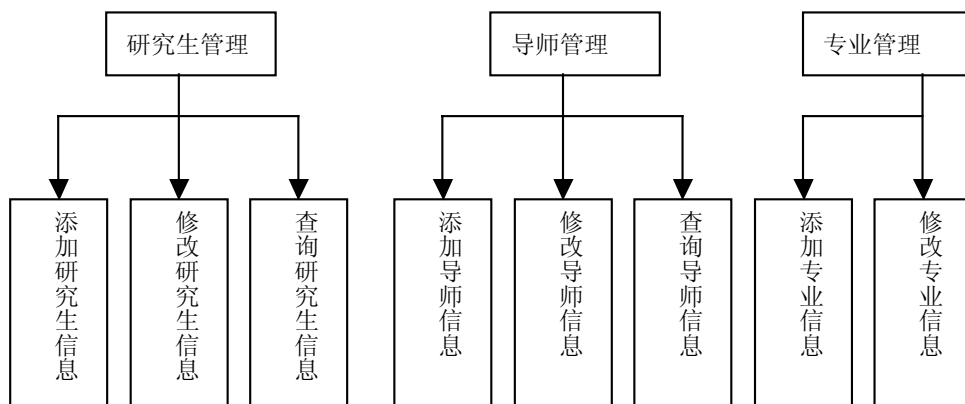


图 10.34 系统功能模块设计

### 10.4.3 数据表设计

#### 1. 研究生信息数据表

数据表名称：graduate\_info。数据表结构如表 10.4 所示。

表 10.4 graduate\_info 数据表的结构

编号	字段名称	字段类型	字段宽度	说明
0	ID	NUMBER	10	学号
1	NAME	VARCHAR2	10	姓名
2	SEX	VARCHAR2	2	性别
3	TYPE	VARCHAR2	10	类别
4	BIRTHDAY	DATE		出生日期
5	CLASSNO	NUMBER	6	班级编号
6	MAJOR	VARCHAR2	30	专业
7	DIRECTOR	VARCHAR2	10	导师姓名
8	SOURCE	VARCHAR2	50	生源性质

#### 2. 专业信息数据表

表 10.5 major\_info 数据表的结构

编号	字段名称	字段类型	字段宽度	说明
0	ID	NUMBER	2	专业编号
1	NAME	VARCHAR2	30	专业名称

#### 3. 导师信息数据表

表 10.6 director\_info 数据表的结构

编号	字段名称	字段类型	字段宽度	说明
0	ID	NUMBER	10	学号
1	NAME	VARCHAR2	10	姓名
2	SEX	VARCHAR2	2	性别
3	ZHICHENG	VARCHAR2	20	职称
4	ZHIWU	VARCHAR2	60	职务
5	MAJOR	VARCHAR2	40	专业



### 10.4.4 索引和视图设计

- 1.研究生信息数据表的索引和视图设计
- 2.专业信息表的索引和视图设计
- 3.导师信息数据表的索引和视图设计

### 10.4.5 创建数据库用户

- (1) 在如图 10.35 所示的创建用户的【一般信息】选项卡中进行如下设置。
- (2) 切换到如图 10.36 所示的创建用户的【角色】选项卡。

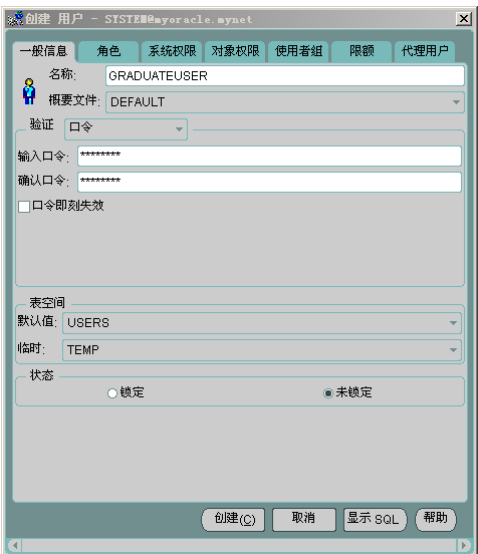


图 10.35 创建用户的【一般信息】选项卡

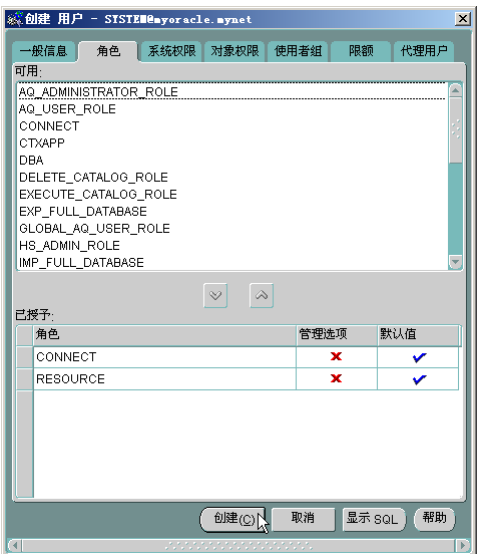


图 10.36 创建用户的【角色】选项卡

- (3) 成功创建上述用户的 SQL 代码如下。

```
CREATE USER "GRADUATEUSER" PROFILE "DEFAULT"  
  IDENTIFIED BY "12345678" DEFAULT TABLESPACE "USERS"  
  TEMPORARY TABLESPACE "TEMP"  
  ACCOUNT UNLOCK;  
  
GRANT UNLIMITED TABLESPACE TO "GRADUATEUSER";  
GRANT "CONNECT" TO "GRADUATEUSER";  
GRANT "RESOURCE" TO "GRADUATEUSER";
```

【参见光盘文件】：第 10 章\creategraduateuser.sql。

(4) 读者可以直接在【SQLPlus Worksheet】中执行 creategraduateuser.sql 文件创建用户。

10.4.6 创建数据表

1.创建研究生信息表

- (1) 在如图 10.37 所示的创建表的【一般信息】选项卡中进行如下操作。
- (2) 切换到如图 10.38 所示的创建表的【约束条件】选项卡。



图 10.37 创建表的【一般信息】选项卡

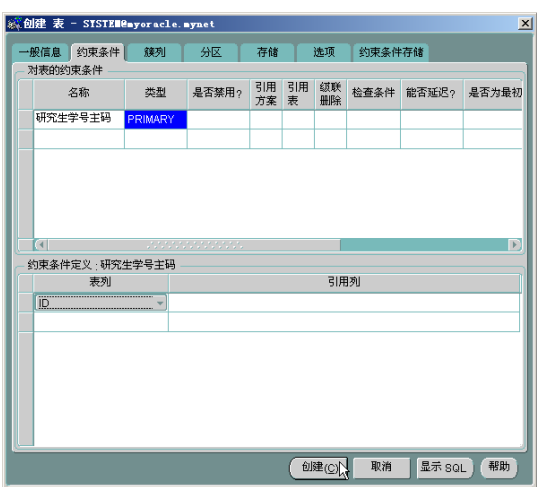


图 10.38 创建表的【约束条件】选项卡

(3) 上述创建数据表 graduateuser.graduate\_info 的 SQL 代码如下。

```
CREATE TABLE "GRADUATEUSER"."GRADUATE_INFO" (  
    "ID"          NUMBER(10)      NOT NULL,  
    "NAME"        VARCHAR2(10)    NOT NULL,  
    "SEX"         VARCHAR2(2)     NOT NULL,  
    "TYPE"        VARCHAR2(10)    NOT NULL,  
    "BIRTHDAY"    DATE             NOT NULL,  
    "CLASSNO"     NUMBER(6)        NOT NULL,  
    "MAJOR"       VARCHAR2(30)    NOT NULL,  
    "DIRECTOR"    VARCHAR2(10)    NOT NULL,  
    "SOURCE"     VARCHAR2(50)     NOT NULL,  
    CONSTRAINT "研究生学号主码" PRIMARY KEY("ID"))  
TABLESPACE "USERS"
```

