**《网络程序设计》实验一**

**报告**

**【宣城校区】**

**设计题目：**

**任务1使用平面设计软件设计并生成Web页面**

**任务2 UI前端设计页面**

**姓名：陈嘉乐**

**班级：计科21-3班**

**学 号：**

**专业班级：13计科02班**

# 2023年4 月

# 一、实验目的

1）掌握静态网页的万维网访问原理；

2）熟悉 HTML 网页的基本结构；

3）掌握 HTML 中表格的使用方法；

4）掌握 HTML 中关于文字、图像以及超级链接的编辑；

5）掌握 Visual Studio/Dreamwaver 等工具的 HTML 编辑能力；

6）熟悉网页的平面设计方法，以及相关工具 Fireworks/Photoshop 的使用；

7）结合主流的前端框架中的组件库，搭建专业的前端网页（Vue.js、Element UI 为例。

# 二、实验环境

1.vs2019下框架版本为4.7.2的asp.net

2.Macromedia Fireworks 8

3.Adobe Photoshop 2021

4.Pc: win10 x64

5.bootstrap3.4.1

6.Sublime Text

# 三、实验内容

（1）设计一款主页（个人主页，或商业公司主页，题材不限，不允许使用我上课用的png等提供的模板，请百度自己搜索合适模板修改成自己的模板），然后利用PS或Fireworks软件切割生成HTML与Images文件导出，并编辑处理。

（2）结合前端组件知识，创建自己的第一个前端组件构建的页面。

**注意啊。是两个任务！！！！！！！！！**

# 四、原理实验步骤

1.先大致在脑海里确定自己的个人主页大致的主题，可以在纸上大概设计和规划下网页的样貌，然后确定主题后就可以按照这个主题去搜集材料和素材。

2.我在找到自己需要的图片模板后，选择使用Fireworks软件切割生成HTML与Images文件并导出，放到一个名为img的文件夹里。

3.要用到的素材都准备好后，就开始利用和结合前端组件知识来创建我的个人主页。

4.利用<div>标签创建容器，在容器里放置其它标签组件，这样方便处理组件在网页的位置。合理使用相应标签实现相应组件，例如<img>里放置图片,<span>、<p> 等标签里放置文本内容。<a>标签可以放置文本点击链接，实现跳转功能。

5.给各个标签里设置选择器，方便给各个组件设置属性、调整样式。我选择使用类选择器（class）。通过使用类选择器，在css文件里给各个类名设置如颜色，宽度，高度，大小，外边距，内边距等属性以达到自己的理想效果。

# 五、关键问题及解决方法

1.学会使用Fireworks或PS软件切割生成HTML与Images文件并导出。

解决方法：先用Fireworks软件打开需要切片加链接的图片，并对其进行合理调整和切片处理。之后选择“文件”>“html 设置”，或者单击“导出”对话框中的“选项”按钮。我在实际运用切片的时候，都是只输出图片，不输出表格的，表格自己来做。若是使用PS，则打开需要切片加链接的图片后，选择“切片工具”，或者按键盘上的Shift+C来切换到“切片工具”。画切片。利用参考线画出的切片会没有分隔符。画好切片后，选择需要添加链接的单个切片区域，点击鼠标右键。选择“编辑切片选项”。在“URL”那一栏的框里填上你需要添加的网址链接。填好后点击“确定”。其他的切片区域也可以进行添加链接。点击“储存”后，弹出“将优化结果储存为”的窗口，这里要把格式更改成“HTML和图像”或者“仅HTML”。HTML是网页格式链接就做里面。

