第 5 章

网络在线考试系统

(Apache+PHP+Ajax+SQL Server 2000 实现)

随着科技的发展,网络技术已经深入到人们的日常生活中,同时带来了教育方式的一次变革。而网络考试则是一个很重要的方向。基于 Web 技术的网络考试系统可以借助于遍布全球的 Internet 进行。因此考试既可以在本地进行,也可以在异地进行,大大拓展了考试的灵活性。并且缩短了传统考试要求老师打印试卷、安排考试、监考、收集试卷、评改试卷、讲评试卷和分析试卷,这个漫长而复杂的过程,使考试更趋于客观、公正。本章介绍了一个具有在线考试、即时阅卷、成绩查询以及考题和考生信息管理等功能的网络在线考试系统。通过阅读本章,读者可以:

- M 掌握网络在线考试的开发过程
- ₩ 掌握应用 SQL Server 2000 创建数据表、数据表
- M 掌握连接 SQL Server 数据库的不同方式
- >> 实时显示时间的两种不同的方法
- m 掌握 Ajax 无刷新技术

5.1 开发背景

随着教育改革的不断深入,近几年来出现了一些新的教育形式,如网上授课、网上考试等。×× ×高校是一所师资力量非常雄厚的学校,随着院校的扩招,学生数量不断增加。为了适应新形式的发展,改变传统的教学模式,方便学生随时随地的对自己的学习情况进行检测,减轻教师的工作压力,现委托其他公司开发网上在线考试系统。这种无纸的网络考试系统,使考务管理突破时空限制,提高考试工作效率和标准化水平,使学校管理者、教师和学生可以在任何时候、任何地点通过网络进行考试。网络在线考试系统已经成为教育技术发展与研究的方向。

5.2 系统分析

5.2.1 需求分析

随着计算机技术的发展和网络技术的日益成熟,通过网络进行信息交流已成为一种快捷的交互方式。在这种网络环境下,学校或考试机构希望通过建立网络在线考试网站来扩大知名度、降低管理成本和减少人力物力的投资,从而为考生提供更全面、更灵活的服务,并全面、准确地对考试进行跟踪和评价。与此同时,考生希望根据自己的学习情况进行测试,并能够得到客观、科学的评价;教务人员希望能够有效地改进现有的考试模式,提高考试效率。

通过实际情况的调查,要求网络在线考试系统具有以下功能:

- ☑ 界面设计美观大方、方便、快捷、操作灵活,树立企业形象。
- ☑ 要求实现在线考试功能,自动核算考试成绩。
- ☑ 要求提供考试时间倒计时功能,使考生实时了解考试剩余时间。
- ☑ 要求系统自动阅卷,保证考试成绩真实有效。
- ☑ 要求考生凭准考证号查询考试成绩,以保证信息安全。
- ☑ 系统运行稳定、安全可靠。
- ☑ 要求对考生及考题信息进行严格管理。

5.2.2 可行性分析

可行性分析的目的就是要用最小的代价在尽可能短的时间内确定问题是否能够解决。通过分析解 法的利弊,来判定系统目标和规模是否现实,系统完成后所能带来的效益是否达到值得去投资开发这 个系统的程度。网络在线考试系统的可行性可从以下两方面考虑。

1. 经济可行性

定期的组织考试是各个院校及时掌握学生学习成绩的有效方式,利用网络在线考试系统,一方面可以节省人力资源,降低考试成本。另一方面,在线考试系统能够快速进行考试和评分,体现出考试的客观与公正性。



2. 技术可行性

开发一个网络在线考试系统,涉及到的最核心的技术问题就是如何实现在不刷新页面的情况下实时显示考试时间及剩余时间,并做到到达考试结束时间时自动提交试卷的功能。通过 Ajax 技术可以轻松实现这些功能,这为网络在线考试系统的开发提供了技术保障。

5.3 系统设计

5.3.1 系统目标

根据前面所作的需求分析及用户的需求可知,网络在线考试系统属于中小型软件,在系统实施后, 应达到以下目标:

- ☑ 采用开放、动态的系统架构,加强用户与网站的动态交互性。
- ☑ 具有空间性。被授权的用户可以在异地登录网络在线考试系统,无须到指定地点进行考试。
- ☑ 操作简单方便、界面简洁美观。
- ☑ 系统提供考试时间倒计时功能,使考生实时了解考试剩余时间。
- ☑ 随机抽取试题。
- ☑ 实现自动提交试卷的功能。当考试时间到达规定时间时,如果考生还未提交试卷,系统将自动交卷,以保证考试严肃、公正地进行。
- ☑ 系统自动阅卷,保证成绩真实准确。
- ☑ 考生可以查询考试成绩。
- ☑ 对考生注册信息进行管理。
- ☑ 系统运行稳定、安全可靠。

5.3.2 系统功能结构

网络在线考试系统的前台功能结构如图 5.1 所示。

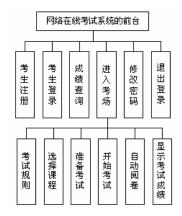


图 5.1 网络在线考试系统的前台功能结构图



网络在线考试系统的后台功能结构如图 5.2 所示。

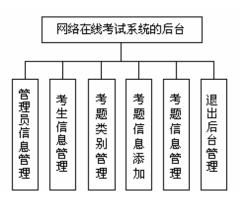


图 5.2 网络在线考试系统的后台功能结构图

5.3.3 系统流程图

为了读者能够更好地学习,下面给出网络在线考试网的网站流程概述及系统流程图。

考生在网络在线考试系统中通过注册为网站用户,登录网站进行相关操作。考生登录后,可以进行在线考试、查询成绩和修改个人密码的操作。在考试前,考生需要阅读考试规则、选择考试套题后开始考试;考试时间结束时,考生提交试卷;提交试卷的同时,系统将自动返回本次考试的考试结果。网站的管理员通过登录模块可以登录到网站的后台系统,对考生信息、考试信息、管理员信息进行管理。网络在线考试系统的系统流程如图 5.3 所示。

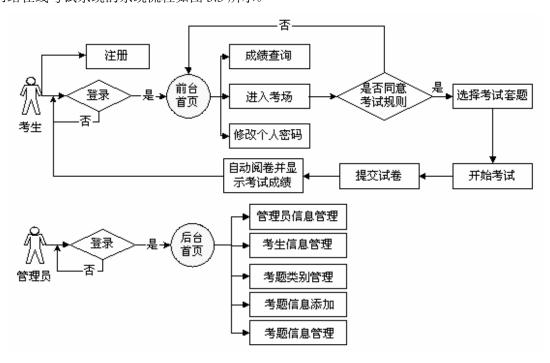


图 5.3 系统流程图



5.3.4 系统预览

网络在线考试系统由多个页面组成,下面仅列出几个典型页面,其他页面参见光盘中的源程序。 考生登录页面如图 5.4 所示,该页面主要用于考生登录,实现在线考试及考试成绩查询等功能。在 线考试页面如图 5.5 所示,该页面用于实现在线答题功能,同时提供了显示考试时间和剩余时间及自动 提交试卷功能。





图 5.4 考生登录页面(光盘\TM\05\online_ks\denglu.php

图 5.5 在线考试页面(光盘\TM\05\online_ks\ksks.php)

考题类别管理页面如图 5.6 所示,该页面主要用于实现显示考题类别的基本信息、添加考题类别和删除考题类别等功能。考题信息管理页面如图 5.7 所示,该页面主要用于管理考试题目信息。





图 5.6 考题类别管理(光盘\TM\05\···\admin\ktlb_gl.php) 图 5.7 考题信息管理(光盘\TM\05\···\admin\ktxx_gl.php)

□ 说明:由于路径太长,因此省略了部分路径,省略的路径是:online_ks\。



5.3.5 开发环境

在开发网络在线考试系统时,该项目使用的软件开发环境如下:

- 1. 服务器端
- ☑ 操作系统: Windows 2003 Server/Linux (推荐)。
- ☑ 服务器: Apache 2.2。
- ☑ PHP 软件: PHP 5.1.6。
- ☑ 数据库: SQL Server 2000。
- ☑ 开发工具: Dreamweaver 8。
- ☑ 浏览器: IE 6.0 及以上版本。
- ☑ 分辨率: 最佳效果 1024×768 像素。
- 2. 客户端
- ☑ 浏览器: IE 6.0 及以上版本。
- ☑ 分辨率: 最佳效果 1024×768 像素。

5.3.6 文件夹组织结构

在编写代码之前,可以把系统中可能用到的文件夹先创建出来(例如,创建一个名为 images 的文件夹,用于保存网站中所使用的图片),这样不但可以方便以后的开发工作,也可以规范网站的整体架构。笔者在网络在线考试系统中设计了如图 5.8 所示的文件夹架构图。在开发时,只需要将所创建的文件保存在相应的文件夹中即可。

