

Id:									Family:		Given:
ITB295/ITN295 – XML									Tutor:	Timeslot:	

Review Questions on Document Type Definition

<http://www.w3.org/TR/REC-xml/>

Q1. Consider the following XML document:

```

1. <!DOCTYPE sql [
2. <!ELEMENT sql (select, from)>
3. <!ELEMENT select (col+)>
4. <!ATTLIST select order CDATA #REQUIRED>
5. <!ELEMENT col (#PCDATA)>
6. <!ELEMENT from (table+)>
7. <!ELEMENT table (#PCDATA)>]>
8. <sql>
9.   <select order="cost">
10.    <col>CarNr</col>
11.    <col>Make</col>
12.    <col>Cost</col>
13.  </select>
14.  <from>
15.    <table>Cars</table>
16.  </from>
17. </sql>

```

- a. The document is both well-formed and valid.
- b. The document is well-formed but not valid.
- c. The document is neither well-formed nor valid.

Hint: See section 1.2 for a discussion of well-formedness and validity constraints. See section 2.1 for the definition of a well-formed XML document. In terms of production [1], are the “prolog” and “element” correct? You will then need to follow up the consequent production rules.

Q2. Consider the following XML:

```

1. <!DOCTYPE Book [
2.   <!ELEMENT Book (TOC, Chapter+, Index)>
3.   <!ELEMENT TOC (#PCDATA)>
4.   <!ELEMENT Chapter (#PCDATA)>
5.   <!ELEMENT Index (#PCDATA) >
6.   <!ENTITY copy "Copyright Adam Smith">
7. ]>
8. <Book><TOC>Herein &copy;</TOC>
9.   <Chapter>The first</Chapter>
10.  <Index>Copyright Adam Smith</Index>
11. </Book>

```

- a. The document is both well-formed and valid.
- b. The document is well-formed but not valid.
- c. The document is neither well-formed nor valid.

Hint: See XML4e, production [1] and its family.

Q3. The following is a valid DTD:

```
1. <!DOCTYPE test>
```

- a. True
- b. False

Hint: See XML4e, production [28].

Q4. Given the following portions of an XML document:

```

1. <!ELEMENT Product>
2. <!ATTLIST Product Code ID #REQUIRED>
1. Product Code="123">Chair</Product>
2. Product Code="124">Sofa</Product>

```

The use of the Code attribute is valid.

- a. True
- b. False

False – Does not start with a letter.

Hint: See XML4e Validity Constraint: ID.

Q5. Consider the following fragment of XML:

```
1. <Name Sex="f" >Sue</Name>
```

This is invalid because there must be no white space after the attribute value.

- a. True
- b. False

Hint: See XML4e, production [40].

Q6. Consider the following fragment of XML:

```
1. <Person Name="David" Name="Dave"/>
```

- a. This is valid because the XML processor will take the last (second, in this case) value assigned to the Name attribute
- b. This is valid because it will take the first and ignore any subsequent appearances of the attribute.
- c. This is invalid because the same attribute may not appear more than once.

Hint: See XML4e, Well-formedness Constraint: Unique Att Spec

Q7. Given the following external entity declaration:

1. `<!ENTITY fit SYSTEM "fit.txt">`

where `fit.txt` is a simple text file containing the characters “Information Technology”. Consider the use of this entity in the following XML fragment:

1. `<faculty name="&fit;">`

This is valid XML.

a. True

b. False

Hint: See XML4e, section 3.1, Well-formedness Constraint: No External Entity Reference.

Q8. Which of the following are valid examples of *mixed* element content specifications.

☒ `(#PCDATA)`

☒ `(#PCDATA|em|br|p)`

☒ `(#PCDATA|em|br)*`

☐ `(student|#PCDATA|mark|unit)*`

☐ `(#PCDATA|student|#PCDATA|mark)*`

Hint: See XML4e, section 3.2.2.

