



## Using jQuery for AJAX Requests

jQuery has a built-in function called "\$.post" that we will use to send all the information into the server. The whitepapers on this function are found here:

<http://api.jquery.com/jQuery.post/>

The \$.post() function call is a shorthand way of calling the \$.ajax() function with a POST type. As you can see from the webpage above, there are 4 arguments that are fed to the function. They are:

url	A string containing the URL to which the request is sent
data	A map or string that is sent to the server with the request
success(data, textStatus, XMLHttpRequest)	A callback function that is executed if the request is successful
dataType	The type of data expected back from the server

## Notes

- You should consider each post to a URL a new (conceptual) browser request
- When the server responds with text, that text has a dataType. The response is one of:
  - xml - the response comes back in an xml format and processed with jQuery
  - html - The response comes back and can be read by a browser
  - script - the response is a set of code written in javascript; jQuery automatically evaluates
  - json - The response is a JSON object and sends the object to the success callback function

## Example Usage

```
$.post('directory/content.html', function(data) {  
    $('#target_div').html(data);  
});
```

## Example Project

Create three static HTML pages, one that contains "Static Content Number 1," one that contains "Static Content Number 2," and one that contains "Static Content Number 3." Create a form on a web page that loads the content from one of the pages listed in a drop-down select element into a previously empty div.