

# XML 树结构

XML 文档形成了一种树结构，它从"根部"开始，然后扩展到"枝叶"。

## 一个 XML 文档实例

XML 文档使用简单的具有自我描述性的语法：

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

第一行是 XML 声明。它定义 XML 的版本（1.0）和所使用的编码（UTF-8：万国码, 可显示各种语言）。

下一行描述文档的**根元素**（像在说："本文档是一个便签"）：

```
<note>
```

接下来 4 行描述根的 4 个**子元素**（to, from, heading 以及 body）：

```
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
```

最后一行定义根元素的结尾：

```
</note>
```

您可以假设，从这个实例中，XML 文档包含了一张 Jani 写给 Tove 的便签。

XML 具有出色的自我描述性，您同意吗？

## XML 文档形成一种树结构

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。

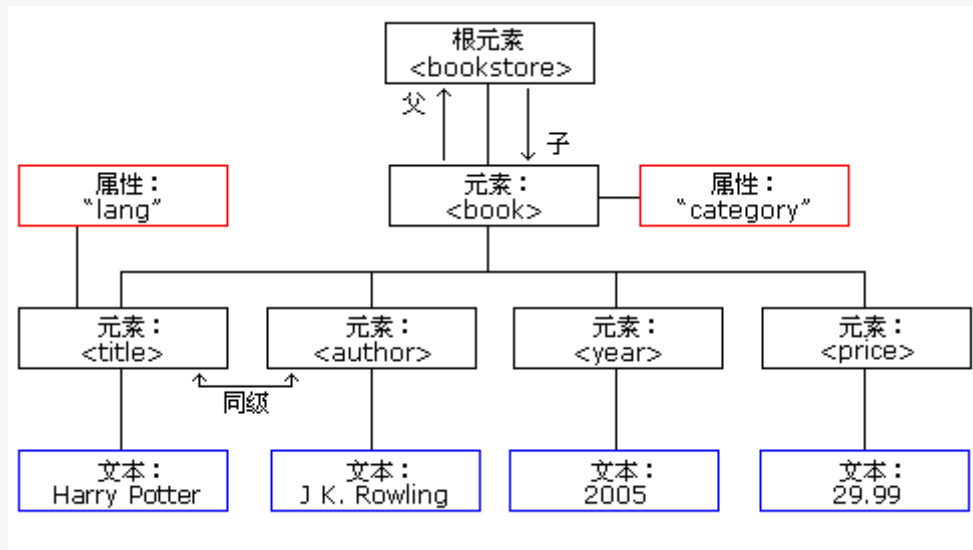
所有的元素都可以有子元素：

```
<root>
  <child>
    <subchild>.....</subchild>
```

```
</child>
</root>
```

父、子以及同胞等术语用于描述元素之间的关系。父元素拥有子元素。相同层级上的子元素成为同胞（兄弟或姐妹）。所有的元素都可以有文本内容和属性（类似 HTML 中）。

## 实例：



上图表示下面的 XML 中的一本书：

### XML 文档实例

```
<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>
```

实例中的根元素是 <bookstore>。文档中的所有 <book> 元素都被包含在 <bookstore> 中。

<book> 元素有 4 个子元素：<title>、<author>、<year>、<price>。



## 1 篇笔记

写笔记



XML将数据组织成为一棵树,DOM 通过解析 XML 文档,为 XML 文档在逻辑上建立一个树模型,树的节点是一个个的对象。这样通过操作这棵树和这些对象就可以完成对 XML 文档的操作,为处理文档的所有方面提供了一个完美的概念性框架。

如下XML文档:

```
<line id="1"> the <bold>First</bold>line</line>
```

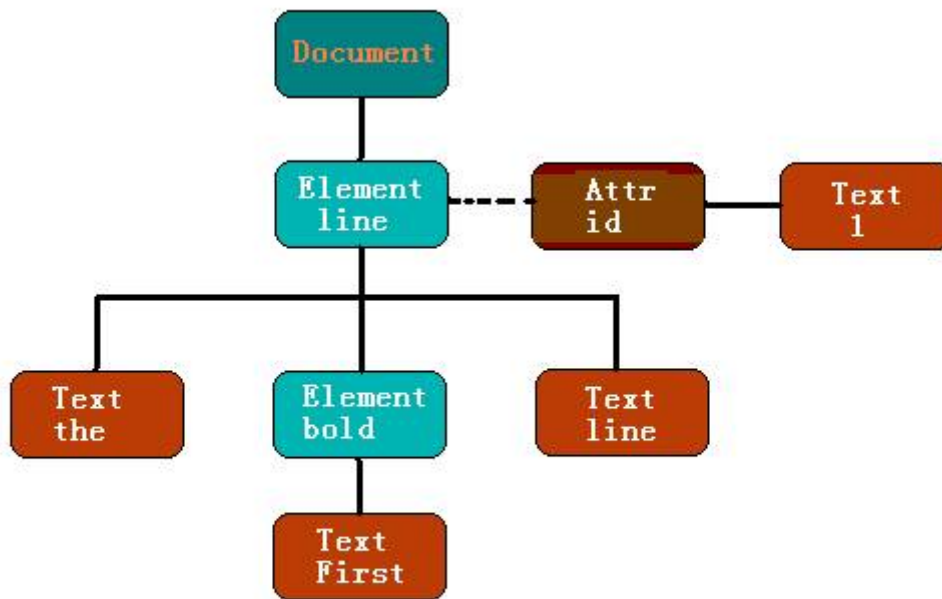


图1 XML 文档的 DOM 表示

由于DOM“一切都是节点 ( everything-is-a-node )”，XML树的每个 Document、Element、Text、Attr和Comment都是 DOM Node。

由上面例子可知，DOM 实质上是一些节点的集合。由于文档中可能包含有不同类型的信息，所以定义了几种不同类型的节点，如：Document、Element、Text、Attr、CDATASection、ProcessingInstruction、Notation、EntityReference、Entity、DocumentType、DocumentFragment 等。

Yvonne 1年前 (2018-03-14)