【参见光盘文件】：第 10 章\createtablegraduateinfo.sql。

(4) 读者可以直接在【SQLPlus Worksheet】中执行 createtablegraduateinfo.sql 文件创建数据表 graduateuser.graduate\_info。

## 2. 创建专业信息表

- (1) 在如图 10.39 所示的创建表的【一般信息】选项卡中按照如图所示进行设置。
- (2) 切换到如图 10.40 所示的创建表的【约束条件】选项卡。



图 10.39 创建表的【一般信息】选项卡

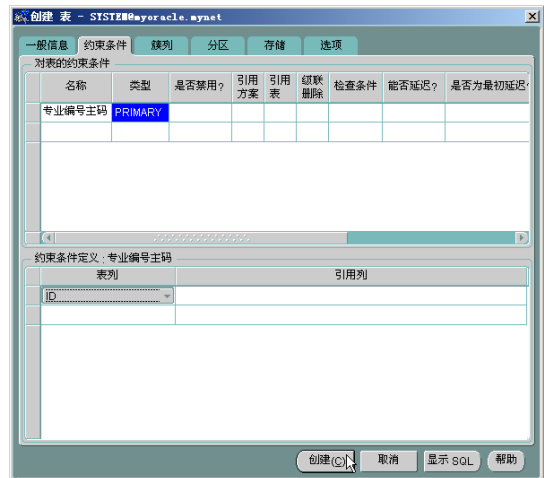


图 10.40 创建表的【约束条件】选项卡

- (3) 上述创建数据表 graduateuser.major\_info 的 SQL 代码如下。

```
CREATE TABLE "GRADUATEUSER"."MAJOR_INFO" (
    "ID"          NUMBER(2)      NOT NULL,
    "NAME"        VARCHAR2(30)  NOT NULL,
    CONSTRAINT "专业编号主码" PRIMARY KEY("ID"))
TABLESPACE "USERS"
```

【参见光盘文件】：第 10 章\createtablemajorinfo.sql。

- (4) 读者可以直接在【SQLPlus Worksheet】中执行 createtablemajorinfo.sql 文件创建数据表 graduateuser.major\_info。

## 3. 创建导师信息表

- (1) 在如图 10.41 所示的创建表的【一般信息】选项卡中按照如图所示进行设置。
- (2) 切换到如图 10.42 所示的创建表的【约束条件】选项卡。



图 10.41 创建表的【一般信息】选项卡

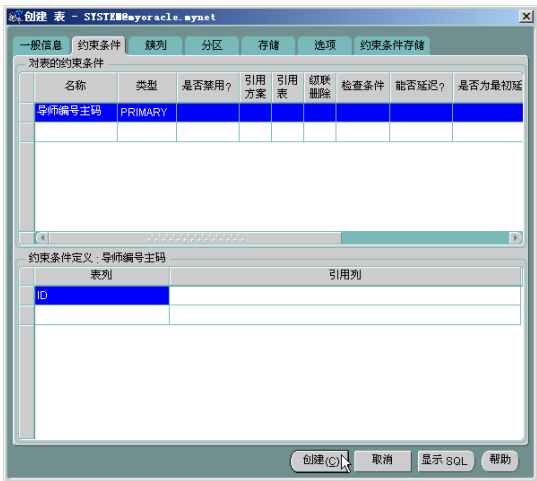


图 10.42 创建表的【约束条件】选项卡

(3) 上述创建数据表 graduateuser.director\_info 的 SQL 代码如下。

```
CREATE TABLE "GRADUATEUSER"."DIRECTOR_INFO" (  
    "ID"          NUMBER(10)    NOT NULL,  
    "NAME"        VARCHAR2(10)  NOT NULL,  
    "SEX"         VARCHAR2(2)   NOT NULL,  
    "ZHICHENG"    VARCHAR2(20)  NOT NULL,  
    "ZHIWU"       VARCHAR2(20)  NOT NULL,  
    "MAJOR"       VARCHAR2(40)  NOT NULL,  
    CONSTRAINT "导师编号主码" PRIMARY KEY("ID"))  
TABLESPACE "USERS"
```

【参见光盘文件】：第 10 章\createtabledirectorinfo.sql。

(4) 读者可以直接在【SQLPlus Worksheet】中执行 createtabledirectorinfo.sql 文件创建数据表 graduateuserdirector\_info。

10.4.7 创建索引

1.为研究生信息表的 NAME 字段建立索引

- (1) 在如图 10.43 所示的创建索引的【一般信息】选项卡中按照图示进行操作。
- (2) 其他选项卡按照默认设置即可。



图 10.43 创建索引的【一般信息】选项卡

(3) 上述创建索引的 SQL 代码如下。

---

```
CREATE INDEX "GRADUATEUSER"."研究生姓名字段索引"
ON "GRADUATEUSER"."GRADUATE_INFO"("NAME")
TABLESPACE "INDX"
```

---

【参见光盘文件】：第 10 章\createindexgraduate.sql。

(4) 读者可以直接在【SQLPlus Worksheet】中执行 createindexgraduate.sql 文件创建数据表 graduateuser.graduate\_info 的索引“研究生姓名字段索引”。

## 2. 为导师信息表的 NAME 字段建立索引

(1) 在如图 10.44 所示的创建索引的【一般信息】选项卡中按照图示进行操作。

(2) 其他选项卡按照默认设置即可。

(3) 上述创建索引的 SQL 代码如下。

---

```
CREATE INDEX "GRADUATEUSER"."导师姓名字段索引"
ON "GRADUATEUSER"."DIRECTOR_INFO"("NAME")
TABLESPACE "INDX"
```

---

【参见光盘文件】：第 10 章\createindexdirector.sql。

(4) 读者可以直接在【SQLPlus Worksheet】中执行 createindexdirector.sql 文件创建数据表 graduateuser.director\_info 的索引“导师姓名字段索引”。



图 10.44 创建索引的【一般信息】选项卡

### 10.4.8 创建视图

#### 1.为研究生信息表建立视图

(1) 在如图 10.45 所示的创建视图的【一般信息】选项卡中按照图示进行操作。

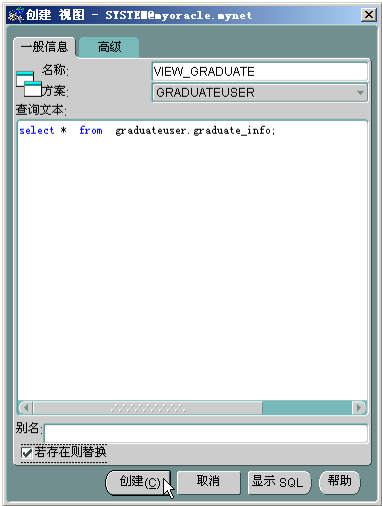


图 10.45 创建视图的【一般信息】选项卡

- (2) 其他选项卡按照默认设置即可。
- (3) 上述创建视图的 SQL 代码如下。

CREATE OR REPLACE VIEW "GRADUATEUSER"."VIEW\_GRADUATE" AS

```
select *
from graduateuser.graduate_info WITH READ ONLY
```

【参见光盘文件】：第 10 章\createviewgraduate.sql。

(4) 读者可以直接在【SQLPlus Worksheet】中执行 createviewgraduate.sql 文件创建数据表 graduateuser.graduate\_info 的视图“VIEW\_GRADUATE”。

