DOM, SAX и StAX

Цели на упражнението:

- Използване на основни DOM, SAX и StAX интерфейси

Среди поддържащи DOM, SAX и StAX XML парсъри:

За реализация на това упражнение може да използвате някоя от следните Интернет базирани среди, поддържащи компилатори за Java, C#, JavaScript и др.:

repl.it (за предпочитане, понеже позволява създаването на началния XML документ)
ideone (началният XML документ трябва да се въведе като стринг)

Задача 1: Като използвате SAX, принтирайте дадения по-долу XML документ в стандартната конзола, спазвайки следните условия:

1. Йерархията на елементите да бъде запазена
2. Всеки елемент да бъде разпечатан на нов ред заедно с включените в него атрибути
3. Текстовите стойности на елементите да бъдат разпечатани с главни букви

XML документ:

```
<?xml version="1.0" encoding="UTF-8"?>
<rss version="0.91">
  <channel location="remote" company="N.A.">
     <title>Linux Today</title>
     <link>http://linuxtoday.com</link>
     <language>en-us</language>
     <description>Linux Today News Service</description>
     <image author="anonymous">
        <title>Linux Today</title>
        <url>http://linuxtoday.com/pics/ltnet.png</url>
        <link>http://linuxtoday.com</link>
     </image>
     <item>
        <title>CNET News.com: USB 2 arrives in Linux test version</title>
        <link>
```

```
    http://linuxtoday.com/news_story.php3?ltsn=2002-01-16-008-20-NW-KN-DV
  </link>
  <description>
    "The USB support in the world of Linux is much more freewheeling than at
    Microsoft. The Linux USB software has been created by a largely
    self-appointed team of programmers, who feed batches of code to the
    main kernel project."
  </description>
</item>
<item>
  <title>
    The Register: Open source developers face new warranty threat
  </title>
  <link>
    http://linuxtoday.com/news_story.php3?ltsn=2002-01-16-007-20-NW-LL
  </link>
  <description>
    "If there's one thing free software developers hate more than writing
    documentation, it's fighting a long-drawn out and unglamorous legal
    battle. But the latest episode in the UCITA saga bodes ill for any
    free software author based in the United States."
  </description>
</item>
<item>
  <title>Mandrake Linux Security Update Advisory: sudo</title>
  <link>
    http://linuxtoday.com/news_story.php3?ltsn=2002-01-16-012-20-SC-MD
  </link>
  <description>
    "The SuSE Security Team discovered a vulnerability in sudo that can be
    exploited to obtain root privilege because sudo is installed setuid
    root. An attacker could trick sudo to log failed sudo calls
    executing the sendmail (or equivalent mailer) program with root
    privileges and an environment that is not completely clean."
  </description>
</item>
<item>
  <title>NewsForge: An Open Source adventure at MacWorld</title>
  <link>
    http://linuxtoday.com/news_story.php3?ltsn=2002-01-16-006-20-PS-BD
  </link>
  <description>
    "'For me, what Darwin brings is not so much another alternative to Linux
    or FreeBSD or whatever, as those systems I have that are running
    some Open Source BSD are likely to continue doing that. It's the
    fact that there is an Open Source kernel and utility suite and
    libraries that will support proprietary commercial applications that
    I'm willing to pay for but can't run on most of my other systems.'"
  </description>
```

```xml
  </item>
  <item>
    <title>
      Linux Journal: Sysadmin Corner: Unsung Heroes, Part 2
    </title>
    <link>
      http://linuxtoday.com/news_story.php3?ltsn=2002-01-16-005-20-RV-SW
    </link>
    <description>
      "It seems that several people decided I should show off their web photo
      album generation tool of choice. So, in response to your
      suggestions, I'm going to put off the cool network tool for today;
      let's go on the premise that you all took thousands of pictures over
      the holidays and are dying to make them available on the Web."
    </description>
  </item>
  <item>
    <title>Conectiva Linux Security Announcement: sudo</title>
    <link>
      http://linuxtoday.com/news_story.php3?ltsn=2002-01-16-004-20-SC
    </link>
    <description>
      "Sebastian Krahmer from SuSe found a vulnerability in the sudo package which
      could be used by a local attacker to obtain root privileges.
      Versions prior to and including 1.6.3p7 remove a few potentially
      dangerous environment variables prior to executing a command as