2.如何精确设定元素位置

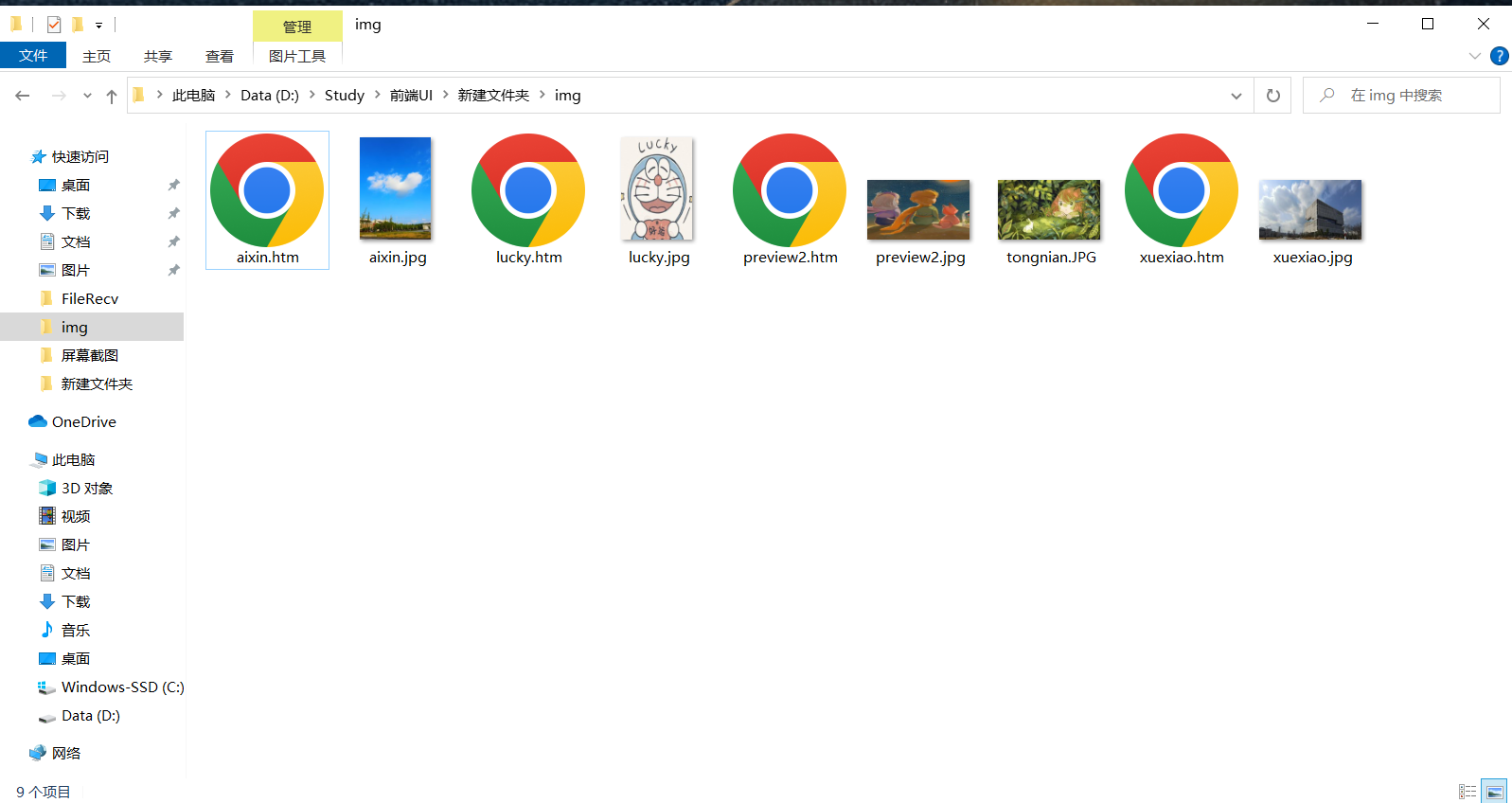
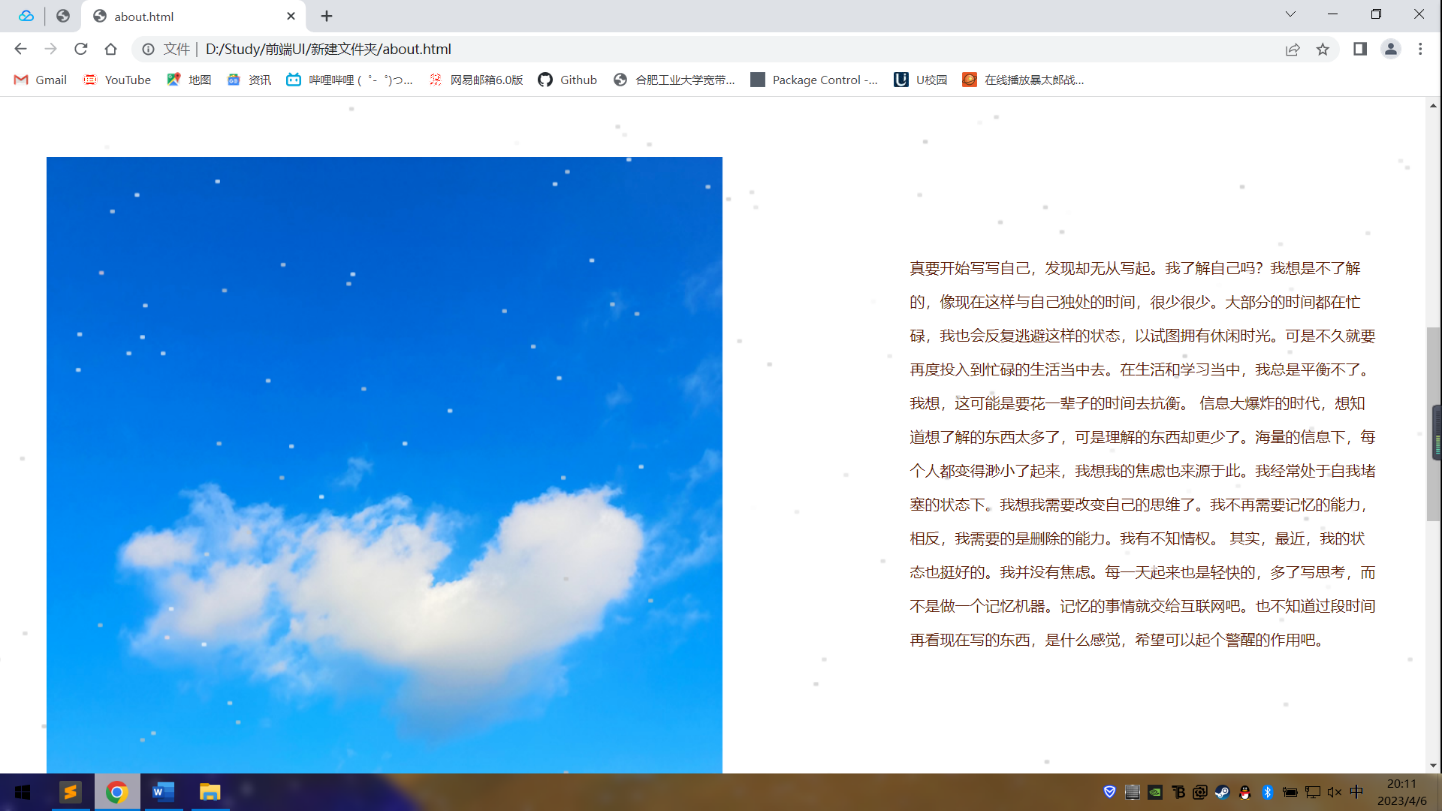
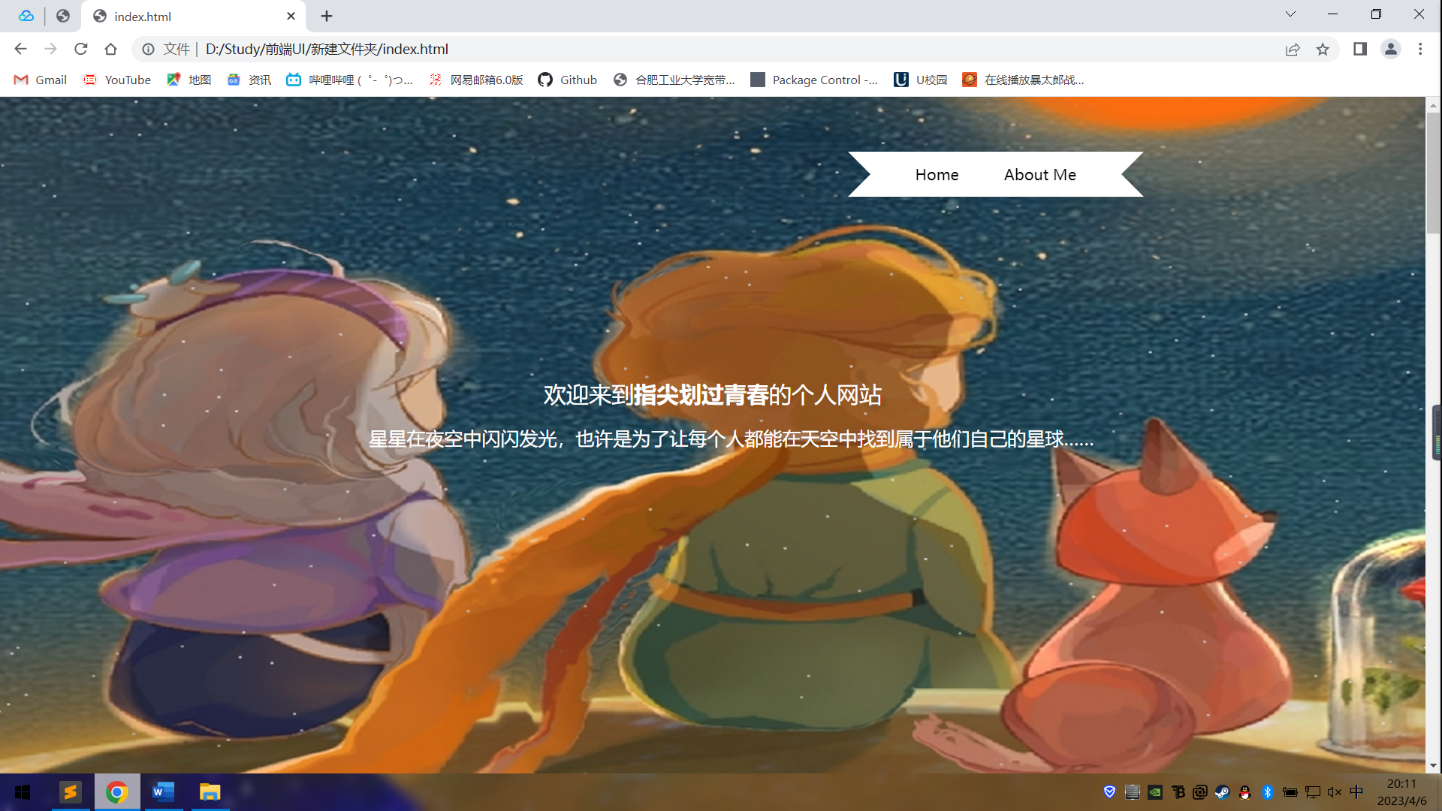
解决方法：合理使用<div>标签。<div>标签可定义文档中的分区或节，可以把文档分割为独立的、不同的部分。它可以用作严格的组织工具，并且不使用任何格式与其关联。如果用 id 或 class 来标记，那么该标签的作用会变得更加有效。当我们使用CSS-P的时候，我们主要把它用在DIVtag上。当你把文字，图象，或其他的放在DIV中，它可称作为“DIV block”。而中文我们把它称作“层次”。如果单独使用 DIV 而不加任何 CSS-P,那么它在网页中的效果和使用是一样的。但当我们把CSS-P用到 DIV中去以后，那么就可以严格设定它的位置。

