

PROBE Klausur XML-Technologien, Sommersemester 2011		
Name	Matrikelnummer	
Vorname		
korrigiert von	Gesamtpunktzahl	von 30
	Note	

<p>Sitzordnung</p> <ul style="list-style-type: none"> Sitzt bequem... <p>Bearbeitungszeit</p> <ul style="list-style-type: none"> 30 Minuten – stellt euch einen Wecker! <p>Hilfsmittel</p> <ul style="list-style-type: none"> <u>keine</u> Hilfsmittel erlaubt. Kein Internetzugriff außer ihr wollt euch selbst betrügen! <p>Sprache</p> <ul style="list-style-type: none"> Antworten auf Deutsch oder Englisch erlaubt. 	Art der Aufgabe	Aufgabe	max. erreichbare Punkte	erzielte Punkte	
	<p style="text-align: center;">14 Aufgaben</p> <p style="text-align: center;">30 Punkte</p> <p style="text-align: center;">30 Minuten empfohlen</p>		1	1	
			2	3	
			3	1	
			4	3	
			5	1	
			6	1	
			7	1	
			8	1	
			9	1	
			10	1	
			11	5	
			12	8	
			13	1	
			14	2	
	Insgesamt	30			

Multiple Choice Aufgaben und freie Aufgaben

Anmerkung: Mehrfachauswahl möglich! Um ein Punkt pro Aufgabe zu bekommen, MÜSSEN alle korrekten Antworten in der Aufgabe angekreuzt sein.

mögliche Punkte	1	erreichte Punkte	
Aufgabe 1: Ist die Aussage korrekt? „Valide XML Dokumente sind immer wohlgeformt“			
<input type="checkbox"/>	nein		
<input checked="" type="checkbox"/>	ja		

mögliche Punkte	3	erreichte Punkte	
Aufgabe 2: Nennen Sie mindestens drei Regeln für Wohlgeformtheit von XML-Dokumenten			
<ul style="list-style-type: none">• Nur genau ein Wurzelement• Korrekt verschachtelt• Namenskonventionen eingehalten• Attribute kommen jeweils nur einmal vor• Attributwerte sind in Anführungszeichen			

mögliche Punkte	1	erreichte Punkte	
Aufgabe 3: Wir betrachten das folgende DTD-Element: <!ELEMENT Auto (Reifen+, Fenster*, Dach?, Fahrer) > Markieren Sie die richtigen Aussagen.			
<input checked="" type="checkbox"/>	Cabrios sind zulässig (Autos ohne Dach)		
<input type="checkbox"/>	Es muss keinen Fahrer geben		
<input type="checkbox"/>	Autos ohne Reifen sind zulässig		
<input checked="" type="checkbox"/>	Die Anzahl der Fenster ist unbeschränkt		

mögliche Punkte	4	erreichte Punkte	
Aufgabe 4: Markiere 4 Fehler in Bezug auf Wohlgeformtheit in diesem XML-Dokument: <?xml version="1.0" standalone="yes" encoding="UTF-8"?> <verzeichnis> <titel>Wikipedia Städteverzeichnis</titel> <eintrag> <stichwort>Genf</stichwort> <eintragstext>Genf ist der Sitz von ...</eintragstext> </eintrag> <eintrag> <stichwort umlaute="ja">Köln</stichwort> <eintragstext>Köln ist eine Stadt, die ...</eintragstext> </verzeichnis>			

mögliche Punkte	1	erreichte Punkte	
Aufgabe 5: Welche der folgenden XML-Elemente sind nicht zulässig?			
<input checked="" type="checkbox"/>	<xmltest> Test </xmltest>		
<input type="checkbox"/>	<first_name> John </first_name>		
<input checked="" type="checkbox"/>	<Datenbank Eintrag> John </ Datenbank Eintrag >		
<input type="checkbox"/>	<ελληνική> ἑλληνική γλώσσα </ελληνική>		

mögliche Punkte	1	erreichte Punkte	
Aufgabe 6: Was stimmt bei diesem XML-Dokument nicht?			
<pre> <book> <title>Mein Buch</title> </book> <book> <title>Anderes Buch</title> </book> </pre>			

mögliche Punkte	1	erreichte Punkte	
Aufgabe 7: Wir betrachten den folgenden Ausschnitt aus einem XML-Schema:			
<pre> <xsd:complexType name="BookType"> <xsd:all> <xsd:element name="Title" type="xsd:string"> <xsd:element name="Author" type="xsd:string"> <xsd:element name="Date" type="xsd:string"> <xsd:element name="ISBN" type="xsd:string"> <xsd:element name="Publisher" type="xsd:string"> </xsd:all> </xsd:complexType> </pre>			
Ist die Reihenfolge der Elemente in einer Instanz des Typs <i>BookType</i> beliebig oder vorgegeben?			
<input checked="" type="checkbox"/>	Beliebige Reihenfolge der Elemente möglich		
<input type="checkbox"/>	Reihenfolge ist fest (Title, Author, Date, ISBN, Publisher)		

mögliche Punkte	1	erreichte Punkte	
<p>Aufgabe 8: Werden Entities in dem Attribut „schwer“ des Elements testfrage durch den Parser verarbeitet oder nicht?</p> <pre><!ELEMENT Klausur (Testfrage)> <!ELEMENT Testfrage ANY> <!ATTLIST Testfrage Schwer CDATA #IMPLIED „nein“></pre>			
<p><input checked="" type="checkbox"/> Ja <input type="checkbox"/> Nein</p>			

