

JSON WITH AJAX

AJAX is Asynchronous JavaScript and XML, which is used on the client side as a group of interrelated web development techniques, in order to create asynchronous web applications. According to the AJAX model, web applications can send and retrieve data from a server asynchronously without interfering with the display and the behavior of the existing page.

Many developers use JSON to pass AJAX updates between the client and the server. Websites updating live sports scores can be considered as an example of AJAX. If these scores have to be updated on the website, then they must be stored on the server so that the webpage can retrieve the score when it is required. This is where we can make use of JSON formatted data.

Any data that is updated using AJAX can be stored using the JSON format on the web server. AJAX is used so that javascript can retrieve these JSON files when necessary, parse them, and perform one of the following operations –

- Store the parsed values in the variables for further processing before displaying them on the webpage.
- It directly assigns the data to the DOM elements in the webpage, so that they are displayed on the website.

Example

The following code shows JSON with AJAX. Save it as **ajax.htm** file. Here the loading function loadJSON is used asynchronously to upload JSON data.

```
<html>
  <head>
    <meta content = "text/html; charset = ISO-8859-1" http-equiv = "content-type">

    <script type = "application/javascript">
      function loadJSON(){
        var data_file = "http://www.tutorialspoint.com/json/data.json";
        var http_request = new XMLHttpRequest();
        try{
          // Opera 8.0+, Firefox, Chrome, Safari
          http_request = new XMLHttpRequest();
        }catch (e){
          // Internet Explorer Browsers
          try{
            http_request = new ActiveXObject("Msxml2.XMLHTTP");

          }catch (e) {

            try{
              http_request = new ActiveXObject("Microsoft.XMLHTTP");
            }catch (e){
              // Something went wrong
              alert("Your browser broke!");
              return false;
            }
          }
        }

        http_request.onreadystatechange = function(){

          if (http_request.readyState == 4 ){
            // Javascript function JSON.parse to parse JSON data
            var jsonObj = JSON.parse(http_request.responseText);

            // jsonObj variable now contains the data structure and can
            // be accessed as jsonObj.name and jsonObj.country.
          }
        }
      }
    </script>
  </head>
</html>
```

```

        document.getElementById("Name").innerHTML = jsonObj.name;
        document.getElementById("Country").innerHTML = jsonObj.country;
    }
}

http_request.open("GET", data_file, true);
http_request.send();
}

</script>

<title>tutorialspoint.com JSON</title>
</head>

<body>
    <h1>Cricketer Details</h1>

    <table class = "src">
        <tr><th>Name</th><th>Country</th></tr>
        <tr><td><div id = "Name">Sachin</div></td>
        <td><div id = "Country">India</div></td></tr>
    </table>

    <div class = "central">
        <button type = "button" onclick = "loadJSON()">Update Details </button>
    </div>

</body>

</html>

```

Given below is the input file **data.json**, having data in JSON format which will be uploaded asynchronously when we click the **Update Detail** button. This file is being kept in <http://www.tutorialspoint.com/json/>

```

{"name": "brett", "country": "Australia"}

```

The above HTML code will generate the following screen, where you can check AJAX in action –

CRICKETER DETAILS

Name	Country
------	---------

Sachin	India
--------	-------

:

When you click on the **Update Detail** button, you should get a result something as follows. You can try [JSON with AJAX](#) yourself, provided your browser supports Javascript.

CRICKETER DETAILS

Name	Country
------	---------

brett	Australia
-------	-----------

Loading [Mathjax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js