



# C/C++学习指南（补充篇）之 XML编程

## 1 XML文档格式

# XML

XML, EXtensible Markup Language  
可扩展标记语言

实际上，XML的应用一般比较简单，还够不上一个“语言”的级别。

XML是一个数据格式，符合这种格式的文件后缀名通常为\*.xml

演示：在C:\下搜索\*.xml...

# 入门示例

下面给出一个简单的例子：

```
<?xml version="1.0" encoding="GBK"?>
<root>
    <host> afanihao.cn </host>
    <port> 8080 </port>
</root>
```

优点：可读性强！

缺点：解析费时、占用空间大、效率低！

# XML文档格式

XML文档(document)用于传输数据  
总体格式为:

<code>&lt;?xml version="1.0" encoding="GBK"?&gt;</code>	声明
<code>&lt;!DOCTYPE ... xxx.dtd ...&gt;</code>	DTD(常省略)
<code>&lt;root&gt;     &lt;host&gt; afanihao.cn &lt;/host&gt;     &lt;port&gt; 8080 &lt;/port&gt; &lt;/root&gt;</code>	内容

参考教程: [w3c school](http://www.w3school.com.cn/xml/)

<http://www.w3school.com.cn/xml/>

# XML声明

XML文档的**第一行**必须是声明语句

例如，

```
<?xml version="1.0" encoding="GBK"?>
```

`version` 版本号：总是1.0

`encoding` 文字编码：GBK, UTF-8等，参考补充篇之《中文问题》

# XML语法

```
<?xml version="1.0" encoding="GBK"?>
```

```
<root>
```

```
    <host> afanihao.cn </host>
```

```
    <port> 8080 </port>
```

```
</root>
```

简言之，XML文档是一个由**元素**构成的树型结构。。。root, host, port都是元素

# XML的元素

示例：

```
<Contact id="1001" page="1">  
    <Name> 邵发 </Name>  
    . . .  
</Contact>
```

1. 名称Name：必须配对！

```
<Name> shaofa </Name>  
<name> shaofa </name>
```

以下写法不对：

```
<NAME> shaofa </name>
```

NAME与name不配对

# XML的元素

## 2. 元素的属性

<Contact id="1001" page="1">

- (1) 采用 key="value" 形式
- (2) key值唯一
- (3) 多个属性以空格分开



# XML的元素

3. 元素的内容可以是Text（文本）

```
<host> afanihao.cn </host>
```

也可以是子元素 Child Element，支持往下嵌套。。

```
<note>
```

```
    <from> 小张 </from>
```

```
    <time> 2016/1/22 16:20 </time>
```

```
    <content> 去哪吃饭 ? </content>
```

```
</note>
```

最终构成一个树型结构。

# XML文档结构

1. 必须有一个唯一的根元素

```
<?xml version="1.0" encoding="GBK"?>
```

```
<root>
```

... 所有内容放在根元素里 ...

```
</root>
```

元素的名称可以任意指定。这里是叫root。

最终形成一个树形结构

# XML文档结构

## 2. 元素内容

元素的内容可以是Text（文本）

```
<host> afanihao.cn </host>
```

也可以是子元素 Child Element，支持往下嵌套。。

```
<note>
```

```
    <from> 小张 </from>
```

```
    <time> 2016/1/22 16:20 </time>
```

```
    <content> 去哪吃饭 ? </content>
```

```
</note>
```

# XML文档结构

## 3. 兄弟元素Sibling

处于同一元素底下的子元素，彼此为兄弟关系。  
兄弟元素可以同名。

```
<Contact>
```

```
    <Name> 邵发 </Name>
```

```
    <Phone> 13810012345 </Phone>
```

```
    <Phone> 18600099999 </Phone>
```

```
</Contact>
```

一个联系人Contact可以有多个手机号。。

# 形式良好的 与 内容合法的

形式良好的 Well Formed

如果语法检查没有问题，那就是形式良好的。言下之意，不管你内容是否合法。

```
<?xml version="1.0" encoding="GBK"?>
```

```
<root>
```

```
    <host> afanihao.cn </host>
```

```
    <port> bu hao yi si </port>
```

```
</root>
```

这个XML文档是形式良好的，但内容有问题！

（ 由DTD负责验证内容是否合乎要求 ）

# 小节

一、介绍了XML的特点：

一种用文本形式传输数据的文档格式

二、介绍了元素的相关语法

三、本篇教程只介绍形式良好的XML

至少内容是否良好，则可以由程序来验证。

总体上很简单！ DTD比较复杂，但我们一般只用简单的！