

概述

什么是XML?

XML 指可扩展标记语言（eXtensible Markup Language）。XML 被设计用来传输和存储数据。

XML是一套定义语义标记的规则，这些标记将文档分成许多部件并对这些部件加以标识。

它也是元标记语言，即定义了用于定义其他与特定领域有关的、语义的、结构化的标记语言的句法语言。

xml构成

XML由3个部分构成，它们分别是：

- 文档类型定义（Document Type Definition，DTD），即XML的布局语言
- 可扩展的样式语言（Extensible Style Language，XSL），即XML的样式表语言
- 可扩展链接语言（Extensible Link Language，XLL）

Python解析xml的方法

常见的XML编程接口有DOM和SAX，这两种接口处理XML文件的方式不同，当然使用场合也不同。

python有三种方法解析XML，SAX，DOM，以及ElementTree:

1. SAX (simple API for XML)

python 标准库包含SAX解析器，SAX用事件驱动模型，通过在解析XML的过程中触发一个个的事件并调用用户定义的回调函数来处理XML文件。

2. DOM(Document Object Model)

将XML数据在内存中解析成一个树，通过对树的操作来操作XML。

3. ElementTree(元素树)

ElementTree就像一个轻量级的DOM，具有方便友好的API。代码可用性好，速度快，消耗内存少。

本文只介绍ElementTree方式解析xml。

一个基本xml片段

下面我们尝试解析下面这一段xml:

```
<?xml version="1.0"?>
<data>
  <country name="Liechtenstein">
    <rank>1</rank>
    <year>2008</year>
    <gdppc>141100</gdppc>
    <neighbor name="Austria" direction="E"/>
    <neighbor name="Switzerland" direction="W"/>
  </country>
</data>
```

```

</country>
<country name="Singapore">
  <rank>4</rank>
  <year>2011</year>
  <gdppc>59900</gdppc>
  <neighbor name="Malaysia" direction="N"/>
</country>
<country name="Panama">
  <rank>68</rank>
  <year>2011</year>
  <gdppc>13600</gdppc>
  <neighbor name="Costa Rica" direction="W"/>
  <neighbor name="Colombia" direction="E"/>
</country>
</data>

```

或是将上面这段xml内容保存至xml_data.xml中。

代码示例

下面通过一段完整的代码演示如何读取、修改和保存xml。

```

#-*- coding:utf-8 -*-

__author__ = "苦叶子"

# 导入ElementTree
import xml.etree.ElementTree as ET

if __name__ == "__main__":
    print("python xml解析实例")

    data = """<data>
<country name="Liechtenstein">
  <rank>1</rank>
  <year>2008</year>
  <gdppc>141100</gdppc>
  <neighbor name="Austria" direction="E"/>
  <neighbor name="Switzerland" direction="W"/>
</country>
<country name="Singapore">
  <rank>4</rank>
  <year>2011</year>
  <gdppc>59900</gdppc>
  <neighbor name="Malaysia" direction="N"/>
</country>
<country name="Panama">
  <rank>68</rank>
  <year>2011</year>
  <gdppc>13600</gdppc>

```

```

        <neighbor name="Costa Rica" direction="W"/>
        <neighbor name="Colombia" direction="E"/>
    </country>
</data>
"""
# 载入xml的两种方式，一种从文件，一种从xml字符串
# 注意区别：从xml字符串加载的xml直接返回root元素对象
# 而从文件加载xml返回是xml树

# 大家根据实际情况来决定用哪种方式即可
# 本示例从xml字符串载入进行演示

# 从文件加载xml，获取xml tree节点
# tree = ET.parse('xml_data.xml')

# 获取根节点
# root = tree.getroot()

# 从字符串加载xml
root = ET.fromstring(data)

# 打印下根节点的节点tag， 输出data
print(root.tag)

# 遍历下根节点的所有子节点及其属性
print("---" * 10)
for child in root:
    print(child.tag, " ", child.attrib)

# 找所有的year节点玩下
print("---" * 10)
for child in root.iter("year"):
    # 打印出year节点的tag和text
    print(child.tag, " ", child.text)

# 修改下节点的text试试， 把year节点所有2011修改为2017
print("---" * 10)
for child in root.iter("year"):
    if child.text == "2011":
        child.text = "2017"
        child.set('updated', 'yes')

# 打印下修改后的xml所有的year节点
print("将2011 -> 2017")
for child in root.iter("year"):
    # 打印出year节点的tag和text
    print(child.tag, " ", child.text)

# 给每个country节点新增一个<wx>开源优测</wx>的节点试试
print("---" * 10)
for child in root.iter("country"):
    wx = ET.SubElement(child, "wx")

```

```

wx.text = "开源优测"

# 遍历wx节点，并打印
for child in root.iter("wx"):
    print(child.tag, " ", child.text)

# 下面演示删除所有的neighbor节点
# 当然你自己可以加判断条件删除指定的节点，自行尝试吧
print("---" * 10)
for child in root.findall("neighbor"):
    root.remove(child)

# 保存上述操作后的xml至xml_write_data.xml
xml_update_data = ET.tostring(root, encoding="unicode")

# 写入xml_write_data.xml
import codecs
fp = codecs.open("xml_write_data.xml", "w", "utf-8")

fp.write(xml_update_data)

fp.close()

```

经过上述一系列代码操作后保存至xml_write_data.xml中的xml内容如下:

```

<data>
  <country name="Liechtenstein">
    <rank>1</rank>
    <year>2008</year>
    <gdppc>141100</gdppc>
    <neighbor direction="E" name="Austria" />
    <neighbor direction="W" name="Switzerland" />
  <wx>开源优测</wx></country>
  <country name="Singapore">
    <rank>4</rank>
    <year updated="yes">2017</year>
    <gdppc>59900</gdppc>
    <neighbor direction="N" name="Malaysia" />
  <wx>开源优测</wx></country>
  <country name="Panama">
    <rank>68</rank>
    <year updated="yes">2017</year>
    <gdppc>13600</gdppc>
    <neighbor direction="W" name="Costa Rica" />
    <neighbor direction="E" name="Colombia" />
  <wx>开源优测</wx></country>
</data>

```

小结

本文所述仅仅是ElementTree的极小部分功能，更多的功能请参见官方文档学习。

扫一扫关注微信公众号：

