# AJAX & JSON

# Introduction

- **JSON (JavaScript Object Notation), is a simple and easy to read and write data exchange format.**
  - It is easy for humans to read and write.
  - It is easy for machines to parse and generate.
  - It is based on a subset of the JavaScript, Standard ECMA-262
  - JSON is a text format that is completely language independent; can be used with most of the modern programming languages.
  - The filename extension is .json
  - JSON Internet Media type is application/json
  - It's popular and implemented in countless projects worldwide, for those don't like XML, JSON is a very good alternative solution.

# Values supported by JSON

- **Strings : double-quoted Unicode, with backslash escaping**
- **Numbers: double-precision floating-point format in JavaScript**
- **Booleans : true or false**
- **Objects: an unordered, comma-separated collection of key:value pairs**
- **Arrays : ordered, comma-separated values enclosed in square brackets**
- **Null : A value that isn't anything**

```
var obj = {
    "name":"Amit",
    "age": 37.5,
    "married": true,
    "address":{ "city":"Pune" , "state":"Mah" },
    "hobbies":["swimming","reading","music"]
}
```

```
<script language="javascript">
var JSONObject = {  "name" : "Amit",
     "address"  : "B-123 Bangalow",
     "age"  : 23,
     "phone"   : "011-4565763",
     "MobileNo"  : 0981100092
  };
var str =
"<h2><font color='blue'>Name </font>::"        +JSONObject.name+"</h2>" +
 "<h2><font color='blue'>Address </font>::"   + JSONObject.address+"</h2>" +
"<h2><font color='blue'>Age </font>::"            +JSONObject.age+"</h2>" +
"<h2><font color='blue'>Phone No </font>::"  +JSONObject.phone+"</h2>" +
"<h2><font color='blue'>Mobile No </font>::"  + JSONObject.MobileNo+"</h2>";

document.write(str);
</script>
```

**Name ::Amit**

**Address ::B-123 Bungalow**

**Age ::23**

**Phone No ::011-4565763**

**Mobile No ::981100092**

# Demo

```
<script >
var students = {
    "Students": [
            { "Name": "Amit Goenka",
              "Major": "Physics"
            },
            { "Name": "Smita Pallod",
              "Major": "Chemistry"
            },
            { "Name": "Rajeev Sen",
              "Major": "Mathematics"
            }] }
var i=0
document.writeln("students.Students.length : " + students.Students.length);
for(i=1;i<students.Students.length+1;i++)  {
    document.writeln("<b>Name : </b>" + students.Students[i].Name + "   ");
    document.writeln("<b>Majoring in : </b>" + students.Students[i].Major);
    document.writeln("</br>");
}
</script>
```

students.Students.length : 3
**Name :** Amit Goenka **Majoring in** : Physics
**Name :** Smita Pallod **Majoring in** : Chemistry
**Name :** Rajeev Sen **Majoring in** : Mathematics

# What is Ajax ?

- **"Asynchronous JavaScript And XML"**
  - AJAX is not a programming language, but a technique for making the user interfaces of web applications more responsive and interactive
  - It provide a simple and standard means for a web page to communicate with the server without a complete page refresh.
- **Why Ajax?**
  - Intuitive and natural user interaction
    - No clicking required. Call can be triggered on any event
    - Mouse movement is a sufficient event trigger
  - "Partial screen update" replaces the "click, wait, and refresh" user interaction model
    - Only user interface elements that contain new information are updated (fastresponse)
  - The rest of the user interface remains displayed as it is without interruption (no loss of operational context)

# XMLHttpRequest

- JavaScript object - XMLHttpRequest object for asynchronously exchanging the XML data between the client and the server
- XMLHttpRequest Methods
  - open("method", "URL", syn/asyn) : Assigns destination URL, method, mode
  - send(content) : Sends request including string or DOM object data
  - abort() : Terminates current request
  - getAllResponseHeaders() : Returns headers (labels + values) as a string
  - getResponseHeader("header") : Returns value of a given header
  - setRequestHeader("label","value") : Sets Request Headers before sending
- XMLHttpRequest Properties
  - Onreadystatechange : Event handler that fires at each state change
  - readyState values – current status of request
  - Status : HTTP Status returned from server: 200 = OK
  - responseText : get the response data as a string
  - responseXML : get the response data as XML data

# Creating an AJAX application

- **Step 1: Get an instance of XHR object**

```
if (window.XMLHttpRequest) { // Mozilla, Safari, IE7+ ...
    xhr = new XMLHttpRequest();
} else if (window.ActiveXObject) { // IE 6 and older
    xhr = new ActiveXObject("Microsoft.XMLHTTP");
}
```

