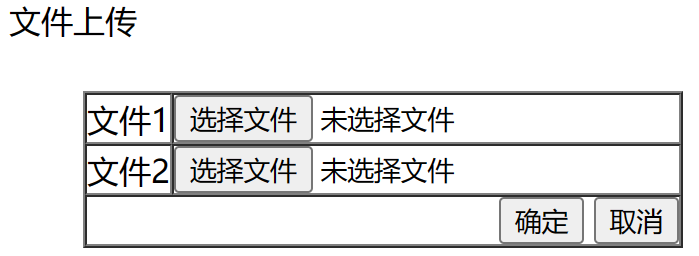
Fileupload.html 文件上传



<body>

<table id="divContainer" style="HEIGHT: 100%; WIDTH: 380" border="0">

<!--

<form> 标签的 enctype 属性规定了在提交表单时要使用哪种内容类型。

在表单需要二进制数据时，比如文件内容，请使用 "multipart/form-data"。

<input> 标签的 type="file" 属性规定了应该把输入作为文件来处理。

举例来说，当在浏览器中预览时，会看到输入框旁边有一个浏览按钮。

-->

<form method="post" name="frmUpload" enctype="multipart/form-data" action="upload\_file.php">

<tr height="35"><td align="left" valign="bottom">文件上传</td></tr>

<tr>

<td align="center" valign="top">

    <table class="Transex" border="1" cellspacing="0" cellpadding="0">

        <tr>

<td nowrap><label class="FieldLabel">文件1</label></td>

<!--nowrap 属性规定表格单元格中的内容不换行-->

        <td><input type="file" class="TxtInput" tabindex="1" name="afile1"></td>

        </tr>

        <tr>

<td nowrap><label class="FieldLabel"> 文件2</label></td>

<td><input type="file" class="TxtInput" tabindex="1" name="afile2"></td>

  </tr>

        <tr>

        <td colspan="2" align="right">

        <!--<button name="btnOk" id="btnOk" >确定</button>-->

        <input type="submit" name="submit" value="确定" />

        <!--<button name="btnCancel" id="btnCancel" onclick="window.close();">取消</button> -->

        <button type="button" name="btnCancel" id="btnCancel" onclick="window.close();"> 取消</button>

        </td>

        <input type="hidden" name="ifupload" value=1>&nbsp;&nbsp;

        </tr>

        </table>

        </td>

  </tr>

</form>

</table>

</body>

Uploadfile.php 上传并修改存储路径

<?php

header("Cache-Control: no-cache, post-check=0, pre-check=0");

header("Content-type:text/html;charset=gb2312");

//filename:upldfile.php

if($\_POST["ifupload"]=="1") {

/\*

addslashes() 函数在指定的预定义字符前添加反斜杠。

这些预定义字符是：

单引号 (')

双引号 (")

反斜杠 (\)

NULL

\*/

//在程序中，有时我们会看到这样的路径写法，"D:\\Driver\\Lan" 也就是两个反斜杠来分隔路径。

$path=AddSlashes(dirname(\_\_FILE\_\_))  . "\\\\upload\\\\";

//echo AddSlashes(dirname(\_\_FILE\_\_)); die();

echo $path;//die();

for($i=1;$i<=2;$i++) {

$files="afile$i";

    echo $files. " Kb<br />";

//用户只能上传pdf文件，文件大小必须大于 20 kb：

//if (($\_FILES[$files]["type"] == "application/pdf")

//&& ($\_FILES[$files]["size"] > 20000))

//{

    if ($\_FILES[$files]["error"] > 0)

    {

    echo "Error: " . $\_FILES[$files]["error"] . "<br />";

    }

    else

    {

    echo "Upload: " . $\_FILES[$files]["name"] . "<br />";

        echo "Type: " . $\_FILES[$files]["type"] . "<br />";

        echo "Size: " . ($\_FILES[$files]["size"] / 1024) . " Kb<br />";

        echo "Stored in: " . $\_FILES[$files]["tmp\_name"] . " Kb<br />";

