



SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with ‘A’ Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING-IOT Including CS&BCT

**COURSE NAME : 19SB602 FULL STACK DEVELOPMENT FOR NEXT
GENERATION IOT**

III YEAR / VI SEMESTER

Unit II- FRONT-END MODULES

Topic : JQuery Syntax, AJAX.



jQuery

- ✓ Java script library
- ✓ Light weight → "write less, do more", JavaScript library.
- ✓ Easy to make website
- ✓ JQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

Library Consists of

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



Many of the biggest companies on the Web use jQuery, such as:

- ✓ Google
- ✓ Microsoft
- ✓ IBM
- ✓ Netflix

jQuery run in all platform of browser

Adding jQuery to Your Web Pages

There are several ways to start using jQuery on your web site.

- ✓ Download the jQuery library from jquery.com
- ✓ Include jQuery from a CDN, like Google



jQuery Syntax

The jQuery syntax is tailor-made for **selecting** HTML elements and performing some **action** on the element(s).

Basic syntax is: **`$(selector).action()`**

- ✓ A \$ sign to define/access jQuery
- ✓ A (selector) to "query (or find)" HTML elements
- ✓ A jQuery action() to be performed on the element(s)

Example

- ✓ `$(this).hide()` - hides the current element.
- ✓ `$("p").hide();`



jQuery Selectors

jQuery selectors are one of the most important parts of the jQuery library.

jQuery selectors allow you to **select and manipulate HTML element(s)**.

jQuery selectors are used to "find" (or select) HTML elements based on their **name, id, classes, types, attributes, values of attributes** and much more.

It's based on the existing CSS Selectors, and in addition, it has some own custom selectors.

All selectors in jQuery start with the **dollar sign and parentheses: \$()**.



The element Selector

The jQuery element selector selects elements based on the element name.

You can select all <p> elements on a page like this:

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

Example

When a user clicks on a button, all <p> elements will be hidden:

Example

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



```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<script
```

```
src="https://ajax.googleapis.com/aja  
x/libs/jquery/3.7.1/jquery.min.js"></
```

```
script>
```

```
<script>
```

```
$(document).ready(function(){
```

```
  $("button").click(function(){
```

```
    $("p").hide();
```

```
  });
```

```
});
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<h2>This is a heading</h2>
```

```
<p>This is a paragraph.</p>
```

```
<p>This is another paragraph.</p>
```

```
<button>Click me to hide  
paragraphs</button>
```

```
</body>
```

```
</html>
```



This is a heading

This is a paragraph.

This is another paragraph.

Click me to hide paragraphs

This is a heading

Click me to hide paragraphs



The #id Selector

The jQuery #id selector uses the id attribute of an HTML tag to find the specific element.

An id should be unique within a page, so you should use the #id selector when you want to find a single, unique element.

To find an element with a specific id, write a hash character, followed by the id of the HTML element:

```
$("#test")
```

Example

When a user clicks on a button, the element with id="test" will be hidden:



```
$(document).ready(function(){  
    $("button").click(function(){  
        $("#test").hide();  
    });  
});
```



```
<!DOCTYPE html>
<html>
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jq
uery/3.7.1/jquery.min.js"></script>
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("#test").hide();
  });
});
</script>
</head>
```

This is a heading

This is a paragraph.

This is another paragraph.

Click me

```
<body>
```

```
<h2>This is a heading</h2>
```

```
<p>This is a paragraph.</p>
```

```
<p id="test">This is another
paragraph.</p>
```

```
<button>Click me</button>
```

```
</body>
```

```
</html>
```

This is a heading

This is a paragraph.

Click me



The .class Selector



```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<script
```

```
src="https://ajax.googleapis.com  
/ajax/libs/jquery/3.7.1/jquery.mi  
n.js"></script>
```

```
<script>
```

```
$(document).ready(function(){  
  $("button").click(function(){  
    $(".test").hide();
```

```
  });
```

```
});
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<h2 class="test">This is a  
heading</h2>
```

```
<p class="test">This is a  
paragraph.</p>
```

```
<p>This is another paragraph.</p>
```

```
<button>Click me</button>
```

```
</body>
```

```
</html>
```

This is a heading

This is a paragraph.

This is another paragraph.

Click me

This is another paragraph.

Click me



AJAX



AJAX = Asynchronous JavaScript And XML.

AJAX is not a programming language.

