

# Ajax

AJAX is an acronym for **Asynchronous JavaScript and XML**.

AJAX allows you to send and receive data asynchronously without reloading the web page. So it is fa

## XMLHttpRequest

An object of XMLHttpRequest is used for asynchronous communication between client and server.

### Properties of XMLHttpRequest object

onReadyStateChange : It is called whenever readystate attribute changes. It must not be used with synchronous requests.

readyState : Represents the state of the request. It ranges from 0 to 4.

0 UNOPENED

1 OPENED open is called but send() is not called.

2 HEADERS\_RECEIVED send() is called, and headers and status are available.

3 LOADING Downloading data; responseText holds the data.

4 DONE The operation is completed fully.

responseText : returns response as text.

responseXML : returns response as XML.

### Methods of XMLHttpRequest object

Method	Description
void open(method, URL)	opens the request specifying get or post method and url.
void open(method, URL, async)	same as above but specifies asynchronous or not.
void open(method, URL, async, username, password)	same as above but specifies username and password.
void send()	sends get request.
void send(string)	send post request.
setRequestHeader(header,value)	it adds request headers.

### **Asynchronous XMLHttpRequest template**

```
var ajax = new XMLHttpRequest();
ajax.onreadystatechange = function() {
  if (ajax.readyState == 4) {
    // 4 means request is finished do something with ajax.responseText;
  } };
ajax.open("get", url, true);
ajax.send(null);
```

### **Checking for request errors**

```
if (ajax.readyState == 4) {
  if (ajax.status == 200) {
    // 200 means request succeeded do something with ajax.responseText;
  }
  else {
    code to handle the error;
  }
}
```

## Ajax syntax

```
$.ajax ({
    method: 'post',
    url   : 'register.php',
    data : data,
    async : true,
    beforeSend : function()
    {
    }
    success : function (response)
    {
        // do with response
    }
});
```

# jQuery Tutorial

jQuery is a lightweight, "write less, do more", JavaScript library. The purpose of jQuery is to make it much easier to use JavaScript on your website.

The jQuery library contains the following features:

- HTML/DOM manipulation
- CSS manipulation
- HTML event methods
- Effects and animations
- AJAX
- Utilities
- 
- 
-

## jQuery Syntax

**selecting** HTML elements and performing some **action** on the element(s).

`$(selector).action()`

Examples:

`$(this).hide()` - hides the current element.

`$("p").hide()` - hides all <p> elements.

`$(".test").hide()` - hides all elements with class="test".

## The Document Ready Event

```
$(document).ready(function(){  
    // jQuery methods go here...  
});
```

## Commonly Used jQuery Event Methods

`$(document).ready()`

`click()`

`dblclick()`

`mouseenter()`

`mouseleave()`

`mousedown()`

`mouseup()`

`hover()`

`focus()` : The function is executed when the form field gets focus

`blur()` : The function is executed when the form field loses focus:

```
$(document).ready(function(){  
    $("input").focus(function(){  
        $(this).css("background-color", "#cccccc");  
    });  
    $("input").blur(function(){  
        $(this).css("background-color", "#eee");  
    });  
});
```

## jQuery Effects

### Hide and Show

```
$(document).ready(function(){  
  $("#hide").click(function(){  
    $("p").hide();  
  });  
  
  $("#show").click(function(){  
    $("p").show();  
  });  
});
```

#### syntax

```
$(selector).hide(speed,callback); // hide(1000)  
  
$(selector).show(speed,callback); // show(1000)  
  
$(selector).toggle(speed,callback);
```

### Fading

- fadeIn()
- fadeOut()
- fadeToggle()
- fadeTo()