     }

/\*

通过使用 PHP 的全局数组 $\_FILES，你可以从客户计算机向远程服务器上传文件。

第一个参数是表单的 input name，

第二个下标可以是 "name", "type", "size", "tmp\_name" 或 "error"。就像这样：

$\_FILES["file"]["name"] - 被上传文件的名称

$\_FILES["file"]["type"] - 被上传文件的类型

$\_FILES["file"]["size"] - 被上传文件的大小，以字节计

$\_FILES["file"]["tmp\_name"] - 存储在服务器的文件的临时副本的名称

$\_FILES["file"]["error"] - 由文件上传导致的错误代码

\*/

echo $\_FILES[$files]['tmp\_name'];

//is\_uploaded\_file() 函数判断指定的文件是否是通过 HTTP POST 上传的

    if (is\_uploaded\_file($\_FILES[$files]['tmp\_name'])) {

        $filename = $\_FILES[$files]['name'];

        $localfile = $path . $filename;

        echo $localfile;

        //move\_uploaded\_file() 函数将上传的文件移动到新位置。

        move\_uploaded\_file($\_FILES[$files]['tmp\_name'], $localfile);

    }

//}

}

echo "<br><b>You have uploaded files successfully</b><br>";

exit;

}

?>

Upload\_compressedfile.php 多加了一个压缩的过程，应该不会考

<?php

header("Cache-Control: no-cache, post-check=0, pre-check=0");

header("Content-type:text/html;charset=gb2312");

//filename:upldfile.php

if($\_POST["ifupload"]=="1") {

//在程序中，有时我们会看到这样的路径写法，"D:\\Driver\\Lan" 也就是两个反斜杠来分隔路径。

$path=AddSlashes(dirname(\_\_FILE\_\_))  . "\\\\upload\\\\";

echo $path;

for($i=1;$i<=2;$i++) {

$files="afile$i";

    echo $files. " Kb<br />";

//用户只能上传pdf文件，文件大小必须大于 20 kb：

//if (($\_FILES[$files]["type"] == "application/pdf")

//&& ($\_FILES[$files]["size"] > 20000))

//{

    if ($\_FILES[$files]["error"] > 0)

    {

    echo "Error: " . $\_FILES[$files]["error"] . "<br />";

    }

    else

    {

    echo "Upload: " . $\_FILES[$files]["name"] . "<br />";

        echo "Type: " . $\_FILES[$files]["type"] . "<br />";

      echo "Size: " . ($\_FILES[$files]["size"] / 1024) . " Kb<br />";

      echo "Stored in: " . $\_FILES[$files]["tmp\_name"] . " Kb<br />";

}

/\*

通过使用 PHP 的全局数组 $\_FILES，你可以从客户计算机向远程服务器上传文件。

第一个参数是表单的 input name，

第二个下标可以是 "name", "type", "size", "tmp\_name" 或 "error"。就像这样：

$\_FILES["file"]["name"] - 被上传文件的名称

$\_FILES["file"]["type"] - 被上传文件的类型

$\_FILES["file"]["size"] - 被上传文件的大小，以字节计

$\_FILES["file"]["tmp\_name"] - 存储在服务器的文件的临时副本的名称

$\_FILES["file"]["error"] - 由文件上传导致的错误代码

\*/

echo $\_FILES[$files]['tmp\_name']."<br>";

 //is\_uploaded\_file() 函数判断指定的文件是否是通过 HTTP POST 上传的

    if (is\_uploaded\_file($\_FILES[$files]['tmp\_name'])) {

    $filename = $\_FILES[$files]['name'];

        $src\_type = $\_FILES[$files]['type'];

        $src\_type = explode('/',$src\_type)[count($src\_type)];

      $localfile = $path . $filename;

        $localfilesmall = $path . "s". $filename;//小图路径

        echo $localfile;

        $src\_fnc = 'imagecreatefrom'.$src\_type ;//创建图像资源函数 对应函数 imagecreatefrompng 或者 imagecreatefromjpeg

        $src\_image = $src\_fnc($\_FILES[$files]['tmp\_name']);//临时图片资源

        $src\_w = imagesx($src\_image);//上传图片的宽

        $src\_h = imagesy($src\_image);//高

        if (src\_w > 600)

        {

            $dst\_w = 300;//固定300，

            $dst\_h = floor((300.0/$src\_w)\*$src\_h);//同样的比例缩放

        }

        else

        {

            $dst\_w = floor($src\_w/3);//这里 目标图片的宽是源图片的三分之一，

            $dst\_h = floor($src\_h/3);//同样的

        }

        $dst\_image = imagecreatetruecolor($dst\_w, $dst\_h);//此时的画布为原来图片的 1/3或比例分之一

        imagecopyresampled($dst\_image, $src\_image, 0, 0, 0, 0, $dst\_w, $dst\_h, $src\_w, $src\_h);