mögliche Punkte	1	erreichte Punkte	
<p>Aufgabe 9: Markieren Sie jeden XPath-Ausdruck der alle schweren (schwer=“ja“ Attribut) Testfrage Elemente in einer Klausur ermittelt.</p> <pre><Klausur> <Testfrage schwer="nein">Frage 1</Testfrage> <Testfrage schwer="ja">Frage 2</Testfrage> <Testfrage schwer="ja">Frage 3</Testfrage> <Zusatzaufgaben> <Testfrage schwer="ja">Frage 4</Testfrage> </Zusatzaufgaben> </Klausur></pre>			
<p><input type="checkbox"/> /Klausur/Testfrage[schwer='ja'] <input type="checkbox"/> //Testfrage[schwer='ja'] <input checked="" type="checkbox"/> Testfrage[@schwer='ja'] <input checked="" type="checkbox"/> /Klausur//Testfrage [@schwer='ja']</p>			

mögliche Punkte	1	erreichte Punkte	
<p>Aufgabe 10: Unterstreichen Sie alle Attribute die sich im Namensraum http://www.test.de/ befinden.</p> <pre><Klausur xmlns="http://www.test.de/" xmlns:test="http://www.test.de/"> <Testfrage schwer="nein">Frage 1</Testfrage> <Testfrage schwer="ja">Frage 2</Testfrage> <Testfrage schwer="ja">Frage 3</Testfrage> <Zusatzaufgaben bonusPunkte="5" xmlns="http://www.test.de/"> <Testfrage test:schwer="ja">Frage 4</Testfrage> </Zusatzaufgaben> </Klausur></pre>			

mögliche Punkte	5	erreichte Punkte	
<p>Aufgabe 11: Wie ist die Ausgabe wenn Sie das XSLT-Stylesheet auf das angegebene XML-Dokument anwenden?</p> <pre> <?xml version="1.0" encoding="ISO-8859-1"?> <Klausur> <Hinweise> Bitte nicht schummeln! </Hinweise> <Testfrage schwer="nein">Frage 1</Testfrage> <Testfrage schwer="ja">Frage 2</Testfrage> <Testfrage schwer="ja">Frage 3</Testfrage> <Zusatzaufgaben> <Testfrage schwer="ja">Frage 4</Testfrage> </Zusatzaufgaben> </Klausur> </pre> <p>Stylesheet:</p> <pre> <?xml version="1.0" encoding="ISO-8859-1"?> <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"> <xsl:template match="/Klausur//Testfrage[@schwer='ja']"> Schwere Frage: <xsl:value-of select="." />
 </xsl:template> <xsl:template match="Testfrage"> Irgendeine Frage: <xsl:value-of select="." />
 </xsl:template> </xsl:stylesheet> </pre>			
<p>Bitte nicht schummeln! Irgendeine Frage: Frage 1 Schwere Frage: Frage 2 Schwere Frage: Frage 3 Schwere Frage: Frage 4</p>			

mögliche Punkte	1+1+3+3	erreichte Punkte	
-----------------	---------	------------------	--

Aufgabe 12: Schreiben Sie XML Schema Definitionen für die folgenden Typen:

a) Ein Element „geburtstag“ zum Speichern eines Datums

```
<xs:element name="geburtstag" type="xs:date"/>
```

b) Ein Element “farbe” zum Speichern von Zeichenketten mit Standard-Wert “rot”

```
<xs:element name="farbe" type="xs:string" default="rot"/>
```

c) Ein Element “alter” das Werte zwischen 0 und 120 enthält

```
<xs:element name="alter">
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="120"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

d) Ein Element “Auto” das als zulässige Werte nur „Audi“, „Golf“ und „BMW“ hat

```
<xs:element name="car">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Audi"/>
      <xs:enumeration value="Golf"/>
      <xs:enumeration value="BMW"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

mögliche Punkte	1	erreichte Punkte									
<p>Aufgabe 13: Markieren Sie alle Elemente „gender“ die valide sind (unter Berücksichtigung des angegebenen XML Schemas)</p> <pre> <xs:element name="gender"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="male female"/> </xs:restriction> </xs:simpleType> </xs:element> </pre>											
<table border="0"> <tr> <td><input type="checkbox"/></td> <td><gender>male female</gender></td> </tr> <tr> <td><input type="checkbox"/></td> <td><gender>xs:male</gender></td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td><gender>male</gender></td> </tr> <tr> <td><input type="checkbox"/></td> <td><gender>female</Gender></td> </tr> </table>				<input type="checkbox"/>	<gender>male female</gender>	<input type="checkbox"/>	<gender>xs:male</gender>	<input checked="" type="checkbox"/>	<gender>male</gender>	<input type="checkbox"/>	<gender>female</Gender>
<input type="checkbox"/>	<gender>male female</gender>										
<input type="checkbox"/>	<gender>xs:male</gender>										
<input checked="" type="checkbox"/>	<gender>male</gender>										
<input type="checkbox"/>	<gender>female</Gender>										

mögliche Punkte	2	erreichte Punkte	
<p>Aufgabe 14: Welcher Parser eignet sich besonders gut zur Verarbeitung von defektem HTML-Code und warum?</p>			
<p>SAX-Parser (1 P) => Kann mit Strukturfehlern umgehen (1 P)</p>			