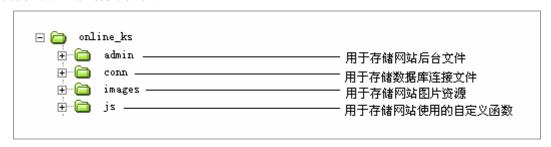


图 5.8 网络在线考试系统的文件夹组织结构

5.4 数据库设计

5.4.1 数据库分析

由于网络在线考试系统对于数据的安全性及完整性要求比较高,并且为了增加程序的适用范围,还要保证系统可以拥有存储足够多数据的能力。SQL Server 2000 是一种高性能的关系型数据库管理系

统,不仅安全可靠而且易于操作,已成为在线事务进程和数据仓库等最好的数据库平台之一。综上所述,本系统采用 SQL Server 2000 数据库。

5.4.2 数据库概念设计

根据以上各节对系统所做的需求分析、系统设计,规划出本系统使用的数据库实体分别为考生信息实体、管理员实体、考题类别实体和考题信息实体。下面将介绍这几个实体的 E-R 图。

1. 考生信息实体

考生信息实体包括编号、考生姓名、联系方式、准考证号、考试成绩、考题类别、考试时间、联系地址、考试状态和考生密码等属性。考生信息实体的 E-R 图如图 5.9 所示。



图 5.9 考生信息实体 E-R 图

2. 管理员实体

管理员实体包括编号、管理员名称和加密密码属性。管理员实体的 E-R 图如图 5.10 所示。

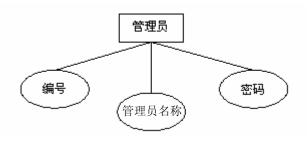


图 5.10 管理员实体 E-R 图

3. 考题类别实体

考题类别实体包括编号和考题类别名称。考题类别实体的 E-R 图如图 5.11 所示。





图 5.11 考题类别实体 E-R 图

4. 考题信息实体

考题信息实体包括编号、考题类别、考试成绩、考题内容、考题答案、考题正确答案及考题所属套题等属性。考试题目实体的 E-R 图如图 5.12 所示。

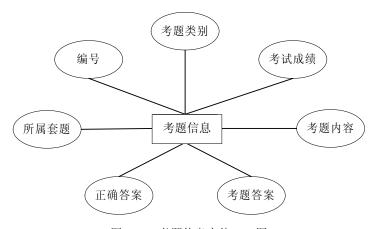


图 5.12 考题信息实体 E-R 图

5.4.3 数据库物理结构

根据 5.4.2 节的数据库概念设计,可以创建与实体对应的数据表。为了使读者对本系统的数据库的结构有一个更清晰的认识,下面给出数据库中所包含的数据表的结构图,如图 5.13 所示。

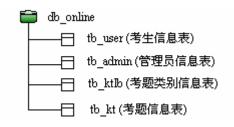


图 5.13 db_online 数据库所包含数据表的结构图

本系统共包含4张数据表,下面进行详细介绍。

1. tb user (考生信息表)

考生信息表用来保存考生信息。该数据表结构如图 5.14 所示。





图 5.14 考生信息表结构

2. tb_admin (管理员信息表)

管理员信息表用来保存管理员的用户名和密码。该数据表结构如图 5.15 所示。



图 5.15 管理员信息表结构

3. tb_ktlb (考题类别信息表)

考题类别信息表用来保存考题类别。该数据表结构如图 5.16 所示。



图 5.16 考题类别信息表结构

4. tb_kt (考题信息表)

考题信息表用来保存考试题目信息和考题答案等相关信息。该数据表结构如图 5.17 所示。



b_kt								
	列名	数据类型	长度	允许至	你况	标识种子	描述	Ų
8	kt_i d	int	4		V	1	自动编号	
	kt_lb	varchar	50				考题类别	
	kt_lx	int	4				考题类型	
	kt_fs	int	4				考试成绩	
	kt_nr	varchar	2000				考题内容	
	kt_daan	varchar	2000	V			考题答案	
	kt_zqdaan	varchar	2000				考题正确答案	
	kt_small_lb	varchar	50				考题所属套题	

图 5.17 考题信息表结构

5.5 前台首页设计

5.5.1 前台首页概述

考生通过"考生登录"模块的验证后,可以登录到网络在线考试系统的前台首页。前台首页主要用于实现前台功能导航,该页面主要包括考生注册、考生登录、修改密码、成绩查询、进入考场和退出信息6个导航链接。

下面看一下本案例中提供的前台首页,该页面在本书光盘中的路径为 $\TM\05\$ nline_ks $\$ index.php,如图 5.18 所示。



图 5.18 网络在线考试系统前台首页



5.5.2 前台首页技术分析

PHP 连接 SQL Server 数据库的方法有两种: 一是通过 ADO 方式连接; 二是通过 mssql_connect() 函数连接。下面分别介绍这两种连接方式。

1. 通过 ADO 方式连接 SQL Server 数据库

利用 ADO 方式建立与 SQL Server 数据库的连接,与使用 ADO 方式建立与 Access 数据库连接的 方法类似,建立连接是在本实例的 conn.php 文件中实现的。其代码如下:

<?php

\$conn=new com("adodb.connection");

\$connstr="provider=sqloledb;data source=localhost;uid=sa;pwd=;database=db_online";

\$conn->open(\$connstr);

?>

2. 通过 mssql_connect()函数连接 SQL Server 数据库

通过 mssql_connect()函数建立与 SQL Server 数据库的连接首先应该配置 php.ini 文件,将 extension=php_mssql.dll 前的注释去掉,然后重新启动 Apache 服务器。通过 mssql_connect()函数建立与数据库连接的代码如下:

<?php

• \$conn=mssql_connect("localhost", "sa", "");

//建立与SQL Server数据库的连接

mssql_select_db("db_online ",\$conn);

//选择数据库

?>

✔ 代码贴士

● mssql_connect()函数: 打开一个到 SQL Server 服务器的连接。如果成功则返回一个 SQL Server 连接标识; 失败则返回 false。

resource mssql_connect ([string servername [, string username [, string password [, bool new_link]]]])

参数 servername 指 SQL Server 服务器;参数 usernamer 指用户名;参数 password 指用户密码;参数 new_link 指连接标识。

❷ mssql_select_db()函数:选择 SQL Server 数据库。如果成功则返回 true;失败则返回 false。

bool mssql_select_db(string database_name [,resource link_identifier])

参数 database_name 指数据库名称;参数 link_identifier 指连接标识。

5.5.3 前台首页的实现过程

网络在线考试系统前台首页主要实现了考生登录功能,考生通过准考证号和密码进行登录。 考生登录页面中所涉及到的重要表单元素如表 5.1 所示。



名 元素类型 重要属性 表单 form1 method="post" action="denglu_ok.php" form online_number text id="online_number" size="20" 准考证号码 考生密码 online_pass password id="online_pass" size="20" "提交"按钮 value="提交" Submit submit

表 5.1 考生登录页面所涉及的重要表单元素

考生在考生登录页面录入准确的准考证号和密码后,单击"提交"按钮,提交表单信息到数据处理页,应用 mssql_query()函数检索考生是否成功登录。如果查询结果为真,则将准考证号存储在 session变量中:否则弹出提示信息。代码如下:

例程01 代码位置: 光盘\TM\05\online_ks\denglu_ok.php

```
<?php
                                                                  //连接数据库文件
include("conn/conn.php");
$online_number=$_POST[online_number];
                                                                  //获取准考证号
                                                                  //获取考生密码
$online_pass=$_POST[online_pass];
/* ***************************检索考生的准考证号和密码是否正确*
\$query = \pmb{mssql\_query} ("select * from tb\_user where online\_number = '\$online\_number' and online\_pass = '\$online\_pass''');
if(mssql_num_rows($query)>0){
                                                                  //如果查询结果为真
                                                                  //将准考证号存储在session变量中
   session_register(online_number);
   echo "<script> alert('登录成功!'); window.location.href='index.php?online=考试规则';</script>";
}else{
                                                                  //如果没有检索到数据
   echo "登录失败!";
                                                                  //弹出提示信息
?>
```

《》 代码贴士

● mssql_query: 向与指定的 SQL Server 服务器中的当前活动数据库发送一条 SQL 查询。语法如下:

resource mssql_query (string query [, resource link_identifier])

参数 query 为传入的是 SQL 的指令; 参数 link_identifier 传入的是由 mssql_connect()函数或 mssql_pconnect()函数返回的连接号,如果省略该参数,则会使用最后一个打开的 MySQL 数据库连接。

- ❷ mssql_num_rows: 获取结果集中行的数目。值得注意的是,该函数仅对 SELECT 语句有效。
- ❸ session_register: 在整个域范围内增加一个 session 变量。语法如下:

boolean session_register(string name)

