# JSON学习

**从结构上看，所有的数据（data）最终都可以分解成三种类型**：

任何疑问都可以咨询本人：148334733

第一种类型是**标量**（scalar），也就是一个单独的字符串（string）或数字（numbers），比如"北京"这个单独的词。

第二种类型是**序列**（sequence），也就是若干个相关的数据按照一定顺序并列在一起，又叫做数组（array）或列表（List），比如"北京，上海"。

第三种类型是**映射**（mapping），也就是一个名/值对（Name/value），即数据有一个名称，还有一个与之相对应的值，这又称作散列（hash）或字典（dictionary），比如"首都：北京"。

1） 并列的数据之间用逗号（", "）分隔。

2） 映射用冒号（": "）表示。

3） 并列数据的集合（数组）用方括号("[]")表示。

4） 映射的集合（对象）用大括号（"{}"）表示。

[  
　　{"城市":"北京","面积":16800,"人口":1600},  
　　{"城市":"上海","面积":6400,"人口":1800}  
]

成组的键/值数组对，用[]

[

{ 数组 },

{ 数组 },

{ 数组 },

]

JSON具有以下这些形式：

**对象（*object*）** 是一个无序的“‘名称/值’对”集合。一个对象以“{”（左括号）开始，“}”（右括号）结束。每个“名称”后跟一个“:”（冒号）；“‘名称/值’ 对”之间使用“,”（逗号）分隔。



**数组（*array*）** 是值（value）的有序集合。一个数组以“[”（左中括号）开始，“]”（右中括号）结束。值之间使用“,”（逗号）分隔。



**值（*value*）** 可以是双引号括起来的字符串（*string*）、数值(number)、true、false、 null、对象（object）或者数组（array）。这些结构可以嵌套。



**字符串（*string*）** 是由双引号包围的任意数量Unicode字符的集合，使用反斜线转义。一个字符（character）即一个单独的字符串（character string）。

JSON的字符串（*string*）与C或者Java的字符串非常相似。



**数值（*number*）** 也与C或者Java的数值非常相似。只是JSON的数值没有使用八进制与十六进制格式。



## 拿到了服务器的JSON数据，我们应该如何使用？

页面中使用JSON

function handleJson() {

var j={"name":"Michael","address":

{"city":"Beijing","street":" Chaoyang Road ","postcode":100025}

};

document.write(j.name);

document.write(j.address.city);

}

假定服务器返回的 JSON 数据是上文的：

{"name":"Michael","address":

{"city":"Beijing","street":" Chaoyang Road ","postcode":100025}

}

只需将其赋值给一个 JavaScript 变量，就可以立刻使用该变量并更新页面中的信息了，相比 XML 需要从 DOM 中读取各种节点而言，JSON 的使用非常容易。我们需要做的仅仅是发送一个 Ajax 请求，然后将服务器返回的 JSON 数据赋值给一个变量即可。有许多 Ajax 框架早已包含了处理 JSON 数据的能力，例如 Prototype（一个流行的 JavaScript 库：http://prototypejs.org）提供了 evalJSON() 方法，能直接将服务器返回的 JSON 文本变成一个 JavaScript 变量：

new Ajax.Request("http://url", {

method: "get",

onSuccess: function(transport) {

var json = transport.responseText.evalJSON();

// TODO: document.write(json.xxx);

}

});