## 2. 为专业信息表建立视图

(1) 在如图 10.46 所示的创建视图的【一般信息】选项卡中按照图示进行操作。

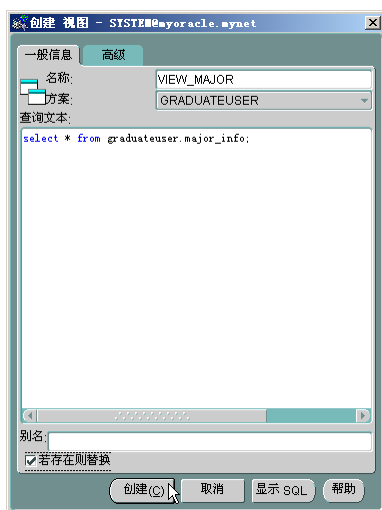


图 10.46 创建视图的【一般信息】选项卡

(2) 其他选项卡按照默认设置即可。

(3) 上述创建视图的 SQL 代码如下。

```
CREATE OR REPLACE VIEW "GRADUATEUSER"."VIEW_MAJOR" AS
select *
from graduateuser.major_info WITH READ ONLY
```

【参见光盘文件】：第 10 章\createviewmajor.sql。

(4) 读者可以直接在【SQLPlus Worksheet】中执行 createviewmajor.sql 文件创建数据表 graduateuser.major\_info 的视图“VIEW\_MAJOR”。

## 2. 为导师信息表建立视图

(1) 在如图 10.47 所示的创建视图的【一般信息】选项卡中按照图示进行操作。

(2) 其他选项卡按照默认设置即可。

(3) 上述创建视图的 SQL 代码如下。

```
CREATE OR REPLACE VIEW "GRADUATEUSER"."VIEW_DIRECTOR" AS
select *
from graduateuser.director_info WITH READ ONLY
```

【参见光盘文件】：第 10 章\createviewdirector.sql。

(4) 读者可以直接在【SQLPlus Worksheet】中执行 createviewdirector.sql 文件创建数据表 graduateuser.director\_info 的视图“VIEW\_DIRECTOR”。

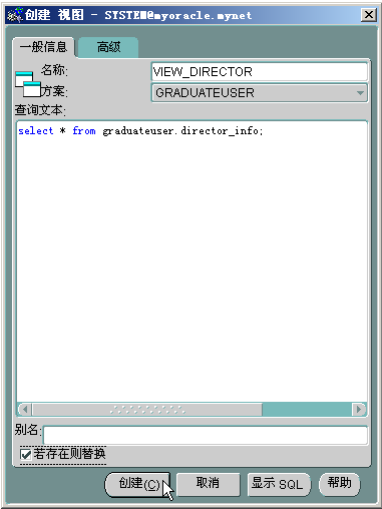


图 10.47 创建视图的【一般信息】选项卡

10.4.9 VB 工程总体框架

开发完成的研究生管理信息系统的总体框架如表 10.7 所示。

【参见光盘文件】：第 10 章\研究生管理信息系统.vbp。

表 10.7 研究生管理信息系统工程各成分

成分名称	成分类型	作用
Frmmain	窗体	主界面的主菜单调用各子窗体完成各项功能
FormInsertDirector	窗体	添加导师信息窗体
FormInsertGraduate	窗体	添加研究生信息窗体
FormInsertMajor	窗体	添加专业信息窗体
FormSelectDirector	窗体	查询导师信息窗体
FormSelectGraduate	窗体	查询研究生信息窗体
FormUpdateGraduate	窗体	修改研究生信息窗体



FormUpdateDirector	窗体	修改导师信息窗体
FormUpdateMajor	窗体	修改专业信息窗体
Module1	模块	全局性的公用函数

10.4.10 系统是如何运行的

- (1) 启动 VB，打开配套光盘的工程文件。
- (2) 如图 10.48 所示。



图 10.48 查看工程的属性

- (3) 出现如图 10.49 所示的工程属性的【通用】选项卡。
- (4) 打开模块 Module1，查看名为 Main 的过程的代码如下。说明 Main 过程将启动名为 frmMain 的主窗体。

```
Public fMainForm As frmMain
Sub Main()
    Set fMainForm = New frmMain
    fMainForm.Show
End Sub
```

显示主窗体

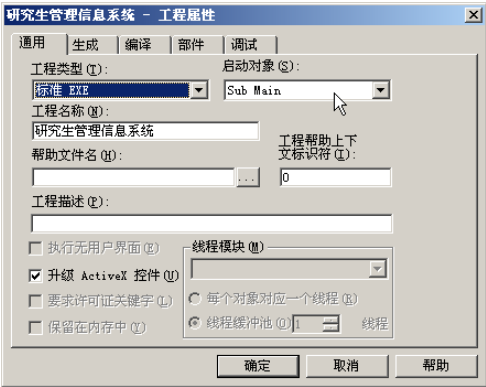


图 10.49 工程属性的【通用】选项卡

(5) 编辑名为 frmMain 的主窗体，如图 10.50 所示。

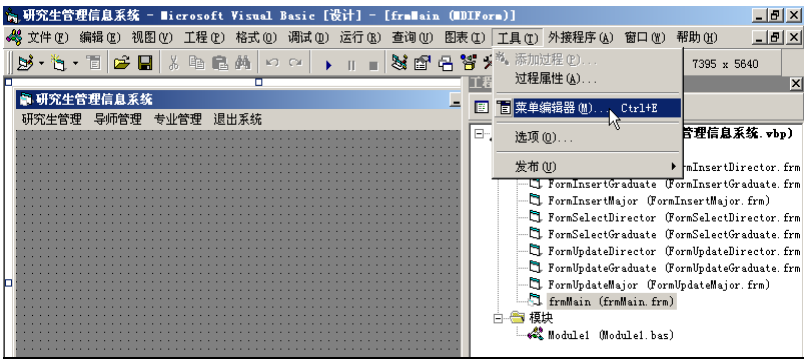


图 10.50 选择查看主窗体的菜单设计

(6) 出现如图 10.51 所示的【菜单编辑器】界面，可以查看主窗体的菜单设计。

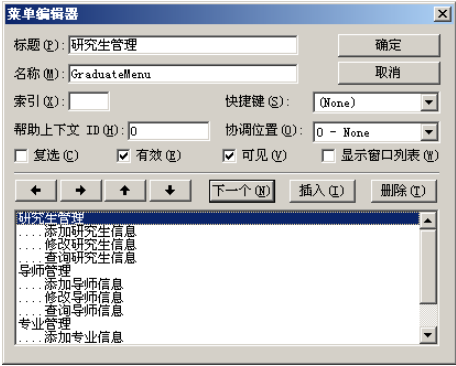


图 10.51 【菜单编辑器】界面

工程中设计的菜单如表 10.8 所示。

表 10.8 主窗体设计的菜单

菜单名称	菜单分级	使用的变量	执行的操作
------	------	-------	-------

研究生管理	1	GraduateMenu	
添加研究生信息	2	GraduateInsertMenu	调用窗体 FormInsertGraduate
修改研究生信息	2	GraduateUpdateMenu	调用窗体 FormUpdateGraduate
查询研究生信息	2	GraduateSelectMenu	调用窗体 FormSelectGraduate
导师管理	1	DirectorMenu	
添加导师信息	2	DirectorInsertMenu	调用窗体 FormInsertDirector
修改导师信息	2	DirectorUpdateMenu	调用窗体 FormUpdateDirector
查询导师信息	2	DirectorSelectMenu	调用窗体 FormSelectDirector
专业管理	1	MajorMenu	
添加专业信息	2	MajorInsertMenu	调用窗体 FormInsertMajor
修改专业信息	2	MajorUpdateMenu	调用窗体 FormUpdateMajor
退出系统	1	ExitMenu	退出主窗体