Q9. The same (child element) name must not appear more than once in a “mixed” specification. This prevents an element from appearing twice in any usage of the element being specified.

a. True

b. False

Hint: See XML4e, section 3.2.2: in particular, what does the “*” signify?

Q10. Consider the following XML fragment:

1. `<Name age="25" height="170">Sue</Name>`

To define the attributes for this element, we could write:

1. `<!ELEMENT Name`

2. `<!ATTRIBUTE age CDATA #REQUIRED>`

3. `<!ATTRIBUTE height CDATA #REQUIRED>`

4. `>`

a. True

b. False

Hint: See XML4e, production [29], and follow the chain of rules.

Q11. An element must be introduced and defined *before* any attributes can be attached.

a. True

b. False

Hint: See XML4e, production [28b].

Q12. The three types of attribute are:

StringType

TokenizedType

EnumeratedType

Hint: See XML4e, production [54].

Q13. When we want to specify that an attribute can take on any string of characters, we can use the following form(s)?

☐ `#PCDATA`

☐ `PCDATA`

☒ `CDATA`

☐ `#CDATA`

Hint: See XML4e, productions [54] and [55].

Q14. Suppose we have the following tables defined in SQL:

1. `create table products`
2. `(pcode char[4] not null,`
3. `qty integer not null,`
4. `primary key pcode)`
- 5.
6. `create table orders`
7. `(ocode char[6] not null,`
8. `opcode char[4] not null,`
9. `oqty integer not null,`
10. `primary key ocode,`
11. `foreign key opcode references products(pcode))`

A corresponding XML definition could be:

1. `<!ELEMENT product (EMPTY)>`
2. `<!ELEMENT products (product)*>`
3. `<!ATTLIST product pcode ID #REQUIRED`
4. `qty CDATA #REQUIRED>`
- 5.
6. `<!ELEMENT orders (order)*>`
7. `<!ELEMENT order (EMPTY)>`
8. `<!ATTLIST order ocode ID #REQUIRED`
9. `opcode IDREF #REQUIRED`
10. `oqty CDATA #REQUIRED>`
- 11.

a. True

b. False

Hint:

Q15. Which of the following are allowable values for an attribute of type ID?

- ☒ `_250`
- ☐ `25_0`
- ☐ `250`
- ☒ `BHP`
- ☐ `anz:com`
- ☐ `xsd:element`

Hint: See XML4e, section 3.3.1, Validity Constraint: ID, and production [5].

Q16. An element `abc` has an attribute `stations` of type IDREFS. The following usage is legitimate.

1. `<abc stations="612 4QR 692 4RN">`

- a. True
- b. False

Hint: See XML4e, section 3.3.1, Validity Constraint: IDREF.

Q17. Suppose there is a one-to-one correspondence between the call signs and the frequencies of a set of radio stations. The following declaration is valid.

1. `<!ATTLIST Station CallSign ID #REQUIRED`
 2. `Frequency ID #REQUIRED>`

- a. True
- b. False

Hint: See XML4e, section 3.3.1, Validity Constraint: One ID per Element Type.

Q18. The following declaration is valid for an enumerated attribute.

1. `<!ATTLIST Station CallSign`
 2. `(4QR|4ZZZ|3AW|5CL)`
 3. `#REQUIRED>`

- a. True
- b. False

Hint: See XML4e, production [59].

Q19. Consider the following declaration:

1. `<!ATTLIST Person Extension CDATA #FIXED "4">`

Therefore all usages of this attribute should be fixed four characters long.

a. True

b. False

Hint: See XML4e, section 3.3.2, Validity constraint: Fixed Attribute Default.

Q20. Consider the following sequence found in an external entity:

1. `
He&xA;l&xD;&xD;p`

The XML processor will pass which of the following character strings to an application?

- a. `Help`
- b. `&xA;&xA;He&xA;l&xA;p`
- c. `&xA;He&xA;lp`
- d. `&xA;He&xA;l&xA;&xA;p`
- e. `&xA;He&xA;l&xA;p`

Hint: See XML4e, section 2.11.