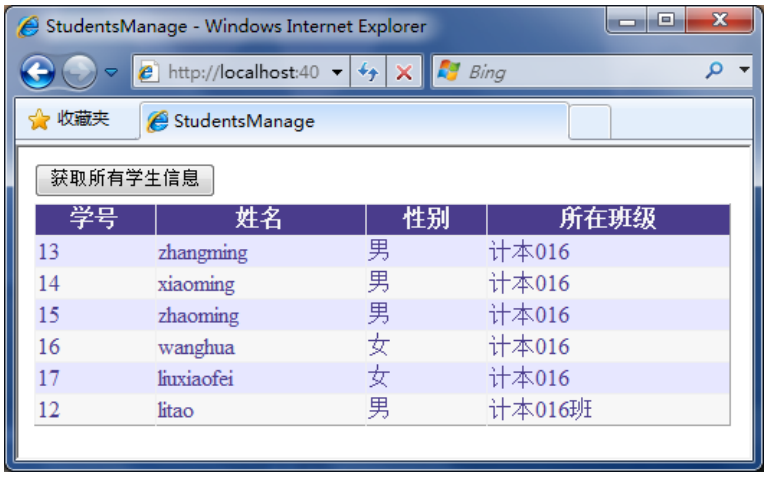
如何使用 Nhgdb 组件做.net 开发

瀚高基础软件股份有限公司版权所有
仅允许不作任何修改的转载和转发

功能概述

1. 从 HighGo DB 的学生表中查询出所有学生信息，结果以表格形式显示。该表由学号、姓名、性别、所在班级组成。
2. 界面显示如下，比较简单，一个按钮和数据表格。



学号	姓名	性别	所在班级
13	zhangming	男	计本016
14	xiaoming	男	计本016
15	zhaoming	男	计本016
16	wanghua	女	计本016
17	liuxiaofei	女	计本016
12	litao	男	计本016班

项目开发步骤

1. 数据库建立学生表和初始化测试数据。
2. 建立基于 asp.net web 应用程序 StudentsManage 项目。
3. 使用 Nhgdb 组件。
4. 修改 Web.config 设置 HighGo DB 数据库连接串。
5. 增加数据访问类 dataccess，处理与 HighGo DB 数据库访问。
6. 增加样式表 main.css，控制按钮和 gridview 的样式
7. 编写首页界面和对应的功能代码。
8. 运行 StudentsManage 项目,进行功能测试。

创建学生表并初始化测试数据

1. 在 HighGo DB 数据库中创建学生表。

```
CREATE TABLE student
(
            stunum integer NOT NULL,
            "name" character varying(30),
            sex integer,
            inclass character varying(60)
);
```

2. 插入些测试数据。

```
insert into student values(12,'liming',1,'计本 016');
insert into student values(13, 'zhangming',1,'计本 016');
insert into student values(14,'xiaoming',1,'计本 016');
insert into student values(15,'zhaoming',1,'计本 016');
insert into student values(16,'wanghua',0,'计本 016');
insert into student values(17,'liuxiaofei',0,'计本 016');
```

建立 StudentsMange 项目

1.在 D:\下创建 hgdbApp 目录，作为项目的存储位置。

2.打开 VS2008 的集成开发工具,如图 1-1。

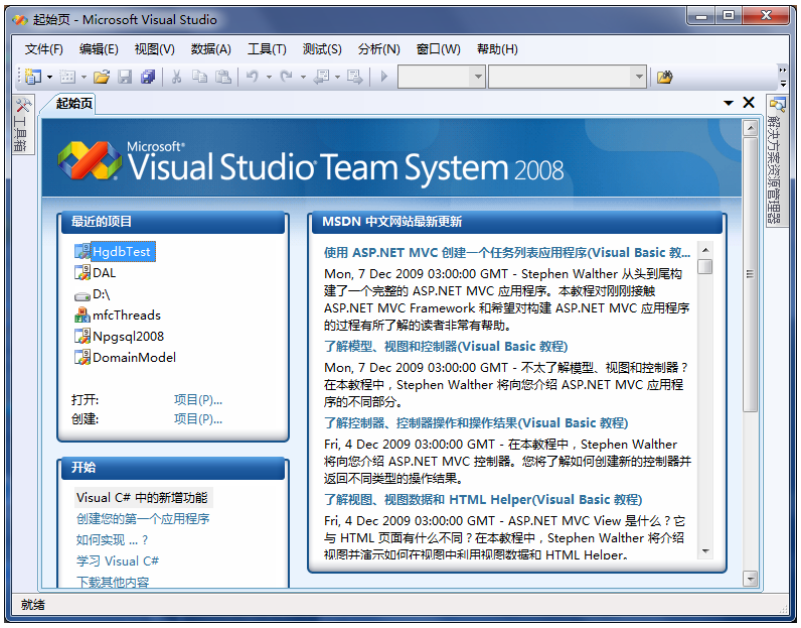


图 1-1

3．文件->新建->项目，弹出新建项目对话框，左侧项目类型中，选择 Visual C#->Web，选择右侧显示的“ASP.NET Web 应用程序”模板，另外还需要设置项目名称，选择项目位置和解决方案名称，解决方案名称会自动与项目名称一致，如图 1-2。

