**数据类型**

1，[主要类型](http://baike.sogou.com/v53199.htm?fromTitle=%E6%95%B0%E6%8D%AE%E7%B1%BB%E5%9E%8B#para2)

[**数字型**](http://baike.sogou.com/v53199.htm?fromTitle=%E6%95%B0%E6%8D%AE%E7%B1%BB%E5%9E%8B#para3)

[**字节型**](http://baike.sogou.com/v53199.htm?fromTitle=%E6%95%B0%E6%8D%AE%E7%B1%BB%E5%9E%8B#para4)

[**文本型**](http://baike.sogou.com/v53199.htm?fromTitle=%E6%95%B0%E6%8D%AE%E7%B1%BB%E5%9E%8B#para5)

2，[操作方法](http://baike.sogou.com/v53199.htm?fromTitle=%E6%95%B0%E6%8D%AE%E7%B1%BB%E5%9E%8B#para6)

[**转换**](http://baike.sogou.com/v53199.htm?fromTitle=%E6%95%B0%E6%8D%AE%E7%B1%BB%E5%9E%8B#para7)

[**定植**](http://baike.sogou.com/v53199.htm?fromTitle=%E6%95%B0%E6%8D%AE%E7%B1%BB%E5%9E%8B#para8)

**简要介绍**

数据类型的出现是为了把数据分成所需内存大小不同的数据，编程的时候需要用大数据的时候才需要申请大内存，就可以充分利用内存。

ECMAScript中有5种简单数据类型：Undefined,Null,Boolea,Number和String。还有一种复杂数据类型—Object

“undefined”    ----------   如果值未定义                       Undefined

“boolean”      ----------     如果这个值是布尔值              Boolean

“string”        ----------     如果这个值是字符串              String

“number”      ----------     如果这个值是数值类型           Number

“object”        ----------     如果这个值是对象或null        Object

“function”       ----------     如果这个值是函数                 Function

**JavaScript值类型和引用类型有哪些**

（1）值类型：数值、布尔值、字符串、null、undefined。

（2）引用类型：对象、数组、函数。

**什么是 JSON**

JSON概念很简单，JSON 是一种轻量级的数据格式，他基于 javascript 语法的子集，即数组和对象表示。由于使用的是 javascript 语法，因此JSON 定义可以包含在javascript 文件中，对其的访问无需通过基于 XML 的语言来额外解析。不过在使用 JSON 之前，很重要的一点是理解 javascript 中数组及对象字面量的特殊语法

# XML的定义

    扩展标记语言 (Extensible Markup Language, XML) ，用于标记电子文件使其具有结构性的标记语言，可以用来标记数据、定义数据类型，是一种允许用户对自己的标记语言进行定义的源语言。 XML是标准通用标记语言 (SGML) 的子集，非常适合 Web 传输。XML 提供统一的方法来描述和交换独立于应用程序或供应商的结构化数据。

**[ XML ]**  
使 用XML作为传输格式的优势：  
1. 格式统一, 符合标准  
2. 容易与其他系统进行远程交互, 数据共享比较方便  
  
缺点:  
1. XML文件格式文件庞大, 格式复杂, 传输占用带宽  
2. 服务器端和客户端都需要花费大量代码来解析XML, 不论服务器端和客户端代码变的异常复杂和不容易维护  
3. 客户端不同浏览器之间解析XML的方式不一致, 需要重复编写很多代码  
4. 服务器端和客户端解析XML花费资源和时间  
  
  
**[ JSON ]**  
那么除了XML格式, 还有没有其他格式, 有一种叫做JSON (JavaScript Object Notation) 的轻量级数据交换格式能够替代XML的工作.  
  
优点:  
1. 数据格式比较简单, 易于读写, 格式都是压缩的, 占用带宽小  
2. 易于解析这种语言, 客户端JavaScript可以简单的通过eval\_r()进行JSON数据的读取  
3. 支持多种语言, 包括ActionScript, C, C#, ColdFusion, Java, JavaScript, Perl, PHP, Python, Ruby等语言服务器端语言, 便于服务器端的解析  
4. 在PHP世界, 已经有PHP-JSON和JSON-PHP出现了, 便于PHP序列化后的程序直接调用. PHP服务器端的对象、数组等能够直接生JSON格式, 便于客户端的访问提取.  
5. 因为JSON格式能够直接为服务器端代码使用, 大大简化了服务器端和客户端的代码开发量, 但 是完成的任务不变, 且易于维护  
  
缺点:  
1. 没有XML格式这么推广的深入人心和使用广泛, 没有XML那么通用性  
2. JSON格式目前在Web Service中推广还属于初级阶段

# JSON 和 XML 优缺点的比较

1.       在可读性方面，JSON和XML的数据可读性基本相同。JSON和XML的可读性可谓不相上下，一边是建议的语法，一边是规范的标签形式，很难分出胜负。

2.       在可扩展性方面，XML天生有很好的扩展性，JSON当然也有，没有什么是XML能扩展，JSON不能的。

3.       在编码难度方面，XML有丰富的编码工具，比如Dom4j、JDom等，JSON也有json.org提供的工具，但是JSON的编码明显比XML容易许多，即使不借助工具也能写出JSON的代码，可是要写好XML就不太容易了。

