

ICS313 :7.0



Web Programming

Lecturer: Toby Daniel

AJAX

- So far we have been looking at AJAX using an XML file... however we have really only been treating the XML file as any other text file.
 - This could have been a .txt .htm or some text generated by a server side scripting engine (PHP, ASP etc.)
- We were using:
 - `XMLHttpRequest.responseText` (for most browsers)
 - `ActiveXObject.responseText` (for Internet Explorer)

Example responseText code

```
if (receiveReq.readyState == 4) {  
document.getElementById('div1').innerHTML=receiveReq.responseText;  
}
```

Where:

- `receiveReq` is the browser specific XMLHttpRequest object
- `div1` is the element ID to be updated in the HTML document
- `responseText` is the text extracted from the text file

Text as a single block into HTML (responseText)

```
<?xml version="1.0" ?>
```

```
<root>
```

```
<store>
```

```
  Megabucks International
```

```
</store>
```

```
<address>
```

```
  52 Acacia Avenue
```

```
</address>
```

```
<director>
```

```
  Joe Blogs
```

```
</address>
```

```
</root>
```

```
<html>
```

```
<head></head>
```

```
<body>
```

```
<div id="div1">
```

```
  Megabucks International
```

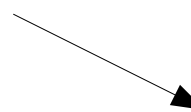
```
  52 Acacia Avenue
```

```
  Joe Blogs
```

```
</div>
```

```
</body>
```

```
</html>
```



responseXML

- `responseXML` is similar in function to `responseText`, however rather than holding the data as a string of text, it includes the XML DOM structure.
- Having the XML DOM means that we can select text/data from within specific tags.
- Once selected, specific data can be updated to different parts of the HTML page (using element ID's)

XML into specific HTML locations (responseXML)

```
<?xml version="1.0" ?>
```

```
<root>
```

```
<store>
```

```
  Megabucks International
```

```
</store>
```

```
<address>
```

```
  52 Acacia Avenue
```

```
</address>
```

```
<director>
```

```
  Joe Blogs
```

```
</address>
```

```
</root>
```

```
<html>
```

```
<head></head>
```

```
<body>
```

```
<div id="name">
```

```
</div>
```

```
<div id="company">
```

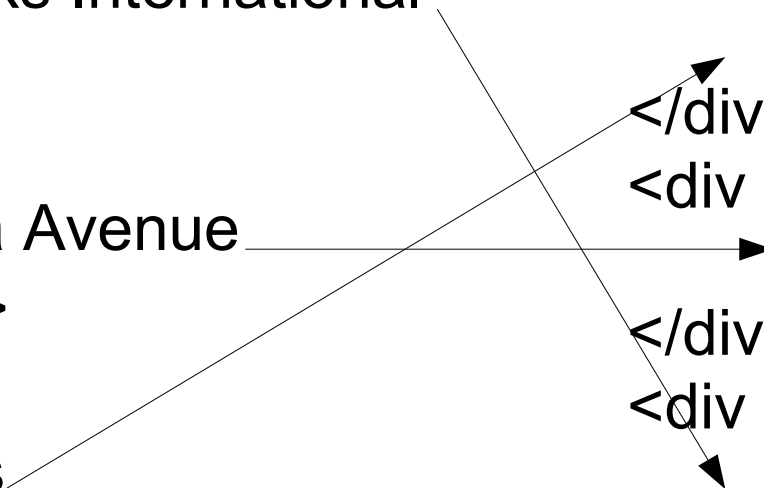
```
</div>
```

```
<div id="location">
```

```
</div>
```

```
</body>
```

```
</html>
```



responseXML

- To use responseXML is very similar to responseText
 - `XMLHttpRequest.responseXML` (for most browsers)
 - `ActiveXObject.responseXML` (for Internet Explorer)
- In addition we have to specify which tag we are taking the data from using:
`.getElementsByTagName('name').item(0).firstChild.data`
- where `name` is the tag name in the XML file.

Example responseXML code

```
if (receiveReq.readyState == 4) {  
    document.getElementById('div1').innerHTML =  
        receiveReq.responseXML.getElementsByTagName('data1').  
        item(0).firstChild.data;  
}
```

Where:

- `receiveReq` is the browser specific XMLHttpRequest object
- `div1` is the element ID to be updated in the HTML document
- `data1` is the tag in the XML file where the data is stored

Problems with IE

The responseXML object works in IE **however...**

- The file has to be sent to the browser with the HTTP header content type set as

Content-Type: application/xml

- ...otherwise IE will throw a JavaScript error, 'Object Expected'
- This means that your AJAX and XML files will not work with IE if they are stored in a folder on your hard drive – they need to be served by a web server!

responseXML Examples

- Have a look at the following examples that use the responseXML object to dynamically update a number of HTML elements using data within the XML file.

- Open
 - ajax3.htm
 - newdata.xml
 - ajax4.htm



Lab. 8

- Have a look at Lab 1 (the web pages about the CIS courses you are taking).
- This time you are going to store the information about the different courses as individual XML files and dynamically update them into a single web page.
- Use the responseXML to place the information in the right place in your web page.