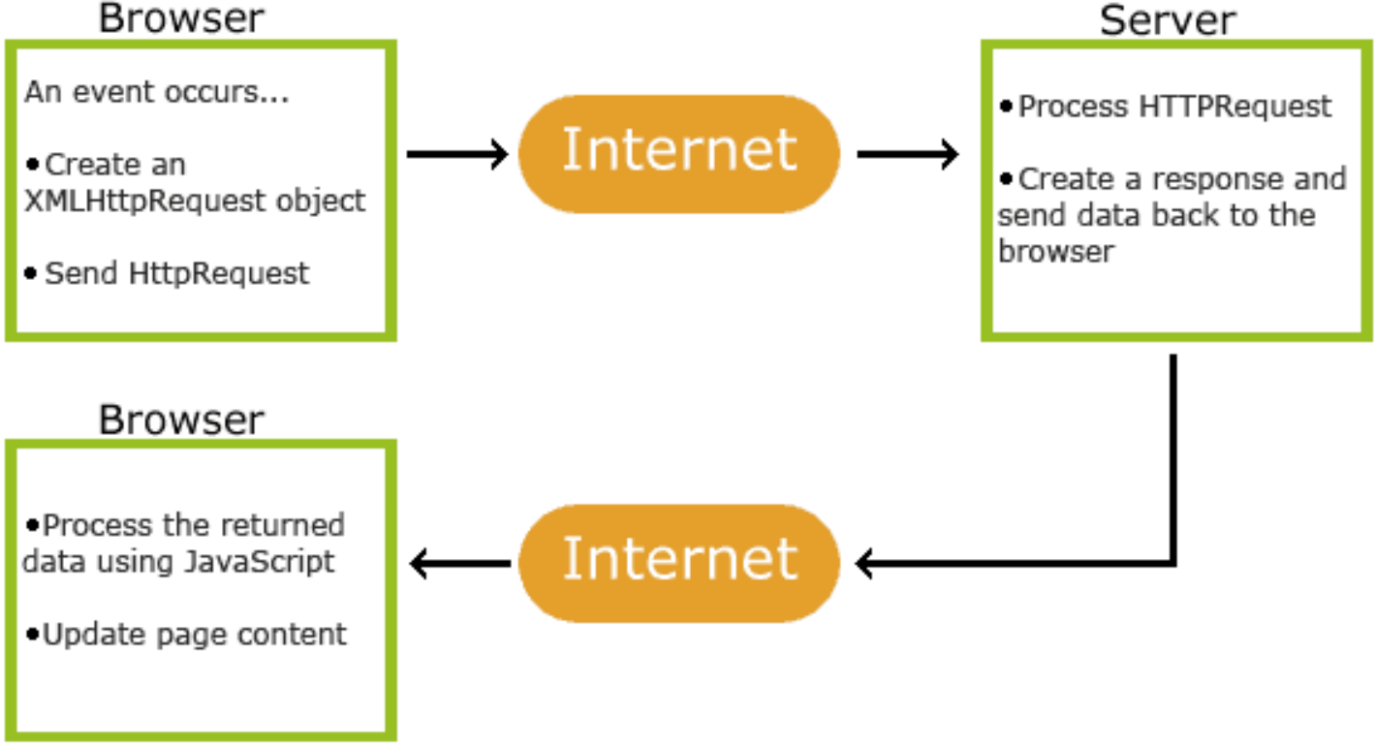
AJAX = Asynchronous JavaScript And XML. AJAX just uses a combination of:

* A browser built-in XMLHttpRequest object (to request data from a web server)
* JavaScript and HTML DOM (to display or use the data)

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



AJAX Process

The XMLHttpRequest object can be used to exchange data with a web server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page. Steps:

* Create an XMLHttpRequest object. Ex.  
  const xhttp = new XMLHttpRequest();
* A callback function is a function passed as a parameter to another function. In this case, the callback function should contain the code to execute when the response is ready. Ex:  
  xhttp.onload = function() {  
    // What to do when the response is ready  
  }
* To send a request to a server, you can use the open() and send() methods of the XMLHttpRequest object. Ex:  
  xhttp.open("GET", "ajax\_info.txt");  
  xhttp.send();

**XMLHttpRequest Object Methods**

* new XMLHttpRequest(): Creates a new XMLHttpRequest object
* abort(): Cancels the current request
* getResponseHeader(): Returns specific header information
* getAllResponseHeaders(): Returns all the header information from the server resource
* open(method, url, async, user, psw): Specifies the request method: the request type GET or POST. Url is the file location. Async is a boolean flag where true (asynchronous) or false (synchronous). User is an optional user name. Psw is an optional password.
* send(): Sends the request to the server(Used for GET requests)
* send(string): Sends the request to the server(Used for POST requests)
* setRequestHeader(): Adds a label/value pair to the header to be sent

**XMLHttpRequest Object Properties**

* onload: Defines a function to be called when the request is received (loaded)
* onreadystatechange: Defines a function to be called when the readyState property changes
* readyState: Holds the status of the XMLHttpRequest. 0: request not initialized, 1: server connection

established, 2: request received, 3: processing request 4: request finished and response is ready

* responseText: Returns the response data as a string
* responseXML: Returns the response data as XML data status.
* status: Returns the status-number of a request. Ex. 200: "OK", 403: "Forbidden", etc.
* statusText: Returns the status-text (e.g. "OK" or "Not Found")

<!DOCTYPE html>

<html>

<body>

<div id="demo">

<h1>The XMLHttpRequest Object</h1>

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

</div>

<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>