      root, but other variables could be abused and used to obtain root
      privileges."
    </description>
  </item>
  <item>
    <title>IBM developerWorks: Introducing XFS</title>
    <link>
      http://linuxtoday.com/news_story.php3?ltsn=2002-01-16-003-20-PS-KN
    </link>
    <description>
      "Up until now, choosing the appropriate next-generation Linux
      filesystem has been refreshingly straightforward. Those who were
      looking for raw performance generally leaned towards ReiserFS, while
      those more interested in meticulous data integrity features
      preferred ext3. However, with the release of XFS for Linux, things
      have suddenly become much more confusing. In particular, it's no
      longer clear that ReiserFS is still the next-gen performance
      leader."
    </description>
  </item>
  <item>
    <title>Red Hat Security Advisory: sudo</title>
    <link>
    http://linuxtoday.com/news_story.php3?ltsn=2002-01-16-002-20-SC-RH
  </link>
  <description>
    "Versions of sudo prior to 1.6.4 would not clear the environment before
    sending an email notification about unauthorized sudo attempts,
    making it possible for an attacker to supply parameters to the mail
    program. In the worst case, this could lead to a local root
    exploit."
  </description>
</item>
<item>
  <title>
    ZDNet: SuSE 7.3 offers solid server reach and desktop usability
  </title>
  <link>
    http://linuxtoday.com/news_story.php3?ltsn=2002-01-16-001-20-RV-SS
  </link>
  <description>
    "SuSE Linux 7.3 is well prepared for corporate computing, offering
    support for a broad range of server architectures and significant
    advances in usability for both administrators and end users.
    Companies looking for a solid server platform--and perhaps even a
    desktop replacement for Windows--would be well advised to evaluate
    SuSE 7.3's stellar offerings."
  </description>
</item>
<item>
  <title>Linux 2.4.18-pre4 Released</title>
  <link>
    http://linuxtoday.com/news_story.php3?ltsn=2002-01-15-025-20-NW-KN
  </link>
  <description>Changelog, link within.</description>
</item>
<item>
  <title>Linux 2.5.3-pre1 Released</title>
  <link>
    http://linuxtoday.com/news_story.php3?ltsn=2002-01-15-024-20-NW-KN
  </link>
  <description>Changelog, link within.</description>
</item>
<item>
  <title>
    LinuxProgramming: Tcl-URL! - weekly Tcl news and links (Jan 15)
  </title>
  <link>
    http://linuxtoday.com/news_story.php3?ltsn=2002-01-15-023-20-PS
  </link>
  <description>All the latest news from the Tcl world.</description>
</item>
```

```
<item>
    <title>MLUG.ca: Up2date with RedHat</title>
    <link>
        http://linuxtoday.com/news_story.php3?ltsn=2002-01-15-022-20-PS-RH-SW
    </link>
    <description>
        "What up2date does is similar to what Debian has been doing for years
        now, namely, finding updates and installing packages without having
        dependency issues. For those who have used Debian I'm refering to
        the program apt-get which does exactly what up2date does. Some might
        argue that apt-get is a better program, but I certainly don't want
        to start a holy war."
    </description>
</item>
<item>
    <title>
        Call for papers out for Ottawa Linux Symposium 2002
    </title>
    <link>
     http://linuxtoday.com/news_story.php3?ltsn=2002-01-15-021-20-NW-CY-DV
    </link>
    <description>
        "OLS is Linux and Linux related software developers symposium with a
        strong focus on emerging technologies, research projects, and works
        in progress. We invite contribution from the free software and
        commercial developers."
    </description>
</item>
<item>
    <title>
        Newsforge: Community commentary: The case for 'lagom' copyright
    </title>
    <link>
        http://linuxtoday.com/news_story.php3?ltsn=2002-01-15-021-20-OP-LL
    </link>
    <description>
        "One of the big issues of free software during 2001 was whether Richard
        M Stallman was for or against a codified GNU GPL. Hence, did
        Stallman --the father of Free Software -- propagate a law to support
        his beliefs?"
    </description>
</item>
<textinput>
    <title>Search</title>
    <description>Search Linux Today:</description>
    <name>query</name>
    <link>http://linuxtoday.com/search.php3</link>
</textinput>
```

```
  </channel>
</rss>
```

