

Aufgabe 5 (Namensräume)

Gegeben sei folgendes Fragment eines XML-Dokuments. Geben Sie für jedes Element an, zu welchem Namensraum es gehört.

```
01 <AD:Buch xmlns:AD="http://www.URL1.com" xmlns="http://www.URL2.com">
02   <Titel>Mein Titel</Titel>
03   <AD:Inhalt>
04     <Element1 xmlns="http://www.URL3.com">
05       <Element2 />
06     </Element1>
07     <AD:Titel>Anderer Titel</AD:Titel>
08     <Element1>
09       <Element2 />
10     </Element1>
11   </AD:Inhalt>
12 </AD:Buch>
```

01: <http://www.URL1.com> (wegen xmlns:AD)

02: <http://www.URL2.com> (Default-Namensraum)

03: <http://www.URL1.com> (wegen xmlns:AD)

04: <http://www.URL3.com> (geänderter Default-Namensraum)

05: <http://www.URL3.com> (geänderter Default-Namensraum)

06: <http://www.URL3.com> (geänderter Default-Namensraum)

07: <http://www.URL1.com> (wegen xmlns:AD)

08: <http://www.URL2.com> (Default-Namensraum)

09: <http://www.URL2.com> (Default-Namensraum)

10: <http://www.URL2.com> (Default-Namensraum)

11: <http://www.URL1.com> (wegen xmlns:AD)

12: <http://www.URL1.com> (wegen xmlns:AD)

Aufgabe 6 (XML-Struktur in Baumstruktur darstellen)

Aufgrund seines hierarchischen Aufbaus entspricht die logische Struktur jedes XML-Dokuments einem Baum.

Stellen Sie das XML-Dokument, das Sie als Lösung von Aufgabe 2 erstellt haben, als Baumstruktur dar, und zwar mit Knoten für Elemente, Attribute und Werte. Die Werte sind zum einen die Attributwerte, zum anderen die in Tags eingeschlossenen Daten.

Lösung aus 2:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes" ?>
<literatur>
  <inproceedings id="bruns93">
    <author>Bruns, Ralf</author>
    <title>Direct Chromosome Representation and Advanced Genetic Operators for
Production Scheduling</title>
    <bookTitle>Proceedings of the Fifth International Conference on Genetic
Algorithms</bookTitle>
    <year>1993</year>
    <editor>Forrest, S.</editor>
    <pages>231--237</pages>
    <publisher>Morgan Kaufmann Publishing</publisher>
    <address>USA</address>
  </inproceedings>
  <article id="sauer97">
    <author>Sauer, Jürgen</author>
    <author>Bruns, Ralf</author>
    <title>Knowledge-Based Scheduling Systems in Industry and Medicine</title>
    <journal>IEEE Expert</journal>
    <year>1997</year>
    <volume>12</volume>
    <number>1</number>
    <pages>24-31</pages>
    <note>Darstellung wissensbasierter Ablaufplanungsverfahren und -systeme
sowie deren Anwendungen für industrielle Maschinenbelegungsplanung und
medizinische Bettenbelegungsplanung.</note>
  </article>
</literatur>
```

literatur

inproceedings id="bruns93"
author
Bruns, Ralf
title
Direct Chromosome Representation and Advanced Genetic Operators for Production Scheduling
bookTitle
Proceedings of the Fifth International Conference on Genetic Algorithms
year
1993
editor
Forrest, S.
pages
231-237
publisher
Morgan Kaufmann Publishing
address
USA

article id="sauer97"
author
Sauer, Jürgen
Bruns, Ralf
title
Knowledge-Based Scheduling Systems in Industry and Medicine
journal
IEEE Expert
year
1997
volume
12
number
1
pages
24-31
note
Darstellung wissensbasierter Ablaufplanungsverfahren und -systeme sowie deren Anwendungen für industrielle Maschinenbelegungsplanung und medizinische Bettenbelegungsplanung

Aufgabe 7 (DTD in passendes XML Schema umschreiben)

1. Erstellen Sie ein äquivalentes XML Schema zu folgender DTD. Die DTD ist die Lösung von Aufgabe 1.

```
<!ELEMENT adressen (adresse)*>  
  
<!ELEMENT adresse (anrede?, name, (postfach | wohnanschrift), plzort)>  
  
<!ELEMENT anrede (#PCDATA)>  
<!ELEMENT name (#PCDATA)>  
<!ELEMENT postfach (#PCDATA)>  
<!ELEMENT wohnanschrift (#PCDATA)>  
<!ELEMENT plzort (#PCDATA)>
```

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<!-- Namensraumdeklaration fuer ein Schema -->  
<!-- kopiert vom Skript, Seite 222 -->
```

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
targetNamespace="http://www.inform.fh-hannover.de/~mustermann"
```

```

xmlns="http://www.inform.fh-hannover.de/~mustermann"
elementFormDefault="qualified">

<!-- URSPRUNGS-DTD: <!ELEMENT adressen (adresse)*> -->

<xsd:element name="adressen" type="adressenType" />

<xsd:complexType name="adressenType">
  <xsd:sequence>
    <xsd:element ref="adresse" minOccurs="0" maxOccurs="unbounded" />
  </xsd:sequence>
</xsd:complexType>

<!-- URSPRUNGS-DTD: <!ELEMENT adresse (anrede?, name, (postfach |
wohnanschrift), plzort)> -->

<xsd:element name="adresse" type="adressType" />
<xsd:complexType name="adressType">
  <xsd:sequence>
    <xsd:element ref="anrede" minOccurs="0" />
    <xsd:element ref="name" />
    <xsd:choice>
      <xsd:element ref="postfach" />
      <xsd:element ref="wohnanschrift" />
    </xsd:choice>
    <xsd:element ref="plzort" />
  </xsd:sequence>
</xsd:complexType>

<!-- URSPRUNGS-DTD: <!ELEMENT anrede (#PCDATA)> -->

<xsd:element name="anrede" type="xsd:string" />

<!-- URSPRUNGS-DTD: <!ELEMENT name (#PCDATA)> -->

<xsd:element name="name" type="xsd:string" />

<!-- URSPRUNGS-DTD: <!ELEMENT postfach (#PCDATA)> -->

<xsd:element name="postfach" type="xsd:string" />

<!-- URSPRUNGS-DTD: <!ELEMENT wohnanschrift (#PCDATA)> -->

<xsd:element name="wohnanschrift" type="xsd:string" />

<!-- URSPRUNGS-DTD: <!ELEMENT plzort (#PCDATA)> -->

<xsd:element name="plzort" type="xsd:string" />

```