# XML基概念和基本语法

## 1-1 什么是XML？都有哪些版本？

XML为可扩展[标记语言](https://baike.baidu.com/item/%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80" \t "_blank)，全称为Extensible Markup Language。XML语法上和HTML比较相似，比如都有类似这样的符号：<xx字符</xx>，作用上也都能用来传输与显示数据。

W3C在1998年2月发布1.0版本，2004年2月又发布1.1版本但因为1.1版本不能向下兼容1.0版本，所以1.1没有人用。我们现在实际项目开发中所使用的都是1.0版本，1.1版本不予考虑。

## 1-2 XML与HTML区别

1. 在HTML中不区分大小写，而XML对大小写十分敏感，必须严格区分 <AA></AA> <BB></Bb>

1. 在HTML中，有时不严格(单目)，如果上下文清楚地显示出段落或者列表在何处结尾，那 么就可以省略</p>或者</li>之类的结束标签<br/> <br>
2. 而在XML中，省略关闭标签是非法的，所有元素都必须有关闭标签。而像<br />这种拥 有单个标签而没有匹配的结束标签的必须用一个/字符作为结尾
3. 在HTML中，引导属性值的引号是可用可不用的。在XML中，属性值必须被引号包围
4. HTML会把多个连续的空格字符合并为一个。在XML中，文档中的空格不会被删减
5. HTML使用固有的标签，而且都是预定义的XML中并没有固有的标签，所有的标签都是 自定义的
6. XML文档和HTML文档都会形成一种树结构。XML 文档必须有一个元素是所有其他元素 的父元素，该元素称为根元素

8. 通俗地说就是HTML旨在显示数据，而XML旨在传输数据。

## 1-3 XML用处

1.数据传输(数据传输媒介)

2.配置文件

xml作为配置文件来使用,在每一项技术中,xml的配置文件的作用都不相同.

比如在学习servlet/jsp的时候,我们要使用到的是叫做web.xml的配置文件.这个配置文件的作用是web开发的主配置文件.所有的servlet类必须经过在web.xml中进行注册之后,才能够使用.

在我们以后的框架的学习中,每一款框架都会要求有xml配置文件的应用,不同的框架对于xml配置的作用都是不同的.

## 1-4 XML语法

（1）创建普通文本文件，后缀名为xml

（2）声明

语法：

<?xml version="1.0" encoding="UTF-8"?>

解析：

a、文档声明必须以<?xml开头，必须以?>结尾

b、文档声明必须在文档中的0行0列

c、文档声明的3个属性

versioin属性：指定XML文档版本.必须属性，因为我们 不会选择1.1， 只会选择1.0；

encoding属性：指定当前文档的编码。可选属性，默认 值是utf-8；

（3）元素

语法：

<servlet></servlet>

解析

a、元素是XML文档中最重要的组成部分，

b、普通元素的结构开始标签、元素体、结束标签组成。例如： <hello>你好</hello>

c、元素体：元素体可以是元素，也可以是文本，例如：

<b><a> 你好</a></b>

d、空元素：空元素只有开始标签,而没有结束标签,但元素必 须 自己闭合.例如：<c/>

<br/>

<input/> <hr/>

e、元素命名：

区分大小写

不能使用空格，不能使用冒号:

不建议以XML、xml、Xml开头

良好的XML文档，必须有一个根元素。

（4）属性

语法：

<web-app version="2.5">

解析：

a、属性是元素的一部分，它必须出现在元素的开始标签中

b、属性的定义格式：属性名=属性值，其中属性值必须使用 单 引或双引

c、一个元素可以有0~N个属性，但一个元素中不能出现同名 属性

d、属性名不能使用空格、冒号等特殊字符，且必须以字母开 头

（5）注释

语法：

<!-- 注释掉的内容 -->

解析：

与html一样，以“<!--”开始，以“-->”结束。xml解析器会 自动忽 略掉注释掉的xml代码。

（6）转义字符

<p>

5&lt;3

</p>

语法：

&lt; 对应<小于号   
 &gt;对应>大于号   
 &amp; 对应&和   
 &apos; 对应' 单引号   
 &quot; 对应" 双引号

解析：

a、因为很多符号已经被XML文档结构所使用，所以在元素 体或属性值中想使用这些符号就必须使用转义字符

b、xml中的转义字符与html中的使用方式一样

（7）CDATA区

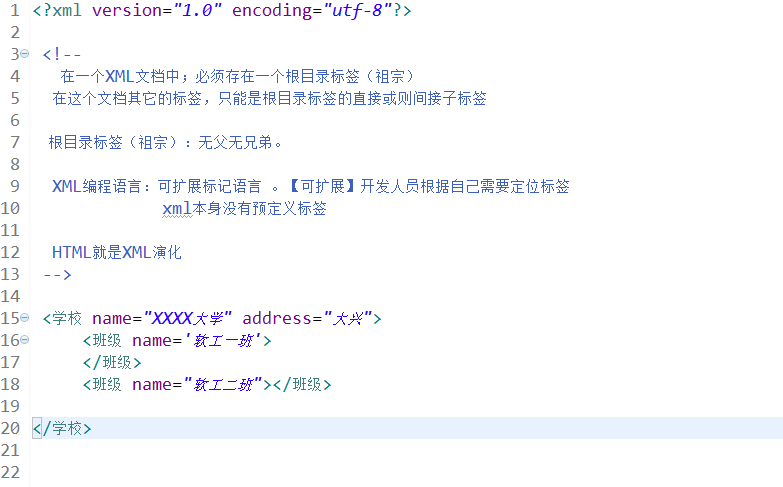
如果我们表述的文本中出现了大量的需要转义的字符,如果一个一个进行转换,那么会非常麻烦,我们可以使用CDATA来完成,写在CDATA中的内容是不需要进行转义操作的.

语法：

<![CDATA[

<><><><><><>>>>>><><><><><><&&&&&&’’’””””””””””””’>>

]]>



# 2 XML文档解析

## 2-1什么是XML文档解析

将xml文档中的元素、属性、文本内容进行拆解，我们获得 这些组件的过程就叫做xml的解析工作。

## 2-2 XML文档解析方式(面试)

（1）sax解析:逐行解析,只能查询。

（2）dom解析:一次性将文档加载到内存中,增删改查的操作都可以。

SAX和dom区别（面试）

SAX 优点：

节省内存

缺点：

运行效率慢

dom 优点：

运行速度快

缺点：

一次性将xml文档所有内容加载到内存，大量消耗内存。

## 2-3 常见的XML解析开发包

JAXP：sun公司提供支持DOM和SAX开发包

JDom：dom4j前身

dom4j：实际开发中最常用的解析开发包（Apache组织）

JAVASE

[Apache组织] ------------ sun -------- oracle

## 2-4 使用dom4j技术解析XML案例

（1）导入jar包

dom4j-1.6.1.jar

（2）操作流程

dom4j 必须使用核心类SaxReader加载xml文档获得 Document，通过Document 对象获得文档的根元素，然后就可 以继续操作了。

（3）常用API

a、SaxReader对象

read(…) 加载执行xml文档

Document对象

getRootElement() 获得根元素

b、Element对象

elements(…) 获得指定名称的所有子元素。可以不指定 名称

element(…) 获得指定名称第一个子元素。可以不指定 名称

getName() 获得当前元素的元素名

getText() 获得当前元素的文本内容

（4）具体代码实现方式：

a.导入jar包

b.创建一个核心对象 SAXReader

new SAXReader();

c.将xml文档加载到内存中形成一棵树

Document doc=reader.read(文件)

d.获取根节点

Element root=doc.getRootElement();

e.通过根节点就可以获取其他节点(文本节点,属性节点, 元素节点)

获取所有的子元素

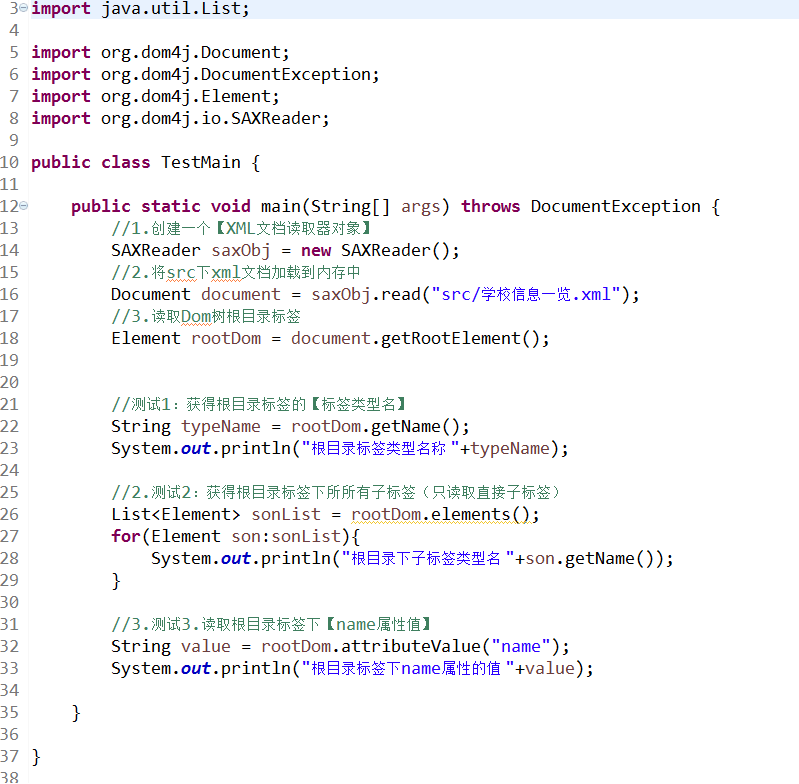
List<Element> list=root.elements()