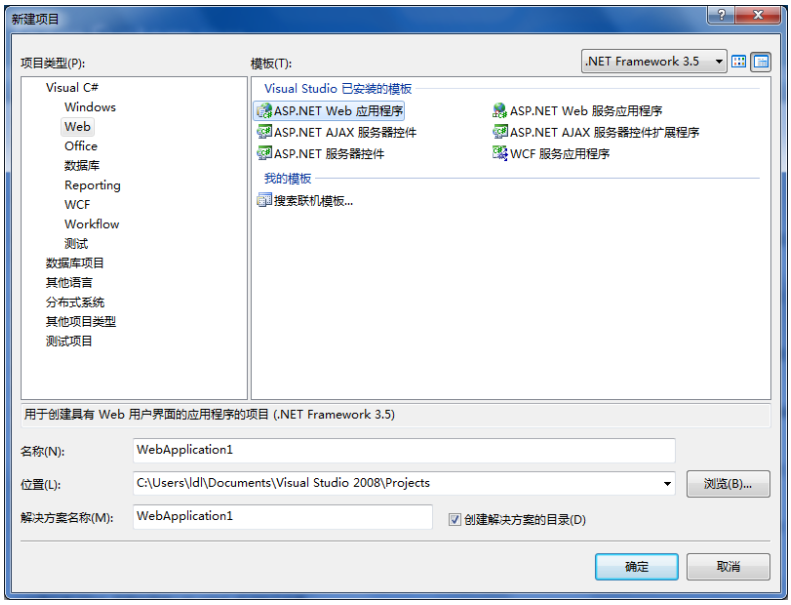


图 1-2

4．将项目名称设置为：StudentsMange，位置 D:\hgdbApp，点击“确定”按钮，就可以了。如图 1-3。

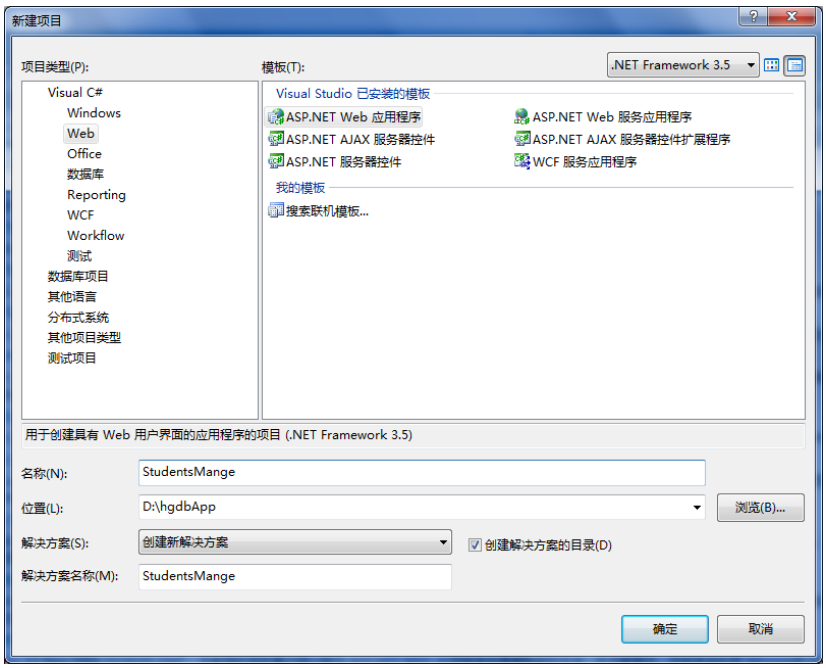


图 1-3

5. 项目创建后，会自动打开 Default.aspx 文件，从右侧浮动面板中点击“解决方案资源管理器”或者从菜单栏上点击视图->解决方案资源管理器。

1）Default.aspx：项目的主页面

2）web.config： 项目的配置文件。

3）引用： 项目使用微软或第三方提供的动态组件

4）App_Data：项目使用的数据存放目录。如图 1-4。

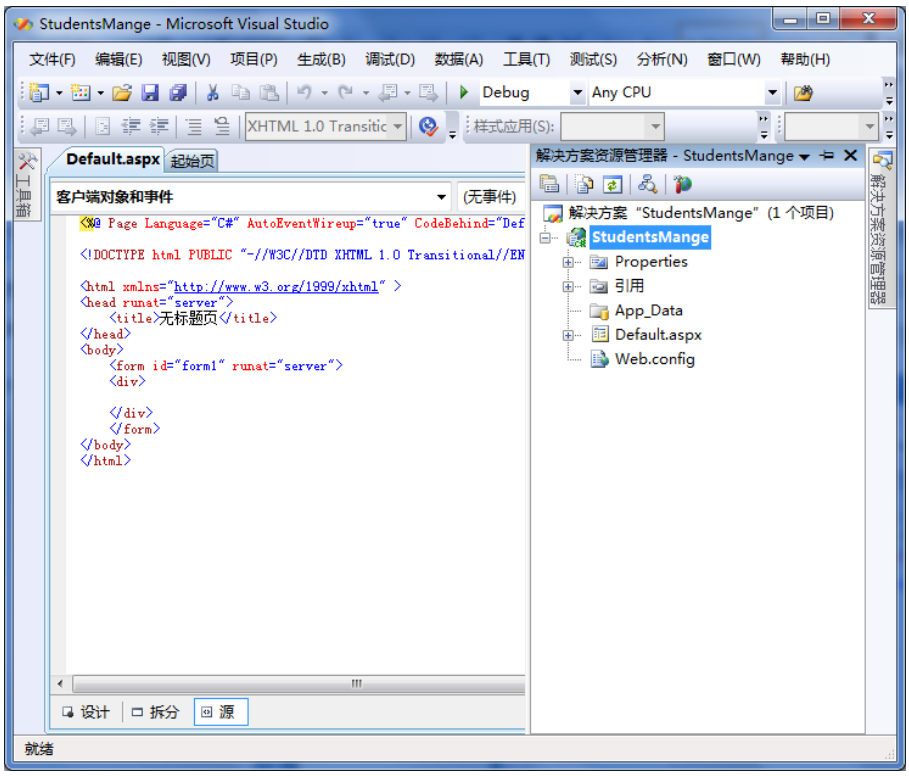


图 1-4

使用 Nhgdb 组件

如果没有 Nhgdb 1.1，请从 www.highgo.com.cn 这里下载，解压文件。例如 解压到 D:\ Nhgdb 1.1。

1. 从 VS2008 的菜单栏上点击视图->解决方案管理器,将项目展开,右键点击引用->添加引用。如图 1-5。

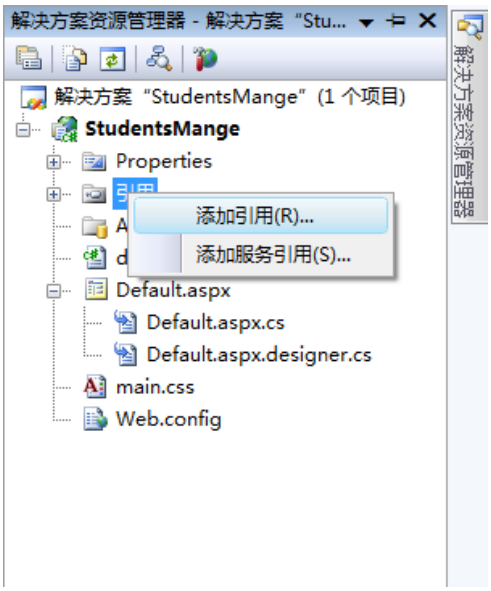


图 1-5

2. 在弹出的“添加引用”的界面，点击“浏览”选项卡， D:\ Nhgdb 1.1 目录下，选择 Nhgdb.dll 和 Mono.Security.dll,点击“确定”，就 OK。如图 1-6。