参数 name 用于指定新 session 变量的名称。

5.6 考生信息模块设计

5.6.1 考生信息模块概述

考生信息模块主要包括考生注册、考生登录、修改密码3个功能。考生首先要注册为网站用户,



然后才被授权登录网站进行一系列操作的权限;登录后考生还可以修改个人的密码。考生信息模块的系统流程如图 5.19 所示。

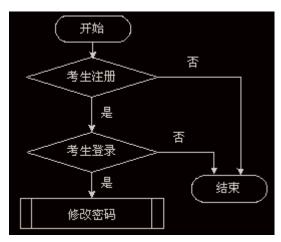


图 5.19 考生信息模块的系统流程图

5.6.2 考生信息模块的技术分析

在考生注册信息模块中应用到 Ajax 无刷新技术获取考生的准考证号码和密码。在介绍该功能的实现过程之前,先对 Ajax 技术中的 XmlHttpRequest 对象的属性和方法进行详细的讲解。

XmlHttpRequest 对象是 Ajax 技术的核心,有关该对象的属性和方法的详细介绍如表 5.2 和表 5.3 所示。

属性	描述
readyState	返回当前的请求状态
onreadystatechange	当readyState属性改变时就可以读取此属性值
status	返回Http状态码
responseText	将返回的响应信息用字符串表示
ResponseBody	返回响应信息正文,格式为字节数组
ResponseXML	将响应的domcoment对象解析成XML文档并返回

表 5.2 XmlHttpRequest 对象的属性

表53	XmlHttpRequest	的方法
æ つ.ა	Aminubrequest	的刀法

方 法	描述
Open	初始化一个新请求
Send	发送请求
GetAllReponseHeaders	返回所有Http头信息
GetResponseHearder	返回指定的Http头信息
SetRequestHeader	添加指定的Http头信息
Abort	停止当前的Http请求



下面对 XmlHttpRequest 对象的常用属性和方法进行详细介绍。

1. readystate 属性

readystate 属性用于返回当前的请求状态,请求状态共有5种,如表5.4所示。

表 5.4 readystate 属性值

属性值	描述
0	表示尚未初始化,即未调用open()方法
1	建立请求,但还未调用send()方法发送请求
2	发送请求
3	处理请求
4	完成响应,返回数据

2. status 属性

status 属性用于返回 Http 状态码,常用 Http 状态码如表 5.5 所示。

表 5.5 Http 状态码

属性名	描述
200	操作成功
404	没有发现文件
500	服务器内部错误
505	服务器不支持或拒绝请求中指定的Http版本

3. responseText 属性

responseText 属性将返回的响应信息用字符串来表示。在默认情况下,返回的响应信息的编码格式为 utf-8。

4. responseXML 属性

responseXML 属性用于将响应的 domcoment 对象解析成 XML 文档并返回。

5. open 方法

open 方法用于初始化一个新的请求。

语法:

open(String method, String url, Boolean asyn, String user, String password)

其中, method 和 url 是必选参数, asyn、user 和 password 是可选参数。open 方法各参数如表 5.6 所示。

表 5.6 open 方法参数

参数名称	描述
method	此参数指明了新请求的调用方法,其取值有get和post
url	表示要请求页面的url地址。格式可以是相对路径、绝对路径或网络路径



续表

参数名称	描述
asyn	说明该请求是异步传输还是同步传输,默认值为true(允许异步传输)
user	服务器验证时的用户名
password	服务器验证时的密码

6. send 方法

send 方法用于发送请求到服务器。

语法:

send(body)

如果没有要发送的内容,则 body 可以省略或为 Null。

7. GetAllReponseHeaders 方法

GetAllReponseHeaders 方法用于获取响应的所有 Http 头信息。获取到的信息是按照"名称/键值"排列的,信息之间是用":"冒号进行分隔的。

语法:

GetAllReponseHeaders ()

8. GetResponseHeader 方法

GetResponseHeader 方法是获取响应中指定的 Http 头信息。

语法:

GetResponseHeader(String head)

9. SetRequestHeader 方法

SetRequestHeader 方法用于添加一个 Http 头信息。

语法:

SetRequestHeader(name,value)

其中, name 表示 Http 头名称, value 表示信息内容。

10. Abort 方法

Abort 方法用于取消一个请求。

语法:

Abort()

5.6.3 考生注册的实现过程

■ 考生注册使用的数据表: tb_user

在网络在线考试系统首页,单击"考生注册"超链接,即可进入考生注册页面,输入个人的真实资料后,单击"注册"按钮,系统将根据输入的个人资料自动生成准考证号和考生密码,如图 5.20



所示。



图 5.20 考生登录页面的运行结果

🔲 说明:考生需要通过准考证号和密码登录网络在线考试的核心模块,即在线考试及成绩查询等。

在考生注册页面 zhuce.php 页中实现考生注册信息的提交,并且通过 Ajax 的页面无刷新技术获取 考生的准考证号和密码。zhuce.php 页中的关键代码如下:

例程02 代码位置: 光盘\TM\05\online_ks\zhuce.php

```
<!---调用一个javascript脚本文件-->
<script type="text/javascript" src="js/zhuce_js.js"></script>
<!---通过onclick事件调用一个自定义函数-->
<input type="button" name="Submit" value="注册" onClick="process()">
<!---输出一个div标签中的值-->
<div id="divMessage" />
```

在 zhuce_js.js 文件中首先定义一个 createXmlHttpRequestObject()对象,并获取 XMLHttpRrequest 对象,然后定义 xmlHttp 用来存储将要使用的 XMLHttpRrequest 对象。关键代码如下:

例程03 代码位置: 光盘\TM\05\online_ks\js\zhuce_js.js



然后使用 XMLHttpRequest 对象创建异步 Http 请求,定义函数 process(),对表单中提交的数据进行判断,并且获取表单中输入的信息,在服务器端执行 zhuce_ok.php 文件,向服务器发送请求。关键代码如下:

例程04 代码位置: 光盘\TM\05\online_ks\js\zhuce_js.js

```
//使用XMLHttpRequest对象创建异步HTTP请求
    function process(){
       if(form1.username.value==""){
                                                   //如果考生姓名为空
           alert("请输入姓名!");
                                                   //则弹出提示
               form1.username.select();
                                                   //返回焦点到考生姓名文本框
               return(false);
                                                   //如果考生联系电话为空
       if(form1.tel.value==""){
           alert("请输入电话号码!");
                                                   //则弹出提示
           form1.tel.select();
                                                   //返回焦点到考生联系方式文本框
           return(false);
       if(checkphone(form1.tel.value)!=true){
                                                   //如果考生联系方式格式不正确
           alert("您输入的电话号码的格式不正确!");
                                                   //则弹出提示
           form1.tel.select();
                                                   //返回焦点到考生联系方式文本框
           return(false);
       if(form1.address.value==""){
                                                   //如果考生联系地址为空
           alert("请输入联系地址!");
                                                   //则弹出提示
                                                   //返回焦点到考生联系地址文本框
           form1.address.select();
           return(false);
       if(xmlHttp.readyState==4 || xmlHttp.readyState==0){
                                                   //在xmlHttp对象不忙时进行处理
       names = document.getElementById("username").value;
                                                   //获取用户在线表单中输入的姓名
       tels = document.getElementById("tel").value;
                                                   //获取用户在线表单中输入的电话
       addresss =document.getElementById("address").value;
                                                   //获取用户在线表单中输入的地址
       xmlHttp.open("GET","zhuce_ok.php?online_user="+names+"& online_tel="+tels+"&
                                                           //在服务器端执行zhuce_ok.php
online_address="+addresss,true);
       xmlHttp.onreadystatechange=handleServerResponse;
                                                           //定义获取服务器端响应的方法
```

```
//向服务器发送请求
   xmlHttp.send(null);
   }else
                                                     //如果服务器忙,1秒后重试
   setTimeout('process()',1000);
最后获取从服务器端返回的消息。关键代码如下:
例程05 代码位置: 光盘\TM\05\online_ks\js\zhuce_js.js
//当收到服务器端的消息时自动执行
function handleServerResponse(){
   if(xmlHttp.readystate==4){
                                                     //在处理结束时进入下一步
                                                     //状态为200表示处理成功结束
       if(xmlHttp.status==200){
          xmlResponse=xmlHttp.responseXML;
                                                     //获取服务器端发来的XML信息
          xmlDocumentElement=xmlResponse.documentElement;
                                                     //获取XML中的文档对象(根对象)
                                                     //获取第一个文档子元素的文本信息
          helloMessage=xmlDocumentElement.firstChild.data;
          //使用从服务器端发来的消息更新客户端显示的内容
          document.getElementById("divMessage").innerHTML='<i>'+helloMessage+'</i>';
          setTimeout('process()',1000);
                                                     //重新开始
       }else{
          //如果Http的状态不是200表示发生错误
          alert("There was a problem accessing the server:"+xmlHttp.statusText);
       }
```

zhuce_ok.php 文件在服务器端被执行,将获取到的用户信息和随机生成的准考证号、密码添加到数据库中,并且将准考证号和密码返回到考生注册页面中。关键代码如下:

