

The background of the slide is a photograph of a modern, multi-story office building with a glass facade. The building is viewed from a low angle, looking up. The word "CYBAGE" is visible on the upper part of the building's facade. A semi-transparent white box with a blue square is overlaid on the right side of the image, containing the title and author information.

CYBAGE

# jQuery Day 2

By Asfiya Khan (Sr. Technical Trainer)

## Agenda



jQuery Lookups



Ajax in jQuery



Difference between  
attr(), props(), data()

## Careful with jQuery lookups



- jQuery lookups aren't magic. It uses native browser lookups where it can.
- And when it can't ... it iterates over each and every DOM element based on your context.
- `.getElementbyID()` and `.getElementsByName()` operate on the document object whereas `.getElementsByTagName()` and `.getElementsByClassName()` can be used by any DOM object.
- So leverage the tag name lookup with jQuery's `.find()` and `.children()` where appropriate.

## Check if jQuery is available on a page



```
if (!window.jQuery) {  
    console.log('jQuery is unavailable!');  
} else {  
    console.log('jQuery is available');  
}
```

## Check if element is available on the page



```
var jqElement = $('#mydiv');  
  
if (jqElement.length) {  
    console.log('Element exists!');  
}
```

## Go from jQuery object to DOM object, and back



```
•var jqDiv = $('#mydiv');
```

•.  
•.  
•.

```
•var div = jqdiv.get(0);
```

```
var div = document.getElementById('mydiv');
```

•  
•  
•

```
var jqDiv = $(div);
```

## jQuery Effects



- jQuery library provides several techniques for adding animation to a web page.
- These include simple, standard animations that are frequently used, and the ability to craft sophisticated custom effects.

Eg. `.slideDown()`, `.slideUp()`, `.slideToggle()`, `fadeIn()`, `fadeOut()`, `fadeToggle()` etc.

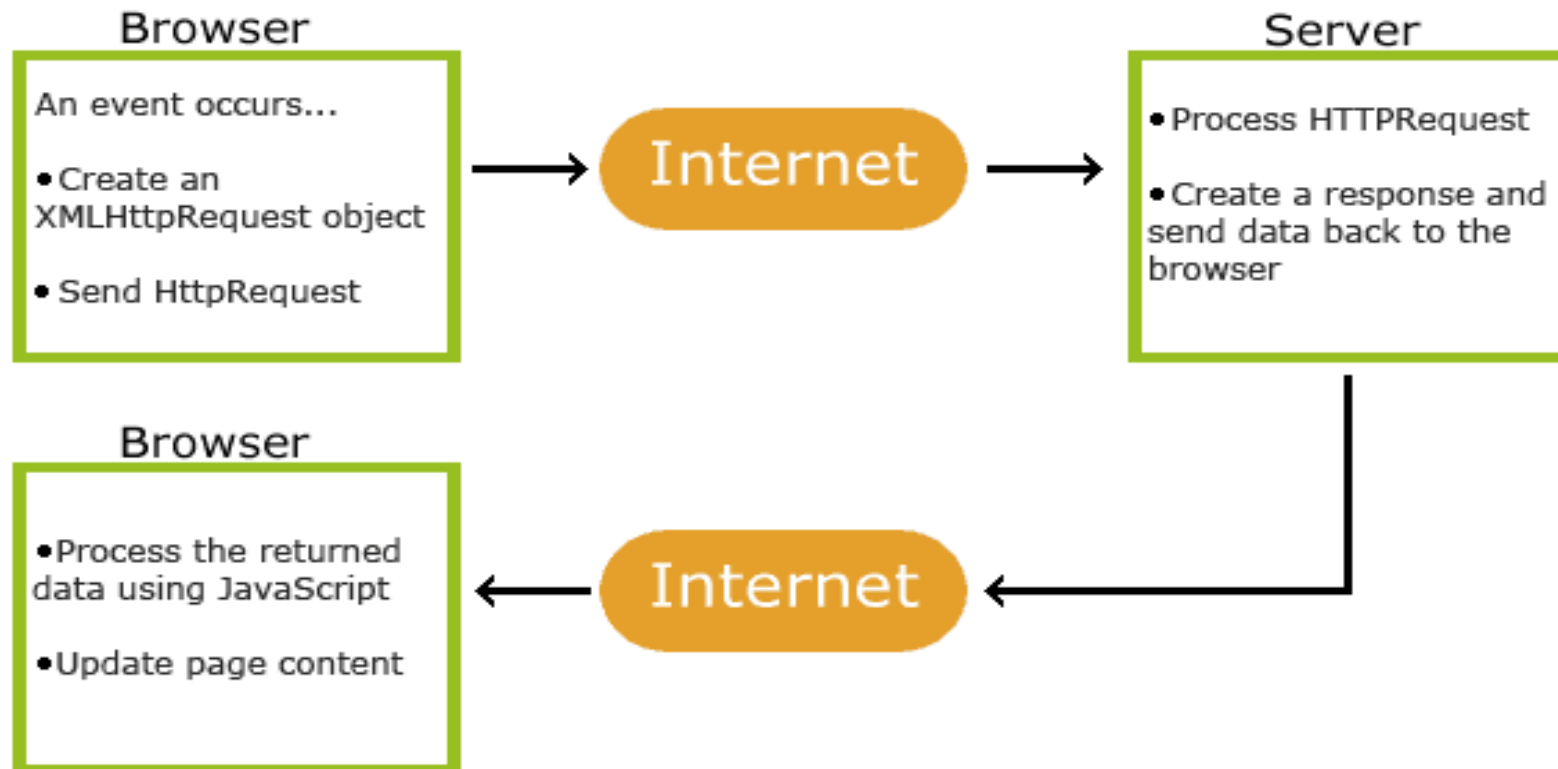
## AJAX - .ajax()



