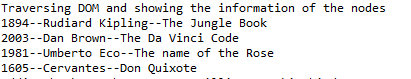
PRACTICE 2: TRAVERSE DOM AND SHOW ELEMENTS.

OBJECTIVE: after the DOM tree is created in memory, we will traverse it working with the information inside the nodes. In this practice, we will only show all the information of the nodes.

We want to see this result from our XML file.:



# PRACTICE:

* Classes used: Node and NodeList, importing org.w3c.dom.
* Methods:

getNodeType, getAttributes, getNodeValue

**public** **class** AccessXMLDOM {

Document doc;

//code written in Practice1

**public** **void** traverseDOMandShow() {

String[] data=**new** String[3];

Node node=**null**;

Node root=doc.getFirstChild();

NodeList nodelist=root.getChildNodes();

//traverse or loop through DOM tree

**for**(**int** i=0;i<nodelist.getLength();i++){

node=nodelist.item(i);//node has the childs of root

**if**(node.getNodeType()==Node.***ELEMENT\_NODE***){//for nodes <Book>

Node ntemp=**null**;

**int** contador=1;

//get the attribute “published”

data[0]=node.getAttributes().item(0).getNodeValue();

//get the values of the sub-child <Title> <Author>

NodeList nl2=node.getChildNodes();//get the list of sub-child

**for**(**int** j=0;j<nl2.getLength();j++){//iterate over the sub-child

ntemp=nl2.item(j);

**if**(ntemp.getNodeType()==Node.***ELEMENT\_NODE***){

//to get the text of title and author, acces the child node and getNodeValue()

data[contador]=ntemp.getChildNodes().item(0).getNodeValue();

contador++;

}

}

//the String array data[] has now the values we needed

System.***out***.println(data[0]+"--"+data[2]+"--"+data[1]);

}//

}

}