例程06 代码位置: 光盘\TM\05\online_ks\zhuce_ok.php

```
//创建一个XML格式输出
    header('Content-Type: text/xml');
    echo '<?xml version="1.0" encoding="gb2312" standalone="yes" ?>';
                                                                //创建XML头
                                                                //创建<response>元素
    echo '<response>';
    echo $online_user=$_GET[online_user];
                                                                //获取考生姓名
                                                                //获取考生联系电话
    $online_tel=$_GET[online_tel];
    $online_address=$_GET[online_address];
                                                                //获取考生联系地址
                                                                //生成6位随机的准考证号码
    $online_number=substr(mt_rand(100000,999999),0,6);
    $online_pass=substr(mt_rand(100000,999999),0,6);
                                                                //生成6位随机的考生密码
    include("conn/conn.php");
    $query=mssql_query("insert into tb_user(online_user,online_tel,online_address,online_number,online_pass)
values('$online_user', '$online_tel', '$online_address', '$online_number', '$online_pass')");
    if($query==true){
       echo $online_user;
       echo "用户注册成功,这是您的准考证号码$online_number.和密码$online_pass.";
       echo '</response>';
                                                                //关闭<response>元素
```



✔ 代码贴士

● \$_GET 预定义变量: PHP 使用\$_GET 预定义变量自动保存通过 get 方法传过来的值,使用格式为:

\$_GET[name]

❷ substr()函数: 从指定的字符串中按照指定的位置截取一定长度的字符。语法如下:

string substr(string str,int start,int length)

参数 str 指定字符串对象。参数 start 指定开始截取字符串的位置。如果参数 start 为负数,则从字符串的末尾开始 截取。参数 length 是可选参数,指定截取字符的个数,如果 length 为负数,则表示取到倒数第 length 个字符。

5.6.4 单元测试

在应用 Ajax 技术实现在线考试的用户注册模块后,为保证程序能够正常运行,必须对其进行测试。运行考生注册模块,运行结果如图 5.21 所示。



图 5.21 考生注册模块运行出现的错误

从弹出的错误信息来看,是程序中与 JavaScript 脚本有关的内容出现问题,经过反复测试和查找发现,问题不是出现在 zhuce_js.js 文件中,因为在该文件中定义的自定义函数 process()是可以运行的,所以问题应该是出现在对服务器返回的信息进行输出中,即 zhuce_ok.php 文件中。对该文件进行分析发现,原来在应用 Ajax 技术中,服务器端返回的消息采用的是 XML 格式。可是在客户端接收 XML 文档时却忽略了"header('Content-Type: text/xml');"头部信息的使用,从而导致客户端用于解析 XML的 API 抛出一个错误信息。只有在设置头部信息(header('Content-Type: text/xml');)后,客户端才可以正确的接收到服务器端传递的 XML 文档信息。正确的程序代码如下:

<?php
//创建一个XML格式输出
header('Content-Type: text/xml'); //标记输出的是XML文档</pre>



//创建XML头 echo '<?xml version="1.0" encoding="gb2312" standalone="yes" ?>'; ?>

5.7 在线考试模块设计

5.7.1 在线考试模块的概述

在线考试模块的主要功能是允许考生在网站上针对指定的课程进行考试。在该模块中,考生首先需要登录到本系统中,阅读考试规则,在同意所列出的考试规则后,才能选择考试类别和套题,然后进入考试页面进行答题,当考生提交试卷或者到达考试结束时间时,系统将自动对考生提交的试卷进行评分,并给出最终考试成绩。在线考试模块的系统流程如图 5.22 所示。



图 5.22 在线考试模块的系统流程图

5.7.2 在线考试模块的技术分析

在线考试模块中最核心的功能是如何输出考题、如何对提交的答案进行判断,并且将答案输出到 当前页中,这是在线考试最关键之处。下面就讲解一下该技术的实现方法,由于上述 3 方面的内容都 是在同一页面中显示,所以在技术实现过程中这 3 个方面的内容是以一个整体形式出现。这里以单选 题为例进行讲解。



首先以上一页表单中提交的值为条件,执行查询语句,通过 while 循环语句输出查询结果;然后根据输出的查询结果创建单选按钮组,生成考题;最后,将考题答案提交到本页,对提交的答案与数据库中存储的正确答案进行比较,并且将答案输出到本页中。单选题输出、答案判断和考题答案输出的关键代码如下:

例程07 代码位置: 光盘\TM\05\online_ks\ksks.php

```
<?php
/*根据提交的变量执行查询语句,从数据库中读取数据,指定考题的类型为0,表示为单选题*/
$query(0=mssql_query("select * from tb_kt where kt_lb='$kt_lbes' and kt_lx='0' and kt_small_lb='$kt_small_lb'");
x=1;
                                    //定义变量$x=1,用于输出题号
$fen0=0:
                                    //定义变量$fen0=0,用于考试分数的计算
                                    //执行while循环语句
while($myrow0=mssql_fetch_array($query0)){
<!--- 输出单选题的题号、考题内容以及该道体的分数--->
   <?php echo $x.".".$myrow0["kt_nr"]?> 
  <?php echo $myrow0[kt_fs];?>
<?php
   /*****获取考题提交的答案,将提交的考题答案进行分割,并赋给变量$str0******/
                                    //将数据库中的考题答案以*进行分割,并写入到数组中
  $array0=explode("*",$myrow0["kt_daan"]);
  if($_POST[Submit]!=""){
                                    //判断表单提交是否为空
     for($a=0;$a<count($array0);$a++){
                                    //循环输出数组中的答案
        if($array0[$a]!=""){
                                    //数组中的值不为空
           if($array0[$a]==$_POST[$myrow0[kt_id]]) {//如果数组中的值与表单提交的值相等
              $str0=$_POST[$myrow0[kt_id]];
                                   //将表单提交的值赋给变量$str0
        }
     }
  /*****************************
  for($a=0;$a<count($array0);$a++){
                                    //执行for循环语句,根据考题答案来决定循环的次数
    if($array0[$a]!=""){
                                    //判断如果考题答案不为空,则执行下面的内容
  
     <!--- 创建单选按钮组,用于提交答案--->
     <input type="radio" name="<?php echo $myrow0[kt id];?>" value="<?php echo $array0[$a];?>">
     <?php echo $array0[$a];?>
    
  if($_POST[$myrow0[kt_id]]==true){
                                       //判断表单提交的值是否为空
                                       //如果变量$a==0则输出下面的内容
                                       //判断提交的答案是否与数据库中的正确答案相同
        if($myrow0["kt zqdaan"]==$str0){
           //如果提交的答案正确,则输出下面的内容
           echo "您输入的答案 ";
           echo "<font color='#FF0000'>".substr($str0,0,1)."</font>";
                                                //获取答案的选项
```

```
echo " 正确  分数:";
echo "<font color='#FF0000'>".$myrow0[kt_fs]."</font>"; //输出考题的分数
$fen0+=$myrow0["kt_fs"]; //对考题的分数进行合算
}else{ //如果提交的答案不正确则输出下面的内容
echo "您输入的答案&nbsp;";
echo "<font color='#FF0000'>".substr($str0,0,1)."</font>";
echo "&nbsp;错误&nbsp;&nbsp;";
echo "正确答案:&nbsp;<font color='#FF0000'>".substr($myrow0[kt_zqdaan],0,1)."</font>";
}

}

?>
</rr>
//实现考题题号的循环
}
$xx++; //实现考题题号的循环
```

5.7.3 应用 Ajax 在线答题的实现过程

在线答题是本项目中最核心的技术,也是本系统开发的最终目的。在线答题模块的主要功能是实现在线考试,在当前页面中输出考题答案和考试成绩,并且采用计时控制考试时间,如果考试超过考试时间则系统自动提交考题答案,并给出考试成绩。

考试时间计时和剩余时间的计算是应用 Ajax 技术来实现的,有关该技术的实现方法可以参考本章 开发技巧与难点分析中的通过 Ajax 技术实现计时与显示剩余时间一节。这里不讲解。

在线答题技术中对单选题的处理方法已经在技术分析中进行了详细的讲解,这里就不再赘述,下面讲解在线答题中多选题的处理方法,其实现的原理与单选题是相同的,只是在输出考题的答案时使用的是复选框,不再是单选按钮组,并且在对复选框名称进行设置时增加一个变量来对不同考题的答案进行区分。首先以上一页表单中提交的值为条件,执行查询语句,通过 while 语句输出查询结果;然后根据输出的查询结果创建复选框,生成考题;最后,将考题答案提交到本页,对提交的答案与数据库中存储的正确答案进行比较,并且将答案输出到本页中。

首先从数据库中读取数据,定义输出考题时应用到的变量值,并且输出考题的编号和内容。关键 代码如下:

例程08 代码位置: 光盘\TM\05\online_ks\ksks.php

```
<?php
  //根据表单提交的值执行查询语句,从数据库中读取符合条件的数据
  $query1=mssql_query("select * from tb_kt where kt_lb='$kt_lbes' and kt_lx='1' and kt_small_lb='$kt_small_lb'");
  y=1;
                                          //定义变量$v用于生成考题编号
  $fen1=0:
                                          //定义变量$fen1用于分数的统计
                                          //定义变量$ii用于控制考题答案输出的次数
  $ii=0;
   while($myrow1=mssql_fetch_array($query1)){
                                           //循环输出数据
2>
<!--
                   ------输出考题编号和考题的内容------
     <?php echo $y.".".$myrow1["kt_nr"]?> 
   <?php echo $myrow1[kt_fs];?>
```

然后,对表单中提交的答案进行处理,与数据库中存储的正确答案相比较,如果答案正确则将表 单提交的答案写入到一个空是数组中。代码如下:

例程09 代码位置: 光盘\TM\05\online_ks\ksks.php

```
<?php
       $array_a1=array();
                                         //定义一个空数组
  //将数据库中考题的答案以*进行分割,存储到数组$array1中
  $array1=explode("*",$myrow1["kt_daan"]);
                                         //判断表单提交的值是否为空
  if($_POST[Submit]!=""){
     for($i=0;$i<count($array1);$i++){
                                         //循环读取数组$array1中的考题答案
        if($array1[$i]!=""){
                                         //判断数组中的答案是否为空
           //对表单中提交的答案与数据库中的答案进行比较
           if($array1[$i]==$_POST[$myrow1[kt_id]."-".$i]) {
              //如果答案相同则将表单提交的答案写入到空数组$array_a1中
              array_push($array_a1,$_POST[$myrow1[kt_id]."-".$i]);
      }
```

✔ 代码贴士

● explode()函数:将字符串依指定的字符串或字符 separator 切开。

array explode(string separator, string string, [int limit])

返回由字符串组成的数组,每个元素都是 string 的一个子串,它们被字符串 separator 作为边界点分隔出来。如果 设置了 limit 参数,则返回的数组包含最多 limit 个元素,而最后那个元素将包含 string 的剩余部分;如果 separator 为空 字符串 (""), explode()函数将返回 false;如果 separator 所包含的值在 string 中找不到,那么 explode()函数将返回包含 string 单个元素的数组;如果参数 limit 是负数,则返回除了最后的-limit 个元素外的所有元素。

❷ array_push()函数:将数组当成一个栈,并将传入的变量压入该数组的末尾。该数组的长度将增加入栈变量的数 目。返回数组新的单元总数。

int array_push (array array, mixed var [, mixed ...])

参数 array: 指输入的数组; 参数 var: 必要参数, 用来压入数组的。

接着创建考题提交的复选框,根据考题的 ID 和变量\$I 定义复选框的名称。代码如下:

例程10 代码位置: 光盘\TM\05\online_ks\ksks.php

```
<?php
  $str1=implode("*",$array_a1);
                        //以*来分割数组$array_a1中的数据,返回值为变量$str1
                         //根据数据库中存储的答案执行循环语句
  for($i=0;$i<count($array1);$i++){
     if($array1[$i]!=""){
                         //判断答案不为空
?>
  
  <input type='checkbox' name='<?php echo $myrow1[kt_id]."-".$i?>' value='<?php echo $array1[$i];?>'> <?php echo</pre>
```

再次对提交的答案进行判断,其中主要应用的是 explode()、list()和 substr()函数来实现答案的判断和输出。程序代码如下:

例程11 代码位置: 光盘\TM\05\online_ks\ksks.php

```
if($_POST[$myrow1[kt_id]."-".$i]==true){
                                          //判断提交的答案是否为真
      if($ii==0){
                                          //判断变量$ii的值是否为0,控制答案输出的次数
                                          //判断表单提交的答案是否与数据库中的正确答案相等
         if($myrow1["kt_zqdaan"]==$str1){
             /* *************如果答案正确,则输出提交的答案,并且给出该题的得分************ */
             echo "您输入的答案 ";
             $arrayesg=explode("*",$str1);
                                                             //以*来分隔提交的答案
             for($gg=0;$gg<count($arrayesg);$gg++){
                                                             //循环读取答案的内容
                list($name,$valuesg)=each($arrayesg);
                                                             //输出答案
                echo "<font color='#FF0000'>".substr($valuesg,0,1)."</font>";
                                                             //截取答案的内容
             echo " 正确  分数:";
             echo "<font color='#FF0000'>".$myrow1[kt_fs]."</font>";
                                                             //统计考题的分数
             fen1+=myrow1["kt_fs"];
         }else{
         /* ********如果输出的答案错误,则提示您输入的答案错误,并给出正确的答案***********/
             echo "您输入的答案 ";
             $arrayes=explode("*",$str1);
             for($g=0;$g<count($arrayes);$g++){
                list($name,$values)=each($arrayes);
                echo "<font color='#FF0000'>".substr(values,0,1)."</font>";
             echo " 错误  正确答案: ";
             $arrayes=explode("*",$myrow1[kt_zqdaan]);
             for($g=0;$g<count($arrayes);$g++){}
                list($name,$values)=each($arrayes);
                echo "<font color='#FF0000'>".substr($values,0,1)."</font>";
      }$ii=1;
                                          //控制每道题的答案只输出一次数
  ?>
<?php } }
                                          //控制每道题的题号
   $y++;
```



《 代码贴士

● count()函数: 返回数组中的单元数目,用来计算数组中值的个数。

int count (mixed array [, int mode])

参数 array: 必要参数,输入的数组;参数 mode: 可选参数,此参数的默认值是 0。

❷ list()函数: 把数组中的值赋给一些变量。与 array()函数类似,这不是真正的函数,而是语言结构。list()函数仅能用于数字索引的数组,且数字索引从 0 开始。

```
void list ( mixed ...)
```

参数 mixed 为被赋值的变量名称。

each()函数: 返回数组中当前指针位置的键名和对应的值,并向前移动数组指针。

array each (array array)

参数 array 为输入的数组。

5.7.4 分数统计和成绩保存的实现过程

在线考试模块中不但完成考试的答题和判卷的操作,而且可以直接对考试的分数进行统计,获取 考生的最终得分,将考生的成绩保存到数据库中,并且更改数据库中考生的考试信息,说明该考生已 经完成本次考试,不可以再进行本类别的考试。考试分数统计和存储考试成绩的关键代码如下:

例程12 代码位置: 光盘\TM\05\online_ks\ksks.php

```
<?php
     zf=fen0+fen1+fen2+fen3;
                                    //考试题分数统计
                                    //输出字符串
     echo "您的总成绩是:";
                                    //输出考试成绩
     echo $zf;
  ?>
  <?php
                 if($Submit=="提交"){
                                    //首先判断考生是否提交考题的答案
  $data=date("Y-m-d H:i:s");
                                    //获取考试的时间
                 $grade="update tb_user set online_grade='$zf',online_subject='$_POST[kt_lbes]',online_pt='1',online_date='$data' where
online_number='$_SESSION[online_number]'";
  $grade_result=mssql_query($grade);
  }
  ?>
```

5.7.5 单元测试

在完成在线考试模块的设计后对该模块进行测试,发现在执行多选题提交的过程中,没有任何值返回。但是在执行单选题提交时可以正常运行,查看该文件的源文件发现,原来错误出现在复选框名



称的设置上,所有复选框都是相同的值,其运行结果如图 5.23 所示。

图 5.23 查看源文件的结果

在设置复选框的名称时必须为每一个题设置不同的复选框名称,否则将不能获取到表单提交的数据,这就是多选题与单选题进行输出和判断的主要区别,多选题中有多个答案,而单选题只有一个,所以单选题的单选按钮组的名称可以直接使用考题的 ID 进行命名,但是多选题必须在考题 ID 的基础之上在定义一个变量来区分每个复选框的名称。在程序中,错误的复选框名称定义的代码如下:

<input type="radio" name="<?php echo \$myrow0[kt_id];?>" value="<?php echo \$array0[\$a];?>">

正确的复选框名称定义应该是在考题 ID 的基础之上加一个变量\$I,程序代码如下:

<input type='checkbox' name='<?php echo \$myrow1[kt_id]."-".\$i?>' value='<?php echo \$array1[\$i];?>'>