3.如何具体对某个标签设置属性且代码不冗余杂乱

解决方法：使用类选择器，类选择器只要你在前面style标签中设置好了类名和相关属性，那么你想要的一些属性就好比成了一件装备，在后面的标签中直接加在class后面的属性值里就行了。这样设置的好处在于具有“特异性”，那些标签想加属性，那些标签不想加都可以很快做到。而且这些标签可以相同，突破了基础选择器的局限。而当你想给局部一小部分相同标签（比如p标签，在网页文章排班中），可以可以再用一个div或span标签将这一小群相同标签“包禳”起来，这样只需在div标签中添加属性，那么这被包括的全部小标签都被添加了相关属性了，可以省去给一个个给小标签添加属性的麻烦。

类选择器允许以一种独立于文档元素的方式来指定样式。该选择器可以单独使用，也可以与其他元素结合使用。只有适当地标记文档后，才能使用这些选择器，所以使用这两种选择器通常需要先做一些构想和计划。在使用类选择器之前，需要修改具体的文档标记，以便类选择器正常工作。为了将类选择器的样式与元素关联，必须将 class 指定为一个适当的值。注意:class里面的属性值不能以数字开头,如果以符号开头,只能是'\_'或者'-'这两个符号,其他的符号不可以,一个class里面可以有多个属性值。

# 六、实验结果



# 七、实验体会

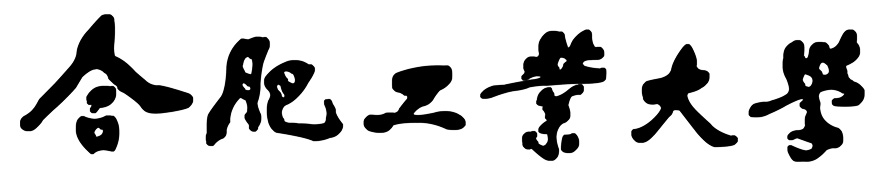
（特别需要阐述与使用bootstrap或Vue.js或实验课提供给同学们的admin-UI 设计Web UI的体会。本实验，不需要附HTML以及Images源文件）

关于网页的布局问题，以典型的登录界面为例：一是用css样式来不断的调整；二是直接在vs的所见即所得的界面中进行调节，使用空格来进行调整；三是利用表格，可以先写一行出来，然后往下添加，通过合并和把组件拜访在不同的位置来达到调整布局的目的。

关于切片的问题，切片一定不要切得太碎，否则导致表格太散、太乱。一是后续很难再改变设计区域，二是组件的大小、或者是图片的大小都要非常的严格，不然就会导致整个切片的网格布局乱掉。

最后是关于网页的美化，很多框架可以直接使用。以bootstrap为例子，从官网下载压缩包后解压，将里面的重要的css、fonts、js文件夹拖到我们项目创建的一个文件夹下面，然后使用link标签引入bootstrap框架的外部样式表，script标签引入bootstrap框架的外部脚本。我们在官网上查看需要组件的样式，找到合适的、美观的样式之后要观察官网给出的源码中的样式，然后修改对应的样式再运用到自己的控件上。

通过这次实验，我更加熟练了前端三件套html、css和js的使用，在Web应用中， HTML定义了一套语法规则，来告诉浏览器如何把一个丰富多彩的页面显示出来。由于HTML是富文档模型，所以，还有一系列的Tag用来表示链接、图片、表格、表单等等。CSS是用来美化利用html制作出来的页面的，用来控制HTML里的所有元素如何展现。JavaScript是为了让HTML具有交互性而作为脚本语言添加的，JavaScript既可以内嵌到HTML中，也可以从外部链接到HTML中，通过本次实验，我更加深刻的了解了这方面的知识，为以后的前端开发奠定了基础。



**【宣城校区】**

**《网络程序设计》实验二、三**

**报告**

**设计题目：AOD.Net用户WebSite登录验证及检索当前用户选课信息**

**学生姓名：陈嘉乐**

**学 号：2021218152**

**专业班级：13计科02班**

# 2023年4 月

# 一、实验目的

1）熟悉 Visual Studio 开发环境，学会使用 VS 创建网站基本结构

2）熟悉 HTML 服务器控件，特别是 FORM 表单服务器控件的设置

3）掌握表单处理方法

4）掌握服务器内置对象：Request、Response、Session 等内置对象的使用

5）掌握 ASP.NET 日期时间类

6）掌握在 MS SQL 创建数据库、数据表，并能使用 SQL 语句进行表操作（SELECT、

INSERT、UPDATE、DELETE）

7）掌握 Web.config 文件中变量的创建与读写

8）掌握 ADO.NET 提供的连接式数据访问方式，了解非连接式数据访问方式

9）掌握数据源 Web 服务器控件，数据 Web 服务器控件

# 二、实验环境

1. vs2019下框架版本为4.7.2的asp.net
2. Pc: win10 x64
3. bootstrap3.4.1
4. sql server 2022
5. SSMS 19.0.2

