



Ajax with JSON

JavaScript Object Notation

A standard way to format data

“HTML”

```
<p>Your vacation to Paris,  
France has been booked for  
$2,196.00 for 4 nights.  
Confirmation #345feab.</p>
```

*Server decides how the
HTML is formatted*

“JSON”

```
{  
  totalPrice: 2196.00,  
  nights: 4,  
  location: 'Paris, France',  
  confirmation: '345feab'  
}
```

*Server returns details,
we can use it as we wish*

Our old code wouldn't work

application.js

```
$( 'form' ).on( 'submit', function(event) {  
    event.preventDefault();  
    var form = $(this);  
    $.ajax( '/book', {  
        type: 'POST',  
        data: form.serialize(),  
        success: function(result) {  
            form.remove();  
            $( '#vacation' ).hide().html(result).fadeIn();  
        }  
    });  
});
```



This code is expecting HTML to get returned, and injecting the result right into the page.

\$.ajax with POST and JSON

```
var form = $(this);
$.ajax('/book', {
  type: 'POST',
  data: form.serialize(),
  dataType: 'json',
  success: function(result) {
    form.remove();
    $('#vacation').hide().html(result).fadeIn();
  },
  contentType: 'application/json'
});
```

Parse the response as JSON

Result isn't HTML, we'll need to create a DOM node here

Ask the server to respond with JSON

Reading the JSON results and creating HTML

```
success: function(result) {  
    form.remove();  
    var msg = $("<p></p>");  
    msg.append("Destination: " + result.location + ". ");  
    msg.append("Price: " + result.totalPrice + ". ");  
    msg.append("Nights: " + result.nights + ". ");  
    msg.append("Confirmation: " + result.confirmation + ".");  
  
    $('#vacation').hide().html(msg).fadeIn();  
}
```



*JSON is converted into a
JavaScript Object*

```
{  
  location: 'Paris',  
  totalPrice: 1196.00,  
  nights: 4,  
  confirmation: '345feab'  
}
```

Create and add a DOM node

<p>Destination: Paris. Price: \$2196.00. Nights: 4. Confirmation: 345feab.</p>

Final event handler with POST and JSON

```
$( 'form' ).on( 'submit', function(e) {  
    event.preventDefault();  
    var form = $(this);  
    $.ajax( '/book', {  
        type: 'POST',  
        contentType: 'application/json',  
        dataType: 'json',  
        data: form.serialize(),  
        success: function(result) {  
            form.remove();  
            var msg = $("<p></p>");  
            msg.append("Destination: " + result.location + ". ");  
            msg.append("Price: " + result.totalPrice + ". ");  
            msg.append("Nights: " + result.nights + ". ");  
            msg.append("Confirmation: " + result.confirmation + ".");  
            $( '#vacation' ).hide().html(msg).fadeIn();  
        },  
        contentType: 'application/json'  
    });  
});
```



Don't Repeat Yourself

application.js

```
$.ajax('/book', { ... });
```



index.html

```
<form action='/book' method='POST'>
  ...
</form>
```

Form URL is in both HTML and JavaScript

application.js

```
$.ajax($('form').attr('action'), { ... });
```



jQuery Object Methods

attr(<attribute>)

attr(<attribute>, <value>)

Will use the action from the form HTML

Final event handler with POST and JSON

```
$( 'form' ).on( 'submit', function(e) {  
    event.preventDefault();  
    var form = $(this);  
    $.ajax( $( 'form' ).attr( 'action' ), {  
        type: 'POST',  
        contentType: 'application/json',  
        dataType: 'json',  
        data: form.serialize(),  
        success: function(result) {  
            form.remove();  
            var msg = $( "<p></p>" );  
            msg.append( "Destination: " + result.location + ". " );  
            msg.append( "Price: " + result.totalPrice + ". " );  
            msg.append( "Nights: " + result.nights + ". " );  
            msg.append( "Confirmation: " + result.confirmation + ". " );  
            $( '#vacation' ).hide().html( msg ).fadeIn();  
        }  
    } );  
});
```



jQuery Travels - Trip Booker

Destination:

Tickets:

BOOK TRIP

Call us at 555-25937 to make a reservation today!