对程序代码进行修改, 重新保存, 程序可以正常运行。

5.8 后台首页设计

5.8.1 后台首页概述

网络在线考试系统的后台首页是管理员对网站信息进行管理的首页面。在该页面中,管理员可以 清楚地了解网站后台管理系统包含的基本操作。网络在线考试网后台首页包含的主要模块如下:

- ☑ 管理员信息管理:主要用于修改管理员信息。
- ☑ 考生信息管理:主要包括查看注册考生信息列表和考生信息查询、考试成绩查询和删除已注 册的考生信息。
- ☑ 考题类别管理:主要包括查看考题类别列表、添加考题类别信息和删除考题类别信息。
- ☑ 考题信息添加:主要用于添加为各类套题添加单选题、多选题、问答题和论述题,并设置每题的分数及内容。
- ☑ 考题信息管理:主要包括查看考题类别列表、修改套题信息和删除套题信息。
- ☑ 包括杳看考试题目列表、添加考试题目、修改考试题目和删除考试题目。
- ☑ 退出管理:主要用于退出后台管理系统。



下面看一下本案例中提供的后台首页,该页面在本书光盘中的路径为\TM\05\online_ks\admin\index.php,如图 5.24 所示。



图 5.24 网络在线考试系统的后台首页

5.8.2 后台首页技术分析

网络在线考试后台应用 switch 语句实现的一种类似于框架的网页嵌套技术。为了能够更好地理解这个技术,下面进行详细介绍。

switch 语句一行接一行地执行(实际上是语句接语句),开始时没有代码被执行。仅当一个 case 语句中的值和 switch 表达式的值匹配时,PHP 才开始执行语句,直到 switch 的程序段结束或者遇到第一个 break 语句为止。如果不在 case 的语句段最后写上 break 的话,PHP 将继续执行下一个 case 中的语句段。

switch 语句的语法如下:

```
switch(expr) {
    case value : statement; break;
    .....

default : statement; break;
}
```



switch 语句的参数说明如表 5.7 所示。

表 5.7 switch 语句的参数说明

参 数	说明
expr	必要参数。定义判断的条件表达式
statement	必要参数。满足条件后,循环执行的语句

5.8.3 后台首页的实现过程

网络在线考试后台主要应用了 switch 语句和 include 包含语句实现了类似于框架的网页嵌套技术。 其中,应用 switch 语句,根据变量标识\$htgl 提交的值进行判断,应用 include 包含语句调用不同功能 模块的脚本文件。代码如下:

例程13 代码位置: 光盘\TM\05\online_ks\admin\index.php

```
<?php
switch($htgl){
                                               //判断后台超链接名称
   case "考生信息管理":
                                               //如果名称为"考生信息管理"
      include("ksxx_gl.php");
                                               //则应用include()语句调用ksxx_gl.php文件
                                               //退出语句段的执行
   case "考题类别管理":
                                               //如果名称为"考题类别管理"
      include("ktlb_gl.php");
                                               //则应用include()语句调用ktlb_gl.php文件
      break:
                                               //退出语句段的执行
   case "考题信息添加":
                                               //如果名称为"考题信息添加"
                                               //则应用include()语句调用ktxx_tj.php文件
      include("ktxx_tj.php");
      break;
                                               //退出语句段的执行
   case "考题信息管理":
                                               //如果名称为"考题信息管理"
      include("ktxx_gl.php");
                                               //则应用include()语句调用ktxx_gl.php文件
                                               //退出语句段的执行
      break;
}
```

5.9 考题信息管理模块设计

5.9.1 考题信息管理模块的概述

考题信息管理模块主要包括查询考题信息、添加考题信息、修改考题信息和删除考题信息等 4 个功能。考题信息管理模块的框架如图 5.25 所示。





图 5.25 考题信息管理模块的框架图

5.9.2 考题信息管理模块的技术分析

在实现考题信息管理模块时,为了更好地管理,因此把考题类别单独存储于一个数据表 tb_lb 中,这样在录入考题信息时就可以把考题类别以下拉列表的形式从数据库中读取出来。这种从下拉列表中动态显示数据表某列的字段值的方法,不仅方便管理员更快捷、灵活地操作网络在线考试系统,更大大提高了工作效率,达到事半功倍的效果。

下拉列表是一种最节省空间的数据显示方式,正常状态下只能看到一个选项,单击控制按钮后,可以显示一定数量的选项,如果超出这个数量,会自动显示滚动条,管理员可以通过拖动滚动条来选择各选项。

下面介绍从下拉列表中动态显示数据表某列的字段值的方法。

首页, 创建与数据库的连接。代码如下:

```
<?php
$id=mssql_connect('localhost','sa',");
mssql_select_db("db_online ",$id);
?>
```

然后应用下拉列表框和 select 查询语句相结合实现在下拉列表中显示数据表中的 online_ktlb 字段的值,通过 while 循环语句进行输出。代码如下:

5.9.3 考题信息添加的实现过程

■ 添加考试题目信息使用的数据表: tb_Lesson、tb_taoTi和tb_Questions

考试题目添加包含两个步骤:一是为添加的考试题目选择专业、课程以及套题,二是将填写的考



试题目信息插入到数据库中。考题信息添加页面的运行结果如图 5.26 所示。



图 5.26 考题信息添加页面的运行结果

考题信息添加页面涉及到的 HTML 表单的重要元素如表 5.8 所示。

表 5.8 考题信息添加页面涉及到的 HTML 表单的重要元素

名 称	类 型	重 要 属 性	含 义
form2	form	method="post" action="ktxx_tj_ok.php"	表单
kt_lb	select	<pre><?php \$query=mssql_query("select * from tb_ktlb"); while(\$myrow=mssql_fetch_array(\$query)){ ?></pre>	考题类别



			续表
名 称	类 型	重 要 属 性	含 义
kt_small_lb	select	<pre><select> <option value="第一套题">第一套题</option> <option value="第二套题">第二套题</option> <option value="第三套题">第三套题</option> <option value="第四套题">第四套题</option> </select></pre>	所属套题
kt_lx	select	<select> <option value="2">简答</option> <option value="3">论述</option> <option value="0">单选</option> <option value="1">多选</option> </select>	考题类型
kt_fs	text	size="10"	考试成绩
kt_nr	textarea	cols="60" rows="5"	考题内容
kt_daan	textarea	cols="60" rows="5"	考题答案
kt_zqdaan	textarea	cols="60" rows="5"	考题正确答案
Submit2	submit	value="提交考题"	"提交考题"按钮

添加考试题目首先要选择考题的类别,然后所择所属套题,再选择考题类型,其实现过程在表 5.8 所示的表单的重要属性里给出了完整的代码,这里不再赘述,最后根据考题类别将考题内容添加完整。

提交考题表单到数据处理页,程序处理页面首先应用变量获取到表单数据,然后应用 insert...into 语句将其插入到考题信息表 tb_kt 中。如果考题信息添加成功,则弹出提示信息,并重新定位到考题信息添加页面。数据处理页的完整代码如下:

例程14 代码位置: 光盘\TM\05\online_ks\admin\ktxx_tj_ok.php

5.9.4 查询考题信息的实现过程

■ 查看考试题目列表使用的数据表: tb_kt、tb_ktlb

管理员登录后,单击"考题信息管理"超链接,进入到查询考题信息页面,选择考题类别后,单



/+ +

击"考题查找"按钮,将查询出该类别下的所有考题信息。同时提供修改考题信息和删除考题信息的 功能。查询考题信息页面的运行结果如图 5.27 所示。



图 5.27 查询考题信息页面的运行结果

查询考题信息页面涉及到的 HTML 表单的重要元素如表 5.9 所示。

表 5.9 查询考题信息页面涉及到的 HTML 表单的重要元素

名 称	类 型	重 要 属 性	含义
form1	form	method="post" action="index.php?htgl=考题信息管理"	查询考题表单
		<select id="kt_lb" name="kt_lb"></select>	
		php</td <td></td>	
		<pre>\$query=mssql_query("select * from tb_ktlb");</pre>	
		while(\$myrow=mssql_fetch_array(\$query)){	
kt_lb	select	?>	考题类别
		<pre><option value="<?php echo \$myrow[online_ktlb];?>"></option></pre>	
		php echo \$myrow[online_ktlb];?	
		php }?	
Submit	submit	value="考题查找"	"考题查找"按钮

续	7	Ę	

			-2.10
名 称	类 型	重 要 属 性	含义
form2	form	method="post" action="ktxx_gl_ok.php"	考题信息表单
kt_lb	text	value=" php echo \$myrow[kt_lb];? "	考题类别显示
kt_lx	text	value=" php echo \$myrow[kt_lx];? "	考题类型
kt_fs	text	value=" php echo \$myrow[kt_fs];? "	考试成绩
kt_id	hidden	value=" php echo \$myrow[kt_id]? "	考题类别 id
Submit2	submit	value="修改"	"修改"按钮
Submit3	submit	value="删除"	"删除"按钮
kt_nr	textarea	php echo \$myrow[kt_nr];?	考题内容
kt_daan	textarea	php echo \$myrow[kt_daan];?	考题答案
kt_zqdaan	textarea	php echo \$myrow[kt_zqdaan];?	考题正确答案