1.SAXReader:[XML文档读取器]

2.Document:管理在内存中生成的Dom树。相当于JavaScript使用document

3.Element: 管理Dom树中dom对象 Element对象就相当于dom对象

4.Attribute:管理Dom对象中属性



## 2-5 遍历XML文档的方式

遍历XML文档方式：

1.自定义递归规则。

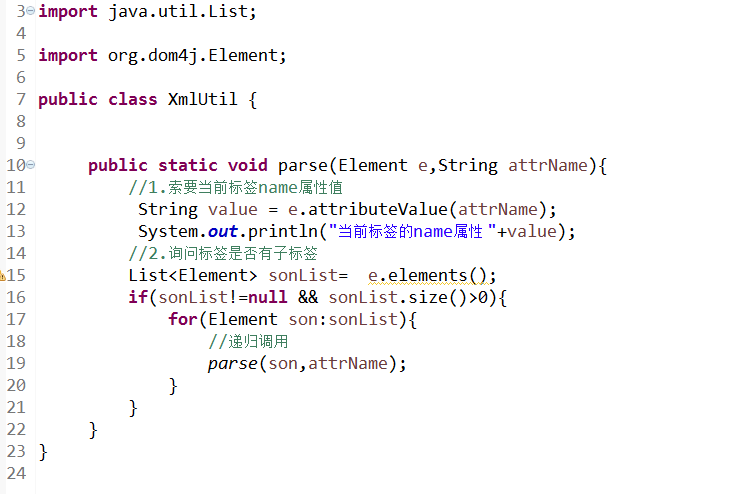
2.借助商用递归工具。 xPath.jar 使用【路径格式】对标签进行定位。jquery（"选择器/过滤器"）

String xPath="//班级"：相当于$("div");定位XML文档所有的【班级标签】

String xPath="//@name" 定位XML文档所有的name属性

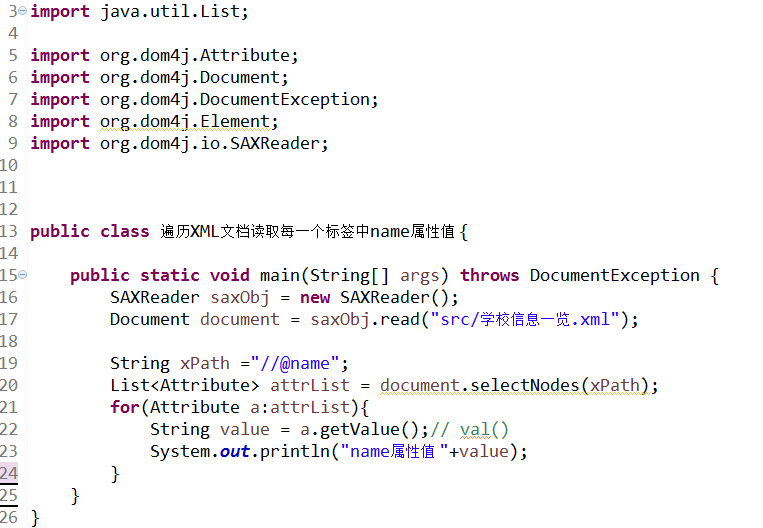
String xPath="//学员[@age>=25]": 定位所有age属性大于等于25的学员标签 $("div[name='one']")

