

HDJS 前端库 Beta2014.1

开发代号：盾友

HDJS 前端库是 100% 自由的，您可以尽情享受和分享

为友爱而生

后盾网 人人做后盾

2012-2014

HDJS 前端库

敏捷开发利器，让开发速度更快

作者：后盾网向军

HDJS 前端库为友爱而生

任何人都可以加入和扩展 HDJS 库，我们会在后盾网论坛公示你的信息，让更多人通过 HDJS 库了解到你

后盾网 www.houdunwang.com HDPHP 官网 www.hdphp.com

社区开发团队 等待你的加入

目录

1. HDJS 前端库.....	5
2. 自动验证 (支持 Ajax)	6
2_1 验证规则	6
2_2 简单的使用	7
2_3 is_validate() 判断验证是否通过	8
2_4 自定义提示信息的验证	9
2_5 Ajax 验证规则	10
2_6 验证两次输入密码是否相等	13
2_7 改变提示信息 span 标签的位置	14
3. CSS 样式	15
3_1 尺寸	15
3_2 表单样式	15
3_3 按钮样式	16
3_4 position-bottom 底部固定	17
3_5 title-header 标题块	17
3_6 wrap 页面背景块	17
4. 简单提示框	18
5. 轮换版 slide	19
6. 弹窗 dialog	21
7. 模态对话框	22
7_1 添加模态对话框	22
7_2 移除模态对话框	25
8. 表格 UI	26

9. Tab 选项卡	27
10. 菜单导航 (menu_list)	29
11. 日历插件	30
12. form 表单 Ajax 提交	31
12_1 hd_submit (没有确定按钮的提交)	31
12_2 hd_submit_confirm (有确定按钮的提交)	32
13. 扩展函数	33
13_1 获得对象在页面中心的位置	33
13_2 关闭窗口	33
13_3 过滤空数组	33
13_4 笛卡尔乘积	33

1. HDJS 前端库

后盾网推出 hdjs 的目的就是帮助开发者更简单、快速的实现前台功能的实现。使用 hdjs 是没有难度的，调用非常简单。同时后盾网会提供视频教程让大家轻松使用 hdjs。

注意：

1. hdjs 可以单独使用（不需要安装 hdphp）
2. 使用 hdphp 框架，请用 `<hdjs/>` 调用标签即可
3. 必须引入 jquery(如果以 hdphp 中的 `<hdjs/>` 标签调用，可忽略这步)

2. 自动验证 (支持 Ajax)

HDJS 提供了自动验证处理机制, 对前台表单的验证提供了非常方便的操作, 提高前台表单验证效率。

需要的文件

```
1 <script type='text/javascript' src='./hdjs/js/jquery-1.8.2.min.js'></script>
```

```
2 <link href='./hdjs/css/hdjs.css' rel='stylesheet'>
```

```
3 <script src='./hdjs/js/hdjs.js'></script>
```

如果使用 HDPHP 框架, 以上 3 行可以替换为下面 1 行

```
<hdjs/>
```

为了节省手册篇幅, 以下示例代码将省略外部文件引用, 请自行引入

注:

1. 使用 `hdphp<hdjs/>` 标签引入, 以上 3 个文件会自动加载, 不需要手动指定
2. 如果页面有多个 form 时, 可以为 form 加 id: 如 `$("#login").validate(...)`

2_1 验证规则

验证规则	说明	参数值
required	是否必填	true 或 false
email	验证是否为邮箱	true
http	验证是否为网址	true
identity	验证身份证	true
tel	验证固定电话	true
phone	验证手机	true
user	验证用户名	true
maxlen	最大数值	maxlen:10 表示不能大于 10
minlen	最小长度	minlen:68 表示不能小于 68
num	数字范围	num:"2,20" 表示只能输入 2 到 20
regex	自定义正则	正则如 <code>/^\d+\$/</code>

验证规则	说明	参数值
confirm	验证两个字段相等	表单 name 名, 如: confirm:"password"
china	验证中文	true
ajax	异步验证	ajax:"hdphp.php" php 脚本返回 1 表示验证通过, 否则为失败

2_2 简单的使用

```
1 <script>
2   $(function () {
3       $("#test").validate({
4           // 验证规则
5           title: {
6               rule: {
7                   required: true
8               },
9               success: ' 操作成功 '
10          }
11      })
12  })
13</script>
14<form id='test' class="hd-form">
15    <table class="table">
16        <tr>
17            <td width="200"> 网址 </td>
18            <td><input type="text" name="title"/></td>
19        </tr>
20    </table>
21    <input type="submit" value="submit"/>
22</form>
```

