Parsing XML using TinyXML

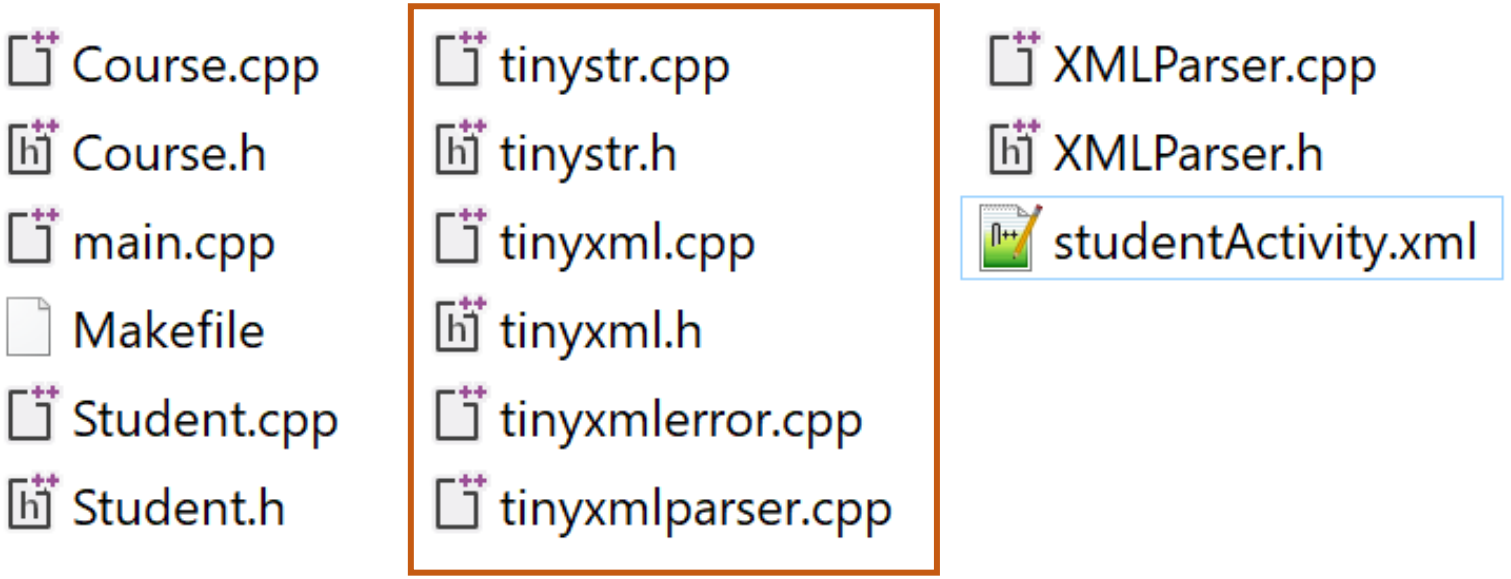
ECE 39595C Object Oriented Programming with C++

September 21st, 2021

Links

- Website:
  - http://www.grinninglizard.com/tinyxml/
- Download:
  - https://sourceforge.net/projects/tinyxml/files/latest/download
- Documentation:
  - http://www.grinninglizard.com/tinyxmldocs/index.html
- Example XML parsing project on Brightspace

Setting up the project



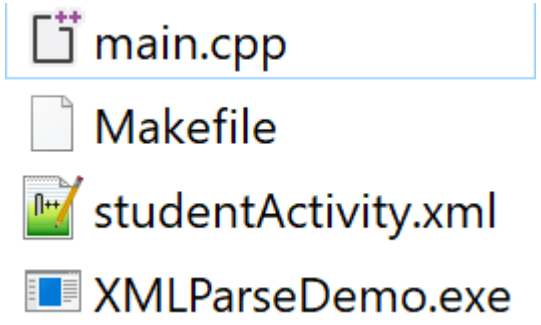
Setting up the Makefile

```
#define TIXML_USE_STL
```

```
EXECUTABLE=XMLParseDemo
CC=g++
RM=rm -f
CFLAGS=-g -std=c++11 -Wall -Werror -DTIXML_USE_STL
OBJECTS=main.o Activity.o Course.o Club.o Student.o XMLParser.o \
    tinystr.o tinyxml.o tinyxmlerror.o tinyxmlparser.o

run: $(EXECUTABLE)
    ./$(EXECUTABLE)

$(EXECUTABLE): $(OBJECTS)
    $(CC) -o $(EXECUTABLE) $(OBJECTS)

.cpp.o: $(HEADERS)
    $(CC) $(CFLAGS) -c $<

clean:
    $(RM) *.o
    $(RM) $(EXECUTABLE)

.PHONY: run clean
```

Reading the file

```xml
<?xml version="1.0" encoding="UTF-8"?>
<Students count="3">
    <Student>...</Student>
</Students>
```




```
#include "tinyxml.h"

void someFunction() {
    std::string filename = "./studentActivity.xml";
    TiXmlDocument doc(filename);
    doc.LoadFile();

    // fetch the root element
    TiXmlElement* rootElement = doc.RootElement();
    if (rootElement != NULL && rootElement->ValueStr() == "Students") {
        // code that parses the XML file
    }
}
```

Iterate Elements

```
<Students count="3">
    <Student name="Student1">...</Student>
    <Student name="Student2">...</Student>
    <Student name="Student3">...</Student>
</Students>
```

```
TiXmlElement* element = ...;
for (TiXmlNode* node = element->FirstChild(); node != NULL;
        node = node->NextSibling()) {
    TiXmlElement* childElement = node->ToElement();
    if (childElement != NULL) {
        // process the element
    }
}
```

Parsing Simple Objects

```
<Course>
    <instructor>Prof Midkiff</instructor>
    <name>Object Oriented Programming</name>
    <credit>3</credit>
    <location>ME 1061</location>
</Course>
```

```
Course* parseCourse(TiXmlElement* element) {
    Course* course = new Course();
    for (TiXmlNode* node = element->FirstChild(); node != NULL;
            node = node->NextSibling()) {
        TiXmlElement* childElement = node->ToElement();
        if (childElement != NULL) {
            std::string name = childElement->ValueStr();
            std::string value = childElement->GetText();
            if (name == "instructor")
                course->setInstructor(value);
            else if (name == "credit")
                course->setCredit(std::stoi(value));
            else if (name == "name")
                course->setName(value);
            else if (name == "location")
                course->setLocation(value);
        }
    }
    return course;
}
```

Parsing Nested Objects

```
<Student name="Purdue Student">
    <Course>
        ...
    </Course>
    <Address>
        ...
    </Address>
    <MealPlan>
        ...
    </MealPlan>
</Student>
```

```
Student* parseStudent(TiXmlElement* element) {
    Student* student = new Student();
    for (TiXmlNode* node = element->FirstChild(); node != NULL;
            node = node->NextSibling()) {
        TiXmlElement* childElement = node->ToElement();
        if (childElement != NULL) {
            std::string name = childElement->ValueStr();
            if (name == "Course")
                student->addCourse(parseCourse(childElement));
            else if (name == "Address")
                student->addAddress(parseAddress(childElement));
            else if (name == "MealPlan")
                student->setMealPlan(parseMealPlan(childElement));
        }
    }
    return student;
}
```

Questions?