在考试题目查询页面中,首先建立用于查询的表单 form1,该表单中包含"考题类别"列表/菜单控件以及"考题查找"按钮。

当管理员选择考题类别后,单击"考题查找"按钮,提交考题 kt_lb 类别到当前页,然后根据获取到的考题类别检索考题信息表 tb_kt,并将该类别下的所有考题信息输出到浏览器。代码如下:

例程15 代码位置: 光盘\TM\05\online_ks\admin\ktxx_gl.php

```
<?php
include("../conn/conn.php");
                                //连接数据库文件
$query=mssql_query("select * from tb_kt where kt_lb='$kt_lb'");
                                //根据选择考题类别检索数据信息
 while($myrow=mssql_fetch_array($query)){
                                //将查询结果应用while循环输出
<form name="form2" method="post" action="ktxx_gl_ok.php">
 考题类别
 <input name="kt_lb" type="text" value="<?php echo $myrow[kt_lb];?>" size="8">
 ------考题类型显示------
 考题类型
   <input name="kt_lx" type="text" value="<?php echo $myrow[kt_lx];?>" size="10">
 分数
    <input name="kt_fs" type="text" value="<?php echo $myrow[kt_fs];?>" size="5">
 <input type="hidden" name="kt_id" value="<?php echo $myrow[kt_id]?>">
    <input type="submit" name="Submit2" value="修改">/<input type="submit" name="Submit3" value="删除">
```

```
考题内容
  <textarea name="kt_nr" cols="60" rows="5"><?php echo
$myrow[kt_nr];?></textarea>
  考题答案
  <textarea name="kt_daan" cols="60" rows="5"><?php echo
$myrow[kt_daan];?></textarea>
  <!--
  考题正确答案
  <textarea name="kt_zqdaan" cols="60" rows="5"><?php echo
</form>
 <?php
                       //while循环语句结束
 }
 ?>
```

《》 代码贴士

- mssql_query()函数:向 SQL Server 服务器中的当前活动数据库发送一条 SQL 查询。
- ❷ mssql fetch array()函数: 返回根据从结果集获取的行生成的数组,如果没有更多行则返回 false。语法如下:

array mssql_fetch_array(resource result [, int result_type])

参数 result: 资源类型的参数,要传入的是由 mssql_query()函数返回的"数据指针"。

参数 result_type: 可选项,整数型参数,要传入的是 MSSQL_ASSOC、MSSQL_NUM 和 MSSQL_BOTH 3 种由 PHP 定义好的常数之一, 默认值是 MSSQL_BOTH。

- (1) 用 MSSQL_ASSOC 只得到关联索引。
- (2) 用 MSSQL_NUM 只得到数字索引。
- (3) 用 MSSQL_BOTH 将得到一个同时包含关联和数字索引的数组。

查询考题信息页面提供了修改考题信息的功能,管理员可对指定的考题信息进行编辑,单击"修 改"按钮后,提交 Submit2 表单信息到数据处理页 ktxx_gl_ok.php。

例程16 代码位置: 光盘\TM\05\online_ks\admin\ktxx_gl_ok.php

```
<?php
                              //连接数据库文件
include("../conn/conn.php");
                              //如果单击了"修改"按钮,则执行下面语句
if($Submit2==true){
$querys=mssql_query("update tb_kt set
```



查询考题信息页面提供了删除考题信息的功能,管理员可对指定的考题信息进行删除,单击"删除"按钮后,提交 Submit3 表单信息到数据处理页 ktxx_gl_ok.php。

例程17 代码位置: 光盘\TM\05\online_ks\admin\ktxx_gl_ok.php

5.10 开发技巧与难点分析

5.10.1 考生登录编号的获取

考生登录编号的生成主要应用的是 mt_rand()函数和 substr()函数。首先通过 mt_rand()函数来获取一个 100000~999999 之间的随机数,然后应用 substr()函数从该随机数中获取 6 个数字,作为考生编号。

mt_rand()函数的功能是获取随机数的值。语法格式如下:

int mt_rand([int min,int max])

证明:如果 mt_rand()函数没有提供可选参数 min 和 max,则返回 0 到 RAND_MAX 之间的伪随机数。

下面应用 mt_rand()函数来获取 100000~999999 之间的随机数。代码如下:

<?php echo mt_rand(100000,999999); ?>

运行结果为: 321864



5.10.2 通过 Ajax 技术实现计时与显示剩余时间

通过 Ajax 技术实现计时与显示剩余时间首先要创建一个 XMLHttpRequest 对象实例,确保其能够 在所有支持XMLHttpRequest的浏览器中运行,将其代码保存在一个名称为xmlHttpRequest.js的文件中, 然后在需要应用 Ajax 技术的页面中,应用以下代码包含该文件。

```
<script type="text/javascript" src="js/xmlHttpRequest.js"></script>
```

xmlHttpRequest.js 文件的代码如下:

例程18 代码位置: 光盘\TM\05\online_ks\js\xmlHttpRequest.js

```
//创建一个布尔型变量,用来检测是否为合法的IE实例
var xmlHttp = false;
                                       //检测是否使用的是IE
try {
  xmlHttp = new ActiveXObject("Msxml2.XMLHTTP");
                                       //如果JavaScript的版本大于5
                                       //如果不是,则使用老版本的ActiveX对象
} catch (e) {
                                       //如果使用的是IE浏览器
  try {
      xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");
   } catch (e2) {}
if (!xmlHttp && typeof XMLHttpRequest != "undefined") {
  try{
      xmlHttp = new XMLHttpRequest();
   }catch(e3){ xmlHttp = false;}
```

接下来编写两个自定义的 JavaScript 函数 ShowTime()和 sparetime(),通过 ShowTime()函数读取显 示时间文件 ShowTime.php 中的数据; 通过 sparetime()函数读取获取剩余时间文件 sparetime.php 中的数 据。ShowTime()和 sparetime()函数的代码如下:

例程19 代码位置: 光盘\TM\05\online_ks\js\xmlHttpRequest.js

```
<script type="text/javascript">
timer = window.setInterval("ShowTime()",1000);
                                            //每隔一秒钟调用一次ShowTime()函数
//定义ShowTime()函数以通过xmlHttpRequest对象读取ShowTime.php文件中的数据
function ShowTime(){
   xmlHttp.open("post", "showtime.php", true);
                                            //以post方法发送一个新请求
   xmlHttp.onreadystatechange = function(){
                                            //如果服务器响应发出的请求,则执行以下操作
   if(xmlHttp.readyState == 4){
      tet = xmlHttp.responseText;
                                            //获取返回的响应信息
             document.getElementById("show\_time").innerHTML = tet;\\
                                            //发送请求
xmlHttp.send(null);
</script>
<script type="text/javascript">
```

```
//每隔一秒钟调用一次sparetime()函数
time = window.setInterval("sparetime()",1000);
function sparetime(){
  xmlHttp.open("post","sparetime.php", true);
                                    //以post方法发送一个新请求
  xmlHttp.onreadystatechange = function(){
                                    //如果服务器响应发出的请求,则执行以下操作
     if(xmlHttp.readyState == 4){
        tet = xmlHttp.responseText;
                                    //获取返回的响应信息
        document.getElementById("sparetime").innerHTML = tet;
        if(tet=="00:00"){}
                                    //判断当剩余时间为00:00时
                                    //提交form1表单中的数据
           form1.submit();
  xmlHttp.send(null);
                                    //发送请求
</script>
```

在 ShowTime.php 文件中实现当前时间的显示。实现的原理是: 获取一个在 session 变量中存储的 考试开始时间的时间戳, 然后再应用 mktime()函数获取当前时间的时间戳, 应用当前时间戳减去考试 开始时间的时间戳, 最后应用 date()函数输出获取的新时间戳的时间值。代码如下:

例程20 代码位置: 光盘\TM\05\online_ks\ShowTime.php

在 sparetime.php 文件中获取考试的剩余时间。实现的原理是:首先设置考试时间为 20 分钟,在考试开始时间的基础上加上 20 分钟,然后用考试时间减去系统的当前时间,获取的就是考试的剩余时间,当剩余时间为 00:00 时表示本次考试结束,将考试题自动提交。sparetime.php 文件的代码如下:

例程21 代码位置: 光盘\TM\05\online_ks\sparetime.php

最后在考试操作页面中通过 DIV 标签来输出考试时间和剩余时间。代码如下:

```
<div id="sparetime"></div> <!--输出考试剩余时间--> </div id="show_time"></div> <!--输出考试时间-->
```





