用TinyXML解析xml文件

1、xml文件内容如下

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE Library SYSTEM "Library.dtd">

<Library>

<Book BookId="978-7-5057-3323-7">

<Title>假如给我三天光明</Title>

<Author>[美] 海伦·凯勒</Author>

<Price>20.00</Price>

<Publicsher>中国友谊出版公司</Publicsher>

<Cost>20.00</Cost>

</Book>

<Member>

<Name>

<FirstName>Xiaohua</FirstName>

<MiddleName>Mary</MiddleName>

<LastName>Li</LastName>

</Name>

<Address>

<HouseNumber>221 B</HouseNumber>

<Street>Baker</Street>

<City>London</City>

</Address>

</Member>

</Library>

2、外部DTD文件内容如下

<?xml version="1.0" encoding="UTF-8"?>

<!ELEMENT Library (Book,Member)+>

<!ELEMENT Book (BookId,Title,Author,Price,Publicsher,Cost)>

<!ELEMENT BookId (#PCDATA)>

<!ELEMENT Title (#PCDATA)>

<!ELEMENT Author (#PCDATA)>

<!ELEMENT Price (#PCDATA)>

<!ELEMENT Publicsher (#PCDATA)>

<!ELEMENT Cost (#PCDATA)>

<!ELEMENT Member (Name,Address)+>

<!ELEMENT Name (FirstName,MiddleName,LastName)>

<!ELEMENT FirstName (#PCDATA)>

<!ELEMENT MiddleName (#PCDATA)>

<!ELEMENT LastName (#PCDATA)>

<!ELEMENT Address (HouseNumber,Street,City)>

<!ELEMENT HouseNumber (#PCDATA)>

<!ELEMENT Street (#PCDATA)>

<!ELEMENT City (#PCDATA)>

3、主要解析代码如下

#include "stdafx.h"

#include <iostream>

#include "tinyxml.h"

#include <tinystr.h>

#include <string.h>

#include <windows.h>

#include <atlstr.h>

using namespace std;

CString GetAppPath()

{//获取应用程序根目录

TCHAR modulePath[MAX\_PATH];

GetModuleFileName(NULL, modulePath, MAX\_PATH);

CString strModulePath(modulePath);

strModulePath = strModulePath.Left(strModulePath.ReverseFind(\_T('\\')));

return strModulePath;

}

bool CreateXmlFile(string& szFileName)

{//创建xml文件,szFilePath为文件保存的路径,若创建成功返回true,否则false

try

{

//创建一个XML的文档对象。

TiXmlDocument \*myDocument = new TiXmlDocument();

//创建一个根元素Library并连接。

TiXmlElement \*RootElement = new TiXmlElement("Library");

myDocument->LinkEndChild(RootElement);

//创建一个Book元素并连接。

TiXmlElement \*LibraryElement = new TiXmlElement("Book");

RootElement->LinkEndChild(LibraryElement);

//设置Book元素的属性。

LibraryElement->SetAttribute("BookId", "978-7-5057-3323-7");

//创建Title元素、Author元素、Author元素、Publicsher元素、Cost元素并连接。

TiXmlElement \*TitleElement = new TiXmlElement("Title");

TiXmlElement \*AuthorElement = new TiXmlElement("Author");

TiXmlElement \*PriceElement = new TiXmlElement("Price");

TiXmlElement \*PublicsherElement = new TiXmlElement("Publicsher");

TiXmlElement \*CostElement = new TiXmlElement("Cost");

LibraryElement->LinkEndChild(TitleElement);

LibraryElement->LinkEndChild(AuthorElement);

LibraryElement->LinkEndChild(PriceElement);

LibraryElement->LinkEndChild(PublicsherElement);

LibraryElement->LinkEndChild(CostElement);

//设置Title元素、Author元素、Price元素、Publicsher元素、Cost元素的内容并连接。

TiXmlText \*TitleContent = new TiXmlText("假如给我三天光明");

TiXmlText \*AuthorContent = new TiXmlText("[美] 海伦·凯勒");

TiXmlText \*PriceContent = new TiXmlText("20.00RMB");

TiXmlText \*PublicsherContent = new TiXmlText("中国友谊出版公司");

TiXmlText \*CostContent = new TiXmlText("18.00RMB");

TitleElement->LinkEndChild(TitleContent);

AuthorElement->LinkEndChild(AuthorContent);

PriceElement->LinkEndChild(PriceContent);

PublicsherElement->LinkEndChild(PublicsherContent);

CostElement->LinkEndChild(CostContent);

//创建一个Member元素并连接。

TiXmlElement \*Library1Element = new TiXmlElement("Member");

RootElement->LinkEndChild(Library1Element);

//创建Member元素的Name元素并连接。

TiXmlElement \*NameElement = new TiXmlElement("Name");

Library1Element->LinkEndChild(NameElement);

//创建FirstName元素、MiddleName元素、LastName元素并连接。

TiXmlElement \*FirstNameElement = new TiXmlElement("FirstName");

TiXmlElement \*MiddleNameElement = new TiXmlElement("MiddleName");