网址

✕ 输入错误

submit

2_3 is_validate() 判断验证是否通过

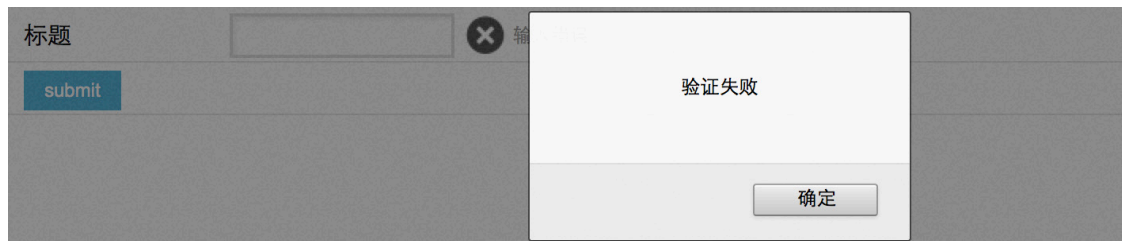
is_validate() 方法用于验证表单是否输入正确，在异步 Ajax 提交时必须使用的方法。

注：在 Ajax 异步提交时，必须首先使用 is_validate() 进行验证

```
1 <script>
2     // 验证规则
3     $(function () {
4         $("#test").validate({
5             // 验证规则
6             title: {
7                 rule: {
8                     required: true
9                 },
10                success: ' 操作成功 '
11            }
12        })
13    })
14    // 提交表单
15    $(function () {
16        $("form").submit(function () {
17            if ($(this).is_validate()) {
18                alert(" 验证通过 ");
19            } else {
20                alert(" 验证失败 ");
21            }
22            return false;
23        })
24    })
25</script>
26<form id='test' action="#" class="hd-form">
27    <table class="table">
28        <tr>
29            <td width="200"> 网址 </td>
```



```
30      <td><input type="text" name="title"/></td>
31    </tr>
32 </table>
```



2_4 自定义提示信息的验证

```
1 <script>
2   $(function () {
3     $("form").validate({
4       username: {
5         rule: {
6           user: "6,20",
7           required: true
8         },
9         error: {
10          user: " 不能小于 6 个字符 ",
11          required: " 用户名不能为空 "
12        },
13        message: " 用户名长度 6 到 20 位 ",
14        success: " 用户名正确 "
15      },
16      email: {
17        rule: {
18          email: true
19        },
20        error: {
21          email: " 邮箱格式错误 "
22        },
23        message: " 请输入邮箱 ",
```

```

24         success: " 邮箱输入正确 "
25     }
26 })
27 })
28 </script>
29 <form action="houdunwang.php" method="post" class="hd-form">
30     <table class="table2">
31         <tr>
32             <td width="100"> 用户名 </td>
33             <td><input type="text" name="username"/></td>
34         </tr>
35         <tr>
36             <td> 邮箱 </td>
37             <td><input type="text" name="email"/></td>
38         </tr>
39     </table>
40     <input type="submit" class="btn1"/>
41 </form>

```

用户名	<input type="text"/>	✖ 用户名不能为空
邮箱	<input type="text" value="hdjs"/>	✖ 邮箱格式错误

2_5 Ajax 验证规则

很多时候我们需要使用 Ajax 异步验证，比如验证码，用户名检测，标题重复检测等，这时我们可以使用 HDJS 的 Ajax 异步验证功能，下面我们会通过实例教会你使用。

只传递表单值：

ajax: 请求的 url 地址

多参数传递

ajax: {

url: "[U:'check']", // 请求的 Url

field:["title","click"] ,// 将字段 title、click 的值做为 post 参数传递

```
data: {webname:"houdunwang"}// 传递 post 数据 webname:houdunwang
}
```

示例（只传递表单值）

```
1  <script>
2      $(function () {
3          $("form").validate({
4              // 验证规则
5              code: {
6                  rule: {
7                      required: true,
8                      ajax: "check_code"
9                  },
10                 error: {
11                     required: " 验证码不能为空 ",
12                     ajax: " 验证码输入错误 "
13                 },
14                 message: " 请输入验证码 ",
15                 success: " 输入正确 "
16             }
17         })
18     })
19 </script>
20 <form action="houdunwang.php" method="post" class="hd-form">
21     <table>
22         <tr>
23             <td width="100"> 验证码 </td>
24             <td>
25                 <input type="text" name="code"/>
26                 
27                 <span id="hd_code"></span>
28             </td>
29         </tr>
```