        $out\_fnc = 'image'.$src\_type; //imagepng   imagejpeg

        if( !($out\_fnc($dst\_image ,$localfilesmall) && file\_exists($localfilesmall))){

        //缩略图保存成功了，文件路径就是$path

        }

else{

        //保存失败了，检查错误

        }

        //move\_uploaded\_file() 函数将上传的文件移动到新位置。

       move\_uploaded\_file($\_FILES[$files]['tmp\_name'], $localfile);

}

//}

}

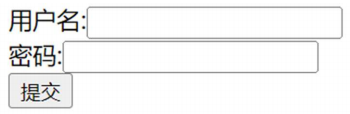
echo "<br><b>You have uploaded files successfully</b><br>";

exit;

}

?>

Loginform(Session) 注册界面



<?

header("Content-type: text/html; charset=GB2312");

?>

<form name="fmlog" method="post" action="logcheck.php?para=para1">

用户名:<input type="text" name="username"><br>

密码:<input type="password" name="password"><br>

<input type="hidden" name="action" value="login">

<input type="submit" value="提交">

</form>

Logcheck(Session) 记录登录信息

<?php

global $c;

  $c = 0;

//isset() 函数用于检测变量是否已设置并且非 NULL。

if (isset($\_POST["action"]) && $\_POST["action"] == "login"){

    // 处理登录的代码

    $user=$\_POST["username"];

    $pwd =$\_POST["password"];

  $para=$\_GET["para"];

    // 记录用户id和password

    session\_start();

    $\_SESSION["USERID"]   = strtoupper($user);

    $\_SESSION["USERPSW"] = $pwd;

    echo $\_SESSION["USERID"] . $para . " logged in. Your are welcome. ";

    echo '<form name="fmlog" method="post">';

    echo '<input type="hidden" name="action" value="logout" >';

    echo '<input type="submit" value="注销">';

    echo '</form>';

print <<<EOT

{$\_SESSION["USERID"]} logged in. Your are welcome.

<form name="fmlog" method="post">

<input type="hidden" name="action" value="logout" >

<input type="submit" value="注销">

</form>

EOT;

}

?>

Mysqltest.php



<?php

header("Cache-Control: no-cache, post-check=0, pre-check=0");

header("Content-type:text/html;charset=gb2312");

$con = mysql\_connect("localhost","root","sql");

mysql\_select\_db("qing\_zhou", $con);

mysql\_query("set names gb2312 ");

$str=$\_POST["str"];

if ($str<>'')

{

$SQL = "delete from php\_admin where userid in ($str)";

//echo $SQL; die();

mysql\_query($SQL);

}

$userid=$\_POST["userid"]; $upassword=$\_POST["upassword"]; $username=$\_POST["username"];

$userage=$\_POST["userage"]; $usermobile=$\_POST["usermobile"]; $useremail=$\_POST["useremail"];

$SQL ="select count(\*) from php\_admin where userid='$userid'";

$RS0 = mysql\_query($SQL);

$RS = mysql\_fetch\_array($RS0);

if ($RS[0]>=1)

{

    echo "用户ID重复，请重新输入!";

    die();

}

$SQL = "insert into php\_admin(userid,password,username,userage,usermobile,useremail)";

$SQL = $SQL . " values('$userid','$upassword','$username',$userage,'$usermobile','$useremail')";

$RS0 = mysql\_query($SQL);

$SQL = "select \* from php\_admin";

$RS0 = mysql\_query($SQL);

?>

<script language = javascript>

<!--

function efg()

{

var str=document.form1.usermobile.value;

    var str2=document.form1.useremail.value;

    var myreg=/^[1][3,4,5,7,8,9][0-9]{9}$/;

    var reg=new RegExp(/^([a-zA-Z0-9.\_-])+@([a-zA-Z0-9\_-])+(\.[a-zA-Z0-9\_-])+/);

    if (!myreg.test(str)) {

    alert("请填写正确的手机号码");

        document.form1.usermobile.focus();

        return false;

    }

    if (!reg.test(str2)) {

        alert("请填写正确的邮箱地址");

        document.form1.useremail.focus();

        return false;