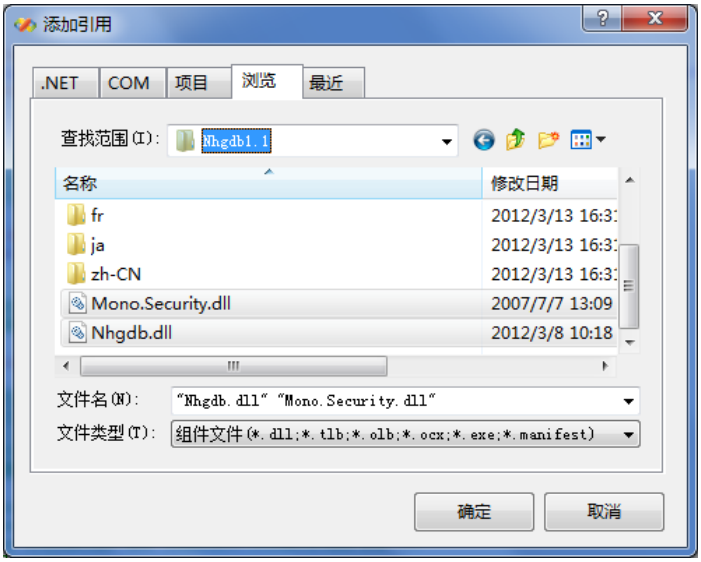


图 1-6

3. 添加引用后，便增加了 Nhgdb 和 Mono.Security 的组件对象，如图 1-7。

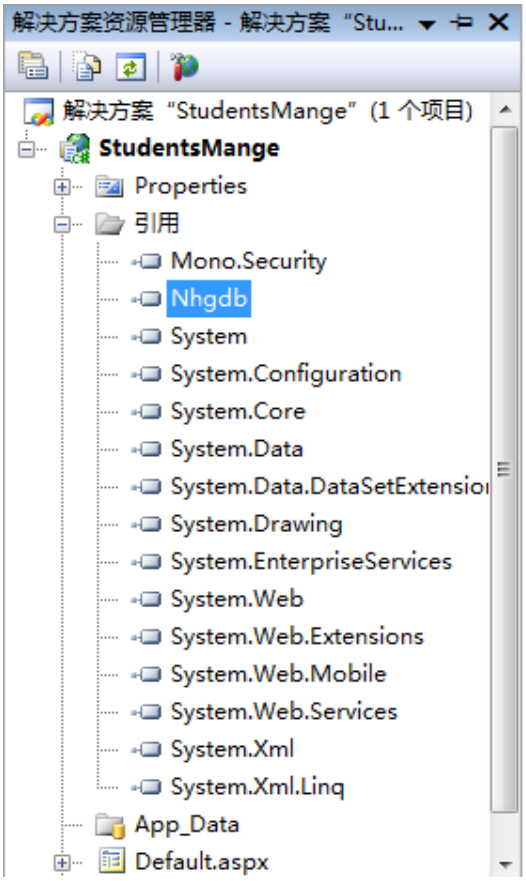


图 1-7

修改 web.config 设置数据库连接串

1．从右侧浮动面板点击“解决方案资源管理器”或者从菜单栏上点击视图->解决方案资源管理器，打开 web.config 文件，添加数据库连接串。将 Configuration 根配置节点下的 connectionStrings 节点修改为：

<!-- 添加数据库的连接串-->

<connectionStrings>

<add name="highgo"

connectionString="Server=127.0.0.1;Port=5866;User Id=ldl;Password=1;Database=highgo;"/>

