



LANZHOU UNIVERSITY OF TECHNOLOGY

结课作业

题 目 XML综述

学生姓名 李广清

学 号 1817700424

专业班级 软件四班

学 院 软件学院

XML综述

一. XML eXtensible Markup Language （可扩展的标记语言） 。

1. XML描述的是文档的内容与语义，而不是文档应当如何显示。

2. 使用 XML可以实现不同语言，平台之间的交互。

3. 每一个 XML 文档都有且只有一个根元素。所谓根元素，就是唯一包含了其他所有元素的元素。

4. XML 中的每个元素都是成对出现的（有开始，有结束），<student> </student>，XML 中的元素嵌套关系要保持正确性，即先开始的标记要先 结束，后开始的标记要后结束。

5. <??>处理指令必须要顶格写，前面不能有任何的空白。

6. XML 元素可以具有属性，属性的形式为：属性名 =‘’属性值‘’ . 多个属性之 间使用空格分开。在一个元素上，相同的属性只能出现一次。

7. XML 的注释： <! --注释内容 --> ，注释不允许嵌套

8. <![CDATA[ CDATA节内容 ]]

二. DTD Document Type Definition （文档类型定义）

1. 格式正规的 XML 文档。符合 XML 语法要求的 XML 文档就是格式正规 的 XML文档。

2. 有效的 XML 文档。首先 XML 文档是个格式正规的 XML 文档，然后又 需要满足 DTD 的要求，这样的 XML文档称为有效的 XML文档 。

3. 创建元素 <!ELEMETN 元素名 数据类型 > 数据类型 :#PCDATA 元素集合 EMPTY ANY 。

4. #PCDATA （Parsed Character Data ） ，可解析的字符数据。

5. 导入外部 DTD文档 : <!DOCTYPE 根元素名 SYSTEM “外部 DTD文件名”

6. 运算符 :

, 顺序

() 一组

| 多选 一

+ 1 个或多个

\* 0 个或多个

? 0 个或 1 个

7. 创建属性 <!ATTLIST 元素名 属性名 类型 属性的属性 >

类型:CDATA ID Enumeration NMTOKEN IDREF

属性的属性 : #REQUIRED #IMPLIED #FIXED “default”

8. schema：其作用与 DTD 一样，也是用于验证 XML 文档的有效性

9. schema也是一个 XML文件，而 DTD有自己的语法。

10. 所有的 schema 文档，其根元素必须为 schema。

11. 定义元素 : 只能包含文本内容， 不能够包含子元素， 也没有属性的元素

<xs:element name=”” type=”” />

12. 定义属性所有的元素属性均被声明为简单类型。只有复杂类型的元素才 可以拥有属性。

<xs:attribute name=" " type=" "/> 所有的属性默认都是可选的，我们可以 通过使用 use 关键字明确的指出是可选 或是必需：

"optional"

"required"

我们可以通过使用 default 或 fixed 为简单类型（简单元素、 属性）指定默认值或固定值

<xs:element name=" " type=" " default=" 这个是默认值 "/>

13. 复杂类型 :复杂类型指包含其他元素 /属性的元素类型。

<xs:complexType>  
 <xs:sequence>  
 <xs:element name="dog" type="xs:string"/>  
 <xs:element name="cat" type="xs:string"/>  
 <xs:element name="snake" type="xs:string"/>  
 </xs:sequence>  
 </xs:complexType>

14. complexType三种子元素

<xs:sequence> 必需都有 ,按顺序

<xs:choice> 多选一 ,只有一个元素

<xs:all> 都有

15. 自定义简单类型 :

<xs:simpleType>

<xs:sequence>

<xs:list>

<xs:list/>

</xs:sequence>

</xs: simpleType >

16. 简单类型的三种子元素 :

<xs:restriction/> 条件型

<xs:list/> 列表型

<xs:union/> 联合

17. group：包括 group 和 attributeGroup，用来定义一组相关的元素。 <xs:group name="grp1">

<xs:sequence>

<xs:element name="a" type="xs:string"/>

<xs:element name="b" type="xs:float"/>

<xs:element name="c" type="xs:int"/>

</xs:sequence>

</xs:group

<xs:attributeGroup name="grp1">

<xs:attribute name="a" type="xs:string"/>

<xs:attribute name="b" type="xs:string"/>

<xs:attribute name="c" type="xs:string "/>

</xs:attributeGroup>