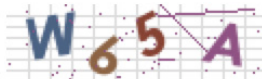
```
30 </table>
31 <input type="submit" class="btn1"/>
32</form>
```

check_code.php 页面

```
1 if(strtoupper($_POST['data'])==$_SESSION['data']){// 验证通过
2     echo 1;
3 }else{// 验证失败
4     echo 0;
5 }
6 exit;
```

验证码

hdjs



验证码输入错误

提交查询

示例（Ajax 验证时传递更多数据）

```
1 <script>
2     $(function () {
3         $("form").validate({
4             // 验证规则
5             title: {
6                 rule: {
7                     required: true,
8                     ajax: {
9                         url: "test.php", // 请求的 Url
10                        field:["title","click"] ,// 将字段 title、click 的值做为 post 参数传递
11                        data: {webname:"houdunwang"}// 传递 post 数据
12                    }
13                },
14                error: {
15                    required: "验证码不能为空 ",
16                    ajax: " 输入错误 "
17                },
```

```
18         message: " 请输入验证码 ",
19         success: " 输入正确 "
20     }
21 })
22 })
23</script>
```

2_6 验证两次输入密码是否相等

```
1 <script>
2     $(function () {
3         $("form").validate({
4             password: {
5                 rule: {
6                     required: true
7                 }
8             },
9             password2: {
10                rule: {
11                    confirm: "password"
12                },
13                error: {
14                    confirm: " 确认密码输入错误 "
15                }
16            }
17        })
18    })
19</script>
20<form action="reg.php" method="post" class="hd-form">
21    <table class="table1">
22        <tr>
23            <td width="100"> 密码 </td>
24            <td><input type="password" name="password"/></td>
```

```
25     </tr>
26     <tr>
27         <td> 确认密码 </td>
28         <td><input type="password" name="password2"/></td>
29     </tr>
30 </table>
31 <input type="submit" class="btn1"/>
32 </form>
```

密码	<input type="password"/>
确认密码	<input type="password"/>
<div>提交查询</div>	

 确认密码输入错误

2_7 改变提示信息 span 标签的位置

默认情况下 hdjs 会自动创建 `` 标签，用于显示提示信息。但有些场景我们想把提示信息放在不同的地方。这样的需求处理起来也很简单。

比如说，我们有表单 `<input type="text" name="web"/>` 需要验证，只需要在页面中创建 `` 标签就可以，验证的信息为自动放入此 `span` 标签中。

注：span 标签的 id 值 就是表单 name 名前加 hd_

3. CSS 样式

3_1 尺寸

在 hdjs 中已经内容了常用的宽度与高度，比如使用 class='w100' 表示设置宽度为 100px

宽度

.w30	.w40	.w50	.w60	.w70	.w80	.w90	.w100
.w150	.w180	.w200	.w250	.w300	.w350	.w400	.w450
.w500	.w550	.w600	.w650	.w700	.w750	.w800	.w850

高度

.h60	.h70	.h80	.h90	.h100	.h120	.h150	.h180
.h200	.h250	.h300	.h350	.h400	.h450	.h500	.h600

浮动样式

class='l' 等于 float:left

class='r' 等于 float:right

3_2 表单样式

使用 hdjs 的表单样式需要在表单的父级（如 form 标签）设置为 hd-form

示例

```
1 <table class="table1 hd-form">
2   <tr>
3     <td width="100"> 栏目 </td>
4     <td>
5       <select name="pid">
6         <option value="0"> 一级栏目 </option>
7       </select>
8     </td>
9   </tr>
10  <tr>
11    <td> 栏目名称 </td>
```

```

12      <td>
13          <input type="text" name="catname" class="w200"/>
14      </td>
15  </tr>
16  <tr>
17      <td> 静态目录 </td>
18      <td>
19          <input type="text" name="catdir" class="w200"/>
20      </td>
21  </tr>
22</table>

```

栏目	<input type="text" value="一级栏目"/>
栏目名称	<input type="text"/>
静态目录	<input type="text"/>