</connectionStrings>，如图 1-8。

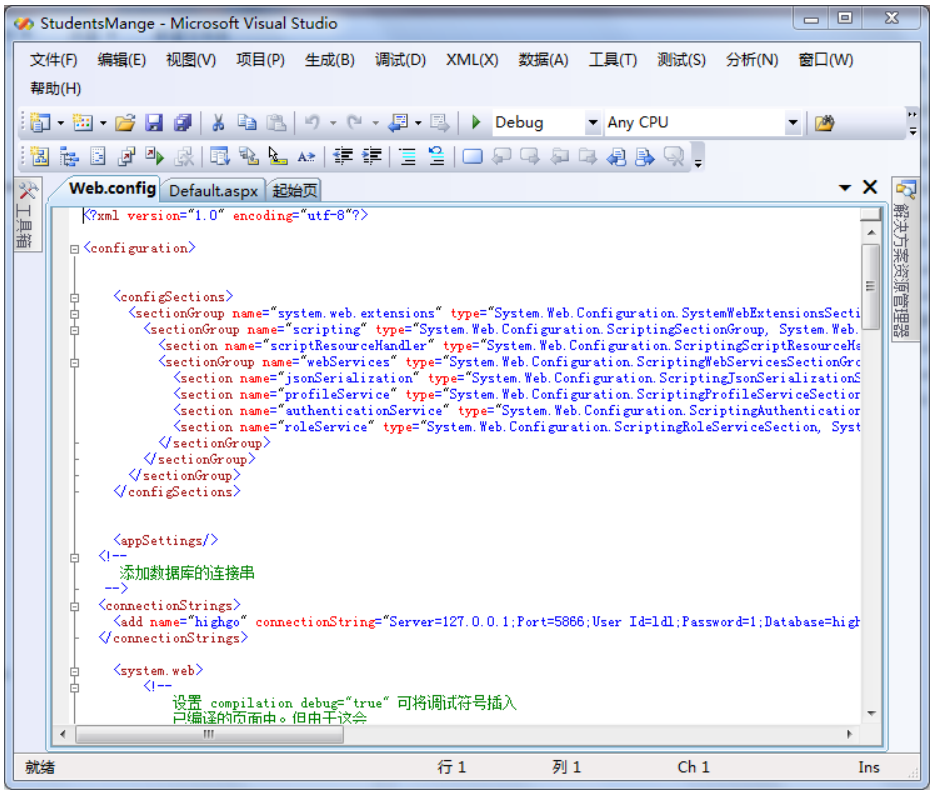


图 1-8

添加数据库访问类

1．在“解决方案资源管理器”中，右键点击项目->添加->类，弹出”添加新项”对话框,输入 dataccess.cs 名，点击“添加”，如图 1-9。

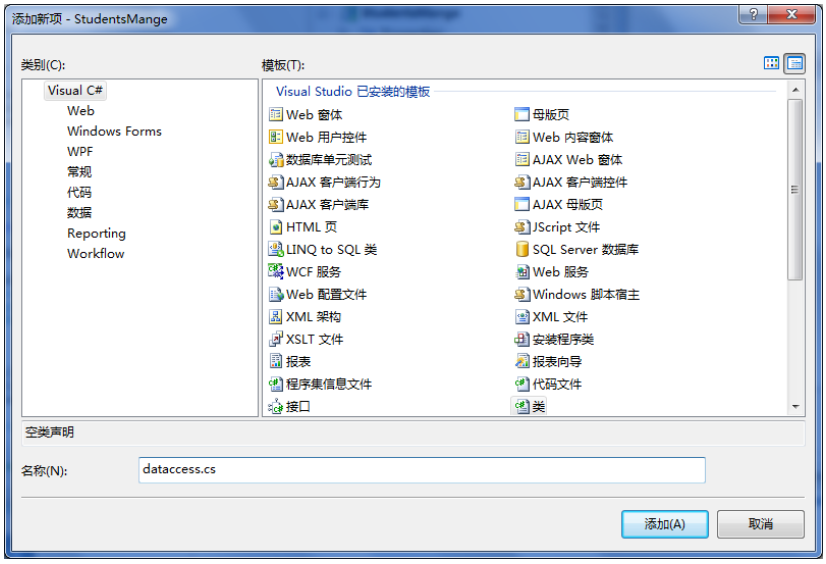


图 1-9

3．完成添加类后，自动打开 dataccess.cs 文件,在该文件中，使用 Nhgdb 名空间，添加静态的数据库连接串成员 strConn 和执行 sql 数据库查询方法 ExeQuery。

dataccess 类主要负责执行 sql 将结果返回到数据表。 如图 1-10。

```
using System;

using System.Data;

using System.Configuration;

using System.Linq;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.HtmlControls;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Xml.Linq;

using Nhgdb;

namespace StudentsMange

{
```

```
    public class dataccess
     {
        // 数据库连接串
        static private string strConn =
ConfigurationManager.ConnectionStrings["highgo"].ConnectionString;
        public DataTable ExeQuery(string sql)