    }

    //showHint(document.form1.userid.value);

    document.form1.submit();

}

function abc(s) {

form1.str.value=s;

alert(form1.str.value);

form1.submit();

}

function edf()

{

    var checkboxes;

    var i;

  //var str = [];

var str='';

    checkboxes = document.getElementsByName("checkbox0");

    for(i=0;i<checkboxes.length;i++){

        if(checkboxes[i].checked){

            //str.push(checkboxes[i].value);

            str = str + '\'' + checkboxes[i].value +'\',';

    }

}

    document.form1.str.value=str.substring(0,str.length-1);

    alert(document.form1.str.value);

    document.form1.submit();

}

//-->

</script>

<form id=form1 name=form1 method=post>

<table>

<tr><td>用户ID:</td><td><input type=text id=userid name=userid></td></tr>

<tr><td>密码:</td><td><input type=password id=upassword name=upassword></td></tr>

<tr><td>用户姓名:</td><td><input type=text id=username name=username></td></tr>

<tr><td>用户年龄:</td><td><input type=text id=userage name=userage></td></tr>

<tr><td>手机号:</td><td><input type=text id=usermobile name=usermobile></td></tr>

<tr><td>EMAIL:</td><td><input type=text id=useremail name=useremail></td></tr>

</table>

<TABLE border=1>

<input type=hidden name=str>

<?php

while($RS = mysql\_fetch\_array($RS0))

{

?>

<tr>

<td><input type=checkbox name=checkbox0 value='<?php echo $RS["userid"]?>'></td>

<td align=left>&nbsp;<?php echo $RS["userid"]?></td>

<td align=left>&nbsp;<?php echo $RS["usertype"]?></td>

<td align=left>&nbsp;<?php echo $RS["password"]?></td>

<td align=left>&nbsp;<?php echo $RS["username"]?></td>

<td align=left>&nbsp;<?php echo $RS["userage"]?></td>

<td align=left>&nbsp;<?php echo $RS["usermobile"]?></td>

<td align=left>&nbsp;<?php echo $RS["useremail"]?></td>

<td> <input type=button value=删除 onclick="abc('<?php echo $RS["userid"]; ?>')"> </td>

</tr>

<?php

}

?>

</table>

<input type=button value=删除 onclick="edf()">

<input type=button value="保存" onclick="return efg();">

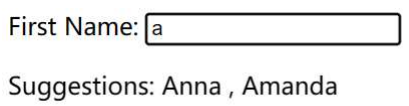
</form>

<?php

mysql\_close($con);

?>

Hint\_demo1.php hint提示，ajax



<!--

该表单是这样工作的：

当用户在输入域中按下并松开按键时，

会触发一个事件

当该事件被触发时，执行名为 showHint() 的函数

表单的下面是一个名为 "txtHint" 的 <span>。

它用作 showHint() 函数所返回数据的占位符。

-->

<html>

<head>

<script src="clienthint.js?abmcjd=gf"></script>

</head>

<body>

<form id="form1">

First Name:

<input type="text" id="txt1" onkeyup="showHint(this.value)">

</form>

<p>Suggestions: <span id="txtHint" name="txtHint">abc</span></p>

</body>

</html>

ClientHint.js ajax 获取提示，将占位符替换为提示

/\*

showHint() 函数

每当在输入域中输入一个字符，该函数就会被执行一次。

如果文本框中有内容 (str.length > 0)，该函数这样执行：

定义要发送到服务器的 URL（文件名）

把带有输入域内容的参数 (q) 添加到这个 URL

添加一个随机数，以防服务器使用缓存文件

调用 GetXmlHttpObject 函数来创建 XMLHTTP 对象，

并在事件被触发时告知该对象执行名为 stateChanged 的函数

用给定的 URL 来打开打开这个 XMLHTTP 对象

向服务器发送 HTTP 请求

如果输入域为空，则函数简单地清空 txtHint 占位符的内容。

\*/

var xmlHttp;

function showHint(str)

{

if (str.length==0)

  {

  document.getElementById("txtHint").innerHTML=""

  return

  }

xmlHttp=GetXmlHttpObject()

if (xmlHttp==null)

  {

  alert ("Browser does not support HTTP Request")

  return

  }

var url="gethint.php";

url=url+"?q="+str;

url=url+"&sid="+Math.random();

xmlHttp.onreadystatechange=stateChanged ;

xmlHttp.open("GET",url,true);

xmlHttp.send(null);

