

Intro to JSON

Dr. Michael Whitney

**Slides are based off W3C ajax materials and the recommended JS/jQuery text

But first: Retrieve HTML with AJAX

```
var xhr = new XMLHttpRequest(); // Create XMLHttpRequest object
xhr.onload = function() { // When response has loaded
    // The following conditional check will not work locally - only on a server
    if(xhr.status === 200) { // If server status was ok
        document.getElementById('content').innerHTML = xhr.responseText;
    }
};
xhr.open('GET', 'data/data.html', true); // Prepare the request
xhr.send(null);
```

Brief XML: Tags

Looks like HTML but the tags contain different words:

`<event>`

`<location>New York, NY</location>`

`<date>May 15</date>`

`<map>img/map-ny.png</map>`

`</event>`

Brief XML: Declaration

Contains info used to parse the document

Optional but if used must be on first line

```
<?xml
    version="version_number"           // all are 1.0
    encoding="encoding_declaration"    // UTF-8 or UTF-16 default
    standalone="standalone_status"     // yes default - no external source needed
?>

<?xml version="1.0" encoding="UTF-8"?>
```

Brief XML: Data

Available in the responseXML property of the object:

```
var xhr = new XMLHttpRequest();  
var xmlData = xhr.responseXML;
```

You need to write JavaScript to convert the XML data into HTML so it can be displayed.

Brief XML: Example

```
if (xhr.status == 200) {  
    xmlData = xhttp.responseXML;  
    txt = "";  
    artistNames = xmlDoc.getElementsByTagName("ARTIST");  
    for (i = 0; i < x.length; i++) {  
        txt = txt + artistNames[i].childNodes[0].nodeValue + "<br>";  
    }  
    document.getElementById("demo").innerHTML = txt;  
}
```

JSON: JavaScript Object Notation

Looks like object literal syntax but it is just data, not an object. Easier alternative to XML.

```
{  
  "location": "New York, NY",  
  "date": "May 30",  
  "map": "img/map-ny.png"  
}
```

JSON: Evaluates to JS Objects

No need for a parser like XML

JavaScript program can use standard JavaScript functions to convert JSON data into JavaScript Objects

e.g., `JSON.parse();`, (more later)

JSON: Data

Available in the **responseText** property of the object:

```
var xhr = new XMLHttpRequest();  
var JSONdata = xhr.responseText;
```

You need to write JavaScript to convert the JSON data into HTML so it can be displayed.

JSON: Syntax

JSON data is made up of **keys** and **values**.

```
{  
  "location": "New York, NY",  
  "date": "May 30",  
  "map": "img/map-ny.png"  
}
```

Why JSON? XML vs JSON

```
{  
  "employees": {  
    "employee": [  
      {"firstName": "John", "lastName": "Doe"},  
      {"firstName": "Anna", "lastName": "Smith"}  
    ]  
  }  
}
```

```
<employees>  
  <employee>  
    <firstName>John</firstName> <lastName>Doe</lastName>  
  </employee>  
  <employee>  
    <firstName>Anna</firstName> <lastName>Smith</lastName>  
  </employee>  
</employees>
```

Why JSON? XML vs JSON

Using XML

- Fetch an XML document
- Use the XML DOM to loop through the document
- Extract values and store in variables

Using JSON

- Fetch a JSON string
- `JSON.Parse` the JSON string

JSON: Syntax Rules

```
{ // object
  "events": [ // array
    { // first object
      "location": "Austin, TX",
      "date": "May 15",
      "map": "img/map-tx.png"
    },
    { // second object
      "location": "New York, NY",
      "date": "May 30",
      "map": "img/map-ny.png"
    }
  ]
}
```

Rules

- Data is in name/value pairs
- Data is separated by commas
- Curly braces hold objects
- Square brackets hold arrays

JSON: Values

JSON values can be:

- A number (integer or floating point)
- A string (in double quotes)
- A Boolean (true or false)
- An array (in square brackets)
- An object (in curly braces)
- null

JSON Object

JavaScript has a JSON object with two important methods:

1: Convert a JavaScript object to a string:

`JSON.stringify();`

2: Convert a string to a JavaScript object:

`JSON.parse();`

JSON: Object from String Example

```
// Create JavaScript string containing JSON syntax:
var text = '{ "employees" : [' +
    '{ "firstName":"John" , "lastName":"Doe" },' +
    '{ "firstName":"Anna" , "lastName":"Smith" },' +

    '{ "firstName":"Peter" , "lastName":"Jones" } ]}';

// Use JSON.parse(text) to convert text into JavaScript Object
var obj = JSON.parse(text);

// Use the javaScript object in your page
<p id="demo"></p>
<script>
    document.getElementById("demo").innerHTML =
        obj.employees[1].firstName + " " + obj.employees[1].lastName;
</script>
```


JSON & AJAX: Same Domain Issue

Ajax - Only works with data from same domain

Get around this issue with **JSONP** (JSON with Padding)

JSONP: How to...

1. Create a callback function in the html to process JSON data and display it on the page.

```
<script>
```

```
    function showEventsCallBack(data) {
```

```
        // code to process & display data
```

```
    }
```

```
</script>
```

JSONP: How to...

2. Create a `<script>` element that calls the JSON data from a remote server:

```
<script src="http://example.org/jsonp"> </script>
```

JSONP: How to...

3. The script then calls the function that was in the browser and passes the data to it as an argument:

```
showEventsCallback({  
  "events": [ {  
    "location": "New York, NY",  
    "date": "May 30",  
    "map": "img/map-ny.png"  
  }... ] });
```

```
function showEventsCallback(data)  
{  
  // code to process &  
  // display data  
}
```

jQuery & AJAX: Request Methods

<code>.load()</code>	Loads HTML fragments into an element.
<code>\$.get()</code>	Loads data using the HTTP GET method. Used to request data from the server
<code>\$.post()</code>	Loads data using the HTTP POST method Used to send data that updates data on server
<code>\$.getJSON()</code>	Loads JSON data using a GET request. Used for JSON data.
<code>\$.getScript()</code>	Loads and executes javaScript data using GET Used for javaScript (e.g., JSONP) data
<code>\$.ajax()</code>	This method is used to perform all requests The above methods all use this under the hood

jQuery & AJAX: load()

This method returns the content into the jQuery selection

```
$('#text').load('ajax.html #text');
```

The element the content will be loaded into:

```
$('#text').load('ajax.html #text');
```

The URL of the file to load comes first in the argument:

```
$('#text').load('ajax.html #text');
```

You can specify a fragment of the page to show (not the whole page):

```
$('#text').load('ajax.html #text');
```

jQuery & AJAX: jqxhr object

The other global Ajax methods return their data in the jqxhr object.

The jqxhr object has the following properties and methods:

The other global Ajax methods return their data in the jqxhr object.

PROPERTIES	METHODS
responseText	.done()
responseXML	.fail()
status	.always()
statusText	.abort()

jQuery & AJAX: 4 Shorthand Methods

`$.get(url[, data][, callback][, type])`

`$.post(url[, data][, callback][, type])`

`$.getJSON(url[, data][, callback])`

`$.getScript(url[, callback])`

url	where the data is fetched from
data	extra information for the server
callback	function to call when data returned
type	type of data to expect from server