        {
            DataTable aTable = new DataTable();
                //建立数据库连接
            NpgsqlConnection conn = new NpgsqlConnection(strConn);
            NpgsqlDataAdapter adpater = new NpgsqlDataAdapter(sql, conn);
            NpgsqlCommandBuilder builder = new NpgsqlCommandBuilder(adpater);
            try
            {
                 conn.Open();//打开数据库连接
                 adpater.Fill(aTable); //执行sql填充结果集.
                return aTable;
            } catch (Exception ex) {
                throw ex;
            }finally {
                conn.Close();
            }
        }
    }
}
```

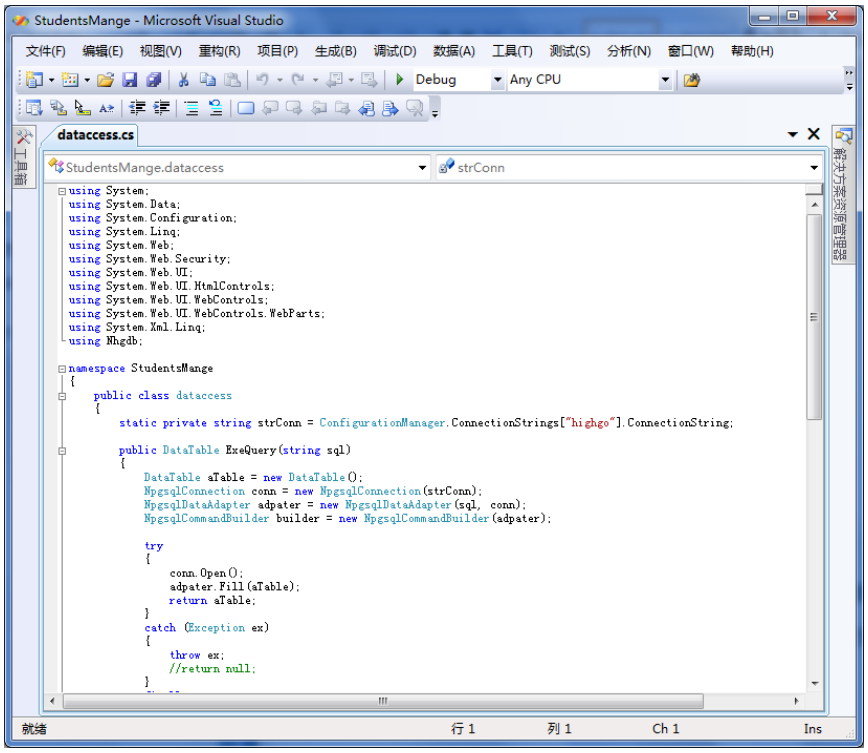


图 1-10

添加 main.css 样式表

1．从右侧浮动面板点击“解决方案资源管理器”或者从 VS2008 的菜单栏上点击视图->解决方案资源管理器，选择项目右键->添加->新建项->添加新项，选择样式表，将名称改为 main.css。如图 1-11。