3_3 按钮样式

HDJS 提供了丰富的按钮样式，只设置相应 class 属性为 btn1、btn2、btn3、btn4、btn5 即可

示例

```

1 <input type="submit" value="确定" class="btn1"/>
2 <input type="submit" value="确定" class="btn2"/>
3 <input type="submit" value="确定" class="btn3"/>
4 <input type="submit" value="确定" class="hd-success"/>
5 <input type="submit" value="确定" class="hd-cancel"/>

```



3_4 position-bottom 底部固定

position-bottom 样式用于将 div 固定到页面底部，主要用于放置按钮。

效果图

	<input type="text"/>
属性类型	<input checked="" type="radio"/> 文本框 <input type="radio"/> 下拉列表框
规格	<input checked="" type="radio"/> 不是 <input type="radio"/> 是
属性值	<input type="text"/>

示例:

```
1 <div class="position-bottom">
2   <input type="submit" class="hd-success" value=" 确定 " />
3 </div>
```

3_5 title-header 标题块

效果图

属性设置

属性名称	<input type="text"/>	<input checked="" type="radio"/> 属性名不能为空
------	----------------------	------------------------------------------

示例

```
<div class="title-header"> 属性设置 </div>
```

3_6 wrap 页面背景块

示例:

```
<div class="wrap"></div>
```

4. 简单提示框

这是最简单的提示信息框，会在指定的时间内自动消失（默认为 2 秒）

示例：

```
1 <script>  
2     $(function () {  
3         dialog_message(' 操作成功 ',10);  
4     })  
5 </script>
```



操作成功

5. 轮换版 slide

前台页面我们经常使用的轮换版，使用 slide.js 可以快速的创建页面轮换版。

说明：

1. 必须引入 slide.js 与 jquery 文件
2. 如果使用了 HDPHP 框架中的 <hdjs/> 标签则不需要引入 slide.js

语法：

`$("#id").slide(参数);`

参数说明

参数	说明
width	轮换版宽度
height	轮换版高度
timeout	切换时间（单位：秒）
bacolor	文字背景颜色

效果图



示例

```
1 <script>
2 $(function () {
3     $('#flash').slide({
4         width: 278,
5         height: 215,
6         timeout: 3,
```

```
7         bgcolor:'#f00f00'
8     });
9 })
10 </script>
11 <div id="flash">
12 <a href="http://www.hdphp.com" title="后盾网 人人做后盾 "></a>
13 <a href="http://www.hdphp.com" title="HDJS 前端开发利器 "></a>
14 </div>
```

6. 弹窗 dialog

语法: `$.dialog({'msg':" 提示信息 ",type:"success"});`

参数说明

msg	提示信息
type	对话框类型, error 错误 success 正确
timeout	自动关闭时间 (单位: 秒)

效果



示例

```
1 <script>
2 $(function(){
3   $.dialog({
4     "msg":" 欢迎使用后盾网 HDJS 前端库 ",
5     "type":"success",
6     "timeout":2,
7     "close_handler":function(){
8       alert(" 这是窗口关闭时执行的回调函数 ");
9     }
10  });
11 })
12 </script>
```

7. 模态对话框

7_1 添加模态对话框

有些业务我们需要弹出一个对话框来输入数据，那么模态是非常好的选择。

语法：\$.modal({width:600,height:300,content:" 后盾网 "})

参数说明

属性名	说明
title	标题内容
width	宽度
height	高度
content	显示内容
button_success	确定按钮文字
button_cancel	关闭按钮文字
success	点击确定后执行的事件
cancel	点击关闭后执行的事件
message	提示信息

示例 1 (标准模态)

```
1 <script>
2     $(function () {
3         $.modal({
4             width: 550,
5             height: 300,
6             title:" 欢迎使用 HDJS 库 ",
7             button: true,
8             button_success: " 确定 ",
```

```

9      button_cancel: " 关闭 ",
10      success: function () {
11          $.removeModal();
12      },
13      content: '<h3> 后盾网人人做后盾 </h3><br/>' +
14          '<form action="reg.php" method="post" class="hd-form">' +
15          '<table class="table1">' +
16          '<tr><td> 盾友名子 : </td><td><input type="text" name="username"/></td></tr>' +
17          '<tr><td> QQ:</td><td><input type="text" name="qq"/></tr>' +
18          '<tr><td colspan="2"><input type="submit" class="btn1"/></td></tr>' +
19          '</table>' +
20          '</form>'
21      });
22  })
23</script>