Упътване

Решение

```
import org.xml.sax.InputSource;
import org.xml.sax.XMLReader;
import org.xml.sax.helpers.XMLReaderFactory;
import java.io.IOException;
import java.io.OutputStreamWriter;
import java.io.StringReader;

import org.xml.sax.Attributes;
import org.xml.sax.ContentHandler;
import org.xml.sax.Locator;
import org.xml.sax.SAXException;

public class SAX1 { //replace SAX1 by Main for pasting it as Main.java in https://repl.it/
  public static void main(String[] args) {
    OutputStreamWriter outputStreamWriter = new OutputStreamWriter(System.out);
    try {
      XMLReader parser = XMLReaderFactory.createXMLReader();
      InputSource source = new InputSource("rss.xml");
      //InputSource source = new InputSource(new StringReader(""));
      parser.setContentHandler(new SAXHandler(outputStreamWriter)
      );
      parser.parse(source);
    } catch (Exception e) {
      e.printStackTrace();
    } finally {
      try {
        outputStreamWriter.close();
      } catch (IOException e) {
        e.printStackTrace();
      }
    }
  }
}


class SAXHandler implements ContentHandler {
  Locator locator;
  Integer indent;
  OutputStreamWriter outputStreamWriter;
  private final Integer TAB_SIZE = 4;
  public SAXHandler(OutputStreamWriter outputStreamWriter) {
    this.outputStreamWriter = outputStreamWriter;
    indent = 0;
  }

  @Override
  public void setDocumentLocator(Locator locator) {
    this.locator = locator;
  }

  @Override
  public void startDocument() throws SAXException {
    printIndented("<?xml version=\"1.0\" encoding=\"UTF-8\"?>", false, false);
  }

  @Override
  public void endDocument() throws SAXException {
    // ...
  }

  @Override
  public void startElement(String uri, String localName, String qName, Attributes atts) throws
SAXException {
    printIndented(String.format("<%s", qName), true, false);
    printAttributes(atts);
    printIndented(">\r\n", false, true);
    ++indent;
  }

  @Override
  public void endElement(String uri, String localName, String qName) throws SAXException {
    --indent;
    printIndented(String.format("</%s>", qName), true, false);
  }

  @Override
  public void characters(char[] chars, int start, int length) throws SAXException {
    String s = new String(chars, start, length).toUpperCase().trim();
    if (s.length() > 0) {
      printIndented(s, false, false);
    }
  }


  @Override
  public void startPrefixMapping(String prefix, String uri) throws SAXException {
    // ...
  }
  @Override
  public void endPrefixMapping(String prefix) throws SAXException {
    // ...
  }

  @Override
  public void ignorableWhitespace(char[] chars, int start, int length) throws SAXException {
    // ...
  }

  @Override
  public void processingInstruction(String target, String data) throws SAXException {
    // ...
  }

  @Override
  public void skippedEntity(String name) throws SAXException {
    // ...
  }

  private void printIndented(String what, boolean isEndOfElement, boolean isElement) {
    try {
      if(isEndOfElement) {
        outputStreamWriter.write("\r\n");
      }
      if (indent > 0 && !isElement) {
        outputStreamWriter.write(String.format("%1$" + (indent * TAB_SIZE) + "s", ""));
      }
      outputStreamWriter.write(what);
      outputStreamWriter.flush();
    } catch (IOException e) {
      e.printStackTrace();
    }
  }

  private void printAttributes(Attributes atts) {
    if (atts.getLength() > 0) {
      ++indent;
      for (int i = 0; i < atts.getLength(); ++i) {
        String name = atts.getQName(i);
        printIndented(String.format(" %s = \"%s\"", name, atts.getValue(i)), false, true);
      }
      --indent;
    }
  }
}
```