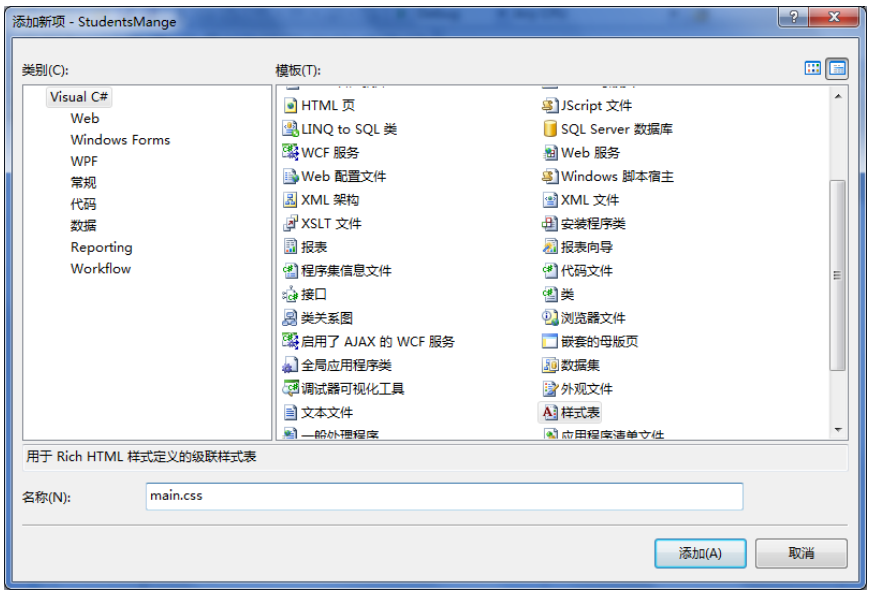


图 1-11

2. 添加完样式表后，自动打 main.css 文件，在该文件中添加如下样式，详见公司提供测试代码示例中的 main.css 文件,如图 1-12。

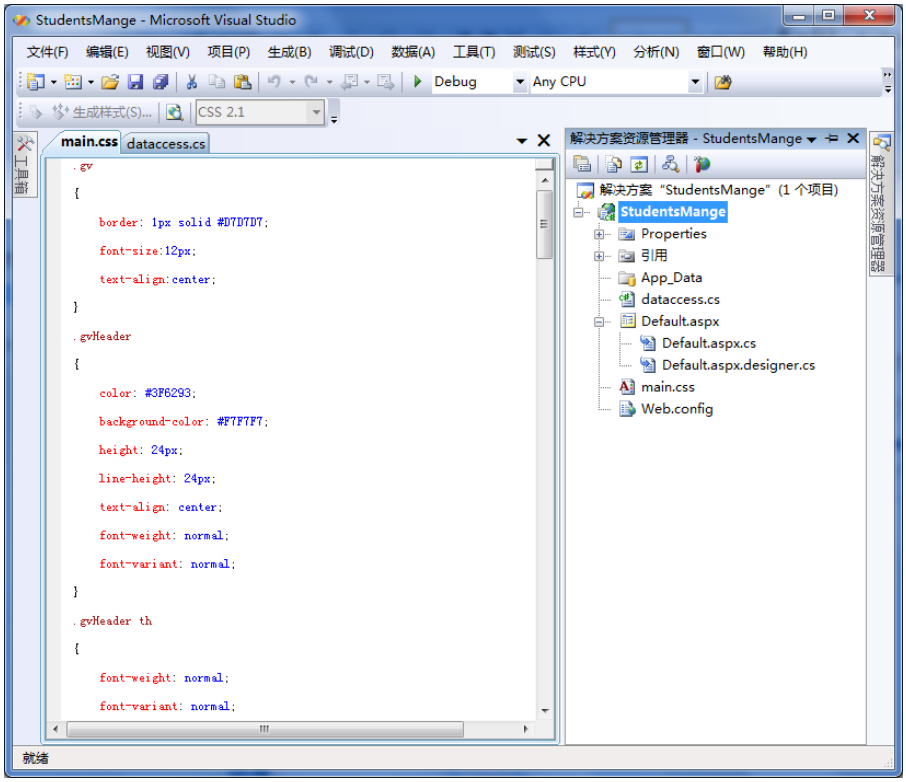


图 1-12

编写主页代码

1. 从右侧浮动面板，点击“解决方案资源管理器”或者从 VS2008 的菜单栏上点击视图->解决方案资源管理器,打开 Default.aspx 文件，添加一个两行一列的表格，第一行添加按钮查询所有学生信息，第二行添加 gridview，显示查询结果。如图 1-13。

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs"
Inherits="HgdbTest._Default" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"  >
<head  runat="server">
    <title> StudentsManage</title>
```