```

效果



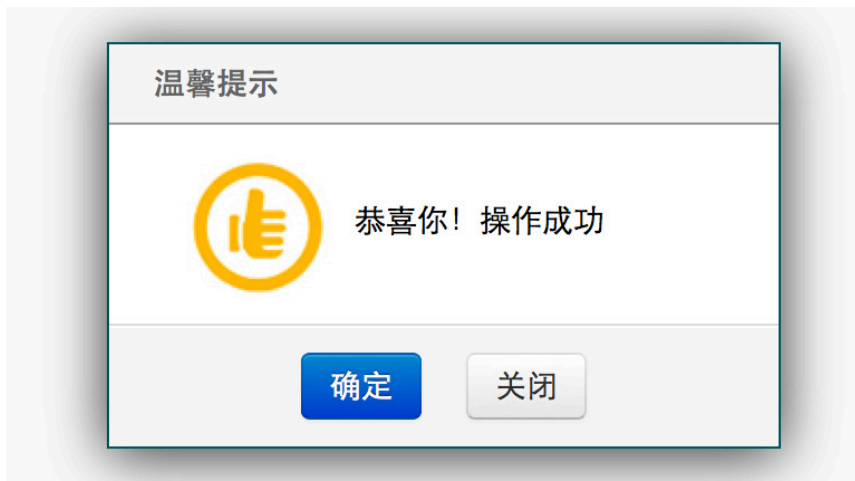
示例 2（确认框）

```

1 <script>
2     $(function () {
3         $.modal({
4             width: 300,

```

```
5      height: 180,  
6      title: " 温馨提示 ",  
7      message: ' 恭喜你! 操作成功 ',  
8      button_success: " 确定 ",  
9      button_cancel: " 关闭 ",  
10     type:'success',// 类型  
11     success: function () {  
12         $.removeModal();// 关闭模态  
13     }  
14 });  
15 })  
16</script>
```



如果将 type 属性设置为 error 显示样式如下



7_2 移除模态对话框

关闭模态对话可以通过调用函数实现。

示例:

```
$.removeModal();
```

8. 表格 UI

Hdjs 提供 3 种表格样式，只需要定义 table 的 class 属性为 table1、table2、table3 中的任意一个就可以，以下是依次为 table1、table2、table3 的效果。

后盾官网	http://www.houdunwang.com
后盾论坛	http://bbs.houdunwang.com

后盾官网	http://www.houdunwang.com
后盾论坛	http://bbs.houdunwang.com

后盾官网	http://www.houdunwang.com
后盾论坛	http://bbs.houdunwang.com

9. Tab 选项卡

Hdjs 提交了方便的 Tab 选项卡 UI，需要 jquery 支持，以下是示例演示。

```
1 <div class="tab">
2   <ul class="tab_menu">
3     <li lab="base">
4       <a> 基本设置 </a>
5     </li>
6     <li lab="tpl" class="action">
7       <a> 后盾网教师管理 </a>
8     </li>
9     <li>
10      <a href="http://houdunwang.com" target="_blank"> 后盾网 </a>
11    </li>
12  </ul>
13  <div class="tab_content">
14    <div id="base">
15      <table class="table1 hd-form">
16        <tr>
17          <td> 标题 </td>
18          <td>
19            <input type="text" name="title"/>
20          </td>
21        </tr>
22        <tr>
23          <td> 作者 </td>
24          <td>
25            <input type="text" name="author"/>
26          </td>
27        </tr>
28        <tr>
29          <td colspan="2">
```

```
30         <input type="submit" value=" 确定 " class="btn1"/>
31     </td>
32 </tr>
33 </table>
34 </div>
35 <div id="tpl">
36     教师管理
37 </div>
38 </div>
39 </div>
```

基本设置

后盾网教师管理

后盾网

标题

作者

确定

10. 菜单导航 (menu_list)

示例

```
1 <div class="menu_list">
2   <ul>
3     <li><a href="{|U:'index'}"> 栏目列表 </a></li>
4     <li><a href="javascript:;" class="action"> 添加栏目 </a></li>
5     <li><a href="javascript:update_cache();"> 更新栏目缓存 </a></li>
6   </ul>
7 </div>
```