(7) 为什么这些菜单可以调用这些子窗体或者执行退出主窗体的操作呢？在图 10.50 的主窗体中用鼠标双击，在出现的代码窗口中的主要代码如下。

```

-----
‘选择菜单项DirectorInsertMenu后将显示子窗体FormInsertDirector
Private Sub DirectorInsertMenu_Click()
    FormInsertDirector.Show
End Sub
‘选择菜单项DirectorSelectMenu 后将显示子窗体FormSelectDirector
Private Sub DirectorSelectMenu_Click()
    FormSelectDirector.Show
End Sub
‘选择菜单项DirectorUpdateMenu后将显示子窗体FormUpdateDirector
Private Sub DirectorUpdateMenu_Click()
    FormUpdateDirector.Show
End Sub
‘选择菜单项GraduateInsertMenu后将显示子窗体FormInsertGraduate
Private Sub GraduateInsertMenu_Click()
    FormInsertGraduate.Show
End Sub
‘选择菜单项GraduateSelectMenu后将显示子窗体FormSelectGraduate
Private Sub GraduateSelectMenu_Click()
    FormSelectGraduate.Show
End Sub
‘选择菜单项GraduateUpdateMenu后将显示子窗体FormUpdateGraduate
Private Sub GraduateUpdateMenu_Click()

```

```
FormUpdateGraduate.Show
End Sub
‘选择菜单项MajorInsertMenu后将显示子窗体FormInsertMajor
Private Sub MajorInsertMenu_Click()
    FormInsertMajor.Show
End Sub
‘选择菜单项MajorUpdateMenu后将显示子窗体FormUpdateMajor
Private Sub MajorUpdateMenu_Click()
    FormUpdateMajor.Show
End Sub
‘选择菜单项ExitMenu后将显示退出主窗体
Private Sub ExitMenu_Click()
    End
End Sub
```

-----

#### 10.4.11 模块的设计

模块 **Module1** 的其他主要代码如下。

-----

```
‘定义了名为ConnectString的无参数函数，连接数据库
Public Function ConnectString() As String
    ConnectString = "DSN=graduateDB;UID=graduateuser;PWD=12345678"
End Function
‘定义了名为ExecuteSQL的带参数函数，参数SQL和MsgString都为字符串型。该函数用于执行SQL语句。
Public Function ExecuteSQL(ByVal SQL As String, MsgString As String) As ADODB.Recordset
    ‘conn为ADODB的connection连接对象
    Dim conn As ADODB.Connection
    ‘rest为ADODB的记录集对象
    Dim rest As ADODB.Recordset
    ‘若执行SQL错误转向ExecuteSQL_Error标记处
    On Error GoTo ExecuteSQL_Error
    Set conn = New ADODB.Connection
    ‘连接数据库
    conn.Open ConnectString
    Set rest = New ADODB.Recordset
    ‘执行SQL字符串语句
    rest.Open Trim$(SQL), conn, adOpenKeyset, adLockOptimistic
    Set ExecuteSQL = rest
    MsgString = "查询到" & rest.RecordCount & " 条记录 "
```

‘执行SQL完毕后关闭连接和记录集

ExecuteSQL\_Exit:

Set rest = Nothing

Set conn = Nothing

Exit Function

‘执行SQL错误时的处理方法

ExecuteSQL\_Error:

MsgString = "查询错误: " & \_

Err.Description

Resume ExecuteSQL\_Exit

End Function

‘定义了名为Checktxt的带参数函数，参数txt为字符串型。该函数用于检测txt字符串是否有内容。

Public Function Checktxt(txt As String) As Boolean

If Trim(txt) = "" Then

Checktxt = False

Else

Checktxt = True

End If

End Function

## 10.4.12 各子窗体的设计

### 1. 添加信息窗体的设计

打开名为 FormInsertGraduate 的子窗体，如图 10.52 所示。

窗体的设计如表 10.8 所示。

表 10.8 子窗体 FormInsertGraduate 的设计

图 10.52 子窗体 FormInsertGraduate

控件名称	类型	主要属性	属性值
FormInsertGraduate	【窗体】	名称	FormInsertGraduate
		Caption	添加研究生信息
		StartPosition	2-屏幕中心

Label1	【Label】	Caption	学号
Label2	【Label】	Caption	姓名
Label3	【Label】	Caption	性别
Label4	【Label】	Caption	类别
Label5	【Label】	Caption	(例如 2002-09-10)
Label6	【Label】	Caption	班级编号
Label7	【Label】	Caption	专业
Label8	【Label】	Caption	导师姓名
Label9	【Label】	Caption	生源性质
Label10	【Label】	Caption	出生日期
CmdCommit	【CommandButton】	名称	CmdCommit
		Caption	确定添加
CmdCancel	【CommandButton】	名称	CmdCancel
		Caption	取消添加
txtID	【TextBox】	名称	TxtID(用于向 graduate_info 数据表的 ID 字段输入数据)
TxtName	【TextBox】	名称	TxtName (用于向 graduate_info 数据表的 NAME 字段输入数据)
txtBirthday	【TextBox】	名称	TxtBirthday (用于向 graduate_info 数据表的 BIRTHDAY 字段输入数据)
TxtClassno	【TextBox】	名称	TxtClassno(用于向 graduate_info 数据表的 CLASSNO 字段输入数据)
TxtSource	【TextBox】	名称	TxtSource (用于向 graduate_info 数据表的 SOURCE 字段输入数据)
		Scrollbars	2-Vertical
ComboSex	【ComboBox】	名称	ComboSex (用于向 graduate_info 数据表的 SEX 字段输入数据)
		Style	0-DropDown Combo
		Text	请选择性别
ComboType	【ComboBox】	名称	ComboType (用于向 graduate_info 数据表的 TYPE 字段输入数据)
		Style	0-DropDown Combo
		Text	请选择类别
ComboMajor	【ComboBox】	名称	ComboMajor (用于向 graduate_info 数据表的 MAJOR 字段输入数据)

CombDirector	【ComboBox】	Style	0-DropDown Combo
		Text	请选择专业
		名称	CombDirector（用于向 graduate_info 数据表的 DIRECTOR 字段输入数据）
		Style	0-DropDown Combo
		Text	请选择导师姓名

窗体的程序代码分析如下。

‘窗体启动时的执行代码

```
Private Sub Form_Load()
```

```
    ‘定义窗体要使用的对象和变量
```

```
    Dim TempRecordset As ADODB.Recordset
```

```
    Dim txtSQL As String
```

```
    Dim MsgText As String
```

```
    Dim i As Integer
```

```
    ‘向【性别】下拉列表框中添加两个选项
```

```
    comboSex.AddItem "男"
```

```
    comboSex.AddItem "女"
```

```
    ‘向【类别】下拉列表框中添加3个选项
```

```
    ComboType.AddItem "硕士"
```

```
    ComboType.AddItem "博士"
```

```
    ComboType.AddItem "博士后"
```

```
    ‘从导师信息表的视图view_director中取出导师姓名，添加到【导师姓名】下拉列表框中供选择
```

```
    txtSQL = "select name from view_director"
```

```
    Set TempRecordset = ExecuteSQL(txtSQL, MsgText)
```

```
    For i = 1 To TempRecordset.RecordCount
```

```
        comboDirector.AddItem TempRecordset.Fields(0)
```

```
        TempRecordset.MoveNext
```

```
    Next i
```

```
    TempRecordset.Close
```

```
    ‘从专业信息表的视图view_major中取出专业姓名，添加到【专业】下拉列表框中供选择
```

```
    txtSQL = "select name from view_major"
```

```
    Set TempRecordset = ExecuteSQL(txtSQL, MsgText)
```

```
    For i = 1 To TempRecordset.RecordCount
```