```
    <link href="main.css" rel="stylesheet" type="text/css" />
</head>
<body>
    <form id="form1" runat="server">
    <table width="500">
        <tr><td>
                <asp:Button ID="btnGetAllStus" runat="server" Text="获取所有学生信息"
                    onclick="btnGetAllStus_Click" />
        </td></tr>
        <tr>
            <td>
                <asp:GridView ID="GridView1" runat="server" Width="98%"
                    style="white-space:nowrap" AutoGenerateColumns="False">
                    <RowStyle BackColor="#E7E7FF" ForeColor="#4A3C8C" CssClass="gvRow" />
                    <HeaderStyle BackColor="#4A3C8C" Font-Bold="True" ForeColor="#F7F7F7"
CssClass="gvHeader" />
                    <AlternatingRowStyle BackColor="#F7F7F7" CssClass="gvAlternatingRow" />
                    <Columns>
                        <asp:BoundField DataField="stunum" HeaderText="学号" />
                        <asp:BoundField DataField="name" HeaderText="姓名" />
                        <asp:BoundField DataField="sex" HeaderText="性别" />
                        <asp:BoundField DataField="inclass" HeaderText="所在班级" />
                    </Columns>
                </asp:GridView>
            </td>
        </tr>
    </table>
    </form>
</body>
</html>
```

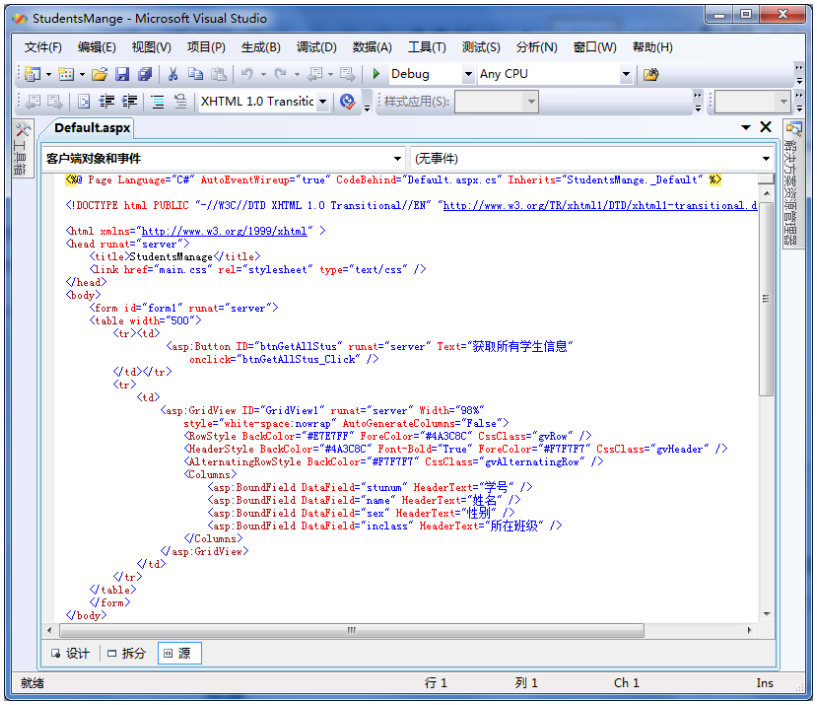


图 1-13

2．从右侧浮动面板，点击“解决方案资源管理器”或者从VS2008的菜单栏上点击视图->解决方案资源管理器,打开Default.aspx.cs文件，在该文件中，添加查询所有学生信息的按钮事件。如图1-14。