Задача 2: Като използвате SAX, имплементирайте валидация на XML документът от задача 1, която проверява дали са изпълнени следните условия:

1. Всеки елемент item трябва да съдържа едно множество от под-елементите title, link и description, всеки от тях срещащ се точно един път

2. Стойността на атрибута version (принадлежащ на елемента rss) трябва да бъде цяло положително число

3. Елементът channel трябва да съдържа поне 2 и неповече от 10 под-елемента item

4. Разпечатайте информация за мястото (номер на ред и колона), на което грешката се среща

Упътване

Решение

```
import java.io.IOException;
import java.io.OutputStreamWriter;
import org.xml.sax.Attributes;
import org.xml.sax.ContentHandler;
import org.xml.sax.Locator;
import org.xml.sax.SAXException;

public class SAXValidator implements ContentHandler {
  Locator locator;
  OutputStreamWriter outputStreamWriter;
  private final Integer TAB_SIZE = 4;
  private String currentElementName;
  private boolean titleDetected = false;
  private int countOfTitle = 0;
  private boolean linkDetected =false;
  private int countOfLink = 0;
  private boolean descriptionDetected = false;
  private int countOfDescription = 0;
  private boolean itemDetected = false;
  private int countOfItem = 0;

  public SAXValidator(OutputStreamWriter outputStreamWriter) {
    this.outputStreamWriter = outputStreamWriter;
  }

  @Override
  public void setDocumentLocator(Locator locator) {
    this.locator = locator;
  }
  @Override
  public void startDocument() throws SAXException {
    //..
  }

  @Override
  public void endDocument() throws SAXException {
    // ...
  }

  @Override
  public void startElement(String uri, String localName, String qName, Attributes atts) throws SAXException {
    currentElementName = qName;
    validateVersion(atts);

    if (qName.equals("item")) {
      titleDetected = false;
      linkDetected = false;
      descriptionDetected = false;
      countOfTitle = 0;
      countOfDescription = 0;
      countOfLink = 0;
      itemDetected = true;
      ++countOfItem;
    }

    if (qName.equals("title")) {
      titleDetected = true;
      ++countOfTitle;
    }

    if (qName.equals("link")) {
      linkDetected = true;
      ++countOfLink;
    }

    if (qName.equals("description")) {
      descriptionDetected = true;
      ++countOfDescription;
    }
  }

  @Override
  public void endElement(String uri, String localName, String qName) throws SAXException {
    if (localName.equals("item")) {
      if (!(titleDetected && linkDetected && descriptionDetected && countOfTitle == 1 && countOfLink == 1 && countOfDescription == 1)) {
        reportError("Item must have one subset of the sequence: title, link, description.");
      }
    }
    if (localName.equals("channel")) {
      if (!(itemDetected && countOfItem >= 2 && countOfItem <= 10)) {
        reportError("Number of elements item must be between 2 and 10: ");
      }
    }
  }

  @Override
  public void characters(char[] chars, int start, int length) throws SAXException {
    //..
  }



  @Override
  public void startPrefixMapping(String prefix, String uri) throws SAXException {
    // ...
  }

  @Override
  public void endPrefixMapping(String prefix) throws SAXException {
    // ...
  }

  @Override
  public void ignorableWhitespace(char[] chars, int start, int length) throws SAXException {
    // ...
  }

  @Override
  public void processingInstruction(String target, String data) throws SAXException {
    // ...
  }

  @Override
  public void skippedEntity(String name) throws SAXException {
    // ...
  }

  private void printIndented(String what) {
    try {
      outputStreamWriter.write(what);
      outputStreamWriter.flush();
    } catch (IOException e) {
      e.printStackTrace();
    }
  }
```

```
  private void reportError(String cause) {
    printIndented(String.format("\r\nError: %s on line %d column %d.", cause, locator.getLineNumber(), locator.getColumnNumber()));
  }

  private void validateVersion(Attributes atts) {
    if (atts.getLength() > 0) {
      try {
        if (currentElementName.equals("rss") && (Integer.parseInt(atts.getValue("version")) < 0)) {
          reportError("Attribute version is expected to have a positive integer value: ");
        }
      } catch (NumberFormatException e) {
        reportError(String.format("Wrong value for version: %s (Attribute version is expected to have a positive integer value):", atts.getValue("version")));
      }
    }
  }
}
```

