

显示某张表中的所有记录

1、打开 SQL Server 2008 R2，在里面创建数据库 test2,然后在数据库创建 users 表,里面有三个字段 id(自动增加),username(varchar(50)),password(varchar(50))，然后使用 insert 插入几条记录。

2、创建一个文件夹，用来存储该程序，假设我放在 e:\test 文件夹下。

3、打开 VS2010，创建解决方案：

步骤：文件--新建--项目，在弹出来的“新建项目”对话框中，已安装的模板中选择“其他项目类型”--“Visual Studio 解决方案”，然后在右边选择“空白解决方案”，其次在名称中输入“Solution1”，然后在位置中选择上面的新建的文件夹：e:\test，然后点击“确定”。

4、创建一个 ASP.NET Web 应用程序，用来显示 users 中的所有内容。

步骤：文件--新建--项目，在弹出来的“新建项目”对话框中，已安装的模板中选择“Visual C#”--“ASP.NET Web 应用程序”，名称处取名叫“myweb”，位置选择:e:\test\Solution1,解决方案中选择“添加到解决方案”--然后点击“确定”。此时，右边的解决方案资源管理器中多出了一个 myweb，里面有许多文件。这里双击里面的 web.config 文件，向里面添加数据库连接代码。

```
<connectionStrings>
<add name="ConnectionString" connectionString="Data Source=.;Initial Catalog=test2;
Integrated Security=True;" providerName="System.Data.SqlClient"/>
</connectionStrings>
```

5、创建一个类库(Data)来连接数据库

步骤：文件--新建--项目，在弹出来的“新建项目”对话框中，已安装的模板中选择“Visual C#”--“类库”，名称:Data；解决方案：选择添加到解决方案；位置：E:\test\Solution1，然后点击“确定”。此时右边的解决方案资源管理器中多了一个 Data。找到 class1.cs 文件，右击重命名改为 DBConn.cs;然后双击这个文件，左边出现它的所有代码。这里开始做引用了，类型 asp 中的 include file 文件，过程比 asp 多一步。操作过程：右击 Data 目录下面的引用--添加引用。在弹出的添加引用对话框中，选择.net 选择卡，找到 System.Configuration，再点击确定。这样，右边的引用中就多了 System.Configuration 了。引用这个文件的目的是获取 web.config 文件中的值，这里主要用来获取连接数据库字符串。此时我们开始处理 DBConn.cs 文件了。先在代码中引用刚才引用的文件，在这个文件顶部 using System.Text;下面加上 using System.Configuration;还要加上一个操作 sql server 的文件 using System.Data.SqlClient;然后开始定义一个连接数据库函数，如下：

```
public SqlConnection getconn()
{
    string connstr =
    System.Configuration.ConfigurationManager.ConnectionStrings["ConnectionString"].ToString();
    SqlConnection conn = new SqlConnection(connstr);
    return conn;
}
```

这段代码要放到

```
public class DBConn
{
    //要放到这个地方
```

```
}
```

要放到这里的两个大括号之间。

6、Model 层,直接对数据库的表中的字段赋值和取出某个字段中的值的层。

创建的方法同 Data 层只是最后的文件名叫 Model 而已。具体步骤:

步骤: 文件--新建--项目, 在弹出来的“新建项目”对话框中, 已安装的模板中选择“Visual C#”--“类库”, 名称:Model; 解决方案: 选择添加到解决方案; 位置: E:\test\Solution1, 然后点击“确定”。此时右边的解决方案资源管理器中多了一个 Model。找到 class1.cs 文件, 右击重命名改为 User.cs;然后双击这个文件, 左边出现它的所有代码。然后再

```
public class User
```

```
{
```

```
}
```

的代码之间添加以下代码:

```
public User() {}
```

```
private string _username;
```

```
public string username
```

```
{
```

```
set { _username = value; }
```

```
get { return _username; }
```

```
}
```

```
private string _password;
```

```
public string password
```

```
{
```

```
set { _password = password; }
```

```
get { return _password; }
```

```
}
```