# 三、实验内容

（1）完成实验报告二中的数据库、表设置等相关内容与操作。

（2）获取与表单用户名、密码后，在注册数据库中验证成功（不用实现实验二的口令、密码验证），则在myhome页面显示当前用户的私有数据；否则，提示请输入正确的用户名、密码。

（3）使用 GridView 等控件，增强数据的显示与编辑 在 myhome.aspx 页面中，添加控件 GridView1，并根据自己对本门课程中，对数据源 服务器控件、数据绑定等掌握情况，合理地设计与编辑 GridView 控件的功能（请显示学号，姓名，性别，出生年月，专业，课程ID，课程名称，课程选修时间，课程成绩等信息）。

仁者见仁，智者见智，此处不再一一详细列要求了。

当然最好时将简短美观相关技术应用到实验页面中（**特别是使用前端技术显示，或者利用前端里面包含的css原始表，来美化微软asp.net控件的外观**）

# 四、原理实验步骤

1. 建立asp.net项目，创建两个web窗口，分别为：login.aspx作为登录页面， myhome.aspx作为登录后跳转的界面。

2. 在login.aspx文件内先利用表格搭建一个比较整齐的的登录界面。

3. 在login.aspx的cs文件中进行登录验证，需要在数据库中新添加一个字段，或者在excel表格中新加一段再导入数据库。查询的时候命令后面加上: and password=xxx即可，之后利用command.ExecuteScalar()方法返回的值转化为int类型后判断是否大于0即可知道关于这段信息的数据是否存在于表中。

4.防止跳过登录访问的登录验证，这部分利用session对象来实现。在连接服务端连接上数据库后确实的确有该学号同学的数据后，进行：Session["user"] = user;

5. 跳转网页的方法：直接使用Response.Redirect("/myhome.aspx");方法进行跳转。

6. 通过服务器控件获取用户输入后，合法用户则创建相应的 Session 变量，并进行合适 的网页重定向，并在 myHome.aspx 网页中显示相应的个人相关记录，如显示当前登录用户 的选修课程。利用相关控件，也可使用数据服务器控件，如 GridView 等进行显示，并编辑等等。判定合法用户依据：未使用数据库/表时，启用静态用户名/口令验证原则：使用数据 库，则通过当前数据库 VS 或 Access 数据库中，相应的表中是否有该用户和密码。

7.创建一个内存中的表接收数据库这张数据表，然后在利用这张数据表得到姓名，显示在文本框中，这一步使用dataTable来找名字而不用GridView是因为GridView绑定的数据我们在显示的时候进行的修改可能影响到选择的目标，所以直接选择最原始的表内容来获取内容

# 五、关键问题及解决方法

1.如何从数据库中读取表的某一个信息

解决方法：利用完整的数据表来读取，

SW\_name.Text = dataTable.Rows[0]["姓名"].ToString();

2.控件的频繁使用

解决方法：为每个空间配置相应的属性如ID和动作Clicked等操作都很繁琐，但是通过集成操作，我们可以利用拆分的双视图进行编辑，这样就可以化繁为简。

3.如何跨页面传递用户名密码等数据

解决方法：利用Session进行跨页面的数据传递进行记录，通过创建session键值对来进行数据的传输，并且对应同一个用户。

4.如何反馈给用户登录验证的提示

解决方法：利用session和ispostback来在main.cs里面发回弹窗给客户端。使用Response.Redirect方法跳转，直接停止了当前的加载，无法将返回的页面发回客户端。

# 六、实验结果

# 七、实验总结

通过本次实验，我学会了使用 VS 创建网站基本结构的方法，学会了一些网站的基本结构构成，以及这些结构是怎么搭建起来的。同时，理解了在 MS SQL 创建数据库、数据表，并能使用 SQL 语句进行表操作，变量的创建和读写等等。

