一.DTD简介

1.文档类型定义(DTD)可定义合法的XML文档构建模块。它使用一系列合法的元素来定义文档的结构。

2.DTD可被成行地声明于XML文档中, 也可作为一个外部引用。

二.内部的DOCTYPE声明

1.假如DTD被包含在您的XML源文件中, 它应当通过下面的语法包装在一个 DOCTYPE 声明中：

|  |
| --- |
| <!DOCTYPE 根元素 [元素声明]> |

2.带有DTD的XML文档实例

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" standalone="yes" ?>  <!DOCTYPE students [  <!ELEMENT students (student\*)>  <!ELEMENT student (name?,age?,sex?)>  <!ELEMENT name (#PCDATA)>  <!ELEMENT age (#PCDATA)>  <!ELEMENT sex (#PCDATA)>  <!ATTLIST student id CDATA #IMPLIED>    ]>  <students>  <student id="10086">  <name>刘邦</name>  <age>18</age>  <sex>男</sex>  </student>  <student id="10087">  <name>吕雉</name>  <age>16</age>  <sex>女</sex>  </student>  </students> |

三.外部文档声明

1.假如DTD位于XML源文件的外部, 那么它应通过下面的语法被封装在一个 DOCTYPE 定义中:

|  |
| --- |
| <!DOCTYPE 根元素 SYSTEM "文件名"> |

2.这个XML文档和上面的XML文档相同, 但是拥有一个外部的DTD:

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" standalone="no" ?>  <!DOCTYPE students SYSTEM "./students.dtd">  <students>  <student id="10086">  <name>刘邦</name>  <age>18</age>  <sex>男</sex>  </student>  <student id="10087">  <name>吕雉</name>  <age>16</age>  <sex>女</sex>  </student>  </students> |

3.这是包含DTD的"students.dtd"文件：

|  |
| --- |
| <!ELEMENT students (student\*)>  <!ELEMENT student (name?,age?,sex?)>  <!ELEMENT name (#PCDATA)>  <!ELEMENT age (#PCDATA)>  <!ELEMENT sex (#PCDATA)>  <!ATTLIST student id CDATA #IMPLIED> |

四.为什么使用DTD

1.通过DTD, 您的每一个XML文件均可携带一个有关其自身格式的描述。

2.通过DTD, 独立的团体可一致地使用某个标准的DTD来交换数据。

3.而您的应用程序也可使用某个标准的DTD来验证从外部接收到的数据。

4.您还可以使用DTD来验证您自身的数据。