

## CSC309 Programming on the Web

### week 7: jquery, ajax

Amir H. Chinnai, Spring 2017

Office Hours: M 3:45-5:45 BA4222

ahchinaei@cs.toronto.edu  
<http://www.cs.toronto.edu/~ahchinaei/>

### review

#### ❖ so far:

- **front-end**
  - structure & semantic, appearance, behavior
  - many design tips
- **back-end**
  - databases
    - structured & semi-structured data
- **interacting with backend**
  - http, rest, node, express

#### ❖ this week:

- using **jquery/ajax** for the interaction

jquery/ajax 7-2

### review

#### ❖ http (underlying protocol for client/server interaction)

GET /allWords.html HTTP/1.1 Host: www.project.org  
HTTP/1.1 200 Ok Content-Type: text/html <html><p>Hello</p></html>

#### ❖ rest (architectural styling/mindset)

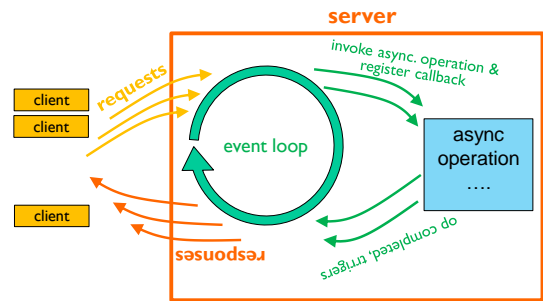
[www.project.org/allWords](http://www.project.org/allWords)  
[www.project.org/words/computer](http://www.project.org/words/computer)

#### ❖ node (asynchronous and event driven)

```
...
fs.readFile("some.txt", readCompletedCallback);
//do other tasks ...
function readCompletedCallback( error, dataBuffer) {
  response.writeHead(200, {'Content-Type': 'text/plain'});
  response.end('<html><p>Hello</p></html>');
  ...
}
```

jquery/ajax 7-3

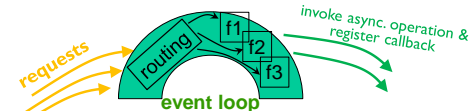
### review



jquery/ajax 7-4

### review

#### ❖ express (robust routing)



#### ❖ routings in lab 6:

```
app.get('/name', getName);
app.post('/name', setName);
app.get('/logout', logout);
app.post('/addmsg', addMessage);
app.get('/messages', getMessages);
```

```
function getName(req, res) {
  if (req.session.name) {
    return res.json({name: req.session.name});
  }
  else {
    return res.json({name: ''});
  }
}
```

jquery/ajax 7-5

### jquery

#### ❖ a js library for

- **traversing and manipulating dom**
- **event handling**

```
<html>
<head>
  <title>jquery example</title>
  <script type = "text/javascript"
    src = "https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js">
  </script>
  <script type = "text/javascript">
    $(function){document.write("Hello World!");};
  </script>
</head>
<body>
  <p>this is dummy</p>
</body>
</html>
```

jquery/ajax 7-6

## \$(selector).action()

### dom selector

```
window.onload = function(){...}
```

cf slide 4.14

```
getElementsByName("p")
getElementById("123")
```

**css selector** (slides 3.4, 3.8, 3.9)

```
p, div {...}
div p {...}
div > p {...}
.welcome {...}
[type] {...}
a[target="_blank"]
```

### jquery selector

```
$(document).ready(function(){...});
or
$(function(){ ... });
```

```
$("#p")
$("#123")

$("p, div")...
$("div p")...
$("div > p")...
$(".welcome")...
$("type")...
$("a[target='_blank']")...
```

jquery/ajax 7-7

## traverse

```
$("#p").parent("div")...
$("#p").parents("div")...
$("#p").parentsUntil("div")...
```

```
$("#div").children("p")...
$("#div").children("p").first()...
$("#div").children("p").last()...
$("#div p:nth-child(2)")...
$("#div").find("p")...
```

```
$("#div").siblings("p")...
$("#div").next("p")...
$("#div").nextAll("p")...
$("#div").nextUntil("p")...
```

```
$("#div").prev("p")...
$("#div").prevAll("p")...
$("#div").prevUntil("p")...
```

```
$("#p:nth-of-type(3)")...
```

```
$("#p").eq(2)...
```

```
$("#p").filter(".welcome")...
$("#p").not(".welcome")...
```

```
$("#p").first()...
```

```
$("#p").last()...
```

jquery/ajax 7-8

## effect

```
$("#p").slideDown(speed, easing, callback)
```

```
$("#p").animate(param, speed, callback)
{left: "250px", "slow", myFunc() }
```

### ❖ queuing

```
$("#p").animate({left: "250px"}, "slow", myFunc());
$("#p").animate({top: "400px"}, "slow", aFunc());
```

### ❖ chaining

```
$("#p").animate({left: "250px"}, "slow", myFunc())
.animate({top: "400px"}, "slow", aFunc());
```

```
$("#a").parent("li").siblings().find("a");
```

jquery/ajax 7-9

## misc.

### ❖ implicit loop

```
$("#p").addClass("intro");
```

### ❖ add/remove styling

```
$("#p").eq(2).removeClass("intro");
$("#p").eq(2).toggleClass("intro");
```

### ❖ explicit loop

```
$("#p").each(function(index){
    $(this).prepend(index+1);
});
```

jquery/ajax 7-10

## recall: ajax

### ❖ asynchronous javascript and xml

#### ❖ idea:

- updating part of the client page, not reloading the whole page
- cf. lecture 4: XMLHttpRequest

```
<button type="button" onclick="loadDoc()">Request data</button>
<p id="demo">I will change after click</p>
<script> function loadDoc() {
    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
        if (this.readyState == 4 && this.status == 200) {
            document.getElementById("demo").innerHTML = this.responseText;
        }
    };
    xhttp.open("GET", "demo_get.asp", true);
    xhttp.send();
}</script>
```

jquery/ajax 7-11

## jquery/ajax

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
<button type="button" onclick="loadDoc()">Request data</button>
<p id="demo">I will change after click</p>
<script> function loadDoc() {
```

```
    $.get("demo_get.asp", function(data, status){
        if (status == "success") {
            $("#demo").replaceWith(data);
        }
    });
}
```

```
</script>
```

### ❖ get syntax:

```
$.get(url, callback);
```

jquery/ajax 7-12

## jquery/ajax

### ❖ post syntax:

`$.post(url, data, callback);`

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    $.post("demo_test_post.asp",
    {
      name:"Donald Duck",
      city:"Duckburg"
    },
    function(data,status){
      success:$("#h1").replaceWith("<h1>"+"data+"+"</h1>");
    });
  });
});
</script>
```

jquery/ajax 7-13

## ajax method

### ❖ ajax syntax:

`$.ajax({name:value, name:value, ... })`

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    $.ajax({
      url:"demo_test_post.asp",
      type:"POST",
      dataType:"json",
      data:JSON.stringify({"name":"Donald Duck","city":"Duckburg"}),
      success:function(data){
        $("#h1").replaceWith("<h1>"+"data+"+"</h1>");
      }
    });
  });
});
</script>
```

jquery/ajax 7-14

## ajax method

### ❖ another example

```
function getName() {
  $.ajax({
    url : "/name",
    dataType : "json",
    contentType: "application/json; charset=utf-8",
    success : function (response) {
      var name = response['name'];
      updateUI(name);
    }
  });
}
```

jquery/ajax 7-15