

LiveCode Lessons (copy)

Search

Topics

[Installing LiveCode Server](#) 6[Using LiveCode Server](#) 4[Interacting with LiveCode Server](#) 6[How do I pass information to LiveCode Server scripts?](#)[How do I handle user input using LiveCode Server?](#)[How Do I Use AJAX with LiveCode Server? \(Current Article\)](#)[How to upload a file with LiveCode Server](#)[How do I use Cookies on LiveCode Server?](#)[Accessing Web Services using LiveCode](#)

Last Updated

May 30, 2018

[Print Article](#)

Other Resources

Getting Started with LiveCode

[Get Up and Running with LiveCode](#)[Getting Started with LiveCode Development](#)

LiveCode Lessons

[How To - Step-By-Step Guides To Tasks In LiveCode](#)[How To - LiveCode Server Tasks](#)[How To - LiveCode Mobile Tasks](#)[How To - LiveCode Sample Scripts](#)[How to - LiveCode Marketplace Products](#)[How to Purchase and License LiveCode](#)

Tutorials

[Creating a Video Library Application](#)

Data Grid

[LiveCode Data Grid](#)[Data Grid Tips & Tricks](#)[Converting the Stock Program](#)

LiveCode Releases

[LiveCode 6.5](#)[LiveCode 6.7](#)[LiveCode 8](#)

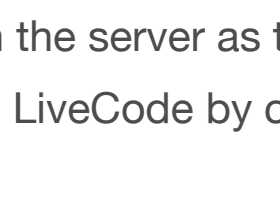
Comments

[Link error: internal](#) for this article[LiveCode Lessons \(copy\)](#) / [LiveCode Lessons](#) / [How To - LiveCode Server Tasks](#) / [Interacting with LiveCode Server](#) / [How Do I Use AJAX with LiveCode Server?](#)

How Do I Use AJAX with LiveCode Server?

This lesson will show you the basics of using AJAX with LiveCode Server by demonstrating how to create a simple auto-complete box. You can see the working demo [here](#).

What is AJAX?

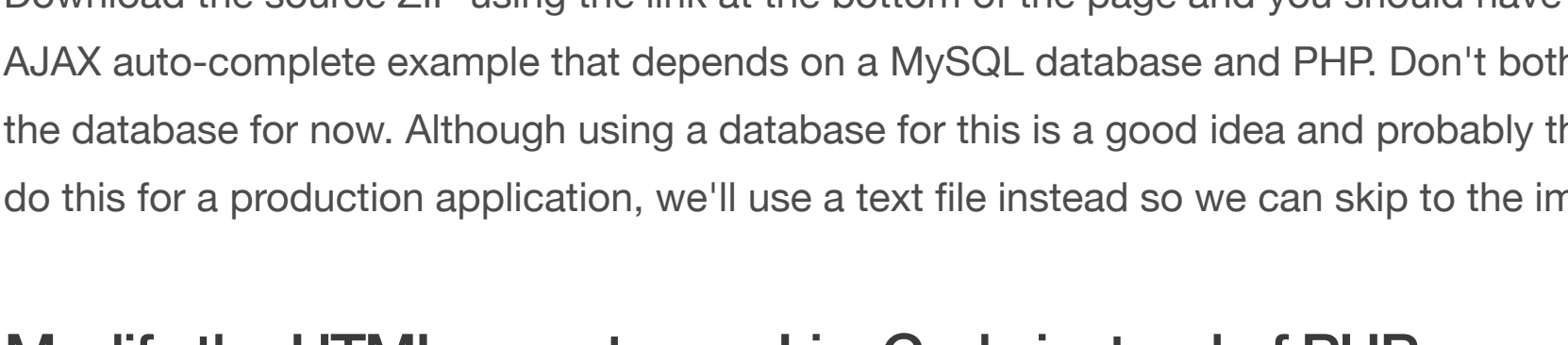


The term "AJAX" refers to a web programming technique where JavaScript is used to perform an HTTP request from a URL. This allows dynamic content to be loaded from a database on the webserver without refreshing the page. It is very common in modern web sites and applications.

One of the most popular uses for AJAX is auto-completion. The user types text into a field, and the JavaScript automatically queries a script on the server as they type to suggest possible completions. In this lesson we show how to do this using LiveCode by creating a text field where the user enters their country.

This lesson uses the jQuery library to make the JavaScript as easy as possible. jQuery is a widely used library with good [documentation at their site](#).

Get a HTML page with an auto-complete box



As this is a LiveCode Server lesson, we don't spend too much time looking at the basic HTML page. Instead we just take it from an existing AJAX auto-complete tutorial from [nostrum.com](#).

Download the source ZIP using the link at the bottom of the page and you should have a fully working AJAX auto-complete example that depends on a MySQL database and PHP. Don't bother setting up the database for now. Although using a database for this is a good idea and probably the best way to do this for a production application, we'll use a text file instead so we can skip to the important parts.

