

PHP Example - AJAX Live Search

[< Previous](#)[Next >](#)

AJAX can be used to create more user-friendly and interactive searches.

AJAX Live Search

The following example will demonstrate a live search, where you get search results while you type.

Live search has many benefits compared to traditional searching:

- Results are shown as you type
- Results narrow as you continue typing
- If results become too narrow, remove characters to see a broader result

Search for a W3Schools page in the input field below:

The results in the example above are found in an XML file ([links.xml](#)). To make this example small and simple, only six results are available.

Example Explained - The HTML Page

When a user types a character in the input field above, the function "showResult()" is executed. The function is triggered by the "onkeyup" event:

```
<html>
<head>
<script>
function showResult(str) {
    if (str.length==0) {
        document.getElementById("livesearch").innerHTML="";
        document.getElementById("livesearch").style.border="0px";
        return;
    }
    var xmlhttp=new XMLHttpRequest();
    xmlhttp.onreadystatechange=function() {
        if (this.readyState==4 && this.status==200) {
            document.getElementById("livesearch").innerHTML=this.responseText;
            document.getElementById("livesearch").style.border="1px solid #A5ACB2";
        }
    }
    xmlhttp.open("GET","livesearch.php?q="+str,true);
    xmlhttp.send();
}
</script>
</head>
<body>

<form>
<input type="text" size="30" onkeyup="showResult(this.value)">
<div id="livesearch"></div>
</form>

</body>
</html>
```

Source code explanation:

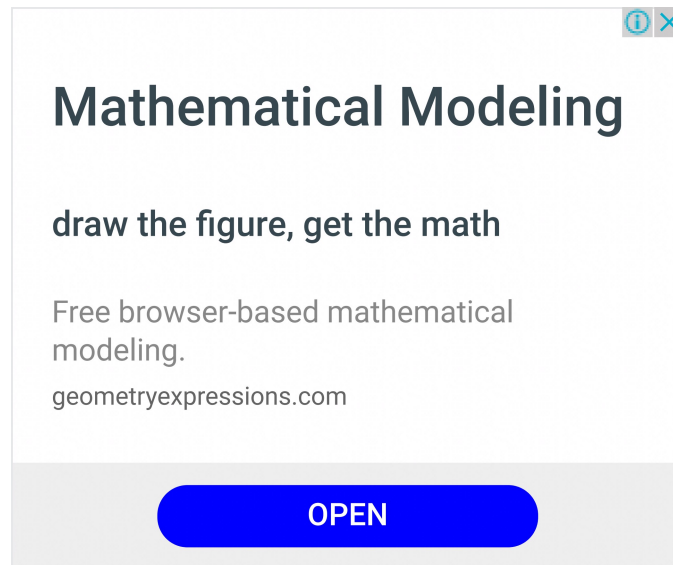
If the input field is empty (str.length==0), the function clears the content of the livesearch placeholder and exits the function.

If the input field is not empty, the showResult() function executes the following:

- Create an XMLHttpRequest object

- Create the function to be executed when the server response is ready
- Send the request off to a file on the server
- Notice that a parameter (q) is added to the URL (with the content of the input field)

ADVERTISEMENT



The PHP File

The page on the server called by the JavaScript above is a PHP file called "livesearch.php".

The source code in "livesearch.php" searches an XML file for titles matching the search string and returns the result:

```
<?php
$xmlDoc=new DOMDocument();
$xmlDoc->load("links.xml");

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

//get the q parameter from URL
$q=$_GET["q"];

//lookup all links from the xml file if length of q>0
if (strlen($q)>0) {
    $hint="";
    for($i=0; $i<($x->length); $i++) {
```

```

$y=$x->item($i)->getElementsByTagName('title');
$z=$x->item($i)->getElementsByTagName('url');
if ($y->item(0)->nodeType==1) {
    //find a link matching the search text
    if (stristr($y->item(0)->childNodes->item(0)->nodeValue,$q)) {
        if ($hint=="") {
            $hint="<a href='" .
            $z->item(0)->childNodes->item(0)->nodeValue .
            "' target='_blank'>" .
            $y->item(0)->childNodes->item(0)->nodeValue . "</a>";
        } else {
            $hint=$hint . "<br /><a href='" .
            $z->item(0)->childNodes->item(0)->nodeValue .
            "' target='_blank'>" .
            $y->item(0)->childNodes->item(0)->nodeValue . "</a>";
        }
    }
}
}
}
}

// Set output to "no suggestion" if no hint was found
// or to the correct values
if ($hint=="") {
    $response="no suggestion";
} else {
    $response=$hint;
}

//output the response
echo $response;
?>

