

ASP 表单和用户输入

Request.QueryString 和 Request.Form 命令用于从表单取回信息，比如用户的输入。



尝试一下 - 实例

[使用 method="get" 的表单](#)

本例演示如何使用 Request.QueryString 命令与用户进行交互。

[使用 method="post" 的表单](#)

本例演示如何使用 Request.Form 命令与用户进行交互。

[使用单选按钮的表单](#)

本例演示如何使用 Request.Form 命令通过单选按钮与用户进行交互。

用户输入

Request 对象可用于从表单取回用户信息。

HTML 表单实例

```
<form method="get" action="simpleform.asp">
First Name: <input type="text" name="fname"><br>
Last Name: <input type="text" name="lname"><br><br>
<input type="submit" value="Submit">
</form>
```

用户输入可通过 Request.QueryString 或 Request.Form 命令取回。

Request.QueryString

Request.QueryString 命令用于收集使用 method="get" 的表单中的值。

使用 GET 方法从表单传送的信息对所有的用户都是可见的（出现在浏览器的地址栏），并且对所发送信息的量也有限制。

如果用户在上面的 HTML 表单中输入 "Bill" 和 "Gates"，发送至服务器的 URL 会类似这样：

```
http://www.w3cschool.cc/simpleform.asp?fname=Bill&lname=Gates
```

假设 "simpleform.asp" 文件包含下面的 ASP 脚本：

```
<body>
Welcome
<%
response.write(request.querystring("fname"))
response.write(" " & request.querystring("lname"))
%>
</body>
```

浏览器将把文档的 body 部分显示如下：

```
Welcome Bill Gates
```

Request.Form

Request.Form 命令用于收集使用 method="post" 的表单中的值。

使用 POST 方法从表单传送的信息对用户是不可见的，并且对所发送信息的量没有限制。

如果用户在上面的 HTML 表单中输入 "Bill" 和 "Gates"，发送至服务器的 URL 会类似这样：

```
http://www.w3cschool.cc/simpleform.asp
```

假设 "simpleform.asp" 文件包含下面的 ASP 脚本：

```
<body>
Welcome
<%
response.write(request.form("fname"))
response.write(" " & request.form("lname"))
%>
</body>
```

浏览器将把文档的 body 部分显示如下：

```
Welcome Bill Gates
```

表单验证

只要有可能，就尽量在浏览器上对用户的输入进行验证（通过客户端脚本）。浏览器的验证速度更快，并可以减少服务器的负载。

如果用户输入会保存到数据库中，那么您应该考虑使用服务器端验证。有一种在服务器端验证表单的好方法，就是将（验证过的）表单传回表单页面，而不是转至不同的页面。用户随后就可以在同一个页面中得到错误的信息。这样做更易于用户发现错误。

[← ASP 程序](#)[ASP Cookies →](#)[✍ 点我分享笔记](#)