- AJAX allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes.
- This means that it is possible to update parts of a web page, without reloading the whole page.



## AJAX - .ajax()



## AJAX - .ajax()



### Using load()

- **`$(selector).load(url,data,callback)`** allows HTML content to be loaded from a server and added into a DOM object:

```
$(document).ready(function(){  
    $('#HelpButton').click(function(){  
        $('#MyDiv').load('HelpDetails.html');  
    });  
});
```

## AJAX - .ajax()



### Using load() With a Selector

- A selector can be added after the URL to filter the content that is returned from calling load():

```
$('#MyDiv').load('HelpDetails.html #MainTOC');
```

## AJAX - .ajax()



### Passing Data using load()

- Data can be passed to the server using `load(url,data)`:

```
$( '#MyDiv' ).load( 'GetCustomers.aspx',  
    {PageSize:25} );
```

## AJAX - .ajax()



### Using a Callback Function with load()

- load() can be passed a callback function:

```
$('#OutputDiv').load('NotFound.html',  
    function (response, status, xhr) {  
        if (status == "error") {  
            alert(xhr.statusText);  
        }  
    });
```

## AJAX - .ajax()

### Using get()

- **`$.get(url,data,callback,datatype)`** can retrieve data from a server:

```
$.get('HelpDetails.html', function (data) {  
    $('#OutputDiv').html(data);  
});
```

### Using getJSON()

- **`$.getJSON(url,data,callback)`** can retrieve data from a server:

```
$.getJSON('CustomerJson.aspx',{id:1},  
    function (data) {  
        alert(data.FirstName + ' ' +  
            data.LastName);  
    });
```

## AJAX - .ajax()



### Using post()

- `$.post(url,data,callback,datatype)` can post data to a server and retrieve results:

```
$.post('GetCustomers.aspx', {PageSize:15},  
      function (data) {  
          $('#OutputDiv').html(data);  
      }  
);
```

## Using the ajax() Function



The ajax() function is configured by assigning values to JSON properties:

```
$.ajax({  
  url: '../CustomerService.svc/InsertCustomer',  
  data: customer,  
  dataType: 'json',  
  success: function (data, status, xhr) {  
    alert("Insert status: " + data.d.Status + '\n' +  
      data.d.Message);  
  },  
  error: function (xhr, status, error) {  
    alert('Error occurred: ' + status);  
  }  
});
```



Thank You!