

# Using AJAX

Unit 5



What the heck is  
AJAX?



# Asynchronous JavaScript and XML



# Making network requests with JavaScript



JavaScript code can...



# JavaScript code can...

- Load text from the server



# JavaScript code can...

- Load text from the server
- Load HTML from the server



# JavaScript code can...

- Load text from the server
- Load HTML from the server
- Load data (XML, JSON, CSV)



Why?



“Asynchronous”



# jQuery Methods



# jQuery Methods

- load



# jQuery Methods

- load
- get, post



# jQuery Methods

- load
- get, post
- getJSON, getScript



# jQuery Methods

- load
- get, post
- getJSON, getScript
- ajax



# jQuery Methods

- load
- get, post
- getJSON, getScript
- ajax
- global event handlers



load



# load

- Simplest - just loads stuff into the DOM



# load

- Simplest - just loads stuff into the DOM
- Can filter to a selector



```
$("#content").load("footer.html");
```



```
$("#content").load("footer.html p");
```



Demo



get and post



# get and post

- Represent different HTTP methods



# get and post

- Represent different HTTP methods
- In general, use get to just fetch something



# get and post

- Represent different HTTP methods
- In general, use get to just fetch something
- In general, use post when you are sending form data



```
$.get("something.html", function(data,  
status, xhr) {  
    //do something with the result  
    $("#content").html(data);  
});
```



```
var me = "Raymond Camden";  
$.get("something.html", {user:me},  
function(data, status, xhr) {  
    //do something with the result  
    $("#content").html(data);  
});
```



```
var me = "Raymond Camden";  
$.get("something.html", {user:me},  
function(data, status, jqxhr) {  
    //do something with the result  
    $("#content").html(data);  
}, "html");
```



Demo



```
var me = "Raymond Camden";  
$.post("something.html", {user:me},  
function(data, status, xhr) {  
    //do something with the result  
    $("#content").html(data);  
});
```



Get vs Post?



Demo



# Working with JSON



# Working with JSON

- JSON is JavaScript data in string format



# Working with JSON

- JSON is JavaScript data in string format
- get and post can be told to work with JSON



# Working with JSON

- JSON is JavaScript data in string format
- get and post can be told to work with JSON
- getJSON is a shortcut



Demo



# Providing Feedback



# Providing Feedback

- Users don't see what's going on



# Providing Feedback

- Users don't see what's going on
- Provide feedback!



Demo



# Working with \$.ajax



# Working with \$.ajax

- Base AJAX functionality



# Working with \$.ajax

- Base AJAX functionality
- Allows for deeper customization



# Working with \$.ajax

- Base AJAX functionality
- Allows for deeper customization
  - headers



# Working with \$.ajax

- Base AJAX functionality
- Allows for deeper customization
  - headers
  - modify the call/result



# Working with \$.ajax

- Base AJAX functionality
- Allows for deeper customization
  - headers
  - modify the call/result
  - username and password



# Working with \$.ajax

- Base AJAX functionality
- Allows for deeper customization
  - headers
  - modify the call/result
  - username and password
  - and more!



```
$.ajax({  
  url: 'something.html',  
  data: {name: 'Ray', age: 41},  
  success: function(data, status, jqxhr) {  
  
    },  
  type: 'post',  
  dataType: 'html',  
});
```



Demo



# Remote Servers



# Remote Servers

- By default - you cannot hit remote servers!



# Remote Servers

- By default - you cannot hit remote servers!
  - `$.get("http://someotherserver")`



# Remote Servers

- By default - you cannot hit remote servers!
  - `$.get("http://someotherserver")`
- You will forget this



# Workarounds



# Workarounds

- Use a proxy via local application server



# Workarounds

- Use a proxy via local application server
- CORS (Cross-origin resource sharing)



# Workarounds

- Use a proxy via local application server
- CORS (Cross-origin resource sharing)
- JSON/P



# JSON/P



# JSON/P

- “JSON with Padding”



# JSON/P

- “JSON with Padding”
- Hack (but a well defined, often used hack)



# JSON/P

- “JSON with Padding”
- Hack (but a well defined, often used hack)
- “Fakes” it by appending a `<script>` tag



# JSON/P

- “JSON with Padding”
- Hack (but a well defined, often used hack)
- “Fakes” it by appending a `<script>` tag
- Remote server has to be aware of this



Demo



# Selectors and Ajax Content



# Selectors and Ajax Content

- You've seen how to select stuff



# Selectors and Ajax Content

- You've seen how to select stuff
- You've seen how to use `$(document).ready`



# Selectors and Ajax Content

- You've seen how to select stuff
- You've seen how to use `$(document).ready`
- But what about stuff loaded via Ajax?



```
$(".ajaxStuff").click(function() {  
    console.log("You clicked me!");  
});
```



Demo



```
$(“parentThing”).on(“event”, “selector”,  
function() {  
    console.log(“You clicked me!”);  
});
```



Demo