

第十组 assignment3

在studentlist.xsd中加入了maxOccurs="unbounded" 以支持多个学生

程序中为了展现附加题的应用，采用先解析xml文档1，然后根据需要处理文档结构后，再生成其它同学的信息。

代码见DomParser.java文件，结果见xml文档2，

Student.java 为学生信息类

以下为代码中开启空白字符和验证的开关。

```
//设置是否跳过空白字符
domParser.setIgnoreWhiteSpace(true);
//
domParser.setIgnoreWhiteSpace(false);

//是否启用验证
Document doc= domParser.parseByValidate(path,xsdPath);
//
Document doc= domParser.parseByNotValidate(path);
```

分别通过调用下图1，2中的代码实现

```
public Document parseByValidate(String xmlPath, String xsdPath){
    File file=new File(xmlPath);
    File xsdFile=new File(xsdPath);

    //启用命名空间插件
    docBuilderFactory.setNamespaceAware(true);
    //启用include功能
    docBuilderFactory.setXIncludeAware(true);
    docBuilderFactory.setIgnoringElementContentWhitespace(ignoreWhiteSpace);

    SchemaFactory schemaFactory =SchemaFactory.newInstance(XMLConstants.W3C_XML_SCHEMA_NS_URI);
    try {
        Schema schema =schemaFactory.newSchema(xsdFile);
        //载入Schema进行验证
        docBuilderFactory.setSchema(schema);

        DocumentBuilder docBuilder = docBuilderFactory.newDocumentBuilder();
        Document doc = docBuilder.parse(file);
```

然后通过DOM操作文档生成要求的信息，同时包括命名空间的写入通过

```
Element root= (Element)
doc.createElementNS("http://www.nju.edu.cn/schema","nju:studentList");
    root.setAttribute("xmlns","http://jw.nju.edu.cn/schema");
    root.setAttribute("xmlns:xsi","http://www.w3.org/2001/XMLSchema-
instance");
    root.setAttribute("xsi:schemaLocation","http://www.nju.edu.cn/schema
StudentList.xsd " +
        " http://jw.nju.edu.cn/schema Student.xsd");
```