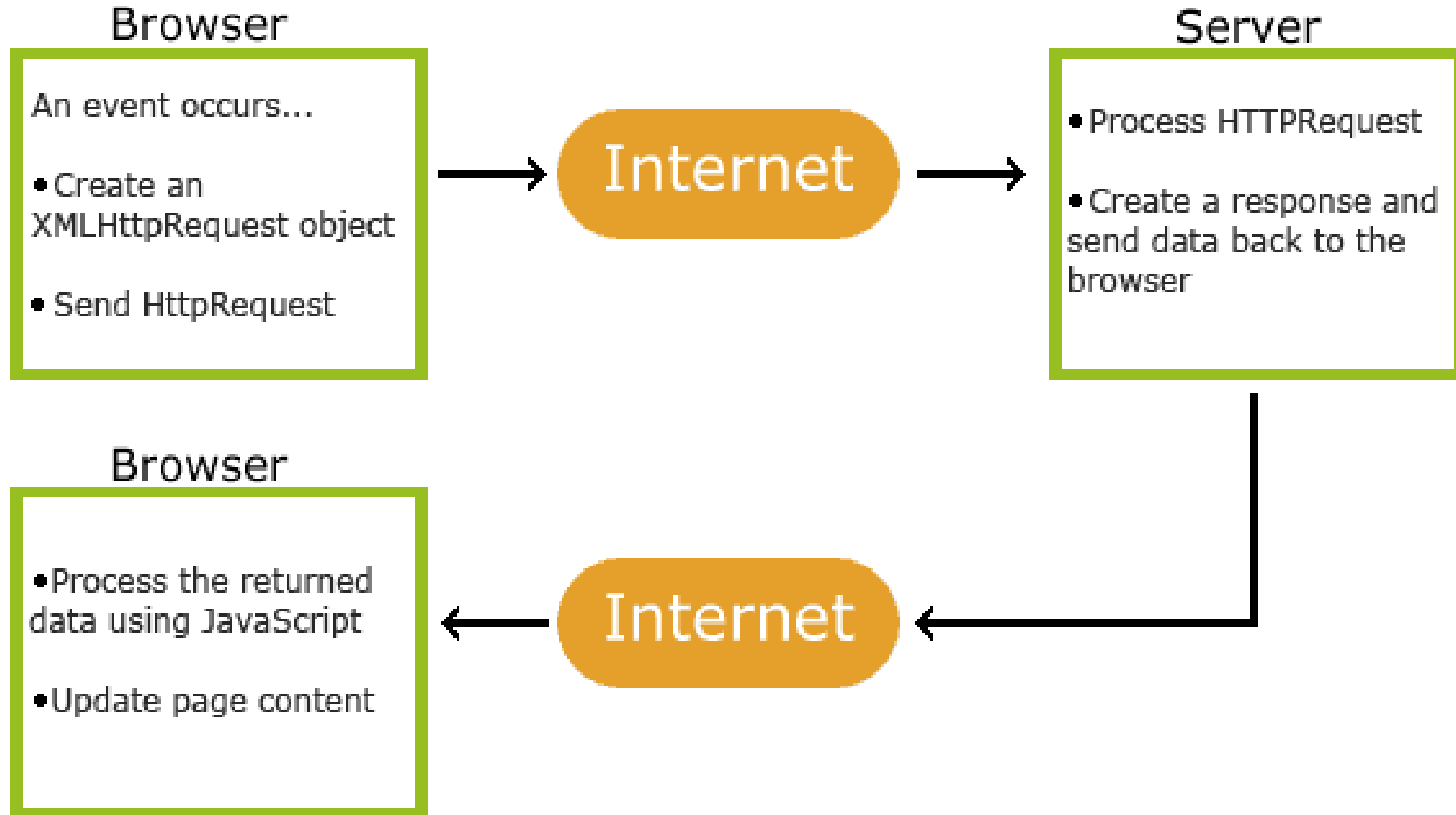
AJAX just uses a combination of:

A browser built-in XMLHttpRequest object (to request data from a web server)

JavaScript and HTML DOM (to display or use the data)



How AJAX Works





1. An event occurs in a web page (the page is loaded, a button is clicked)
2. An XMLHttpRequest object is created by JavaScript
3. The XMLHttpRequest object sends a request to a web server
4. The server processes the request
5. The server sends a response back to the web page
6. The response is read by JavaScript
7. Proper action (like page update) is performed by JavaScript



The XMLHttpRequest Object

All modern browsers support the XMLHttpRequest object.

The XMLHttpRequest object can be used to exchange data with a server behind the scenes.

This means that it is possible to update parts of a web page, without reloading the whole page.



Create an XMLHttpRequest Object

All modern browsers (Chrome, Firefox, Edge (and IE7+), Safari, Opera) have a built-in XMLHttpRequest object.

Syntax for creating an XMLHttpRequest object:

```
variable = new XMLHttpRequest();
```



```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1>The XMLHttpRequest Object</h1>
```

```
<p id="demo">Let AJAX change this text.</p>
```

```
<button type="button" onclick="loadDoc()">Change Content</button>
```

```
<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", "ajax_info.txt", true);  
xhttp.send();  
}  
</script>  
  
</body>  
</html>
```



The XMLHttpRequest Object

Let AJAX change this text.

Change Content



Any Query????

Thank you.....