XML Document

Book xml

```
<?xml version="1.0" encoding="UTF-8"?>
<library>
   <book>
      <title>The Great Gatsby</title>
      <author>F. Scott Fitzgerald</author>
      <year>1925</year>
      <genre>Fiction</genre>
   </book>
   <book>
      <title>To Kill a Mockingbird</title>
      <author>Harper Lee</author>
      <year>1960</year>
      <genre>Fiction</genre>
   </book>
   <book>
      <title>1984</title>
      <author>George Orwell</author>
      <year>1949</year>
      <genre>Dystopian</genre>
   </book>
</library>
```

Java XmlParser

```
import java.io.File;
import java.io.InputStream;
import javax.xml.parsers.*;
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.*;

public class XmlParser {
    public static void main(String[] args) {
        try {
            // Load XML from the src/xmlproject folder
            InputStream inputStream = XmlParser.class.getResourceAsStream("books.xml");

            if (inputStream == null) {
                System.out.println("File not found in package xmlproject!");
                return;
            }

            // Create a DocumentBuilderFactory and parse the XML content
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            DocumentBuilder builder = factory.newDocumentBuilder();
            Document document = builder.parse(inputStream);
```

```
        // Normalize document
        document.getDocumentElement().normalize();

        // Get all <book> elements
        NodeList nodeList = document.getElementsByTagName("book");

        // Loop through each book
        for (int i = 0; i < nodeList.getLength(); i++) {
            Node node = nodeList.item(i);

            if (node.getNodeType() == Node.ELEMENT_NODE) {
                Element element = (Element) node;

                // Extract values for each book
                String title = element.getElementsByTagName("title").item(0).getTextContent();
                String author =
element.getElementsByTagName("author").item(0).getTextContent();
                String year = element.getElementsByTagName("year").item(0).getTextContent();
                String genre = element.getElementsByTagName("genre").item(0).getTextContent();

                // Print book details
                System.out.println("Title: " + title);
                System.out.println("Author: " + author);
                System.out.println("Year: " + year);
                System.out.println("Genre: " + genre);
                System.out.println("----------------------");
            }
        }
        Element firstBook = (Element) nodeList.item(0);
```

```
            firstBook.getElementsByTagName("year").item(0).setTextContent("2023");
            TransformerFactory transformerFactory = TransformerFactory.newInstance();
            Transformer transformer = transformerFactory.newTransformer();
            DOMSource source = new DOMSource(document);
            StreamResult result = new StreamResult(new File("updated_books.xml"));
            transformer.transform(source, result);



        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```