栏目列表

添加栏目

更新栏目缓存

11. 日历插件

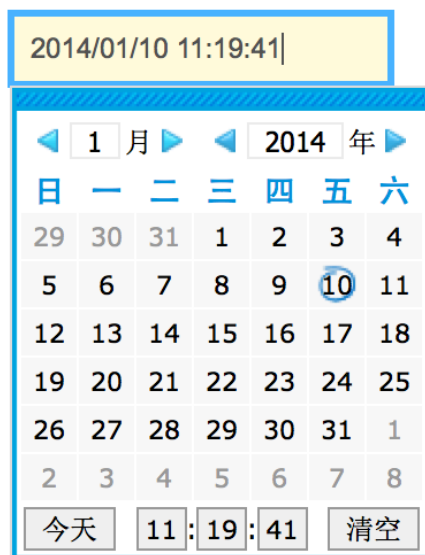
注:

1. 必须加载 JQuery
2. 如果使用 hdphp 框架中的 <hdjs/> 标签时, 则不用引入任何文件

示例

```
1 <script src='jquery-1.8.2.min.js'></script>
2 <script src='org/cal/lhgcalendar.min.js'></script>
3 <input type="text" readonly="readonly" id="updatetime" name="updatetime"
4   value="{[date:'Y/m/d h:i:s']}"
5   class="w150"/>
6 <script>
7   $('#updatetime').calendar({format: 'yyyy/MM/dd HH:mm:ss'});
8 </script>
```

效果图



12. form 表单 Ajax 提交

开发中我们经常需要用到 Ajax 异步提交表单 (form)，为了提高开发效率，HDJS 提供了方便、快速的 Ajax 异步提交机制，你一定会喜欢的。我们通过示例来学习使用方法。

12_1 hd_submit（没有确定按钮的提交）

效果图



前台 html 页面

```
1 <form action="add.php" class="hd-form" onsubmit="return hd_submit(this,'http://bbs.
  houdunwang.com')">
2   <table>
3     <tr>
4       <td><input type="text" name='title' class="w150"/></td>
5     </tr>
6     <tr>
7       <td colspan="2"><input type="submit" class="hd-success"/></td>
8     </tr>
9   </table>
10 </form>
```

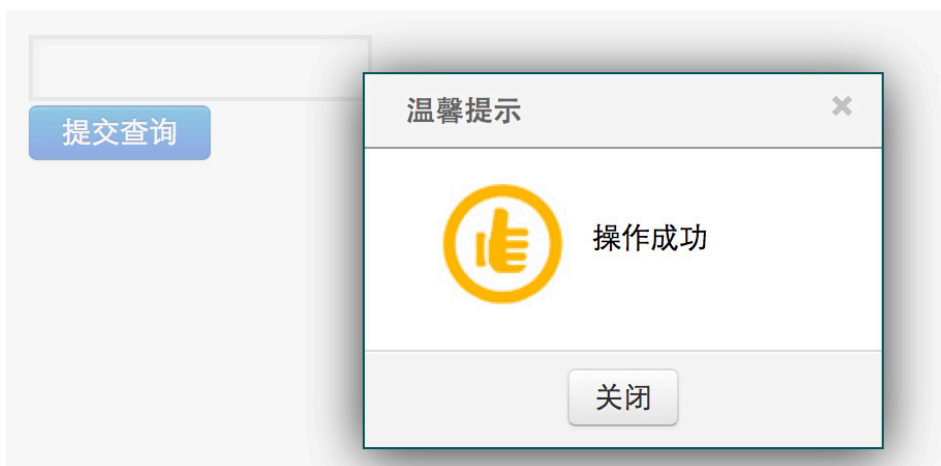
服务器 php 脚本

```
1 class indexControl extends Control
2 {
3     public function add()
4     {
5         $return = array(
6             'state' => 1,
```

```
7      'message' => ' 操作成功 '
8  );
9      $this->ajax($return);
10 }
11 }
```

12_2 hd_submit_confirm（有确定按钮的提交）

效果图



前台 html 页面

与上面的布局一样，只需要将 onsubmit 的方法修改为 hd_submit_confirm 即可

服务器 php 脚本

与上面的 php 代码一样

13. 扩展函数

13_1 获得对象在页面中心的位置

语法: `function center_pos(obj)`

@param obj 用于操作的对象

13_2 关闭窗口

`function _close(msg)`

@param msg 提示信息

13_3 过滤空数组

`$.array_filter([1,,3])`

13_4 笛卡尔乘积

`function descarte(list)`

示例

```
var result = descartes({'a':['a','b'],'b':[1,2]});
```