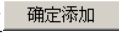
```
        ComboMajor.AddItem TempRecordset.Fields(0)
```

```
        TempRecordset.MoveNext
```

```
    Next i
```

```
    TempRecordset.Close
```

```
End Sub
```

‘单击按钮  后的执行代码

```
Private Sub CmdCommit_Click()  
    Dim TempRecordset As ADODB.Recordset  
    Dim txtSQL As String  
    Dim MsgText As String  
    ‘调用全局模块中的函数Checktxt检查【姓名】文本框中是否输入了内容  
    If Not Checktxt(txtID.Text) Then  
        MsgBox "请输入学号!", vbOKOnly + vbExclamation, "警告"  
        txtID.SetFocus  
        Exit Sub  
    End If  
    If Not Checktxt(txtName.Text) Then  
        MsgBox "请输入姓名!", vbOKOnly + vbExclamation, "警告"  
        txtName.SetFocus  
        Exit Sub  
    End If  
    ‘调用全局模块中的函数Checktxt检查【性别】下拉列表框中是否输入了内容  
    If Not Checktxt(comboSex.Text) Then  
        MsgBox "请选择性别!", vbOKOnly + vbExclamation, "警告"  
        comboSex.SetFocus  
        Exit Sub  
    End If  
    ‘调用全局模块中的函数Checktxt检查【类别】下拉列表框中是否输入了内容  
    If Not Checktxt(ComboType.Text) Then  
        MsgBox "请输入研究生类别!", vbOKOnly + vbExclamation, "警告"  
        ComboType.SetFocus  
        Exit Sub  
    End If  
    ‘调用全局模块中的函数Checktxt检查【出生日期】文本框中是否输入了内容  
    If Not Checktxt(txtBirthday.Text) Then  
        MsgBox "请输入出生日期!", vbOKOnly + vbExclamation, "警告"  
        txtBirthday.SetFocus  
        Exit Sub  
    End If  
    ‘调用全局模块中的函数Checktxt检查【班级编号】文本框中是否输入了内容  
    If Not Checktxt(txtClassno.Text) Then  
        MsgBox "请输入班级编号!", vbOKOnly + vbExclamation, "警告"  
        txtClassno.SetFocus  
        Exit Sub  
    End If
```



‘调用全局模块中的函数Checktxt检查【专业】下拉列表框中是否输入了内容

```
If Not Checktxt(ComboMajor.Text) Then
    MsgBox "请输入专业!", vbOKOnly + vbExclamation, "警告"
    ComboMajor.SetFocus
Exit Sub
```

End If

‘调用全局模块中的函数Checktxt检查【导师姓名】下拉列表框中是否输入了内容

```
If Not Checktxt(comboDirector.Text) Then
    MsgBox "请输入导师姓名!", vbOKOnly + vbExclamation, "警告"
    comboDirector.SetFocus
Exit Sub
```

End If

‘调用VB系统函数IsNumeric检查【学号】文本框中是否输入的数字学号

```
If Not IsNumeric(Trim(txtID.Text)) Then
    MsgBox "请输入数字学号!", vbOKOnly + vbExclamation, "警告"
Exit Sub
txtID.SetFocus
```

End If

‘上述输入信息无误后，查询研究生信息表中是否有重复学号的记录

```
txtSQL = "select * from graduate_Info where ID = '" & Trim(txtID.Text) & "'"
Set TempRecordset = ExecuteSQL(txtSQL, MsgText)
```

‘若发现研究生信息表中有重复学号的记录，提示重新输入数据

```
If TempRecordset.EOF = False Then
    MsgBox "学号重复，请重新输入!", vbOKOnly + vbExclamation, "警告"
    TempRecordset.Close
    txtID.SetFocus
```

Else

```
TempRecordset.Close
```

‘若没有发现研究生信息表中有重复学号的记录，检查【出生日期】文本框中输入的是否是日期型的数据

```
If Not IsDate(txtBirthday.Text) Then
    MsgBox "出生日期应输入日期格式 (yyyy-mm-dd)!", vbOKOnly + vbExclamation, "警告"
    txtBirthday.SetFocus
Else
    txtBirthday = Format(txtBirthday, "yyyy-mm-dd")
    If Not IsDate(txtBirthday.Text) Then
        MsgBox "出生日期应输入日期格式 (yyyy-mm-dd)!", vbOKOnly + vbExclamation, "警告"
        txtBirthday.SetFocus
    Else
        txtBirthday = Format(txtBirthday, "yyyy-mm-dd")
```

```
txtSQL = "select * from graduate_Info"
```

```
‘执行查询语句
```

```
Set TempRecordset = ExecuteSQL(txtSQL, MsgText)
```

‘给记录集赋值，每个数据表的字段对应了窗体中的每个输入数据，注意添加时要和数据表中的字段顺序一致。

```
TempRecordset.AddNew
```

```
TempRecordset.Fields(0) = Trim(txtID.Text)
```

```
TempRecordset.Fields(1) = Trim(txtName.Text)
```

```
TempRecordset.Fields(2) = Trim(comboSex.Text)
```

```
TempRecordset.Fields(3) = Trim(ComboType.Text)
```

```
TempRecordset.Fields(4) = Trim(txtBirthday.Text)
```

```
TempRecordset.Fields(5) = Trim(txtClassno.Text)
```

```
TempRecordset.Fields(6) = Trim(ComboMajor.Text)
```

```
TempRecordset.Fields(7) = Trim(comboDirector.Text)
```

```
TempRecordset.Fields(8) = Trim(txtSource.Text)
```

```
‘更新记录集，数据被插入到数据表中
```

```
TempRecordset.Update
```

```
MsgBox "添加研究生信息成功！", vbOKOnly + vbExclamation, "警告"
```

```
TempRecordset.Close
```

```
Unload Me
```

```
End If
```

```
End If
```

```
End Sub
```

```
End Sub
```

‘单击按钮  后的执行代码

```
Private Sub CmdCancel_Click()
```

```
‘释放窗体，返回到调用的主窗体
```

```
Unload Me
```

```
End Sub
```

## 2.修改信息窗体的设计

打开名为 FormUpdateGraduate 的子窗体，如图 10.53 所示。

图 10.53 子窗体 FormUpdateGraduate

窗体的设计如表 10.10 所示。

表 10.10 子窗体 FormUpdateGraduate 的设计

控件名称	类型	主要属性	属性值
FormUpdateGraduate	【窗体】	名称	FormUpdateGraduate
		Caption	修改研究生信息
		StartPosition	2-屏幕中心
Label1	【Label】	Caption	学号
Label2	【Label】	Caption	姓名
Label3	【Label】	Caption	性别
Label4	【Label】	Caption	类别
Label5	【Label】	Caption	班级编号
Label6	【Label】	Caption	专业
Label7	【Label】	Caption	导师姓名
Label8	【Label】	Caption	生源性质
Label9	【Label】	Caption	出生日期
TxtID	【TextBox】	名称	TxtID（无法修改，惟一编号）
TxtName	【TextBox】	名称	TxtName（NAME 字段）
TxtSource	【TextBox】	名称	TxtSource（SOURCE 字段）
		Scrollbars	2-Vertical
TxtBirthday	【TextBox】	名称	TxtBirthday（BIRTHDAY 字段）
TxtClassno	【TextBox】	名称	TxtClassno（CLASSNO 字段）
ComboSex	【ComboBox】	名称	ComboSex（SEX 字段）