## 2.6递归遍历





## 2.7 商用递归工具xPath.jar





# 3 XML约束

## 3-1 DTD约束(简单约束)

DTD（Document Type Definition），文档类型定义，用来约束xml文档。规定xml文档中元素的名称，子元素的名称及顺序，元素的属性等。



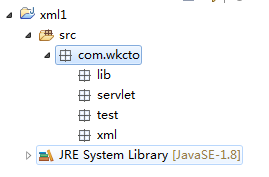
## 3-2 Schema约束（高级约束）

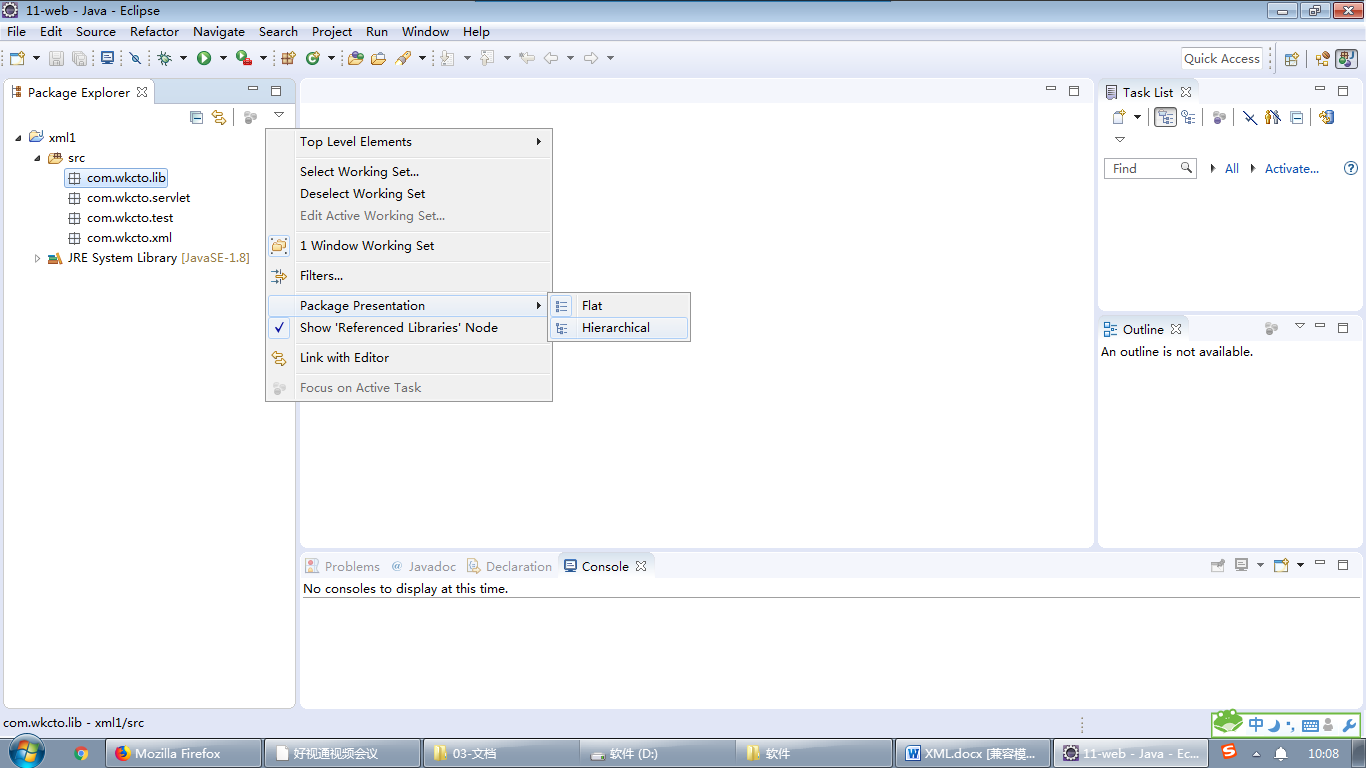
同DTD约束一样，也是用来约束xml文档用的技术。

从功能上讲，Schema约束要比DTD强大很多，是DTD 替代者

Schema本身也是xml文档，但Schema文档的扩展名为xsd，而不是xml。

创建包结构





## 3-3通过使用约束对web.xml进行搭建

见代码

## 3-4 面试题：XML与DTD、Schema的关系是什么

回答:dtd与schema也是xml,是传统xml文件的约束扩展

## 5-6 XML与JSON的比较

XML和JSON都可以完成不同系统之间数据交换。

XML体积较大，解析难度较大，所以在现代开发中通常采用JSON进行数据交换，因为JSON体积小，解析简单，而XML通常充当系统当中的配置文件