**PROGRAMMING 3**

**Exercise 1**

“accounts.xml”

<account number="001">

<accountHolder>Rodrigo</accountHolder>

<accountType>Checking</accountType>

<currentBalance>5000</currentBalance>

</account>

<account number="002">

<accountHolder>Andrea</accountHolder>

<accountType>Savings</accountType>

<currentBalance>4000</currentBalance>

</account>

<account number="003">

<accountHolder>Andrew</accountHolder>

<accountType>Savings</accountType>

<currentBalance>3000</currentBalance>

</account>

<account number="004">

<accountHolder>Roxy</accountHolder>

<accountType>Checking</accountType>

<currentBalance>2000</currentBalance>

</account>

<account number="005">

<accountHolder>Fabio</accountHolder>

<accountType>Foreign Currency</accountType>

<currentBalance>1000</currentBalance>

</account>

</accounts>

**Exercise 2**

**import** java.io.IOException;

**import** javax.xml.parsers.DocumentBuilder;

**import** javax.xml.parsers.DocumentBuilderFactory;

**import** javax.xml.parsers.ParserConfigurationException;

**import** org.w3c.dom.Document;

**import** org.w3c.dom.Element;

**import** org.w3c.dom.Node;

**import** org.w3c.dom.NodeList;

**import** org.xml.sax.SAXException;

**public** **class** UsingDomAccount {

**public** **static** **void** main(String[] args) {

**try** {

DocumentBuilderFactory factory = DocumentBuilderFactory.*newInstance*();

DocumentBuilder builder = factory.newDocumentBuilder();

Document doc = builder.parse("accounts.xml");

doc.getDocumentElement().normalize();

Element root = doc.getDocumentElement();

System.***out***.println(root.getNodeName());

NodeList list = doc.getElementsByTagName("account");

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

Node node = list.item(i);

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

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

//print all employees

Element eElement = (Element)node;

System.***out***.println("Account Number: " + eElement.getAttribute("number"));

System.***out***.println("Account Holder: "+ eElement.getElementsByTagName("accountHolder").item(0).getTextContent());

System.***out***.println("Account Type: "+ eElement.getElementsByTagName("accountType").item(0).getTextContent());

System.***out***.println("Current Balance: "+ eElement.getElementsByTagName("currentBalance").item(0).getTextContent());

}

}

} **catch** (ParserConfigurationException | SAXException | IOException e) {

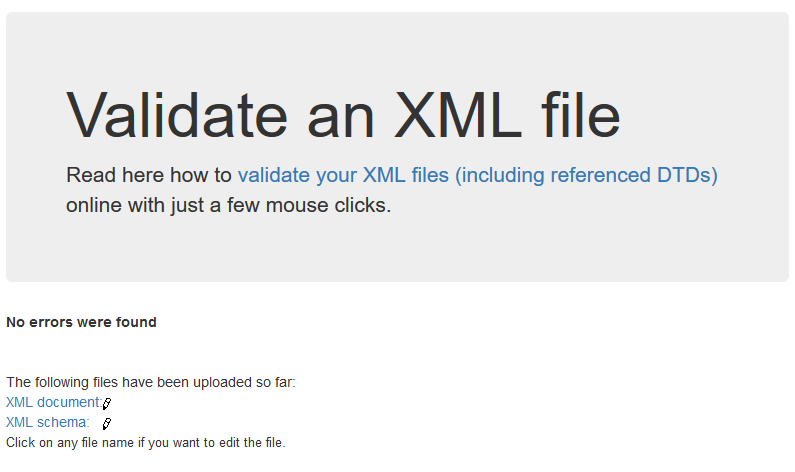
e.printStackTrace();

}

}

}

**Exercise 3**



<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="accounts">

<xs:complexType>

<xs:sequence>

<xs:element name="account" maxOccurs="unbounded" minOccurs="0">

<xs:complexType>

<xs:sequence>

<xs:element type="xs:string" name="accountHolder"/>

<xs:element type="xs:string" name="accountType"/>

<xs:element type="xs:double" name="currentBalance"/>

</xs:sequence>

<xs:attribute type="xs:byte" name="number" use="optional"/>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

<accounts>

<account number="001">

<accountHolder>Rodrigo</accountHolder>

<accountType>Checking</accountType>

<currentBalance>5000</currentBalance>

</account>

<account number="002">

<accountHolder>Andrea</accountHolder>

<accountType>Savings</accountType>

<currentBalance>4000</currentBalance>

</account>

<account number="003">

<accountHolder>Andrew</accountHolder>

<accountType>Savings</accountType>

<currentBalance>3000</currentBalance>

</account>

<account number="004">

<accountHolder>Roxy</accountHolder>

<accountType>Checking</accountType>

<currentBalance>2000</currentBalance>

</account>

<account number="005">

<accountHolder>Fabio</accountHolder>

<accountType>Foreign Currency</accountType>

<currentBalance>1000</currentBalance>

</account>

</accounts>