//document.getElementById("txtHint").innerHTML=xmlHttp.responseText;

}

/\*

stateChanged() 函数

每当 XMLHTTP 对象的状态发生改变，

则执行该函数。

在状态变成 4 （或 "complete"）时，

用响应文本填充 txtHint 占位符 txtHint 的内容。

\*/

/\*

发送一个请求后，

客户端无法确定什么时候

会完成这个请求，

所以需要用事件机制来捕获请求的状态，

XMLHttpRequest对象提供了

onreadyStateChange事件实现这一功能。

这类似于回调函数的做法。

onreadyStateChange

事件可指定一个事件处理函数来处理

XMLHttpRequest对象的执行结果

onreadyStateChange事件是在

readyState属性发生改变时触发的，

readyState的值表示了当前请求的状态，

在事件处理程序中可以根据这个值

来进行不同的处理。

 readyState有五种可取值

 0：尚未初始化，

 1：正在加载，

 2：加载完毕，

 3：正在处理；

 4：处理完毕。

 一旦readyState属性的值变成了4，

 就可以从服务器返回的响应数据

 进行访问了。

\*/

function stateChanged()

{

  var str;

if (xmlHttp.readyState==4 || xmlHttp.readyState=="complete")

  {

    str = xmlHttp.responseText;

    if (str=='用户ID重复，请重新输入!')

    {

      document.getElementById("txtHint").innerHTML= str;

      document.getElementById("userid").focus();

    }

    else

    {

        document.getElementById("form1").submit();

    }

  }

}

/\*

GetXmlHttpObject() 函数

AJAX 应用程序只能运行在完整支持 XML 的 web 浏览器中。

上面的代码调用了名为 GetXmlHttpObject() 的函数。

该函数的作用是解决为不同浏览器创建不同 XMLHTTP 对象的问题。

\*/

function GetXmlHttpObject()

{

var xmlHttp=null;

try{

  // Firefox, Opera 8.0+, Safari

  xmlHttp=new XMLHttpRequest();

  }

catch (e){

  // Internet Explorer

  try{

  xmlHttp=new ActiveXObject("Msxml2.XMLHTTP");

  }

  catch (e){

  xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");

  }

  }

return xmlHttp;

}

Gethint.php 获取提示的具体过程

<?php

// Fill up array with names

$a[]="Anna";

$a[]="Brittany";

$a[]="Cinderella";

$a[]="Diana";

$a[]="Eva";

$a[]="Fiona";

$a[]="Gunda";

$a[]="Hege";

$a[]="Inga";

$a[]="Johanna";

$a[]="Kitty";

$a[]="Linda";

$a[]="Nina";

$a[]="Ophelia";

$a[]="Petunia";

$a[]="Amanda";

$a[]="Raquel";

$a[]="Cindy";

$a[]="Doris";

$a[]="Eve";

$a[]="Evita";

$a[]="Sunniva";

$a[]="Tove";

$a[]="Unni";

$a[]="Violet";

$a[]="Liza";

$a[]="Elizabeth";

$a[]="Ellen";

$a[]="Wenche";

$a[]="Vicky";

//get the q parameter from URL

$q=$\_GET["q"];

//lookup all hints from array if length of q>0

if (strlen($q) > 0)

{

$hint="";

for($i=0; $i<count($a); $i++){

  if (strtolower($q)==strtolower(substr($a[$i],0,strlen($q)))){

    if ($hint==""){

      $hint=$a[$i];

      }

    else

      {

      $hint=$hint." , ".$a[$i];

      }

    }

  }

}

//Set output to "no suggestion" if no hint were found

//or to the correct values

if ($hint == "")

{

$response="no suggestion";

}

else

{

$response=$hint;

}

//output the response

echo $response;

?>

Demo2.php AJAX 有选项框，

<html>

<head>

<script src="selectcd.js?a=2"></script>

</head>

<body>

<form>

Select a CD:

<select name="cds" onchange="showCD(this.value)">

<option value="Bob Dylan">Bob Dylan</option>

<option value="Bee Gees">Bee Gees</option><selected></selected>

<option value="Cat Stevens">Cat Stevens</option>

</select>

</form>

<p>

<div id="txtHint"><b>CD info will be listed here.</b></div>

</p>

</body>

</html>

Selectcd.js(AJAX) 有 xml 应该不考

var xmlHttp

function showCD(str)