Задача 3: Като използвате SAX, трансформирайте XML документът от задача 1 в HTML документ, използвайки за изход стандартната конзола. Новият HTML документ трябва да съдържа таблица със следното съдържание:

1. Три колони с имена title, link и description
2. По един ред за всеки елемент item със стойностите на под-елементите му title, link и description

Решение

```
import java.io.IOException;
import java.io.OutputStreamWriter;
import org.xml.sax.Attributes;
import org.xml.sax.ContentHandler;
import org.xml.sax.Locator;
import org.xml.sax.SAXException;

public class SAXTransformator implements ContentHandler {
  private class Item {
    String title;
    String link;
    String description;
  }

  Locator locator;
  OutputStreamWriter outputStreamWriter;
  private final Integer TAB_SIZE = 4;
  private String currentElement;
  private Integer indent;
  private Item currentItem;
  boolean inItem = false;

  public SAXTransformator(OutputStreamWriter outputStreamWriter) {
    this.outputStreamWriter = outputStreamWriter;
  }

  @Override
  public void setDocumentLocator(Locator locator) {
    this.locator = locator;
    indent = 0;
  }

  @Override
  public void startDocument() throws SAXException {
    printIndented("<!DOCTYPE html>");
    printIndented("<html>");
    ++indent;
    printIndented("<head><title>List of items</title></head>");
    printIndented("<body>");
    ++indent;
    printIndented("<table style=\"border: 1px solid black;\">");
    ++indent;
    printIndented("<thead><tr><th>Title</th><th>Link</th><th>Description</th></tr></thead>");
    printIndented("<tbody>");
    ++indent;
  }

  @Override
  public void endDocument() throws SAXException {
    --indent;
    printIndented("</tbody>");
    --indent;
    printIndented("</table>");
    --indent;
    printIndented("</body>");
    --indent;
    printIndented("</html>");
  }

  @Override
  public void startElement(String uri, String localName, String qName, Attributes atts) throws
SAXException {
    currentElement = qName;

    if ("item".equals(currentElement)) {
      currentItem = new Item();
      inItem = true;
    }
  }

  @Override
  public void endElement(String uri, String localName, String qName) throws SAXException {
    if ("item".equals(localName)) {
      printIndented("<tr><td>" + currentItem.title + "</td><td>" + currentItem.link + "</td><td>" +
currentItem.description + "</td></tr>");
      inItem = false;
    }
  }

  @Override
  public void characters(char[] chars, int start, int length) throws SAXException {
    String s = new String(chars, start, length).trim();

    if (inItem && s.length() > 0) {
      if ("title".equals(currentElement)) {
        currentItem.title = s;
      }
      if ("link".equals(currentElement)) {
        currentItem.link = s;
      }
      if ("description".equals(currentElement)) {
        if(currentItem.description == null) {
          currentItem.description = s;
        } else {
          currentItem.description += s;
        }
      }
    }
  }

  @Override
  public void startPrefixMapping(String prefix, String uri) throws SAXException {
    // ...
  }

  @Override
  public void endPrefixMapping(String prefix) throws SAXException {
    // ...
  }

  @Override
  public void ignorableWhitespace(char[] chars, int start, int length) throws SAXException {
    // ...
  }

  @Override
  public void processingInstruction(String target, String data) throws SAXException {
    // ...
  }

  @Override
  public void skippedEntity(String name) throws SAXException {
    // ...
  }

  private void printIndented(String what) {
    try {
      if (indent > 0) {
        outputStreamWriter.write(String.format("%1$" + (indent * TAB_SIZE) + "s", ""));
      }
      outputStreamWriter.write(what + "\r\n");
      outputStreamWriter.flush();
    } catch (IOException e) {
      e.printStackTrace();
    }
  }
}
```

