Java JDOM Parser - Query XML Document

Java JDOM parser is an API which has classes and methods to build JDOM documents from XML files to query related information. In this chapter, we are going to query elements by text content using getText() method and to query elements by attributes, we are using getAttributeValue() method.

Query XML Using JDOM Parser

Following are the steps we need to follow to query an XML document using JDOM parser –

- **Step 1:** Creating a SAXBuilder Object
- **Step 2:** Reading the XML
- **Step 3:** Parsing the XML Document
- **Step 4:** Querying the Elements

Refer this chapter for first three steps

Step 4: Querying the Elements

After completing the first three steps, we get a JDOM document. Using classes and methods present in org.jdom2 package, we can start querying the elements and their attributes.

Now, we are going to see two examples on how to query elements based on their text content and their attributes. We use the same cars.xml file for both the examples.

Querying Elements by TextContent

We can query elements by their text content, by first getting the root element using getRootElement() method. After we obtain the root element, we can use getChildren() function to get all the child elements. Then, we can query the elements by text content using getText() method.

Example

Consider the following **cars.xml** file with carname elements having company attribute and text content. Now, we are going to query this XML file to find "Bentley 2" car.

```
<?xml version = "1.0"?>
<cars>
      <carname company="Ferarri" >Ferarri 101</carname>
      <carname company="Lamborgini">Lamborgini 001</carname>
      <carname company="Lamborgini">Lamborgini 002</carname>
      <carname company="Lamborgini">Lamborgini 003</carname>
      <carname company="Bentley">Bentley 1</carname>
      <carname company="Bentley">Bentley 2</carname>
      <carname company="Bentley">Bentley 3</carname>
</cars>
```

In the following **QueryXMLElements.java** program, we are parsing the cars.xml file using SAXBuilder to query all the carname elements. After getting the carname elements in an Element list, we are iterating the list to find "Bentley 2" car.

```
import java.io.File;
import org.jdom2.Document;
import org.jdom2.Element;
import org.jdom2.input.SAXBuilder;
import java.util.List;

public class QueryXMLElements {
   public static void main(String args[]) {
      try {

         //Creating a SAXBuilder Object
         SAXBuilder saxBuilder = new SAXBuilder();

         //Reading the XML
         File inputFile = new File("cars.xml");

         //Parsing the XML Document
         Document document = saxBuilder.build(inputFile);

         //Retrieving the Root Element
         Element RootElement = document.getRootElement();
         List<Element> carList = RootElement.getChildren("carname");

         //Finding "Bentley 2" car in the list
         boolean found=false;
         for(int index=0; index<carList.size();index++) {
```

```
            Element car = carList.get(index);
            if(car.getText().equals("Bentley 2")) {
               found=true;
               break;
            }
         }
         if(found) {
            System.out.println("Bentley 2 car is found");
         }
         else {
            System.out.println("Bentley 2 car is not found");
         }
      } catch(Exception e) {
          e.printStackTrace();
      }
   }
}
```

The output window displays that the "Bentley 2" car is found in the XML file.

```
Bentley 2 car is found
```

Learn **Java** in-depth with real-world projects through our **Java certification course**. Enroll and become a certified expert to boost your career.

Querying Elements by Attributes

Elements can also have attributes along with the text content. Now, let use the same cars.xml file to query the carname elements by their company attribute.

The **getAttributeValue("attr_name")** method of Element class takes attribute name as a String argument and returns the corresponding value of the attribute.

Example

In the following **QueryAttributes.java** program, we are getting the list of carname Elements from getChildren() method. Then, we are iterating the list and incrementing the count when the company attribute value is equal to "Bentley".

```
import java.io.File;
import org.jdom2.Document;
import org.jdom2.Element;
import org.jdom2.input.SAXBuilder;
```

```
import java.util.List;

public class QueryAttributes {
   public static void main(String args[]) {
      try {

         //Creating a SAXBuilder Object
         SAXBuilder saxBuilder = new SAXBuilder();

         //Reading the XML
         File inputFile = new File("cars.xml");

         //Parsing the XML Document
         Document document = saxBuilder.build(inputFile);

         //Retrieving the Root Element
         Element RootElement = document.getRootElement();
         List<Element> carList = RootElement.getChildren("carname");

         //Counting Bentley cars
         int count=0;
         for(int index=0; index<carList.size();index++) {
            Element car = carList.get(index);
            if(car.getAttributeValue("company").equals("Bentley")) {
               count++;
            }
         }
         System.out.println("Total number of Bentley cars : " + count);
      } catch(Exception e) {
          e.printStackTrace();
      }
   }
}
```

Output

The count value has the number of Bentley cars in the XML file and is printed on the console.

```
Total number of Bentley cars : 3
```