{

xmlHttp=GetXmlHttpObject()

if (xmlHttp==null)

  {

  alert ("Browser does not support HTTP Request")

  return

  }

var url="getcd.php"

url=url+"?q="+str

url=url+"&sid="+Math.random()

xmlHttp.onreadystatechange=stateChanged

xmlHttp.open("GET",url,true)

xmlHttp.send(null)

}

function stateChanged()

{

  if (xmlHttp.readyState==4 || xmlHttp.readyState=="complete")

  {

        document.getElementById("txtHint").innerHTML=xmlHttp.responseText

  }

}

function GetXmlHttpObject()

{

var xmlHttp=null;

try

{

  xmlHttp=new XMLHttpRequest();

  }

catch (e)

  {

  try

  {

  xmlHttp=new ActiveXObject("Msxml2.XMLHTTP");

  }

  catch (e)

  {

  xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");

  }

  }

return xmlHttp;

}

Getcd.php AJAX XML

<?php

/\*

当请求从 JavaScript 发送到 PHP 页面时，发生：

PHP 创建 "cd\_catalog.xml" 文件的 XML DOM 对象

循环所有 "artist" 元素 (nodetypes = 1)，

查找与 JavaScript 所传数据向匹配的名字

找到 CD 包含的正确 artist

输出 album 的信息，并发送到 "txtHint" 占位符

\*/

/\*

节点

根据 DOM，XML 文档中的每个成分都是一个节点。

DOM 是这样规定的：

整个文档是一个文档节点

每个 XML 标签是一个元素节点

包含在 XML 元素中的文本是文本节点

每一个 XML 属性是一个属性节点

注释属于注释节点

nodeType 属性可返回节点的类型。

最重要的节点类型是：

元素类型 节点类型

元素element 1

属性attr 2

文本text 3

注释comments 8

文档document 9

\*/

$q=$\_GET["q"];

$xmlDoc = new DOMDocument();

$xmlDoc->load("cd\_catalog.xml");

//返回 xmlDoc文档节点下的所有 <ARTIST> 元素节点

$x=$xmlDoc->getElementsByTagName('ARTIST');

for ($i=0; $i<=$x->length-1; $i++)

{

//Process only element nodes

if ($x->item($i)->nodeType==1)

  {

    // 或者 if ($x->item($i)->childNodes->item(0)->nodeValue == $q)

    //echo $x->item($i)->nodeValue.":".$x->item($i)->nodeName."<br>";

    if ($x->item($i)->nodeValue == $q)

    {

    $y=($x->item($i)->parentNode);

    }

  }

}

$cd=($y->childNodes);

//echo $cd->length;

for ($i=0;$i<$cd->length;$i++)

{

//Process only element nodes

if ($cd->item($i)->nodeType==1)

  {

  echo($cd->item($i)->nodeName);

  echo(": ");

  echo($cd->item($i)->nodeValue);

  echo("<br>");

  }

}

?>

Mysql1.php 大杂烩，插入，删除，复选框，正则表达式，ajax



<script src="clienthint.js?abcjd=aaaa"></script>

<script language="javascript">

function efg()

{

var str=document.form1.usermobile.value;

var str2=document.form1.useremail.value;

var myreg=/^[1][3,4,5,7,8,9][0-9]{9}$/;

var reg=new RegExp(/^([a-zA-Z0-9.\_-])+@([a-zA-Z0-9\_-])+(\.[a-zA-Z0-9\_-])+/);

if (!myreg.test(str)) {

alert("请填写正确的手机号码");

document.form1.usermobile.focus();

return false;

}

if (!reg.test(str2)) {

alert("请填写正确的邮箱地址");

document.form1.useremail.focus();

return false;

}

showHint(document.form1.userid.value); //调用 clienthint.js 中的 showHint 来检测 ID 重

复，过程中使用了 ajax

//document.form1.submit();

}

function abc()

{

var checkboxes;

var i;

//var str = [];

var str='';

checkboxes = document.getElementsByName("checkbox0");

for(i=0;i<checkboxes.length;i++){ if(checkboxes[i].checked){

//str.push(checkboxes[i].value);

str = str + checkboxes[i].value +',';

}

}

document.form1.userinfo.value=str.substring(0,str.length-1);

alert(document.form1.userinfo.value);