Задача 4: Решете задача 1 като вместо SAX използвате DOM

Упътване

Решение

```
//import java.io.StringReader;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;

import org.w3c.dom.Attr;
import org.w3c.dom.Document;
import org.w3c.dom.NamedNodeMap;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;
import org.xml.sax.InputSource;


public class Main {
  private static boolean skipNL;

  public static void main(String[] args) throws Exception {
    DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
    dbf.setValidating(false);
    DocumentBuilder builder = dbf.newDocumentBuilder();
    InputSource source = new InputSource("rss.xml");
    //InputSource source = new InputSource(new StringReader(""));
    Document document = builder.parse(source);
    System.out.println(printXML(document.getDocumentElement()));
  }

  private static String printXML(Node rootNode) {
   String tab = "";
   skipNL = false;
   return(printXML(rootNode, tab));
  }
  private static String printXML(Node rootNode, String tab) {
   String print = "";
   if(rootNode.getNodeType()==Node.ELEMENT_NODE) {
   print += "\n"+tab+"<"+rootNode.getNodeName();
   NamedNodeMap attributes = rootNode.getAttributes();
  for (int j = 0; j < attributes.getLength(); j++) {
    Attr attr = (Attr) attributes.item(j);
    if(attr != null) {
      print += " " + attr.getNodeName() + "=\"" + attr.getNodeValue() + "\"";
    }
  }
  print += ">";
   }
   NodeList nl = rootNode.getChildNodes();
   if(nl.getLength()>0) {
   for (int i = 0; i < nl.getLength(); i++) {
   print += printXML(nl.item(i), tab + " "); // \t
   }
   } else {
   if(rootNode.getNodeValue()!=null) {
   print = rootNode.getNodeValue();
   }
   skipNL = true;
   }
   if(rootNode.getNodeType()==Node.ELEMENT_NODE) {
   if(!skipNL) {
   print += "\n"+tab;
   }
   skipNL = false;
   print += "";
   }
   return(print);
  }
}
```

Задача 5: Използвайки DOM, променете XML документът от задача 1, по следния начин:

1. Превърнете под-елемента link на елемента item в негов атрибут

2. Запазете първите 10 item елементи, а всички останали ги изтрийте

3. Добавете нов под-елемент sponsor на елемента channel

Решение

```
import java.io.File;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.transform.OutputKeys;
import javax.xml.transform.Transformer;
import javax.xml.transform.TransformerFactory;
import javax.xml.transform.dom.DOMSource;
import javax.xml.transform.stream.StreamResult;
import org.w3c.dom.Document;
import org.w3c.dom.Element;
import org.w3c.dom.NodeList;
import org.xml.sax.InputSource;

public class DOM2 {
  public static void main(String[] args) throws Exception {
    DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
    dbf.setValidating(false);
    DocumentBuilder builder = dbf.newDocumentBuilder();
    InputSource source = new InputSource("rss.xml");
    //InputSource source = new InputSource(new StringReader(""));
    Document document = builder.parse(source);
    processTree(document);

    TransformerFactory tf = TransformerFactory.newInstance();
    Transformer writer = tf.newTransformer();
    writer.setOutputProperty(OutputKeys.ENCODING, "utf-8");
    writer.transform(new DOMSource(document), new StreamResult(new File("rss_new.xml")));
  }

  private static void processTree(Document doc) {
    NodeList linkList = doc.getElementsByTagName("link");
    NodeList itemList = doc.getElementsByTagName("item");

    for (int i = itemList.getLength() - 1; i >= 10; --i) {
      Element item = (Element)itemList.item(i);
      item.getParentNode().removeChild(item);
    }
    for (int i = linkList.getLength() - 1; i >= 0; --i) {
      Element link = (Element)linkList.item(i);
      Element item = (Element)link.getParentNode();

      if("item".equals(item.getNodeName())) {
        item.setAttribute("link", link.getTextContent().trim());
        item.removeChild(link);
      }
    }

    Element sponsor = doc.createElement("sponsor");
    sponsor.setTextContent("IBM");
    doc.getElementsByTagName("channel").item(0).appendChild(sponsor);
  }
}
```

