PRACTICE 3: ADDING AND DELETING NODES FROM THE DOM TREE

ADDING A NODE:

Create the node from bottom to top, and append it to the father. Repeat it until you get to the root.

**public** **int** addBookToDOM(String title, String author,String year) {

**try**{

System.***out***.println("Adding book to the DOM tree:"+title+";"+author+";"+year);

//create the nodes=>append them to the father, from the leaf to the root

//CREATE TITLE with the text in the middle

Node ntitle=doc.createElement("Title");//creates labels <Title>...</Title>

Node ntitle\_text=doc.createTextNode(title);//creates the text node for the Title

ntitle.appendChild(ntitle\_text);//appends the title to the labels=><Title>title</Title>

//Node nauthor=doc.createElement("Author").appendChild(doc.createTextNode(author));//one line doesn't work

//CREATE AUTHOR

Node nauthor=doc.createElement("Author");

Node nauthor\_text=doc.createTextNode(author);

nauthor.appendChild(nauthor\_text);

//CREATE BOOK, WITH THE ATTRIBUTE, TITLE AND AUTHOR

Node nbook=doc.createElement("Book");

((Element)nbook).setAttribute("published",year);

nbook.appendChild(ntitle);

nbook.appendChild(nauthor);

//APPEND BOOK TO THE ROOT

Node raiz=doc.getFirstChild();//tb. doc.getChildNodes().item(0)

raiz.appendChild(nbook);

System.***out***.println("Book added.");

**return** 0;

}**catch**(Exception e){

System.***out***.println(e);

**return** -1;

}

}

DELETING A NODE:

We will delete the node knowing the title of the book. Traversing the DOM, find the node and delete it.

**public** **int** deleteNode(String tit) {

System.***out***.println("Looking for the book "+tit+" to delete it");

**try**{

//Node root=doc.getFirstChild();

Node root= doc.getDocumentElement();

NodeList nl1=doc.getElementsByTagName("Title");

Node n1;

**for**(**int** i=0;i<nl1.getLength();i++){

n1=nl1.item(i);

**if**(n1.getNodeType()==Node.***ELEMENT\_NODE***){

**if** (n1.getChildNodes().item(0).getNodeValue().equals(tit)){

System.***out***.println("Deleting the node <Book> with title "+tit);

//n1.getParentNode().removeChild(n1); //borra el nododeletes node <Title> tit </Title>, but leaves Book and Author

n1.getParentNode().getParentNode().removeChild(n1.getParentNode());

}

}

}

System.***out***.println("Node deleted");

//Keep the DOM tree in a new file (to keep our original file)

//saveDOMasFile("DeletedBooks.xml");

**return** 0;

}**catch**(Exception e){

System.***out***.println(e);

e.printStackTrace();

**return** -1;

}

}