

Bitly: Asynchronous Forms

Time	~ 3 - 4 hours
Learning Goals	<ul style="list-style-type: none">• Understand the difference between synchronous and asynchronous requests• Use jQuery to make an AJAX request

What is AJAX?

The user experience can often be improved with asynchronous actions. The goal here is to see and feel the difference between the traditional (synchronous) nature of HTML forms and the asynchronous nature of JavaScript and XML (AJAX) with our Bitly form.

For example, when you fill in a form on a website, say Facebook, and click on “post”, in a synchronous action, the site will be reloaded with your updated status. With AJAX, your status gets updated without the page being reloaded. What do you think has happened here? Here is a [good reading](#) of what AJAX means with a layman explanation.

Discuss with your partner on some examples of an AJAX request that you have come across. How does it enhance your user experience compared to a website without AJAX?

Implement an Asynchronous Bitly Submission

For this exercise, all your JavaScript will be in `public/js/application.js`.

We will be using the 5 steps suggested in the reading above:

1. [Bind a callback](#) to be triggered when [your form is submitted](#).
2. Prevent the form to be submitted in the default way ([preventDefault\(\)](#)).
3. [Serialize](#) the data in the form to be submitted.
4. Use jQuery's AJAX API to make a [POST request](#) to the server.
5. When the server responds, update the page accordingly (this is the asynchronous part you don't know when the server will respond) by displaying the new shortened URL link as a new row in the table.

Your code will look something like this:

```
$(document).ready(function(){  
  
    // this ID here refers to the form where the user types in a URL, you may have a different name for the ID. Edit the code accordingly.  
  
    $('#url-form').submit(function(e){  
  
        e.preventDefault();  
  
        $.ajax({
```

Bitly: Asynchronous Forms

```
url: 'urls', //this refers to the route post 'urls'

method: 'POST',

data: $(this).serialize(),

dataType: 'json',

success: function(data){

    // write some code here to display the shortened URL

}

}); // end of function .ajax

}); // end of function .submit

}); // end of function document.ready
```

This is a [good resource](#) to help you get started.

Helpful Reading

1. [AJAX Pros and Cons](#)
2. [What Is Ajax & How Is It Used In Modern Web Development?](#)
3. [How AJAX Works: 10 Practical Uses For AJAX](#)

Extra

Instead of binding the callback to a `.submit()`, let's mimick the [actual Bitly way](#) by returning the shortened link as soon as it's pasted into the input box. What method would you need to make this happen?