

Introduction to AJAX POST with jQuery:

jQuery provides us with the `$.post` method to perform POST requests to a given URL:

```
$.post( "/api/serve.php", data, function(response, status, xhr) { });
```

- The first argument denotes the API or server endpoint URL to where the POST request should be sent.
- The second argument is the actual data that needs to be sent.
- The third argument is the callback function, which would be executed once the request completes.

Creating age calculator with AJAX POST

Open up *ageCalculator_Begin.htm* file before proceeding.

We have a simple user interface that asks the user to type in their name and select their date of birth, using an HTML5 date picker field. When the user presses the "Send" button, this data is sent via a POST request to the *php/ageCalculator.php* PHP script, which then calculates the age of the user and sends it back as a string.

Once again, do ensure that these files are being served from your web server. Also make sure your web server supports PHP.

I've already pre-built the click event handler for the "Send" button. The `serverURI` variable carries the path to the *ageCalculator.php* script, while the `outputBox` variable refers to the `div` on the UI where the resulting output from the AJAX request needs to be shown. We also have a simple validation to ensure that our input fields are not empty when we click on the "Send" button.

Let's start coding:

```
if($("#input[name='text']").val() != '' && $("#input[name='dob']").val() != '') {  
    outputBox.text('Please wait while I compute...');  
    // Send Data via AJAX  
    $.post(serverURI,$("#inputForm").serialize(),function(data) {  
        outputBox.html(data);  
    });  
} else {  
    outputBox.text('Please type in your name and select your date of birth!');  
}
```

We are jQuery's `serialize` method to fetch form's value formatted in the standard URL notation.

In the callback function, we just pull in the data that the PHP script returns and display it within our `outputBox`.

Check this out in the browser!