//return false;

document.form1.submit();

}

</script>

<?php

header("Cache-Control: no-cache, post-check=0, pre-check=0");

header("Content-type:text/html;charset=gb2312");

$userinfo= $\_POST["userinfo"];

$con = mysql\_connect("localhost","root","sql");

mysql\_select\_db("qing\_zhou", $con);

mysql\_query("set names gb2312 ");

$userid=$\_POST["userid"];

$upassword=$\_POST["upassword"];

$username=$\_POST["username"];

$userage=$\_POST["userage"];

$usermobile=$\_POST["usermobile"];

$useremail=$\_POST["useremail"];

$SQL ="select count(\*) from php\_admin where userid='$userid'";

$RS0 = mysql\_query($SQL);

$RS = mysql\_fetch\_array($RS0);

if ($RS[0]>=1)

{

echo "用户 ID 重复，请重新输入!";

die();

}

$SQL = "insert into php\_admin(userid,password,username,userage,usermobile,useremail)";

$SQL = $SQL . "

values('$userid','$upassword','$username',$userage,'$usermobile','$useremail')";

//echo $SQL; //die();

$RS0 = mysql\_query($SQL);

$SQL = "delete from php\_admin where userid in (".$userinfo.")";

//echo $SQL; die();

$RS0 = mysql\_query($SQL);

$SQL = "select \* from php\_admin";

$RS0 = mysql\_query($SQL); ?>

<form id=form1 name=form1 method=POST action="mysql1.php">

<input type=hidden id=userinfo name=userinfo>

<table>

<tr><td>用户 ID:</td><td><input type=text id=userid name=userid></td></tr>

<tr><td>密码:</td><td><input type=password id=upassword name=upassword></td></tr>

<tr><td>用户姓名:</td><td><input type=text id=username name=username></td></tr>

<tr><td>用户年龄:</td><td><input type=text id=userage name=userage></td></tr>

<tr><td>手机号:</td><td><input type=text id=usermobile name=usermobile></td></tr>

<tr><td>EMAIL:</td><td><input type=text id=useremail name=useremail></td></tr>

</table>

<table>

<?php

$i=0;

while($RS = mysql\_fetch\_array($RS0))

{ $i++;

?>

<tr>

<td><input type="checkbox" id=checkbox0 name=checkbox0

value="'<?php echo $RS["userid"] ?>'"></td>

<td align=left> <?php echo $RS["userid"]?></td>

<td align=left> <?php echo $RS["usertype"]?></td>

<td align=left> <?php echo $RS["password"]?></td>

<td align=left> <?php echo $RS["username"]?></td>

<td align=left> <?php echo $RS["userage"]?></td>

<td align=left> <?php echo $RS["usermobile"]?></td>

<td align=left> <?php echo $RS["useremail"]?></td>

<!--

<td align=left><input type=button value="删除" onclick="return abc('<?php

$userinfo=$RS["userid"]; echo $userinfo?>');"> </td>

-->

</tr>

<?php }

mysql\_close($con); $RS=NULL; $Con =NULL;

?>

<tr><td colspan=7 align=left><input type=button value="删除" onclick="return abc();">

<input type=button value="保存" onclick="return efg();"> </td></tr>

<tr><td colspan=7 align=left><span id="txtHint" name="txtHint"></span></td></tr>

</table>

</form>

Clienthint.js 检测重复，返回错误信息

/\*

showHint() 函数

每当在输入域中输入一个字符，该函数就会被执行一次。

如果文本框中有内容 (str.length > 0)，该函数这样执行：

定义要发送到服务器的 URL（文件名）

把带有输入域内容的参数 (q) 添加到这个 URL

添加一个随机数，以防服务器使用缓存文件

调用 GetXmlHttpObject 函数来创建 XMLHTTP 对象，

并在事件被触发时告知该对象执行名为 stateChanged 的函数

用给定的 URL 来打开打开这个 XMLHTTP 对象

向服务器发送 HTTP 请求

如果输入域为空，则函数简单地清空 txtHint 占位符的内容。

\*/

var xmlHttp;

function showHint(str)

{

if (str.length==0)

{

document.getElementById("txtHint").innerHTML=""

return

}

xmlHttp=GetXmlHttpObject()

if (xmlHttp==null)