Задача 6: Решете задача 1, използвайки StAX

Упътване

Решение

```
import java.io.FileNotFoundException;
import java.io.FileReader;
import javax.xml.stream.XMLInputFactory;
import javax.xml.stream.XMLStreamConstants;
import javax.xml.stream.XMLStreamException;
import javax.xml.stream.XMLStreamReader;

public class StAX {
  public static void main(String[] args) {
   XMLInputFactory xmlif = XMLInputFactory.newInstance();
   XMLStreamReader xmlr = null;
    try {
      xmlr = xmlif.createXMLStreamReader(new FileReader("rss.xml"));
     while(xmlr.hasNext()){
     printEvent(xmlr);
     xmlr.next();
     }
     xmlr.close();
    } catch (FileNotFoundException e) {
      e.printStackTrace();
    } catch (XMLStreamException e) {
      e.printStackTrace();
    }
   }
```

```
private static void printEvent(XMLStreamReader xmlr) {
  switch (xmlr.getEventType()) {
    case XMLStreamConstants.START_ELEMENT:
     System.out.print("<");
     printName(xmlr);
     printNamespaces(xmlr);
     printAttributes(xmlr);
     System.out.print(">");
     break;
    case XMLStreamConstants.END_ELEMENT:
     System.out.print("</");
     printName(xmlr);
     System.out.print(">");
     break;
    case XMLStreamConstants.SPACE:
    case XMLStreamConstants.CHARACTERS:
     int start = xmlr.getTextStart();
     int length = xmlr.getTextLength();
     System.out.print(new String(xmlr.getTextCharacters(),
                   start,
                   length));
     break;
    case XMLStreamConstants.PROCESSING_INSTRUCTION:
     System.out.print("<?");
     if (xmlr.hasText())
       System.out.print(xmlr.getText());
     System.out.print("?>");
     break;
    case XMLStreamConstants.CDATA:
     System.out.print("<![CDATA[");
     start = xmlr.getTextStart();
     length = xmlr.getTextLength();
     System.out.print(new String(xmlr.getTextCharacters(),
                   start,
                   length));
     System.out.print("]]>");
     break;
    case XMLStreamConstants.COMMENT:
     System.out.print("<!--");
     if (xmlr.hasText())
       System.out.print(xmlr.getText());
     System.out.print("-->");
     break;
    case XMLStreamConstants.ENTITY_REFERENCE:
     System.out.print(xmlr.getLocalName()+"=");
     if (xmlr.hasText())
       System.out.print("["+xmlr.getText()+"]");
     break;
```

```
      case XMLStreamConstants.START_DOCUMENT:
       System.out.print("<?xml");
       System.out.print(" version='"+xmlr.getVersion()+"'");
       System.out.print(" encoding='"+xmlr.getCharacterEncodingScheme()+"'");
       if (xmlr.isStandalone())
         System.out.print(" standalone='yes'");
       else
         System.out.print(" standalone='no'");
       System.out.print("?>");
       break;
   }
}

private static void printName(XMLStreamReader xmlr){
if(xmlr.hasName()){
 String prefix = xmlr.getPrefix();
 String uri = xmlr.getNamespaceURI();
 String localName = xmlr.getLocalName();
 printName(prefix,uri,localName);
}
}

private static void printName(String prefix,
                     String uri,
                     String localName) {
if (uri != null && !("".equals(uri)) ) System.out.print("['"+uri+"']:");
if (prefix != null && prefix.length() > 0) System.out.print(prefix+":");
if (localName != null) System.out.print(localName);
}

private static void printAttributes(XMLStreamReader xmlr){
for (int i=0; i < xmlr.getAttributeCount(); i++) {
 printAttribute(xmlr,i);
}
}

private static void printAttribute(XMLStreamReader xmlr, int index) {
String prefix = xmlr.getAttributePrefix(index);
String namespace = xmlr.getAttributeNamespace(index);
String localName = xmlr.getAttributeLocalName(index);
String value = xmlr.getAttributeValue(index);
System.out.print(" ");
printName(prefix,namespace,localName);
System.out.print("='"+value+"'");
}

private static void printNamespaces(XMLStreamReader xmlr){
for (int i=0; i < xmlr.getNamespaceCount(); i++) {
 printNamespace(xmlr,i);
```

```
  }
  }

  private static void printNamespace(XMLStreamReader xmlr, int index) {
  String prefix = xmlr.getNamespacePrefix(index);
  String uri = xmlr.getNamespaceURI(index);
  System.out.print(" ");
  if (prefix == null)
   System.out.print("xmlns='"+uri+"'");
  else
   System.out.print("xmlns:"+prefix+"='"+uri+"'");
  }
}
```