Modify the HTML page to use LiveCode instead of PHP

Open up the HTML page in a text editor and find the JavaScript section near the top. There is a function called "lookup". This needs to be replaced with our own version which uses a LiveCode page. The new code is shown below. Note that we've split off the callback into a separate function for clarity and changed the AJAX request a little, but the logic from the original script stays intact.

```
function lookup(inputString)

{

    if (inputString.length == 0)

    {

        // Hide the suggestion box.

        $('#suggestions').hide();

    }

    else

    {

        $.ajax({

            url: "autocomplete.lc?string=" + inputString,

            type: "GET",

            success: function(data) { lookupCallback(data); }

        });

    }

} // lookup


function lookupCallback(data)

{

    if(data.length > 0)

    {

        $('#suggestions').show();

        $('#autoSuggestionsList').html(data);

    }

}
```

Create the LiveCode file that handles the AJAX request

In the autocomplete folder, next to the other files including index.htm, create a file called "autocomplete.lc". This will replace the file "rpc.php". Edit this with a text editor, and place the following LiveCode script into it:

```
<?lc

function getAutoComplete p_string

local tCompletionsFile

put "/home/oliverk/public_html/examples/autocomplete/countries.txt" into tCompletionsFile

local tCompletions

put url ("file:" & tCompletionsFile) into tCompletions

filter tCompletions with (p_string & "**")

local tTemplate

put "<li onClick=fill('[[tCompletion]]');[[quote]]>[[tCompletion]]</li>" into tTemplate

local tOutput

repeat for each line tCompletion in tCompletions

put merge(tTemplate) & return after tOutput

end repeat

delete the last char of tOutput

return tOutput

end getAutoComplete

put getAutoComplete($_GET["string"])

?>
```

This code is called whenever the user enters some text in the field on the HTML page. It loads a text file which contains a list of possible countries (one-per-line) and filters out the countries which the user could be typing. It then returns appropriate HTML to display these.

Create the list of countries and upload to on-rev

Prev: [How do I handle user input using LiveCode Server?](#) Next: [How to upload a file with LiveCode Server](#)

5 Comments

Torsten

Sunday Mar 18 2018 at 09:49 PM

"Download the source ZIP using the link at the bottom of the page"... I cannot find it, who can help me?

Kind regards,

Torsten

Heather Laine

Tuesday Mar 20 2018 at 06:41 PM

Unfortunately i think the link originally used in this lesson is no longer available, and the sample stack used from an external source is gone. You will need to find some equivalent sample to work with perhaps by googling for it...

Marc

Thursday Apr 05 2018 at 08:39 PM

1. Get index.html:
wget <http://oliverk.on-rev.com/examples/autocomplete/index.html>
or use this code:

```
Ajax Auto Suggest

function lookup(inputString)
{
    if(inputString.length == 0)
    {
        // Hide the suggestion box.
        $('#suggestions').hide();
    }
    else
    {
        $.ajax({
            url: "autocomplete.irev?string=" + inputString,
            type: "GET",
            success: function(data) { lookupCallback(data); }
        });
    }
} // lookup


function lookupCallback(data)
{
    if(data.length > 0)
    {
        $('#suggestions').show();
        $('#autoSuggestionsList').html(data);
    }
}


function fill(thisValue) {
    $('#inputString').val(thisValue);
    setTimeout($('#suggestions').hide(), 200);
}

body {
    font-family: Helvetica;
    font-size: 11px;
    color: #000;
}

h3 {
    margin: 0px;
    padding: 0px;
}

.suggestionsBox {
    position: relative;
    left: 30px;
    margin: 10px 0px 0px 0px;
    width: 200px;
    background-color: #212427;
    -moz-border-radius: 7px;
    -webkit-border-radius: 7px;
    border: 2px solid #000;
    color: #fff;
}

.suggestionList {
    margin: 0px;
    padding: 0px;
}

.suggestionList li {
    margin: 0px 0px 3px 0px;
    padding: 3px;
    cursor: pointer;
}

.suggestionList li:hover {
    background-color: #659CD8;
}
```

Type your county:

```
2. Create file autocomplete.irev and input source code:

[[tCompletion]]" into tTemplate
local tOutput
repeat for each line tCompletion in tCompletions
put merge(tTemplate) & return after tOutput
end repeat
delete the last char of tOutput
return tOutput
end getAutoComplete
put getAutoComplete($_GET["string"])
?>
```

3. Optional, download the file countries.txt:
wget <http://oliverk.on-rev.com/examples/autocomplete/countries.txt>
And change line 'put "/home/oliverk/public_html/examples/autocomplete/countries.txt"' into
tCompletionsFile" into 'put "countries.txt" into tCompletionsFile"

4. Open a webbrowser and request the index.html file.

Torsten

Wednesday May 30 2018 at 04:11 PM

The URL of the demo was not correct, the extension was .html but the right one is .htm :-)
Try this: <http://oliverk.on-rev.com/examples/autocomplete/index.htm>

Heather Laine

Wednesday May 30 2018 at 11:04 PM

Thank you Torsten, that does seem to be correct. I will edit the lesson to add the link.

Add your comment

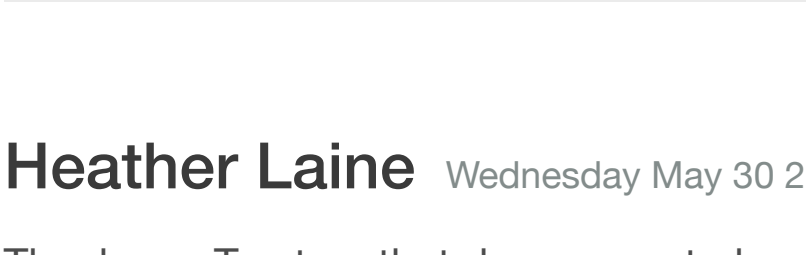
Name*

Email*

Comment*

Subscribe ☐ E-Mail me when someone replies to this comment

Are you human?



Submit Comment