

2011年计算机二级考试JAVA知识点整理 (79) PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/646/2021_2022_2011_E5_B9_B4_E8_AE_A1_c97_646650.htm

3.7 Java与XML联合编程之SAX

篇 SAX概念 SAX是Simple API for XML的缩写，它并不是由W3C官方所提出的标准，可以说是“民间”的事实标准。实际上，它是一种社区性质的讨论产物。虽然如此，在XML中对SAX的应用丝毫不比DOM少，几乎所有的XML解析器都会支持它。与DOM比较而言，SAX是一种轻量型的方法。我们知道，在处理DOM的时候，我们需要读入整个的XML文档，然后在内存中创建DOM树，生成DOM树上的每个Node对象。当文档比较小的时候，这不会造成什么问题，但是一旦文档大起来，处理DOM就会变得相当费时费力。特别是其对于内存的需求，也将是成倍的增长，以至于在某些应用中使用DOM是一件很不划算的事(比如在applet中)。这时候，一个较好的替代解决方法就是SAX。SAX在概念上与DOM完全不同。首先，不同于DOM的文档驱动，它是事件驱动的，也就是说，它并不需要读入整个文档，而文档的读入过程也就是SAX的解析过程。所谓事件驱动，是指一种基于回调(callback)机制的程序运行方法。(如果你对Java新的代理事件模型比较清楚的话，就会很容易理解这种机制了)

在XMLReader接受XML文档，在读入XML文档的过程中就进行解析，也就是说读入文档的过程和解析的过程是同时进行的，这和DOM区别很大。解析开始之前，需要向XMLReader注册一个ContentHandler，也就是相当于一个事件监听器，在ContentHandler中定义了很多方法，比如startDocument()，

它定制了当在解析过程中，遇到文档开始时应该处理的事情。当XMLReader读到合适的内容，就会抛出相应的事件，并把这个事件的处理权代理给ContentHandler，调用其相应的方法进行响应。这样泛泛的说来或许有些不容易理解，别急，后面的例子会让你明白SAX的解析过程。看看这个简单XML文件：Ogden Nash Adam 当XMLReader读到标签时，就会调用ContentHandler.startElement()方法，并把标签名POEM作为参数传递过去。在你实现的startElement()方法中需要做相应的动作，以处理当出现时应该做的事情。各个事件随着解析的过程(也就是文档读入的过程)一个个顺序的被抛出，相应的方法也会被顺序的调用，最后，当解析完成，方法都被调用后，对文档的处理也就完成了。下面的这个表，列出了在解析上面的那个XML文件的时候，顺序被调用的方法：

项目	方法回调
{文档开始}	startDocument()
"\n"	characters("\n...", 6, 1)
startElement(null,"POEM",null,{Attributes})	startElement(null,"AUTHOR",null,{Attributes})
"Ogden Nash"	characters("\n...", 15, 10)
endElement(null,"AUTHOR",null)	characters("\n...", 34, 1)
endElement(null,"TITLE",null)	characters("\n...", 55, 1)
startElement(null,"LINE",null,{Attributes})	"Adam"
characters("\n...", 62, 4)	endElement(null,"LINE",null)
characters("\n...", 67, 1)	endElement(null,"POEM",null)
{文档结束}	endDocument()

100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com