```

If there is any text sent from the JavaScript (`strlen($q) > 0`), the following happens:

- Load an XML file into a new XML DOM object
- Loop through all <title> elements to find matches from the text sent from the JavaScript
- Sets the correct url and title in the "\$response" variable. If more than one match is found, all matches are added to the variable
- If no matches are found, the \$response variable is set to "no suggestion"

< Previous

Next >

ADVERTISEMENT



COLOR PICKER



SHOP

HOW TO

Tabs
Dropdowns
Accordions
Side Navigation
Top Navigation
Modal Boxes
Progress Bars
Parallax
Login Form
HTML Includes
Google Maps
Range Sliders
Tooltips
Slideshow
Filter List
Sort List

SHARE



CERTIFICATES

HTML

CSS

JavaScript

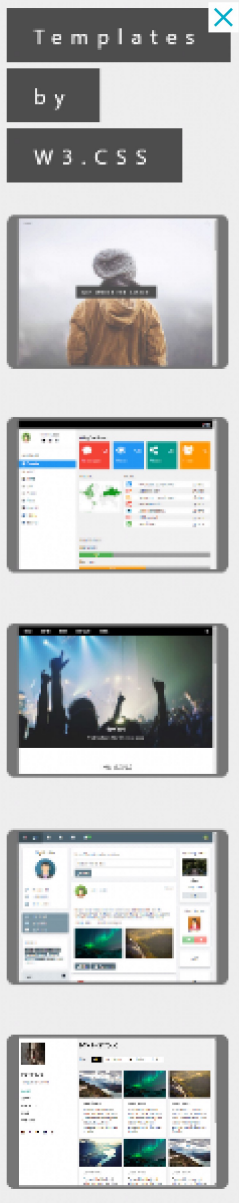
Python

SQL

PHP

And more

ADVERTISEMENT



ADVERTISEMENT

ADVERTISEMENT

[REPORT ERROR](#)[FORUM](#)[ABOUT](#)[SHOP](#)

Top Tutorials

[HTML Tutorial](#)
[CSS Tutorial](#)
[JavaScript Tutorial](#)
[How To Tutorial](#)
[SQL Tutorial](#)
[Python Tutorial](#)
[W3.CSS Tutorial](#)
[Bootstrap Tutorial](#)
[PHP Tutorial](#)
[Java Tutorial](#)
[C++ Tutorial](#)
[jQuery Tutorial](#)

Top References

[HTML Reference](#)
[CSS Reference](#)
[JavaScript Reference](#)
[SQL Reference](#)
[Python Reference](#)
[W3.CSS Reference](#)
[Bootstrap Reference](#)
[PHP Reference](#)
[HTML Colors](#)
[Java Reference](#)
[Angular Reference](#)
[jQuery Reference](#)

Top Examples

[HTML Examples](#)
[CSS Examples](#)
[JavaScript Examples](#)
[How To Examples](#)
[SQL Examples](#)
[Python Examples](#)
[W3.CSS Examples](#)
[Bootstrap Examples](#)
[PHP Examples](#)
[Java Examples](#)

XML Examples
jQuery Examples

Web Certificates

HTML Certificate
CSS Certificate
JavaScript Certificate
SQL Certificate
Python Certificate
PHP Certificate
Bootstrap Certificate
XML Certificate
jQuery Certificate

Get Certified »

W3Schools is optimized for learning and training. Examples might be simplified to improve reading and learning. Tutorials, references, and examples are constantly reviewed to avoid errors, but we cannot warrant full correctness of all content. While using W3Schools, you agree to have read and accepted our terms of use, cookie and privacy policy.

Copyright 1999-2020 by Refsnes Data. All Rights Reserved.
W3Schools is Powered by W3.CSS.