```
protected void btnGetAllStus_Click(object sender, EventArgs e)
{
        string sql =
        "select stunum,name,(case when sex=1 then '男' else '女' end) sex,inclass from student";
            dataccess aDb = new dataccess();
            DataTable aTable = aDb.ExeQuery(sql);
            GridView1.DataSource = aTable;
            GridView1.DataBind();
}
```

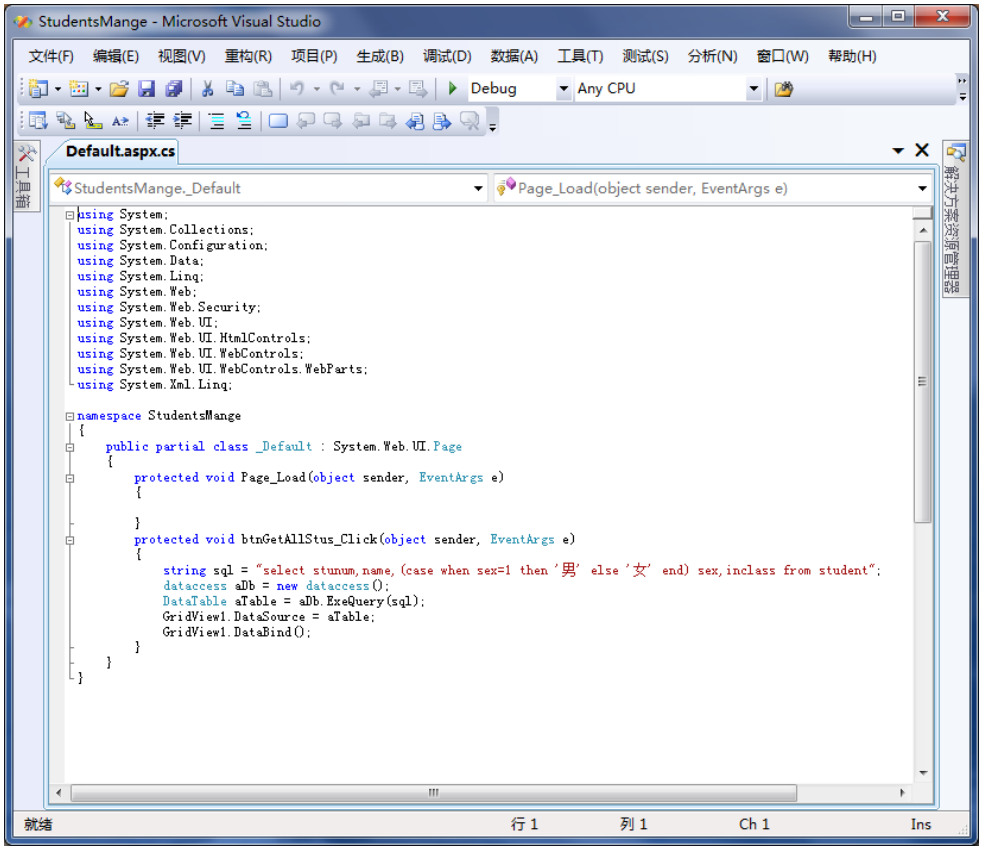


图 1-14

运行测试

1. 从 VS2008 的菜单栏点击调试->开始执行(不调试),运行 StudentsMange 项目。菜单显示如图 1-15。

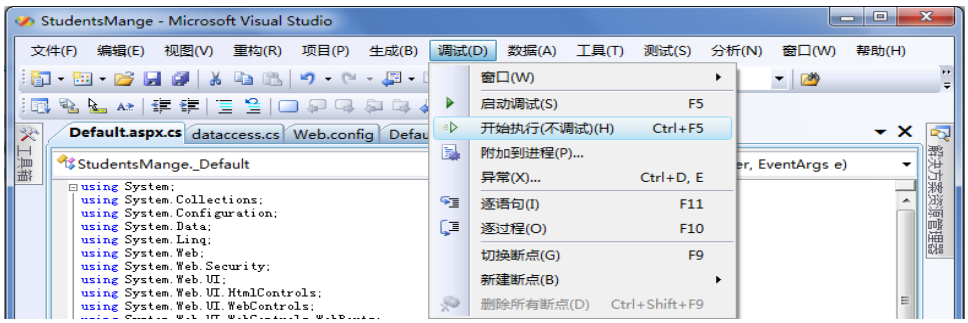


图 1-15

2. 运行后的首页显示如图 1-16 。

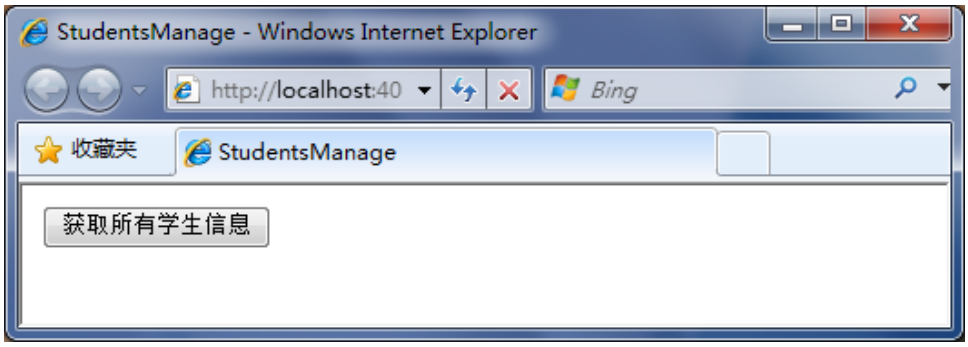


图 1-16

3. 点击获取所有学生信息，显示查询到的学生信息列表。如图 1-17。

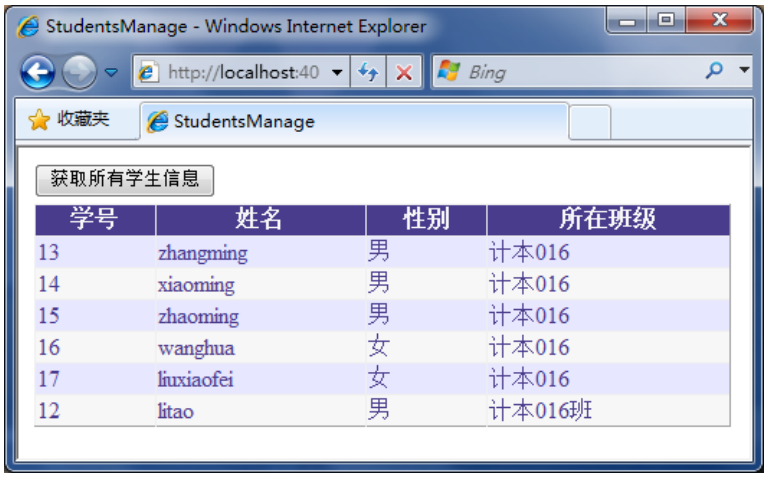


图 1-17