到此为止, 上面还未涉及到 VS 的三层架构, 以上都是为三层架构做准备的, 从下面开始正式开始三层架构。

三层架构是逻辑上的三层, 只是有数据的调取方法的顺序有关, 在开发的过程中没有多大的区别。

7、数据访问层 (Dal), 该层所做事务直接操作数据库, 针对数据的增添、删除、修改、更新、查找等。

创建的步骤如第6条, 名称: Dal; 解决方案: 添加到解决方案; 路径为 e:\test\Solution1

然后将 Dal 目录下面的 class1.cs 改名为 UserDB.cs。这里要对 Model 层进行操作, 所以要引用 Model 层。

方法, 右击引用--添加引用--选择“项目”选项卡--选择 Model--确定; 然后在 UserDB.css 的顶部加上 using Model;由于这层要对数据库进行操作, 需要加上

```
using Data; (数据库连接字符串在这里)
```

```
using System.Data;
```

```
using System.Data.SqlClient;
```

然后在

```
public class UserDB
```

```
{
```

```
}
```

之间添加一个显示 users 表中的所有用户信息的函数,如下:

```
public string getlist()
```

```
{
```

```
string str = "";
```

```

Data.DBConn dbconn = new DBConn();
SqlConnection myconn = dbconn.getconn();
myconn.Open();
string sql = "select * from users order by id desc";
SqlDataAdapter da = new SqlDataAdapter(sql, myconn);
DataSet ds = new DataSet();
da.Fill(ds);
if (ds.Tables[0].Rows.Count > 0)
{
    str = "<table cellpadding=0 cellspacing=0 border=1 width=500 >\r\n";
    string username, password;
    foreach (DataRow m in ds.Tables[0].Rows)
    {
        username = m["username"].ToString();
        password = m["password"].ToString();
        str += "<tr>\r\n";
        str += " <td>" + username + "</td>\r\n";
        str += " <td>" + password + "</td>\r\n";
        str += "</tr>\r\n";
    }
    str += "</table>\r\n";
}
da.Dispose();
da = null;
myconn.Dispose();
myconn.Close();
return str;
}

```

8、业务层（BII），针对具体问题的操作，也可以说是对数据层的操作，对数据业务逻辑处理。

创建的方法同上。名称为 BII；解决方案：添加到解决方案；路径为 e:\test\Solution1；然后将 BII 目录下面的 class1.cs 改名为 UserBII.cs 了。这层要引用 Dal 层，所以要添加引用 Dal，方法同上；然后在 UserBII.cs 顶部加以下代码 using Dal；然后双击 UserBII.cs，在左边

```

public class UserBII
{
}

```

中间添加以下代码

```

public string getuserlist()
{
    Dal.UserDB db = new UserDB();
    return db.getlist();
}

```

9、表现层（UI），通俗讲就是展现给用户的界面，即用户在使用一个系统的时候他的所见所得。

现在开始给页面添加代码了，让查询出来的内容显示到页面上来。步骤：展开 myweb,找到 default.aspx 文件，双击。在左边的 </asp:Content> 上面加上代码 <%=str%>。此时 str 有个红色的下划线，因为没有定

义这个变量。点击 **default.aspx** 前面的加号，展开 **default.aspx**,展开后，双击 **default.aspx.cs**。这里需要引用 **Bll** 文件，右击里面的引用，添加引用，选择“项目”，找到 **Bll**，然后点击“确定”，最后在顶部加上 **using Bll**；。接下来定义 **str** 变量，在 **protected void Page_Load(object sender, EventArgs e)**上面加上 **public string str**；然后在 **protected void Page_Load(object sender, EventArgs e)**两个大括号之间添加以下代码：

```
Bll.UserBll ub = new UserBll();
```

```
str = ub.getuserlist();
```

就可以了

10、右击 **default.aspx**,选择“在浏览器中查看”。