EXPERIMENT: 6

6. Create an xml for the bookstore. Validate the same using both DTD and XSD

Creating an XML for a bookstore involves defining the structure of the XML file to represent information about books, authors, and related metadata. To ensure the XML adheres to a specific structure, you can validate it using Document Type Definition (DTD) or XML Schema Definition (XSD). DTD is simpler but less expressive, while XSD is more robust and supports data types and namespaces.

Let's start by creating an XML file representing a simple bookstore, then validate it with both DTD and XSD.

```
<?xml version="1.0" encoding="UTF-8"?>
<bookstore>
   <book isbn="978-3-16-148410-0">
      <title>Effective Java</title>
      <author>Joshua Bloch</author>
      <price>45.0</price>
   </book>
   <book isbn="978-0-596-52068-7">
      <title>Head First Java</title>
      <author>Kathy Sierra</author>
      <price>30.0</price>
   </book>
</bookstore>
```

Step 2: Create DTD for Validation

To validate the bookstore XML with a DTD, define the expected structure. The following DTD specifies the required elements, their order, and attributes.

```
<!DOCTYPE bookstore [
   <!ELEMENT bookstore (book+)>
   <!ELEMENT book (title, author, price)>
   <!ATTLIST book
      isbn CDATA #REQUIRED
   >
   <!ELEMENT title (#PCDATA)>
   <!ELEMENT author (#PCDATA)>
   <!ELEMENT price (#PCDATA)>
]>
```

This DTD specifies:

<bookstore> contains one or more <book> elements.
<book> contains <title>, <author>, and <price>.
The isbn attribute of <book> is required.
To validate the XML with this DTD, add a reference to it in the XML file:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE bookstore SYSTEM "bookstore.dtd">
<bookstore>
  <book isbn="978-3-16-148410-0">
    <title>Effective Java</title>
    <author>Joshua Bloch</author>
    <price>45.0</price>
  </book>
  <book isbn="978-0-596-52068-7">
    <title>Head First Java</title>
    <author>Kathy Sierra</author>
    <price>30.0</price>
  </book>
</bookstore>
```

Step 3: Create XSD for Validation
An XSD defines a more complex schema with data types, constraints, and namespaces.
Here's an XSD to validate the bookstore XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="bookstore">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="book" minOccurs="1" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="title" type="xs:string"/>
              <xs:element name="author" type="xs:string"/>
              <xs:element name="price" type="xs:float"/>
            </xs:sequence>
            <xs:attribute name="isbn" type="xs:string" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
```

```
        </xs:complexType>
    </xs:element>
</xs:schema>
```

This XSD schema specifies:

<bookstore> contains one or more <book> elements.
<book> has isbn as a required attribute and contains <title>, <author>, and <price>.
The data types are explicitly defined (string, float).
To validate the XML with this XSD, add a reference to it in the XML file:

```
<?xml version="1.0" encoding="UTF-8"?>
<bookstore xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:noNamespaceSchemaLocation="bookstore.xsd">
    <book isbn="978-3-16-148410-0">
        <title>Effective Java</title>
        <author>Joshua Bloch</author>
        <price>45.0</price>
    </book>
    <book isbn="978-0-596-52068-7">
        <title>Head First Java</title>
        <author>Kathy Sierra</author>
        <price>30.0</price>
    </book>
</bookstore>
```

Conclusion

This example demonstrates creating an XML file for a bookstore and validating it using both DTD and XSD. While DTD is suitable for simple validation, XSD offers more complex schema definitions with data types and additional constraints. By referencing the DTD or XSD in the XML file, you ensure the XML structure aligns with the defined schema, providing validation and error-checking.