		Style	0-DropDown Combo
ComboType	【ComboBox】	名称	ComboType (TYPE 字段)
		Style	0-DropDown Combo
ComboMajor	【ComboBox】	名称	ComboMajor (MAJOR 字段)
		Style	0-DropDown Combo
CombDirector	【ComboBox】	名称	ComboDirector (DIRECTOR 字段)
		Style	0-DropDown Combo
Frame1	【Frame】	名称	Frame1
Frame2	【Frame】	名称	Frame2
CmdFirst	【CommandButton】	名称	CmdFirst
		Caption	首记录
CmdPrevious	【CommandButton】	名称	CmdPrevious
		Caption	上记录
CmdNext	【CommandButton】	名称	CmdNext
		Caption	下记录
CmdFirst	【CommandButton】	名称	CmdFirst
		Caption	首记录
CmdLast	【CommandButton】	名称	CmdLast
		Caption	末记录
CmdEdit	【CommandButton】	名称	CmdEdit
		Caption	修改记录
CmdUpdate	【CommandButton】	名称	CmdUpdate
		Caption	更新数据
CmdCancel	【CommandButton】	名称	CmdCancel
		Caption	取消修改
CmdDelete	【CommandButton】	名称	CmdDelete
		Caption	删除记录
CmdExit	【CommandButton】	名称	CmdExit
		Caption	退出修改


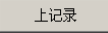

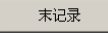
窗体的程序代码分析如下。

‘定义窗体级的变量，这样各模块可以使用

```
Dim TempRecordset As ADODB.Recordset
```

```
Dim TempBookmark As Variant
```

```


Dim Tempclean As Boolean
'启动窗体时执行的代码
Private Sub Form_Load()
    Dim txtSQL As String
    Dim MsgText As String
    '按照ID为顺序，从数据表中获取数据
    txtSQL = "select * from graduate_Info order by id"
    Set TempRecordset = ExecuteSQL(txtSQL, MsgText)
    '调用过程DisplayData，显示第一条记录
    TempRecordset.MoveFirst
    Call DisplayData
    mcbookmark = TempRecordset.Bookmark
    Tempclean = True
End Sub
'单击按钮  后的执行代码
Private Sub CmdFirst_Click()
    TempRecordset.MoveFirst
    Call DisplayData
End Sub
'单击按钮  后的执行代码
Private Sub CmdPrevious_Click()
    TempRecordset.MovePrevious
    If TempRecordset.BOF Then
        TempRecordset.MoveLast
    End If
    Call DisplayData
End Sub
'单击按钮  后的执行代码
Private Sub CmdNext_Click()
    TempRecordset.MoveNext
    If TempRecordset.EOF Then
        TempRecordset.MoveFirst
    End If
    Call DisplayData
End Sub
'单击按钮  后的执行代码
Private Sub CmdLast_Click()
    TempRecordset.MoveLast
    Call DisplayData
End Sub

```

‘单击按钮  后的执行代码

```
Private Sub CmdEdit_Click()  
    Tempclean = False  
    Frame1.Enabled = False  
    CmdFirst.Enabled = False  
    CmdPrevious.Enabled = False  
    CmdNext.Enabled = False  
    CmdLast.Enabled = False  
    txtID.Enabled = False  
    ‘ 激活其他编辑框  
    txtName.Enabled = True  
    comboSex.Enabled = True  
    comboType.Enabled = True  
    txtBirthday.Enabled = True  
    txtClassno.Enabled = True  
    ComboMajor.Enabled = True  
    comboDirector.Enabled = True  
    txtSource.Enabled = True  
    TempBookmark = TempRecordset.Bookmark
```

End Sub

‘单击按钮  后的执行代码

```
Private Sub CmdUpdate_Click()  
    Dim txtSQL As String  
    Dim MsgText As String  
    Dim TempRecordsetc As ADODB.Recordset  
    If Tempclean Then  
        MsgBox "请先选择修改研究生信息", vbOKOnly + vbExclamation, "警告"  
        Exit Sub  
    End If  
    If Not Checktxt(txtID.Text) Then  
        MsgBox "请输入学号!", vbOKOnly + vbExclamation, "警告"  
        txtID.SetFocus  
        Exit Sub  
    End If  
    If Not IsNumeric(Trim(txtID.Text)) Then  
        MsgBox "学号请输入数字!", vbOKOnly + vbExclamation, "警告"  
        txtID.SetFocus  
        Exit Sub  
    End If
```

```
If Not Checktxt(txtName.Text) Then
    MsgBox "请输入姓名!", vbOKOnly + vbExclamation, "警告"
    txtName.SetFocus
Exit Sub
End If

If Not Checktxt(comboSex.Text) Then
    MsgBox "请选择性别!", vbOKOnly + vbExclamation, "警告"
    comboSex.SetFocus
Exit Sub
End If

If Not Checktxt(comboType.Text) Then
    MsgBox "请选择类别!", vbOKOnly + vbExclamation, "警告"
    comboType.SetFocus
Exit Sub
End If

If Not Checktxt(txtBirthday.Text) Then
    MsgBox "请输入出生日期!", vbOKOnly + vbExclamation, "警告"
    txtBirthday.SetFocus
Exit Sub
End If

If Not Checktxt(txtClassno.Text) Then
    MsgBox "请选择班号!", vbOKOnly + vbExclamation, "警告"
    txtClassno.SetFocus
Exit Sub
End If

If Not Checktxt(ComboMajor.Text) Then
    MsgBox "请输入专业!", vbOKOnly + vbExclamation, "警告"
    ComboMajor.SetFocus
Exit Sub
End If

If Not Checktxt(comboDirector.Text) Then
    MsgBox "请输入导师姓名!", vbOKOnly + vbExclamation, "警告"
    comboDirector.SetFocus
Exit Sub
End If

If Not Checktxt(txtSource.Text) Then
    MsgBox "请输入生源性质!", vbOKOnly + vbExclamation, "警告"
    txtSource.SetFocus
Exit Sub
End If
```



```

TempRecordset.Delete
txtSQL = "select * from graduate_Info where ID = '" & Trim(txtID.Text) & "'"
Set TempRecordsetc = ExecuteSQL(txtSQL, MsgText)
If TempRecordsetc.EOF = False Then
    MsgBox "学号重复, 请重新输入!", vbOKOnly + vbExclamation, "警告"
    TempRecordsetc.Close
    txtID.SetFocus
Else
    TempRecordsetc.Close
    If Not IsDate(txtBirthday.Text) Then
        MsgBox "出生时间应输入日期格式 (yyyy-mm-dd)!", vbOKOnly + vbExclamation, "警告"
        txtBirthday.SetFocus
    Else
        txtBirthday = Format(txtBirthday, "yyyy-mm-dd")
        If Not IsDate(txtBirthday.Text) Then
            MsgBox "出生时间应输入日期格式 (yyyy-mm-dd)!", vbOKOnly + vbExclamation, "警告"
            txtBirthday.SetFocus
        Else
            txtBirthday = Format(txtBirthday, "yyyy-mm-dd")
            TempRecordset.AddNew
            TempRecordset.Fields(0) = Trim(txtID.Text)
            TempRecordset.Fields(1) = Trim(txtName.Text)
            TempRecordset.Fields(2) = Trim(comboSex.Text)
            TempRecordset.Fields(3) = Trim(comboType.Text)
            TempRecordset.Fields(4) = Trim(txtBirthday.Text)
            TempRecordset.Fields(5) = Trim(txtClassno.Text)
            TempRecordset.Fields(6) = Trim(ComboMajor.Text)
            TempRecordset.Fields(7) = Trim(comboDirector.Text)
            TempRecordset.Fields(8) = Trim(txtSource.Text)
            TempRecordset.Update
            MsgBox "修改研究生信息成功!", vbOKOnly + vbExclamation, "修改研究生信息"
            Call DisplayData
            Frame1.Enabled = True
            CmdFirst.Enabled = True
            CmdPrevious.Enabled = True
            CmdNext.Enabled = True
            CmdLast.Enabled = True
            txtID.Enabled = False
            txtName.Enabled = False
            comboSex.Enabled = False