关于数据库的建立：本次实验中是利用了今年选课信息的excel文档来添加到数据库中，形成了一张myhome表。表中学号为int类型，其余的均为varchar类型。然后是数据的连接，在web.conflg文件中配置好，再在vs的服务器资源管理器中连接即可，十分方便。

然后是对控件绑定数据的熟悉，以及对GridView控件的熟悉，对GridView控件的行样式加深了理解，也深刻体会了GridView的强大以及方便。熟悉了一些 HTML 服务器控件，例如数据源 Web 服务器控件、数据 Web 服务器控件和HTML 服务器控件，特别是 FORM 表单服务器控件的设置，通过为html控件增加server标签即可将其变成服务器控件，从而让我学会了如何灵活的使用控件。

最后还学习了一些服务器内置对象的使用，如Request、Response、Session ，ASP.NET 中日期时间类，可以控制时间和日期的显示。特别是对整个基于.NET Framework框架下的asp.net开发网站的整个流程有了深刻的体会，，以及各种跳转、服务端和客户端的关系都有了全新的理解和体会。这对我以后的学习受益颇多、影响深远。

# 附录：源代码（仅仅附上核心代码，不要太多）

## 1 数据库生成SQL语句[包括建库、建表语句；表中记录不需要]

SqlConnection conn;

SqlCommand cmd;

SqlDataReader rd;

string conStr = "Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=NetSchool;Data Source=.";

conn = new SqlConnection(conStr);

conn.Open();

string sun;

sun = "select \* from [NetSchool].[dbo].[STUDENT]where USERID='" + txtUserName.Text + "'and PASSWORD='" + txtPwd.Text + "'";

cmd = new SqlCommand(sun, conn);

rd = cmd.ExecuteReader();

string conStr = "Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=NetSchool;Data Source=.";

conn = new SqlConnection(conStr);

String cmd\_test = "SELECT \* FROM STUDENT\_CLASS where USERID = " + Session["USERID"].ToString();

cmd = new SqlCommand(cmd\_test, conn);

conn.Open();

rd = cmd.ExecuteReader();

## 2 网页login.aspx以及 login.aspx

namespace WebApplication3

{

public partial class login : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void btnLogin\_Click(object sender, EventArgs e)

{

string userID = txtUserName.Text;

string userPwd = txtPwd.Text;

SqlConnection conn;

SqlCommand cmd;

SqlDataReader rd;

string conStr = "Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=NetSchool;Data Source=.";

conn = new SqlConnection(conStr);

conn.Open();

if (rd.HasRows)

{

Session["USERID"] = userID;

Session["PASSWORD"] = userPwd;

Response.Redirect("myhome.aspx");

}

protected void btnReset\_Click(object sender, EventArgs e)

{

txtUserName.Text = "";

txtPwd.Text = "";

}

}

}

## 3 网页myhome.aspx 以及 myhome.aspx.cs

namespace WebApplication3

{

public partial class myhome : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

if (Session["USERID"] != null && Session["PASSWORD"] != null)

{

Label2.Text = "用户名+密码为：" + Session["USERID"].ToString() + Session["PASSWORD"].ToString();

Label3.Text = DateTime.Now.ToLocalTime().ToString();

ListBMyClass.Items.Clear();

ListBMyClass.Items.Add("我所选修的课程");

}

protected void Exit\_Click(object sender, EventArgs e)

{

Session.RemoveAll();

Response.Redirect("login.aspx");

}

protected void btnSearchMyClass\_Click(object sender, EventArgs e)

{

SqlConnection conn;

SqlCommand cmd;

SqlDataReader rd;

string conStr = "Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=NetSchool;Data Source=.";

cmd = new SqlCommand(cmd\_test, conn);

conn.Open();

rd = cmd.ExecuteReader();

while (rd.Read())

{

ListBMyClass.Items.Add(new ListItem(rd["CLASSID"].ToString(), rd["CLASSID"].ToString()));

}

if (rd.HasRows == false)

{

ListBMyClass.Items.Add("你没有选修任何一门课程");

}

rd.Close();

conn.Close();

}

}

}