5.11 Ajax 无刷新技术专题

5.11.1 Ajax 概述

Ajax 是 Asynchronous JavaScript and XML 的缩写,意思是异步的 JavaScript 与 XML。

Ajax 并不是一种新技术,或者说它不是一种技术,实际上,它是结合了 Javascript、XHTML 和 CSS、DOM、XML 和 XSTL、XMLHttpRequest 等编程技术以新的强大方式组合而成,可以让开发人员构建基于 PHP 技术的 Web 应用,并打破了使用页面重载的惯例。Ajax 包含:

- (1) XHTML 和 CSS 技术实现标准页面。
- (2) Document Object Model 技术实现动态显示和交互。
- (3) XML 和 XSLT 技术实现数据的交换和维护。
- (4) XMLHttpRequest 技术实现异步数据接收。
- (5) JavaScript 绑定和处理所有数据。

Ajax 是一种运用浏览器的技术,它可以在浏览器和服务器之间得到异步通信机制进行数据通信,从而允许浏览器向服务器获取少量信息而不是刷新整个页面。

5.11.2 Ajax 的优点

Ajax 是使用客户端脚本与 Web 服务器交换数据的 Web 应用开发方法。这样, Web 页面不用打断交互流程进行重新加载,就可以动态地更新。Ajax 优点如下:

(1) 减轻服务器的负担。

Ajax 的原则是"按需取数据",可以最大程度地减少冗余请求,从而减轻对服务器造成的负担。

- (2) 无刷新更新页面,减少用户心理和实际的等待时间。
- "按需取数据"的模式减少了数据的实际读取量。如果说重载的方式是从一个终点回到原点再到另一个终点的话(如图 5.28 所示),那么 Ajax 就是以一个终点为基点到达另一个终点(如图 5.29 所示)。

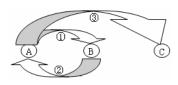


图 5.28 重载方式

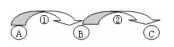


图 5.29 Ajax 方式

其次,即使要读取较大的数据,也不会出现白屏的情况。Ajax 使用 XMLHTTP 对象发送请求并得到服务器响应,在不重新载入整个页面的情况下用 JavaScript 操作 DOM 最终更新页面,所以在读取数据的过程中,用户所面对的不是白屏,而是原来的页面状态;页面只有接收到全部数据后才更新相应部分的内容,而这种更新也是瞬间的,用户几乎感觉不到。

- (3) 带来更好的用户体验。
- (4) 把部分服务器负担的工作转交给客户端,利用客户端闲置的能力来处理任务,从而减轻服务器和带宽的负担,节约空间和宽带租用成本。



- (5) 可以调用外部数据。
- (6) 是一种基于标准化并被广泛支持的技术,不需要下载插件或者小程序。
- (7) 进一步促进 Web 页面展现形式与数据的分离。

Ajax 的工作原理 5.11.3

传统的 Web 模式强制用户进入"提交→等待→重新显示"网页,用户的动作总是与服务器进行同步 思考,客户在网页上的操作转化为 HTTP 请求传回服务器,而服务器接受请求以及相关数据、解析数据 并将其发送给相应的处理单元后,将返回的数据转成 HTML 页返还给客户。而当服务器处理数据的时候, 用户只能等待,每一步操作都需要等待服务器返回新的网页。由于每次应用的交互都需要向服务器发送 请求,应用的响应时间就依赖于服务器的响应时间,这就导致了用户页面的响应比本地应用慢得多。

运用了 Ajax 技术的 Web 应用模型,它的工作原理相当于在客户端和服务器端之间添加了一个中 间层,称为 Ajax 引擎(采用 JavaScript 编写,通常在一个隐藏的框架中),实现了与服务器进行异步 思考的通信能力,从而使用户从请求/响应的循环中解脱出来,向服务器发出异步请求,也就是不用等 待服务器的通信。所以用户不用再打开一个空白窗口,等待服务器完成后再进行响应。Ajax 应用可以 仅向服务器发送并取回必需的数据,它使用 SOAP 或其他一些基于 XML 的 Web Service 接口,并在客 户端采用 JavaScript 处理来自服务器的响应。因为在服务器和浏览器之间交换的数据大量减少,所以 Web 站点看起来是即时响应的。同时很多的处理工作可以在发出请求的客户端机器上完成,所以 Web 服务器的处理时间也减少了。

引入 Ajax 的 Web 模型与传统的 Web 模型比较如图 5.30 所示。

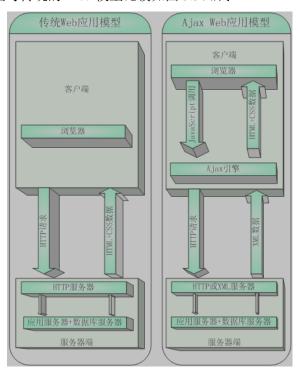


图 5.30 传统 Web 应用模型与 Ajax Web 应用模型的比较



5.11.4 Ajax 的工作流程

使用 Ajax,用户可以创建接近本地桌面应用的直接、更可用、更丰富、更动态的 Web 用户界面。 Ajax 内部工作流程如图 5.31 所示。

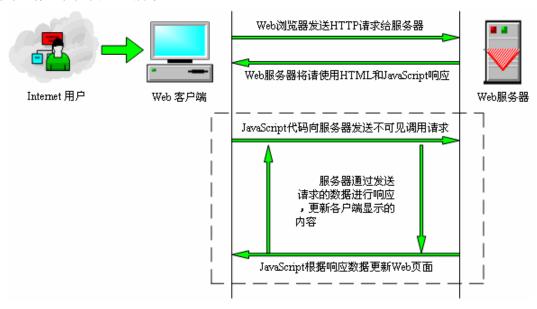


图 5.31 Ajax 内部工作流程图

5.11.5 Ajax 中的核心技术 XMLHttpRequest

Ajax 技术中最核心的技术就是 XMLHttpRequest,它是一个具有应用程序接口的 JavaScript 对象,能够使用超文本传输协议(Http)连接一个服务器,是微软公司为了满足开发者的需要,于 1999 年在 IE 5.0 浏览器中率先推出的。现在许多浏览器都对其提供了支持,不过实现方式与 IE 有所不同。

通过 XMLHttpRequest 对象,Ajax 可以像桌面应用程序一样只同服务器进行数据层面的交换,而不用每次都刷新页面,也不用每次都将数据处理的工作交给服务器来做,这样既减轻了服务器负担又加快了响应速度、缩短了用户等待的时间。

在使用 XMLHttpRequest 对象发送请求和处理响应之前,首先需要初始化该对象,由于 XMLHttpRequest 不是一个 W3C 标准,所以对于不同的浏览器,初始化的方法也是不同的。

1. IE 浏览器

IE 浏览器把 XMLHttpRequest 实例化为一个 ActiveX 对象。具体方法如下:

var http_request = new ActiveXObject("Msxml2.XMLHTTP");

或者

var http_request = new ActiveXObject("Microsoft.XMLHTTP");



- 迎说明:上面语法中的 Msxml2.XMLHTTP 和 Microsoft.XMLHTTP 是针对 IE 浏览器的不同版本而进行设置的,目前比较常用的是这两种。
 - 2. Mozilla、Safari 等其他浏览器

Mozilla、Safari 等其他浏览器把它实例化为一个本地 JavaScript 对象。具体方法如下:

var http_request = new XMLHttpRequest();

为了提高程序的兼容性,可以创建一个跨浏览器的 XMLHttpRequest 对象。创建一个跨浏览器的 XMLHttpRequest 对象其实很简单,只需要判断一下不同浏览器的实现方式,如果浏览器提供了 XMLHttpRequest 类,则直接创建一个实例,否则使用 IE 的 ActiveX 控件。具体代码如下:

```
if (window.XMLHttpRequest) { // Mozilla、Safari······
http_request = new XMLHttpRequest();
} else if (window.ActiveXObject) { // IE浏览器
try {
    http_request = new ActiveXObject("Msxml2.XMLHTTP");
} catch (e) {
    try {
        http_request = new ActiveXObject("Microsoft.XMLHTTP");
    } catch (e) {}
} catch (e) {}
}
```

证明:由于 JavaScript 具有动态类型特性,而且 XMLHttpRequest 对象在不同浏览器上的实例是兼容的,所以可以用同样的方式访问 XMLHttpRequest 实例的属性的方法,不需要考虑创建该实例的方法是什么。

5.12 本章总结

至此,一个完整的网络在线考试系统已经全部完成。在程序的开发过程中,采用了 switch 框架,使整个系统的设计思路更加清晰。同时,为了使程序更加人性化,系统中应用了 Ajax 技术实现在线考试时自动计时等功能。希望读者能认真学习,并做到融会贯通。