{

alert ("Browser does not support HTTP Request")

return

}

var url="gethint.php";

url=url+"?q="+str;

url=url+"&sid="+Math.random();

xmlHttp.onreadystatechange=stateChanged ;

xmlHttp.open("GET",url,true);

xmlHttp.send(null);

//document.getElementById("txtHint").innerHTML=xmlHttp.responseText;

}

/\*

stateChanged() 函数

每当 XMLHTTP 对象的状态发生改变，

则执行该函数。在状态变成 4 （或 "complete"）时，

用响应文本填充 txtHint 占位符 txtHint 的内容。

\*/

/\*

发送一个请求后，

客户端无法确定什么时候

会完成这个请求，

所以需要用事件机制来捕获请求的状态，

XMLHttpRequest 对象提供了

onreadyStateChange 事件实现这一功能。

这类似于回调函数的做法。

onreadyStateChange

事件可指定一个事件处理函数来处理

XMLHttpRequest 对象的执行结果

onreadyStateChange 事件是在

readyState 属性发生改变时触发的，

readyState 的值表示了当前请求的状态，

在事件处理程序中可以根据这个值

来进行不同的处理。

readyState 有五种可取值

0：尚未初始化，

1：正在加载，

2：加载完毕，

3：正在处理；

4：处理完毕。

一旦 readyState 属性的值变成了 4，

就可以从服务器返回的响应数据

进行访问了。

\*/

function stateChanged()

{

var str;

if (xmlHttp.readyState==4 || xmlHttp.readyState=="complete")

{

str = xmlHttp.responseText;

if (str=='用户 ID 重复，请重新输入!')

{

document.getElementById("txtHint").innerHTML= str;

document.getElementById("userid").focus();

}

else

{

document.getElementById("form1").submit();

}

}

} /\*

GetXmlHttpObject() 函数

AJAX 应用程序只能运行在完整支持 XML 的 web 浏览器中。

上面的代码调用了名为 GetXmlHttpObject() 的函数。

该函数的作用是解决为不同浏览器创建不同 XMLHTTP 对象的问题。

\*/

function GetXmlHttpObject()

{

var xmlHttp=null;

try

{

// Firefox, Opera 8.0+, Safari

xmlHttp=new XMLHttpRequest();

}

catch (e)

{

// Internet Explorer

try

{

xmlHttp=new ActiveXObject("Msxml2.XMLHTTP");

}

catch (e)

{

xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");

}

}

return xmlHttp;

}

Gethint.php 连接数据库查重

<?php

$con = mysql\_connect("localhost","root","sql");

mysql\_select\_db("qing\_zhou", $con); mysql\_query("set names gb2312 ");

$q=$\_GET["q"];

$SQL ="select count(\*) from php\_admin where userid='$q'";

$RS0 = mysql\_query($SQL);

$RS = mysql\_fetch\_array($RS0);

if ($RS[0]>=1)

{

echo "用户 ID 重复，请重新输入!";

}

mysql\_close($con); $RS=NULL; $Con =NULL;

?>

Userlogin.php 用户登录 session

<?php

header("Cache-Control: no-cache, post-check=0, pre-check=0");

header("Content-type:text/html;charset=gb2312");

session\_start();

$\_SESSION['userid']='';

$\_SESSION['usertype']='';

$userid = $\_POST['userid'];

$password = $\_POST['password'];

$con = mysql\_connect("localhost","root","sql");

mysql\_select\_db("qing\_zhou", $con);

mysql\_query("set names gb2312 ");

$sql1 = "select userid,usertype from php\_admin where userid='".$userid."' and password='".$password."'";

//echo $sql1; die();

$RS0 = mysql\_query($sql1, $con);

if (mysql\_num\_rows($RS0)>0)

{

    $RS = mysql\_fetch\_array($RS0);

    $\_SESSION['userid'] = $RS['userid'];

    $\_SESSION['usertype'] = $RS['usertype'];

    mysql\_close($con);

    $RS=NULL;

    $con =NULL;

    header("location: main.php");

}

else

{

    mysql\_close($con);

    $RS=NULL;

    $con =NULL;

    header("location: userlogin.htm");

}

?>

Loginexit.php 用户注销session

<?php

    session\_start();

    unset($\_SESSION['userid']);

    unset($\_SESSION['usertype']);

    session\_destroy();

    header("location: userlogin.htm");

?>