一、XML 简介

Q：什么是 XML？

A：

XML 指可扩展标记语言（EXtensible Markup Language）

XML 是一种标记语言，很类似 HTML

XML 的设计宗旨是传输数据，而非显示数据

XML 标签没有被预定义。您需要自行定义标签。

XML 被设计为具有自我描述性。

Q：XML与 HTML 的区别与联系？

A：

XML 不是 HTML 的替代。

XML 和 HTML 为不同的目的而设计：

XML 被设计为传输和存储数据，其焦点是数据的内容。

HTML 被设计用来显示数据，其焦点是数据的外观。

HTML 旨在显示信息，而 XML 旨在传输信息。

Q：XML需要学习语法吗？

A：XML 仅仅是纯文本

XML 没什么特别的。它仅仅是纯文本而已。有能力处理纯文本的软件都可以处理 XML。

不过，能够读懂 XML 的应用程序可以有针对性地处理 XML 的标签。标签的功能性意义依赖于应用程序的特性。

通过 XML 您可以发明自己的标签，如：

<note>

<to>George</to>

<from>John</from>

<heading>Reminder</heading>

<body>Don't forget the meeting!</body>

</note>

123456

上例中的标签没有在任何 XML 标准中定义过（比如 和 ）。这些标签是由文档的创作者发明的。

这是因为 XML 没有预定义的标签。

在 HTML 中使用的标签（以及 HTML 的结构）是预定义的。HTML 文档只使用在 HTML 标准中定义过的标签。

XML 允许创作者定义自己的标签和自己的文档结构。

Q：XML有什么用呢？

A：

XML 把数据从 HTML 分离

如果你需要在 HTML 文档中显示动态数据，那么每当数据改变时将花费大量的时间来编辑 HTML。

通过 XML，数据能够存储在独立的 XML 文件中。这样你就可以专注于使用 HTML 进行布局和显示，并确保修改底层数据不再需要对 HTML 进行任何的改变。

通过使用几行 JavaScript，你就可以读取一个外部 XML 文件，然后更新 HTML 中的数据内容。

XML 简化数据共享

在真实的世界中，计算机系统和数据使用不兼容的格式来存储数据。

XML 数据以纯文本格式进行存储，因此提供了一种独立于软件和硬件的数据存储方法。

这让创建不同应用程序可以共享的数据变得更加容易。

XML 简化数据传输

通过 XML，可以在不兼容的系统之间轻松地交换数据。

对开发人员来说，其中一项最费时的挑战一直是在因特网上的不兼容系统之间交换数据。

由于可以通过各种不兼容的应用程序来读取数据，以 XML 交换数据降低了这种复杂性。

二、XML结构

<?xml version="1.0" encoding="ISO-8859-1"?>

和HTML类似，这行表示XML 声明。它定义 XML 的版本 (1.0) 和所使用的编码 (ISO-8859-1 = Latin-1/西欧字符集)。

XML 文档形成了一种树结构，它从“根部”开始，然后扩展到“枝叶”。

XML 文档必须包含根元素。该元素是所有其他元素的父元素。

XML 文档中的元素形成了一棵文档树。这棵树从根部开始，并扩展到树的最底端。

所有元素均可拥有子元素：

<root>

<child>

<subchild>.....</subchild>

</child>

</root>

12345

下面用XML表示一本书的相关信息：

<bookstore>

<book category="COOKING">

<title lang="en">Everyday Italian</title>

<author>Giada De Laurentiis</author>

<year>2005</year>

<price>30.00</price>

</book>

<book category="CHILDREN">

<title lang="en">Harry Potter</title>

<author>J K. Rowling</author>

<year>2005</year>

<price>29.99</price>

</book>

<book category="WEB">

<title lang="en">Learning XML</title>

<author>Erik T. Ray</author>

<year>2003</year>

<price>39.95</price>

</book>

</bookstore>

1234567891011121314151617181920

例子中的根元素是 bookstore。文档中的所有 book 元素都被包含在 bookstore 中。

三、XML语法

所有 XML 元素都须有关闭标签

如：This is a paragraph

XML 标签对大小写敏感

XML 元素使用 XML 标签进行定义。

XML 标签对大小写敏感。在 XML 中，标签 Letter 与标签 letter 是不同的。

必须使用相同的大小写来编写打开标签和关闭标签：

<Message>这是错误的。</message>

<message>这是正确的。</message>

123

XML 必须正确地嵌套

<b><i>This text is bold and italic</i></b>

1

XML 文档必须有根元素

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。

<root>

<child>

<subchild>.....</subchild>

</child>

</root>

12345

XML 的属性值须加引号

与 HTML 类似，XML 也可拥有属性（名称/值的对）。

在 XML 中，XML 的属性值须加引号。请研究下面的两个 XML 文档。第一个是错误的，第二个是正确的：

<note date=08/08/2008>

<to>George</to>

<from>John</from>

</note>

1234

<note date="08/08/2008">

<to>George</to>

<from>John</from>

</note>

1234

实体引用

在 XML 中，一些字符拥有特殊的意义。

如果你把字符 “<” 放在 XML 元素中，会发生错误，这是因为解析器会把它当作新元素的开始。

这样会产生 XML 错误：

<message>if salary < 1000 then</message>

1

为了避免这个错误，请用实体引用来代替 “<” 字符：

<message>if salary &lt; 1000 then</message>

1

在 XML 中，有 5 个预定义的实体引用：

注释：在 XML 中，只有字符 “<” 和 “&” 确实是非法的。大于号是合法的，但是用实体引用来代替它是一个好习惯。

XML 中的注释

在 XML 中编写注释的语法与 HTML 的语法很相似：

<!-- This is a comment -->

1

在 XML 中，空格会被保留

HTML 会把多个连续的空格字符裁减（合并）为一个：

HTML: Hello my name is David.

输出: Hello my name is David.

12

在 XML 中，文档中的空格不会被删节。

XML 以 LF 存储换行