

Create a new file named books.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>
```

Parsing XML in Java

Create a Java Class for XmlParser.java:

```
package javaapplication8;
import java.io.File;
import org.w3c.dom.*;
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;
/**
 *
 * @author student
 */
public class xmlparser {
    public static void main(String[] args) {
        try {
            // Create a new DocumentBuilderFactory and DocumentBuilder
            DocumentBuilderFactory factory =
            DocumentBuilderFactory.newInstance();
            DocumentBuilder builder = factory.newDocumentBuilder();

            // Parse the XML file

            Document document =
            builder.parse("C:\\Users\\student\\Documents\\NetBeansProjects\\JavaApplication8\\src\\java
            application8\\books.xml");

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

            // Get the root element (library)
            NodeList nodeList = document.getElementsByTagName("book");
```

```

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

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

        // Get and print the details of 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();

        System.out.println("Title: " + title);
        System.out.println("Author: " + author);
        System.out.println("Year: " + year);
        System.out.println("Genre: " + genre);
        System.out.println("-----");
    }
}

// Modify the year of the first book
Element firstBook = (Element) nodeList.item(0);
firstBook.getElementsByTagName("year").item(0).setTextContent("2023");

```

```

// Save the modified document
TransformerFactory transformerFactory =
TransformerFactory.newInstance();
Transformer transformer = transformerFactory.newTransformer();
DOMSource source = new DOMSource(document);
StreamResult result = new StreamResult(new
File("C:\\Users\\student\\Documents\\NetBeansProjects\\JavaApplication8\\src\\javaapplicatio
n8\\updated_books.xml"));
transformer.transform(source, result);
} catch (Exception e) {
e.printStackTrace();
}
}
}

```

Out put



```

Output - xmlproject (run)

run:
Title: The Great Gatsby
Author: F. Scott Fitzgerald
Year: 1925
Genre: Fiction
-----
Title: To Kill a Mockingbird
Author: Harper Lee
Year: 1960
Genre: Fiction
-----
Title: 1984
Author: George Orwell
Year: 1949
Genre: Dystopian
-----
BUILD SUCCESSFUL (total time: 1 second)

```

Modifying XML Data

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><library>

<book>

<title>The Great Gatsby</title>

<author>F. Scott Fitzgerald</author>

<year>2023</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>
```

Out put

Output - xmlproject (run)



```
run:
Updated XML content:
<?xml version="1.0" encoding="UTF-8" standalone="no"?><library>
  <book>
    <title>The Great Gatsby</title>
    <author>F. Scott Fitzgerald</author>
    <year>2023</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>BUILD SUCCESSFUL (total time: 1 second)
```