WSA | Forward looking IT finishing school

AJAX

jQuery

Introduction to AJAX

- **AJAX(Asynchronous JavaScript and XML)** is a mechanism of performing data exchange with the server without reloading the whole code

- It helps to fetch data in form of XML / JSON / HTML

- With vanilla JavaScript, writing AJAX code is quite complex where you need to write multiple lines of code

- With jQuery you can perform it very easily

Example usage – Vanilla JavaScript code

```
<script>
function loadDoc() {
    // Create a request object
    var xhttp = new XMLHttpRequest();
    // Register the function to be called when server responds
    xhttp.onreadystatechange = function() {
        if (this.readyState == 4 && this.status == 200) {
            document.getElementById("demo").innerHTML = this.responseText;
    }
};

// There is a file called ajax_info.txt on server
xhttp.open("GET", "ajax_info.txt", true);
// Send the request
xhttp.send();
}
</script>
```

jQuery AJAX APIs

API & Description

```
$(selector).load(URL,data,callback);
```

Sets or returns the text content of selected elements

- **URL** - The location from where data need to be fetched
- **data** - Query string
- **callback** - To get notification when the load is complete

The callback function can further have three parameters:

- **responseTxt** - contains the resulting content if the call succeeds
- **statusTxt** - contains the status of the call
- **xhr** - contains the XMLHttpRequest object

Example usage – Loading URL contents

```
<script>
$(document).ready(function() {
$("button").click(function() {
    /* Load a page from external URL with response / error handling */
    $("#div1").load("http://jsonplaceholder.typicode.com/posts/1",
         function(responseTxt, statusTxt, xhr){
             if(statusTxt == "success")
                 alert("External content loaded successfully!");
             if(statusTxt == "error")
                 alert("Error: " + xhr.status + ": " + xhr.statusText);
         });
    });
});
</script>
```

jQuery AJAX APIs

Selector	Description
`$.get(URL,callback);`	**URL** - The location from where data need to be fetched **callback** - To get notification when the load is complete
`$.post(URL,data,callback);`	**URL** - The location from where data need to be fetched **data** - That required to be posted **callback** - To get notification when the load is complete

Example usage – HTTP GET

```
<script>
$(document).ready(function() {
    $("button").click(function() {
        $.get("https://api.github.com/users/devendradora",
            function(data, status){
              console.log("Data: " , data , "\nStatus: " , status);
         });
    });
});
</script>
```

Thank you

WebStack Academy

#83, Farah Towers,
1st Floor, MG Road,
Bangalore – 560001

M: +91-809 555 7332
E: training@webstackacademy.com

WSA in Social Media: