

Worksheet 9 – Introduction to AJAX

1. Write an AJAX version of your HitTastic! search functionality. To do this:

a) write a new version of **searchresults.php** called **ajaxsearchresults.php**. This should work the same way as **searchresults.php** but should ONLY echo out HTML containing the actual search results – NOT the full page (<html>, <head>, <body>, stylesheets etc). This is because the AJAX search results will be included within a <div> on another page – **ajax.html** – described below.

(Alternatively, do this as a route in Slim)

b) write an **ajax.html** page containing the search box for the artist, and a <div> to place the results in.

c) write the JavaScript to actually send the AJAX request to your **ajaxsearchresults.php** and show the search results in the <div>. Make sure you link the HTML to the JavaScript as shown in the example on the notes.

2. Advanced question: AJAX download

Have a go at converting your ‘download’ functionality to use AJAX. To do this:

- Change your **ajaxsearchresults.php** to write out ‘download’ buttons. These should link, via a ‘click’ event, to another JavaScript function which should send a download request. For example:

```
echo "<input type='button' value='Download!'  
onclick='ajaxdownload(ID of current song)' />";
```

Note how this is working. We write out a button from PHP, and run the **ajaxdownload()** function on the client side when it’s clicked, passing in the ID of the current song.

In your JavaScript file, write this **ajaxdownload()** function to make an AJAX request to a new version of your **download.php**, called **ajaxdownload.php**. Use an AJAX POST request to send the song ID, and use **\$_POST** in **ajaxdownload.php**, rather than **\$_GET**. Make **ajaxdownload.php** echo out a message to confirm successful download, and in your AJAX callback function, show this to the user in another <div>.