# JSON简介

## 什么是JSON

1）JSON指的是JavaScript Object Notation（JavaScript对象表示法），JSON本质上是一个

JavaScript对象的字符串表现形式。

2）JSON是存储和交换文本信息的语法，类似XML，但JSON是轻量级。

## JSON特点

1. 虽然JSON使用JavaScript语法来描述对象，但是JSON仍然独立于语言和平台，不受编程语言的限制，JSON解析器和JSON库支持许多不同的编程语言。
2. JSON具有自我描述性，更易理解。

## JSON与XML

JSON和XML都可以用作于数据交换与传输或者充当配置文件。JSON实例如下：



XML实例如下：



JSON和XML相同点如下：

·JSON和XML数据都是“自我描述”，都易于理解

·JSON和XML数据都是有层次的结构

·JSON和XML数据可以被大多数编程语言使用

·都可以用作数据传输和充当配置文件

JSON和XML不同点如下：

·JSON不需要结束标签

·JSON更加简短

·JSON读写速度更快

·JSON可以使用数组

为什么JSON比XML更好？

·XML比JSON更难解析

·JSON可以直接使用现有的JavaScript对象解析

·针对Ajax应用，JSON比XML数据加载更快，而且更简单

**使用XML**

1. 读取XML文档
2. 使用XML DOM来循环遍历文档
3. 读取值并存储在变量中

**使用JSON**

1. 读取JSON字符串
2. 用eval( )处理JSON字符串

# 使用JSON

## JSON语法

JSON语法是JavaScript对象表示语法的子集。

·数据通过名称/值定义通过逗号分隔

·大括号{}保存对象

·中括号[]保存数组，数组可以包含多个对象。



## JSON数据类型

JSON的值可以是：数字（整数或浮点数）、字符串（使用双引号包含）、逻辑值（true或false）、数组（在中括号[]中）、对象（在大括号{}中）、null



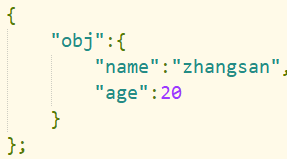
数字：



字符串：

 逻辑值：

 数组：

 对象：

 null：

## JSON对象

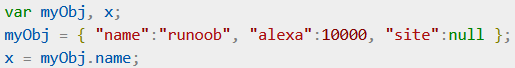
JSON对象使用大括号包含起来，如下：



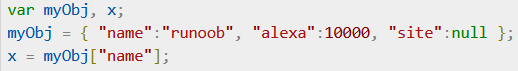
key必须是一个字符串，value是一个合法的JSON数据类型，key和value之间使用:分隔，每个key/value对之间使用，号分隔。

**访问对象值**

可以使用.的方式来访问对象某个属性的值。如下：

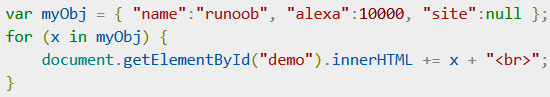


也可以通过中括号[]来访问对象某个属性的值。如下：

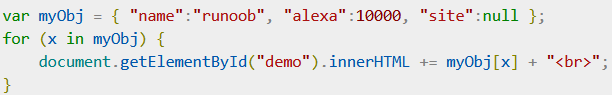


**遍历对象**

可以使用for-in来遍历对象的属性名。如下：

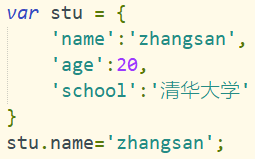


在for-in遍历对象时，使用中括号[]来访问属性的值。如下：



**添加属性或修改对象属性值**

可以使用对象.属性名的方式来修改对象属性的值。如下：



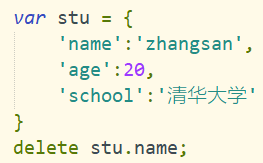
也可以通过对象 [‘属性名’]的方式修改对象属性的值。如下：



在修改对象的属性值时，如果属性不存在，则表示添加属性。

**删除对象属性**

可以使用delete关键字删除对象的属性。如下：



也可以使用中括号的形式表示：



## JSON数组

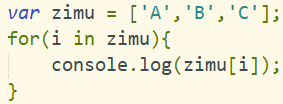
JSON数组使用中括号{}包含。如下：



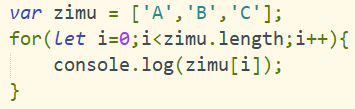
数组的值必须是合法的JSON数据类型，也可以是JavaScript表达式；多个元素使用，分隔。

**遍历数组**

可以使用for-in来遍历数组：



也可以使用for循环遍历数组：



**修改数组值或扩充数组**

使用索引号可以修改指定索引处元素的值。如下：



使用索引号修改数组值时，如果该索引并不在数组长度内，则会将数组长度扩充至此，中间没有给值的元素的值为undfined。如下：



**删除数组元素**

可以使用delete关键字来删除指定索引处的元素：



但并没有删除，而是将删除的元素的值改为了undfined

## JSON文件

JSON文件的后缀名是.json，JSON文本的MIME类型是application/json。

## JSON字符串与JavaScript对象的转换

JavaScript提供了JSON.parse( )方法将JSON字符串转换为JavaScript对象，提供了JSON.stringify( )方法将JavaScript对象转换为JSON字符串。

**JSON.parse( )**

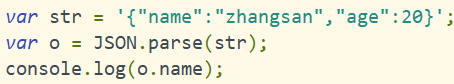
前端在接收后端返回的数据时一般是字符串，此时可以使用JSON.parse( )方法将其转换为JavaScript对象。语法如下：

**JSON.parse(text[,reviver])**

text是一个符合JSON语法规则的字符串；

reviver是可选的，一个转换结果函数，将为对象的每个成员调用此函数。

如下，将JSON字符串转换为JavaScript对象：



**JSON.stringify( )**

前端向服务器发送数据时一般是字符串，使用JSON.stringify( )可以将JavaScript对象转换为JSON字符串。语法如下：

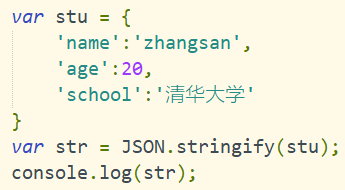
**JSON.stringify(value[,replacer[,space]])**

value是需要转换的JavaScript值，通常为对象或数组；

replacer是可选的，是用于转换结果的函数或数组。如果replacer为函数，则stringify将调用该函数，并传入每个成员的键和值。使用返回值，而不是使用原始值。如果此函数返回undefined，则排除成员。根对象的键是一个空字符串：””。如果replacer是一个数组，则仅转换该数组中具有键值的长远。成员的转换顺序与键在数组中的顺序一样。当value参数为数组时，将忽略replacer数组；

space也是可选参数，文本添加缩进、空格和换行符，如果space是一个数字，则返回值文本在每个级别缩进指定数目的空格，如果space大于10，则文本缩进10个空格，space参数的值也可以使用非数字，如:\t。

如下：将一个JavaScript对象转换为JSON字符串

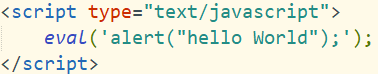


## eval函数

eval函数可以执行字符串形式的JavaScript代码，语法如下：

**eval(str)**

参数str是一个string类型的JavaScript代码。如下：



在浏览器中执行，将弹出一个警告框。