- **Step 2: Make the request**

```
xhr.open('GET', 'http://www.example.org/some.file', true);
xhr.send(null);
```

```
xhr.open("POST", "AddNos.jsp");
xhr.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
xhr.send("tno1=100&tno2=200");
```

- **Step 3 : Attach callback function to xhr object**

```
httpRequest.onreadystatechange = function(){
    // process the server response
};
```

# Ajax Demo

```
<script >
var xhr;
function getData(){
  getHTTPRequestObject();
  if(xhr){
    xhr.open("GET", "Sample.txt", true);
    xhr.send();
    xhr.onreadystatechange = function(){
      if(xhr.readyState == 4 &&  xhr.status == 200){
          document.getElementById("lblresult").innerHTML=xhr.responseText;
      }
    } //end of callback function
}}
function getHTTPRequestObject() {
      if (window.XMLHttpRequest) { // Mozilla, Safari, IE7+ ...
          xhr = new XMLHttpRequest();
      } else if (window.ActiveXObject) { // IE 6 and older
          xhr = new ActiveXObject("Microsoft.XMLHTTP");
 }}
</script> <body>
  <input type="button" onclick="getData()"  value="Getresult"/>
  <div id="lblresult"></div>
</body>
```

//sample.txt
hi how r u this is the response data from file

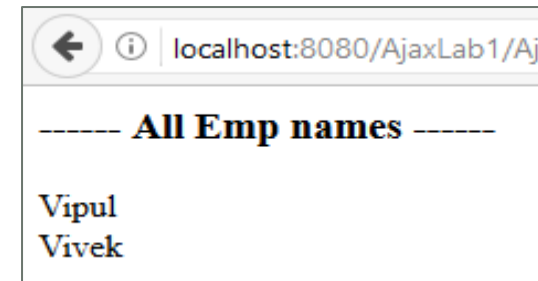Getresult
hi how r u this is the response data from file

# AJAX Demo with XML

```
<script language="javascript" type="text/javascript">
var xmlhttp;
function getData(){
  getHTTPRequestObject();
  if(xmlhttp){
    xmlhttp.open("GET", "Employee.xml", true);
    xmlhttp.send();
    xmlhttp.onreadystatechange = function(){
        if(xmlhttp.readyState == 4 && ixmlhttp.status == 200){
            xmlDoc=xmlhttp.responseXML;
            x=xmlDoc.getElementsByTagName("EmpName");
            document.write("<h3>--- Emp names ----</h3><br>");
            for (i=0;i<x.length;i++){
                document.write(x[i].childNodes[0].nodeValue  + "<br>");
            }
    } }
</script>
<body>
<input type="button" onclick=" getData()" value="Getresult"/>
<div id="lblresult"></div>
</body>
</html>
```

```
<Employees>
  <Employee>
    <empid>1001</empid>
    <EmpName>Vipul</EmpName>
    <Desig>Software Analyst</Desig>
  </Employee>
 <Employee>
  <Empid>1002</Empid>
    <EmpName>Vivek</EmpName>
    <Desig>Software Analyst</Desig>
  </Employee>
</Employees>
```

localhost:8080/AjaxLab1/Aj

------ **All Emp names** ------

Vipul
Vivek

# AJAX Demo with JSON

```
<script >
 var xmlhttp;
 function getData(){
   getHTTPRequestObject();
   if(xmlhttp){
     xmlhttp.open("GET", "EmpJSONData.txt", true);
    xmlhttp.onreadystatechange = function(){
      if(xmlhttp.readyState == 4 && xmlhttp.status == 200){
          var obj = JSON.parse(xmlhttp.responseText);
          var displaytext = "";
          displaytext += "Emp name : " + obj.name + "<br>" +
                   "Designation : " + obj.desig + "<br>" +
                   "Age : " + obj.age + "<br>" +
                   "Salary : " + obj.sal;
        document.getElementById("lblres").innerHTML = displaytext;
      }
}}
</script><body>
<h3 id="lblres">Result</h3>
<input type="button" id="btngetjsondata" onclick="getData()"  value="GetData">
</body>
```

**Emp name : Kapil Verma**
**Designation : ASE**
**Age : 23**
**Salary : 22000**

[ GetJsonData ]

```
{ "name":"Kapil Verma",
  "desig":"ASE",
  "age":23,
  "sal":22000
}
```