4.       在解码难度方面，XML的解析得考虑子节点父节点，让人头昏眼花，而JSON的解析难度几乎为0。这一点XML输的真是没话说。

5.       在流行度方面，XML已经被业界广泛的使用，而JSON才刚刚开始，但是在Ajax这个特定的领域，未来的发展一定是XML让位于JSON。到时Ajax应该变成Ajaj(Asynchronous Javascript and JSON)了。

6.       JSON和XML同样拥有丰富的解析手段。

7.       JSON相对于XML来讲，数据的体积小。

8.       JSON与JavaScript的交互更加方便。

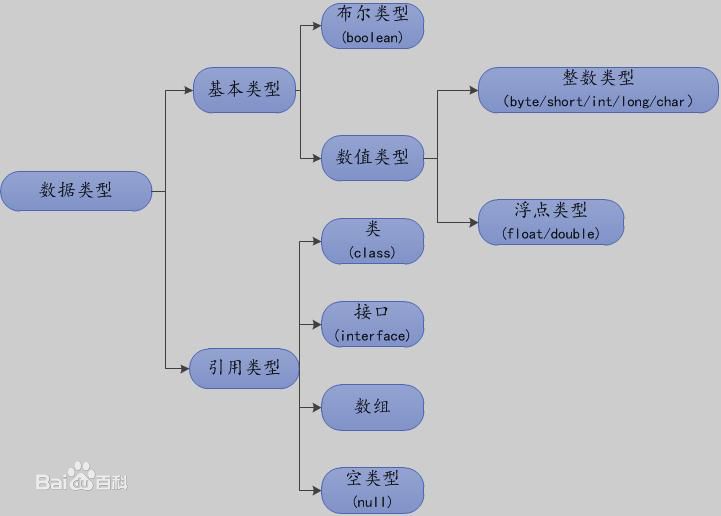
9.       JSON对数据的描述性比XML较差。

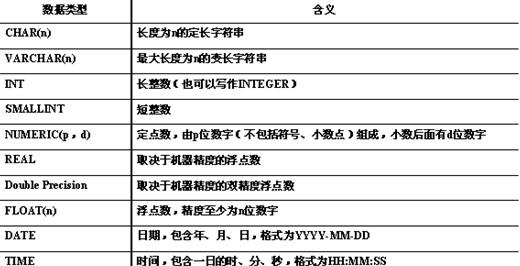
10.   JSON的速度要远远快于XML。

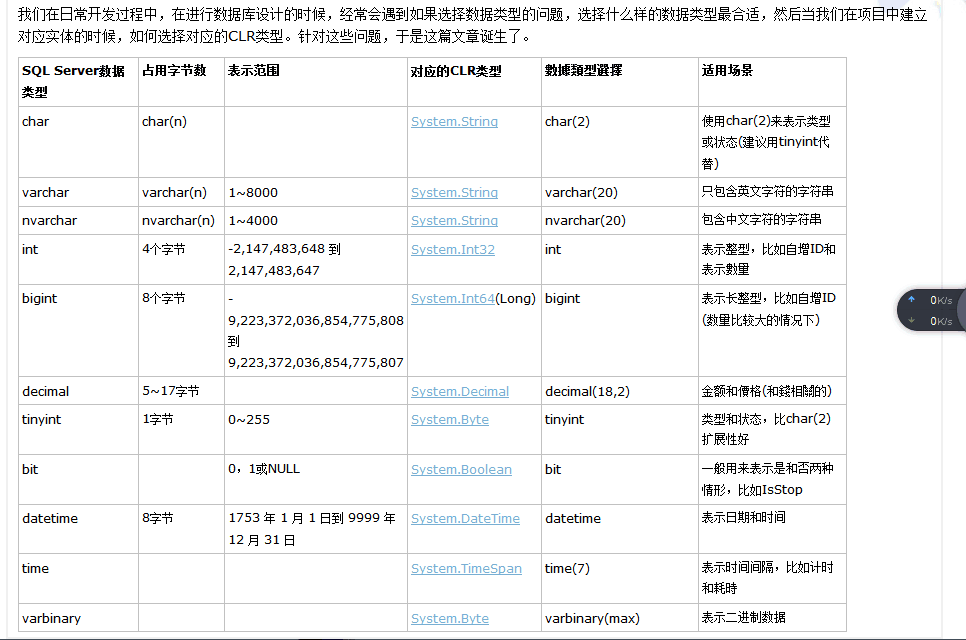
**XML的定义**

    扩展标记语言 (Extensible Markup Language, XML) ，用于标记电子文件使其具有结构性的标记语言，可以用来标记数据、定义数据类型，是一种允许用户对自己的标记语言进行定义的源语言。 XML是标准通用标记语言 (SGML) 的子集，非常适合 Web 传输。XML 提供统一的方法来描述和交换独立于应用程序或供应商的结构化数据。

**[ XML ]**  
使 用XML作为传输格式的优势：  
1. 格式统一, 符合标准  
2. 容易与其他系统进行远程交互, 数据共享比较方便  
  
缺点:  
1. XML文件格式文件庞大, 格式复杂, 传输占用带宽  
2. 服务器端和客户端都需要花费大量代码来解析XML, 不论服务器端和客户端代码变的异常复杂和 不容易维护  
3. 客户端不同浏览器之间解析XML的方式不一致, 需要重复编写很多代码  
4. 服务器端和客户端解析XML花费资源和时间

****

****

****