Задача 7: Използвайки StAX, създайте по-долу дадения XML документ

```
<?xml version="1.0"?>
<bookstore>
  <book category="COOKING">
    <title lang="en">Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
 </book>
 <book category="CHILDREN">
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
  <book category="WEB">
    <title lang="en">Learning XML</title>
    <author>Erik T. Ray</author>
    <year>2003</year>
    <price>39.95</price>
  </book>
</bookstore>
```

Решение

```
import java.io.IOException;
import java.io.StringWriter;
import javax.xml.stream.XMLOutputFactory;
import javax.xml.stream.XMLStreamException;
import javax.xml.stream.XMLStreamWriter;

public class StAX2 {
  public static void main(String[] args) {
    try {
      StringWriter stringWriter = new StringWriter();
      XMLOutputFactory xMLOutputFactory = XMLOutputFactory.newInstance();
      XMLStreamWriter xMLStreamWriter = xMLOutputFactory.createXMLStreamWriter(stringWriter);
      xMLStreamWriter.writeStartDocument();
      xMLStreamWriter.writeStartElement("bookstore");
      xMLStreamWriter.writeStartElement("book");//book 1
      xMLStreamWriter.writeAttribute("category", "COOKING");
      createSimpleElement(xMLStreamWriter, "title", "lang", "en", "Everyday Italian");
      createSimpleElement(xMLStreamWriter, "author", null, null, "Giada De Laurentiis");
      createSimpleElement(xMLStreamWriter, "year", null, null, "2005");
      createSimpleElement(xMLStreamWriter, "price", null, null, "30.00");
      xMLStreamWriter.writeEndElement();//book 1
      xMLStreamWriter.writeStartElement("book");//book 2
      xMLStreamWriter.writeAttribute("category", "CHILDREN");
      createSimpleElement(xMLStreamWriter, "title", "lang", "en", "Harry Potter");
      createSimpleElement(xMLStreamWriter, "author", null, null, "J K. Rowling");
      createSimpleElement(xMLStreamWriter, "year", null, null, "2005");
      createSimpleElement(xMLStreamWriter, "price", null, null, "29.99");
      xMLStreamWriter.writeEndElement();//book 2
      xMLStreamWriter.writeStartElement("book"); //book 3
      xMLStreamWriter.writeAttribute("category", "WEB");
      createSimpleElement(xMLStreamWriter, "title", "lang", "en", "Learning XML");
      createSimpleElement(xMLStreamWriter, "author", null, null, "Erik T. Ray");
      createSimpleElement(xMLStreamWriter, "year", null, null, "2003");
      createSimpleElement(xMLStreamWriter, "price", null, null, "39.95");
      xMLStreamWriter.writeEndElement();//book 3
      xMLStreamWriter.writeEndDocument();
      xMLStreamWriter.flush();
      xMLStreamWriter.close();
      String xmlString = stringWriter.getBuffer().toString();
      stringWriter.close();
      System.out.println(xmlString);
    } catch (XMLStreamException e) {
      e.printStackTrace();
    } catch (IOException e) {
      e.printStackTrace();
    }
  }

  public static void createSimpleElement(XMLStreamWriter xMLStreamWriter, String elementName,
String attributeName, String attributeValue, String strValue) {
    try {
      if(xMLStreamWriter != null && elementName != null) {
        xMLStreamWriter.writeStartElement(elementName);
        if(attributeName != null) {
          xMLStreamWriter.writeAttribute(attributeName, attributeValue);
        }
        if(strValue != null) {
          xMLStreamWriter.writeCharacters(strValue);
        }
        xMLStreamWriter.writeEndElement();
       }
    } catch (XMLStreamException e) {
      e.printStackTrace();
    }
  }
}
```