TiXmlElement \*LastNameElement = new TiXmlElement("LastName");

NameElement->LinkEndChild(FirstNameElement);

NameElement->LinkEndChild(MiddleNameElement);

NameElement->LinkEndChild(LastNameElement);

//设置FirstName元素、MiddleName元素、LastName元素的内容并连接。

TiXmlText \*FirstNameContent = new TiXmlText("Xiaohua");

TiXmlText \*MiddleNameContent = new TiXmlText("Mary");

TiXmlText \*LastNameContent = new TiXmlText("Li");

FirstNameElement->LinkEndChild(FirstNameContent);

MiddleNameElement->LinkEndChild(MiddleNameContent);

LastNameElement->LinkEndChild(LastNameContent);

//创建Member元素的Address元素并连接。

TiXmlElement \*AddressElement = new TiXmlElement("Address");

Library1Element->LinkEndChild(AddressElement);

//创建HouseNumber元素、Street元素、City元素并连接。

TiXmlElement \*HouseNumberElement = new TiXmlElement("HouseNumber");

TiXmlElement \*StreetElement = new TiXmlElement("Street");

TiXmlElement \*CityElement = new TiXmlElement("City");

AddressElement->LinkEndChild(HouseNumberElement);

AddressElement->LinkEndChild(StreetElement);

AddressElement->LinkEndChild(CityElement);

//设置HouseNumber元素、Street元素、City元素的内容并连接。

TiXmlText \*HouseNumberContent = new TiXmlText("221 B");

TiXmlText \*StreetContent = new TiXmlText("Baker");

TiXmlText \*CityContent = new TiXmlText("London");

HouseNumberElement->LinkEndChild(HouseNumberContent);

StreetElement->LinkEndChild(StreetContent);

CityElement->LinkEndChild(CityContent);

CString appPath = GetAppPath();

string seperator = "\\";

string fullPath = appPath.GetBuffer(0) + seperator + szFileName;

myDocument->SaveFile(fullPath.c\_str());//保存到文件

}

catch (string& e)

{

return false;

}

return true;

}

bool ReadXmlFile(string& szFileName)

{//读取Xml文件，并遍历

try

{

CString appPath = GetAppPath();

string seperator = "\\";

string fullPath = appPath.GetBuffer(0) + seperator + szFileName;

//创建一个XML的文档对象。

TiXmlDocument \*myDocument = new TiXmlDocument(fullPath.c\_str());

myDocument->LoadFile();

//获得根元素，即Library。

TiXmlElement \*RootElement = myDocument->RootElement();

//输出根元素名称，即输出Library

cout << RootElement->Value() << endl;

//获得Book节点。

TiXmlElement \*Book = RootElement->FirstChildElement();

//获得Book的ID、Title节点、Author节点、Price节点、Public节点、Cost节点属性。

TiXmlElement \*TitleElement = Book->FirstChildElement();

TiXmlElement \*AuthorElement = TitleElement->NextSiblingElement();

TiXmlElement \*PriceElement = AuthorElement->NextSiblingElement();

TiXmlElement \*PublicElement = PriceElement->NextSiblingElement();

TiXmlElement \*CostElement = PublicElement->NextSiblingElement();

TiXmlAttribute \*BookIdAttribute = Book->FirstAttribute();

cout << BookIdAttribute->Value() << endl;

cout << TitleElement->FirstChild()->Value() << endl;

cout << AuthorElement->FirstChild()->Value() << endl;

cout << PriceElement->FirstChild()->Value() << endl;

cout << PublicElement->FirstChild()->Value() << endl;

cout << CostElement->FirstChild()->Value() << endl;

//获得Member节点。

TiXmlElement \*Member = Book->NextSiblingElement();

//获得Member的Name,Address节点。

TiXmlElement \*Name1Element = Member->FirstChildElement();

TiXmlElement \*Address = Name1Element->NextSiblingElement();

//获得Name的FirstName节点、MiddleName节点、LastName节点的属性。

TiXmlElement \*FirstNameElement = Name1Element->FirstChildElement();

TiXmlElement \*MiddleNameElement = FirstNameElement->NextSiblingElement();

TiXmlElement \*LastNameElement = MiddleNameElement->NextSiblingElement();

cout << FirstNameElement->FirstChild()->Value() << endl;

cout << MiddleNameElement->FirstChild()->Value() << endl;

cout << LastNameElement->FirstChild()->Value() << endl;

//获得Address的HouseNumber节点、Street节点、City节点的属性。

TiXmlElement \*HouseNumberElement = Address->FirstChildElement();

TiXmlElement \*StreetElement = HouseNumberElement->NextSiblingElement();

TiXmlElement \*CityElement = StreetElement->NextSiblingElement();

cout << HouseNumberElement->FirstChild()->Value() << endl;

cout << StreetElement->FirstChild()->Value() << endl;

cout << CityElement->FirstChild()->Value() << endl;

}

catch (string& e)

{

return false;

}

return true;

}

int main()

{

string fileName = "info.xml";

CreateXmlFile(fileName);

ReadXmlFile(fileName);

getchar();

return 0;

}

4、运行结果截图如下：

