Übung XML Design Patterns

Ausgangslage

XML Dokumente sollte für Maschinen gut lesbar und strukturiert aufgebaut werden. XML Design Patterns bieten hier eine Hilfe, um XML Dokumente zu strukturieren und lesbar zu machen.

book.xml

Das folgende XML Dokument könnte als Lösung der Übung Book definiert sein: <?xml version="1.0"
encoding="UTF-8"?>
<document>

<book>
 <title>XSLT</title>

<author>Michael Kay</author>

<publisher>Wrox Press Ltd.</publisher>

<isbn>1-861003-12-9</isbn>
 </book>

<book>
 <title>XML and Java</title>

<author>Hiroshi Maruyama</author>

<author>Kent Tamura</author>

<author>Naohiko Uramoto</author>

<publisher>Addison Wesley</publisher>

<isbn>0201485435</isbn>
 </book>

<book>
 <title>Java and XML</title>

<author>Brett McLaughlin</author>

<author>Mike Loukides</author>

<publisher>O'Reilly</publisher>

<isbn>0596000162</isbn>

</book>
</document>Das Dokument erfüllt die gesuchten Anforderungen, es kann aber noch flexibler aufgebaut werden.

Collection Design Pattern

Das Book Dokument speichert mehrere Bücher und bildet damit eine Menge (Collection): Die Knoten innerhalb einer Collection sollten immer vom gleichen Elementtyp sein. Das XML Design Pattern Collection beschreibt dies. <?xml version="1.0"
encoding="UTF-8"?>
<document>

<booklist>
 <book>

<title>XSLT</title>
 <authorlist>

<author>Michael Kay</author>

</authorlist>
 <publisher>Wrox Press Ltd.</publisher>

<isbn>1-861003-12-9</isbn>
 </book>

<book>
 <title>XML and Java</title>
 <authorlist>

<author>Hiroshi Maruyama</author>

<author>Kent Tamura</author>

<author>Naohiko Uramoto</author>

</authorlist>
 <publisher>Addison Wesley</publisher>

<isbn>0201485435</isbn>
 </book>

<book>
 <title>Java and XML</title>

<authorlist>
 <author>Brett McLaughlin</author>

<author>Mike Loukides</author>
 </authorlist>

<publisher>O'Reilly</publisher>

<isbn>0596000162</isbn>
 </book>

</booklist>
</document>Die booklist-Collection enthält ausschliesslich book-Element.

Head-Body Design Pattern

Oft möchten wir XML Dokumente auch mit Metadaten ergänzen wie z.B. Datum der Erstellung oder den Benutzer ablegen, welcher das Dokument erstellte. Hier ist das Head-Body Design Pattern die richtige Lösung.Das XML Design Pattern Head-Body beschreibt dies. <?xml version="1.0" encoding="UTF-8"?>
<document>
 <head>
 <meta name="creationTime" value="2022-06-06T10:01:44.156Z" />
 <!-- place more meta data here -->
 </head>
 <body>
 <booklist>
 <book>
 <title>XSLT</title>
 <authorlist>
 <author>Michael Kay</author>
 </authorlist>
 <publisher>Wrox Press Ltd.</publisher>
 <isbn>1-861003-12-9</isbn>
 </book>
 <book>
 <title>XML and Java</title>
 <authorlist>
 <author>Hiroshi Maruyama</author>
 <author>Kent Tamura</author>
 <author>Naohiko Uramoto</author>
 </authorlist>
 <publisher>Addison Wesley</publisher>
 <isbn>0201485435</isbn>
 </book>
 <book>
 <title>Java and XML</title>
 <authorlist>
 <author>Brett McLaughlin</author>
 <author>Mike Loukides</author>
 </authorlist>
 <publisher>O'Reilly</publisher>
 <isbn>0596000162</isbn>
 </book>
 </booklist>
 </body>
</document>Das Book Dokument mit dem Head-Body Pattern.

Kontakt

Simtech AG
Finkenweg 23
3110 Münsingen
Schweiz

Impressum

Das Copyright für sämtliche Inhalte dieser Website liegt bei Simtech AG, Schweiz.
Beachten Sie auch unsere Hinweise zum Urheberrecht, Datenschutz und Haftungsauschluss.
Jeder Hinweis auf Fehler nehmen wir gerne entgegen.

Copyright

2024 Simtech AG, All rights reserved, Powered by stack.ch written in Golang by Daniel Schmutz

https://www.simtech-ag.ch/programmierer-xslt-schulung