```




```

        comboType.Enabled = False
        txtBirthday.Enabled = False
        txtClassno.Enabled = False
        ComboMajor.Enabled = False
        comboDirector.Enabled = False
        txtSource.Enabled = False
        Tempclean = True
    End If
End If
End If
End Sub
‘单击按钮  后的执行代码
Private Sub CmdCancel_Click()
    If Not Tempclean Then
        Frame1.Enabled = True
        CmdFirst.Enabled = True
        CmdPrevious.Enabled = True
        CmdNext.Enabled = True
        CmdLast.Enabled = True
        txtID.Enabled = False
        txtName.Enabled = False
        comboSex.Enabled = False
        comboType.Enabled = False
        txtBirthday.Enabled = False
        txtClassno.Enabled = False
        ComboMajor.Enabled = False
        comboDirector.Enabled = False
        txtSource.Enabled = False
        TempRecordset.Bookmark = TempBookmark
        Call DisplayData
    Else
        MsgBox "没有进行任何修改!", vbOKOnly + vbExclamation, "警告"
    End If
End Sub
‘单击按钮  后的执行代码
Private Sub CmdDelete_Click()
    TempBookmark = TempRecordset.Bookmark
    str2$ = MsgBox("是否删除当前记录?", vbOKCancel, "删除当前记录")
    If str2$ = vbOK Then
        TempRecordset.MoveNext
    End If
End Sub

```

```

        If TempRecordset.EOF Then
            TempRecordset.MoveFirst
            TempBookmark = TempRecordset.Bookmark
            TempRecordset.MoveLast
            TempRecordset.Delete
            TempRecordset.Bookmark = TempBookmark
            Call DisplayData
        Else
            TempBookmark = TempRecordset.Bookmark
            TempRecordset.MovePrevious
            TempRecordset.Delete
            TempRecordset.Bookmark = TempBookmark
            Call DisplayData
        End If
    Else
        TempRecordset.Bookmark = TempBookmark
        Call DisplayData
    End If
End Sub

‘单击按钮  后的执行代码
Private Sub CmdExit_Click()
    Unload Me
End Sub

‘显示记录内容的过程DisplayData的执行代码
Public Sub DisplayData()
    txtID.Text = TempRecordset.Fields(0)
    txtName.Text = TempRecordset.Fields(1)
    comboSex.Text = TempRecordset.Fields(2)
    comboType.Text = TempRecordset.Fields(3)
    txtBirthday.Text = Format(TempRecordset.Fields(4), "yyyy-mm-dd")
    txtClassno.Text = TempRecordset.Fields(5)
    ComboMajor.Text = TempRecordset.Fields(6)
    comboDirector.Text = TempRecordset.Fields(7)
    txtSource.Text = TempRecordset.Fields(8)
End Sub

```

---

### 3 查询信息窗体的设计

打开名为 FormSelectGraduate 的子窗体，如图 10.54 所示。窗体的设计如表 10.11 所示。

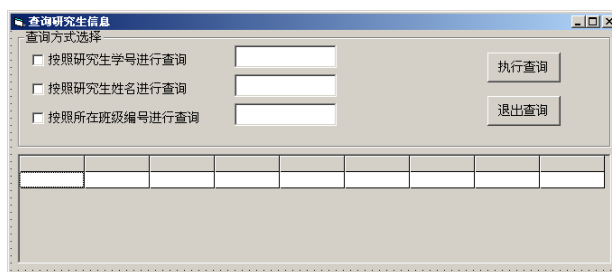


图 10.54 子窗体 FormSelectGraduate

表 10.11

子窗体 FormSelectGraduate 的设计

控件名称	类型	主要属性	属性值
FormSelectGraduate	【窗体】	名称	FormSelectGraduate
		Caption	查询研究生信息
		StartPosition	2-屏幕中心
Frame1	【Frame】	名称	Frame1
		Caption	查询方式选择
myflexgrid	【MSHFlexGrid】	名称	myflexgrid
Check1(0)	【CheckBox】	名称	Check1
		Caption	按照研究生学号进行查询
Check1(1)	【CheckBox】	名称	Check1
		Caption	按照研究生姓名进行查询
Check1(2)	【CheckBox】	名称	Check1
		Caption	按照所在班级编号进行查询
TxtID	【TextBox】	名称	TxtID
TxtName	【TextBox】	名称	TxtName
TxtClassno	【TextBox】	名称	TxtClassno
CmdSelect	【CommandButton】	名称	CmdSelect
		Caption	执行查询
CmdExit	【CommandButton】	名称	CmdExit
		Caption	退出查询

窗体的代码设计如下。

```


-----
'启动窗体时的执行代码
Private Sub Form_Load()
    '在窗体的myflexgrid表格上显示标题栏

```

```

With myflexgrid
    .CellAlignment = 4
    .TextMatrix(1, 0) = "学号"
    .TextMatrix(1, 1) = "姓名"
    .TextMatrix(1, 2) = "性别"
    .TextMatrix(1, 3) = "培养类别"
    .TextMatrix(1, 4) = "生日"
    .TextMatrix(1, 5) = "班级编号"
    .TextMatrix(1, 6) = "专业"
    .TextMatrix(1, 7) = "导师姓名"
    .TextMatrix(1, 8) = "生源性质"
End With
End Sub

```

‘单击  按钮后的执行代码

```


Private Sub CmdSelect_Click()
    Dim txtSQL As String
    Dim MsgText As String
    Dim Condition(3) As Boolean
    Dim TempRecordset As ADODB.Recordset
    ‘组合查询条件
    txtSQL = "select * from view_graduate where "
    If Check1(0).Value Then
        If Trim(txtID.Text) = "" Then
            sMsg = "学号不能为空"
            MsgBox sMsg, vbOKOnly + vbExclamation, "警告"
            txtID.SetFocus
            Exit Sub
        Else
            If Not IsNumeric(Trim(txtID.Text)) Then
                MsgBox "请输入数字学号!", vbOKOnly + vbExclamation, "警告"
                Exit Sub
            txtID.SetFocus
        End If
        Condition(0) = True
        txtSQL = txtSQL & "ID =' " & Trim(txtID.Text) & "' "
    End If
    End If
    If Check1(1).Value Then
        If Trim(txtName.Text) = "" Then
            sMsg = "姓名不能为空"

```

```

        MsgBox sMeg, vbOKOnly + vbExclamation, "警告"
        txtName.SetFocus
    Exit Sub
Else
    Condition(1) = True
    If Condition(0) Then
        txtSQL = txtSQL & "and Name =' " & txtName.Text & "' "
    Else
        txtSQL = txtSQL & "Name =' " & txtName.Text & "' "
    End If
End If
End If
If Check1(2).Value Then
    If Trim(txtClassno.Text) = "" Then
        sMeg = "班级编号不能为空"
        MsgBox sMeg, vbOKOnly + vbExclamation, "警告"
        txtClassno.SetFocus
    Exit Sub
Else
    Condition(2) = True
    If Condition(0) Or Condition(1) Then
        txtSQL = txtSQL & "and classno = ' " & txtClassno.Text & "' "
    Else
        txtSQL = txtSQL & "classno = ' " & txtClassno.Text & "' "
    End If
End If
End If
If Not (Condition(0) Or Condition(1) Or Condition(2)) Then
    MsgBox "请设置查询方式!", vbOKOnly + vbExclamation, "警告"
    Exit Sub
End If
txtSQL = txtSQL & " order by ID "
Set TempRecordset = ExecuteSQL(txtSQL, MsgText)
With myflexgrid
    .Rows = 2
    .CellAlignment = 4
    .TextMatrix(1, 0) = "学号"
    .TextMatrix(1, 1) = "姓名"
    .TextMatrix(1, 2) = "性别"
    .TextMatrix(1, 3) = "培养类型"

```

```
.TextMatrix(1, 4) = "生日"
.TextMatrix(1, 5) = "班级编号"
.TextMatrix(1, 6) = "专业"
.TextMatrix(1, 7) = "导师姓名"
.TextMatrix(1, 8) = "生源性质"
'显示查询结果
Do While Not TempRecordset.EOF
    .Rows = .Rows + 1
    .CellAlignment = 4
    .TextMatrix(.Rows - 1, 0) = TempRecordset.Fields(0)
    .TextMatrix(.Rows - 1, 1) = TempRecordset.Fields(1)
    .TextMatrix(.Rows - 1, 2) = TempRecordset.Fields(2)
    .TextMatrix(.Rows - 1, 3) = TempRecordset.Fields(3)
    .TextMatrix(.Rows - 1, 4) = Format(TempRecordset.Fields(4), "yyyy-mm-dd")
    .TextMatrix(.Rows - 1, 5) = TempRecordset.Fields(5)
    .TextMatrix(.Rows - 1, 6) = TempRecordset.Fields(6)
    .TextMatrix(.Rows - 1, 7) = TempRecordset.Fields(7)
    .TextMatrix(.Rows - 1, 8) = TempRecordset.Fields(8)
    TempRecordset.MoveNext
Loop
End With
TempRecordset.Close
End Sub
'单击  按钮后的执行代码
Private Sub CmdExit_Click()
    Unload Me
End Sub
```

---

## 10.5 如何执行实例程序

### 10.5.1 在服务器上的操作

(1) 以 system 用户、SYSDBA 身份登录【SQLPlus Worksheet】, 执行 creategraduateuser.sql 文件创建用户 graduateuser。

(2) 执行 createtablegraduateinfo.sql 文件创建数据表 graduateuser.graduate\_info。

(3) 执行 createtabledirectorinfo.sql 文件创建数据表 graduateuser.director\_info。

(4) 执行 createtablemajorinfo.sql 文件创建数据表 graduateuser.major\_info。

(5) 执行 createindexgraduate.sql 文件创建数据表 graduateuser. graduate\_info 的姓名字段的索引。

(6) 执行 createindexdirector.sql 文件创建数据表 graduateuser. director\_info 的姓名字段的索引。

(7) 执行 createviewgraduate.sql 文件创建数据表 graduateuser. graduate\_info 的视图。

(8) 执行 createviewdirector.sql 文件创建数据表 graduateuser. director\_info 的视图。

(9) 执行 createviewmajor.sql 文件创建数据表 graduateuser. major\_info 的视图。

## 10.5.2 在客户机上的操作

(1) 在客户机上调用 Oracle 9i 客户机的【网络配置助手】(Net Configuration Assistant) 配置一个能够连接数据库服务器的【本地网络服务名称】，笔者配置的名称为 myoraclelink。这个名称是可以任意取的。

(2) 在客户机的【控制面板】/【32 位 ODBC】中创建一个系统数据源名称，这个名称必须是 graduateDB。当然如果读者要采用自己的名称，可以在 VB 程序的模块 MODULE1 中进行适当的修改。修改的代码如下。

```
-----
'定义了名为ConnectionString的无参数函数，连接数据库
Public Function ConnectString() As String
    ConnectString = "DSN=graduateDB;UID=graduateuser;PWD=12345678"
End Function
-----
```

## 10.5.3 在客户机上运行程序

(1) 将配套光盘上的 Graduate.exe 文件任意复制到客户机上，执行该文件出现如图 10.55 所示界面。



图 10.55 研究生管理信息系统主界面

(2) 出现如图 10.56 所示的【添加专业信息】界面。

(3) 成功添加专业信息后出现如图 10.57 所示的提示界面。

(4) 按照同样的步骤依次添加专业信息数据，这样添加后的专业数据将用于研究生信息录入和导师信息录入窗体使用。

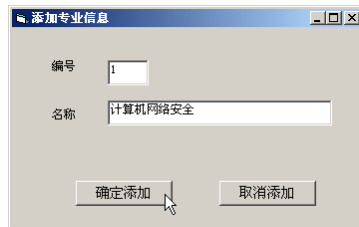


图 10.56 【添加专业信息】界面



图 10.57 成功添加专业信息

(5) 出现如图 10.58 所示的【添加导师信息】界面。

(6) 成功添加导师信息后出现如图 10.59 所示的界面。

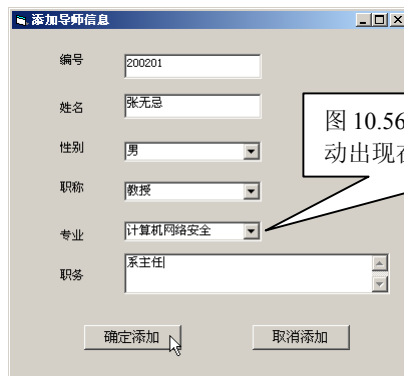


图 10.58 【添加导师信息】界面



图 10.59 成功添加导师信息

(7) 按照同样的步骤依次添加导师信息数据。

(8) 出现如图 10.60 所示的界面。

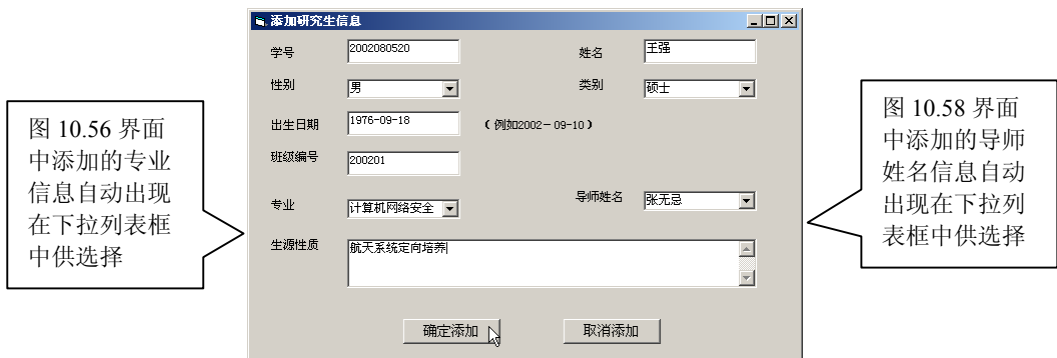


图 10.60 【添加研究生信息】界面

(9) 成功添加研究生信息后出现如图 10.61 所示界面，单击 **确定** 按钮。



图 10.61 成功添加研究生信息



(10) 出现如图 10.62 所示的【修改研究生信息】界面。

图 10.62 【修改研究生信息】界面

(11) 出现如图 10.63 所示的【查询研究生信息】界面。

学号	姓名	性别	培养类型	生日	班级编号	专业	导师姓名	生源性质
2002080520	王强	男	硕士	1976-09-18	200201	计算机网络安全	张无忌	航天系统定向培养
2002080521	赵敏	女	博士	1978-09-11	200201	计算机网络安全	王海洋	国家统招

图 10.63 【查询研究生信息】界面

(12) 在图 10.55 所示的系统主界面的【菜单栏】选择【退出系统】选项将退出系统。

(13) 其他操作读者可以自行学习，这里不再赘述。

## 10.6 习题

- (1) 在构建 VB+Oracle 9i 的开发和运行环境中，有什么主要的步骤？
- (2) ODBC 数据源和本地网络服务名是什么关系？
- (3) 通过试验熟悉 VB 程序中 3 种访问数据库的方法。
- (4) 完成一个完整的管理信息系统的开发。写出开发报告以及可以完整的运行程序。