МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Кафедра информатики и систем управления

**Отчет по лабораторной работе №6**

по дисциплине

Управление данными

«Знакомство с XML»

РУКОВОДИТЕЛЬ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Балашова Т. И.

(подпись)

СТУДЕНТ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Рыкова А.В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

17-АС

Работа защищена «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_

С оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Нижний Новгород, 2020

**Цель лабораторной работы**:

Изучить работу с языком разметки XML.

**Задание**:

Часть1.

Составить свой корректный XML-документ, описывающий иерархическую базу данных.

Часть 2.

Ознакомится с языком запросов XPath. Написать пару запросов к своему XML документу созданном ранее.

**Выполнение работы:**

**Часть 1**

Описание документа:

**//library.dtd**

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<!DOCTYPE library

[

<!ELEMENT library (books,readers)>

<!ELEMENT books (book+)>

<!ELEMENT book (title,author,year )>

<!ATTLIST book id CDATA #REQUIRED>

<!ATTLIST book library\_name CDATA #REQUIRED>

<!ELEMENT title (#PCDATA)>

<!ELEMENT author (name\_author,country\_author)>

<!ELEMENT name\_author (#PCDATA)>

<!ELEMENT country\_author (#PCDATA)>

<!ELEMENT year (#PCDATA)>

<!ELEMENT readers (reader+)>

<!ELEMENT reader (first\_name,last\_name,telephone )>

<!ATTLIST reader number\_ticket CDATA #REQUIRED>

<!ELEMENT first\_name (#PCDATA)>

<!ELEMENT last\_name (#PCDATA)>

<!ELEMENT telephone (#PCDATA)>

]>

Корневым элементом является library, содержащий вложенные элементы books и readers.   
 Элемент books содержит вложенные элементы book. Элемент book содержит вложенные элементы title, author и year, а также имеет атрибуты id и library\_name. Элемент author имеет свои вложенные элементы name\_author и country\_author.

Элемент readers будет иметь вложенные элементы reader. Элемент reader будет иметь вложенные элементы first\_name, last\_name и telephone. Элемент reader содержит атрибут number\_ticket.

**//lab6.xml**

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<!DOCTYPE library

[

<!ELEMENT library (books,readers)>

<!ELEMENT books (book+)>

<!ELEMENT book (title,author,year )>

<!ATTLIST book id CDATA #REQUIRED>

<!ATTLIST book library\_name CDATA #REQUIRED>

<!ELEMENT title (#PCDATA)>

<!ELEMENT author (name\_author,country\_author)>

<!ELEMENT name\_author (#PCDATA)>

<!ELEMENT country\_author (#PCDATA)>

<!ELEMENT year (#PCDATA)>

<!ELEMENT readers (reader+)>

<!ELEMENT reader (first\_name,last\_name,telephone )>

<!ATTLIST reader number\_ticket CDATA #REQUIRED>

<!ELEMENT first\_name (#PCDATA)>

<!ELEMENT last\_name (#PCDATA)>

<!ELEMENT telephone (#PCDATA)>

]>

<library>

<books>

<book id="1" library\_name="Центральная">

<title>1984</title>

<author>

<name\_author>Оруэлл</name\_author>

<country\_author>Великобритания</country\_author>

</author>

<year>1949</year>

</book>

<book id="2" library\_name="Центральная">

<title>Детство</title>

<author>

<name\_author>Толстой</name\_author>

<country\_author>Россия</country\_author>

</author>

<year>1852</year>

</book>

<book id="3" library\_name="Центральная">

<title>Бесы</title>

<author>

<name\_author>Достоевский</name\_author>

<country\_author>Россия</country\_author>

</author>

<year>1871</year>

</book>

<book id="4" library\_name="Центральная">

<title>Вельд</title>

<author>

<name\_author>Брэдбери</name\_author>

<country\_author>США</country\_author>

</author>

<year>1950</year>

</book>

</books>

<readers>

<reader number\_ticket="173133">

<first\_name>Анастасия</first\_name>

<last\_name>Рыкова</last\_name>

<telephone>89159370044</telephone>

</reader>

<reader number\_ticket="173134">

<first\_name>Иван</first\_name>

<last\_name>Иванов</last\_name>

<telephone>89319677888</telephone>

</reader>

<reader number\_ticket="173135">

<first\_name>Петр</first\_name>

<last\_name>Петров</last\_name>

<telephone>89159374632</telephone>

</reader>

</readers>

</library>

**Часть 2**

**//Main.java**

package lab6;

import org.xml.sax.SAXException;

import javax.xml.parsers.DocumentBuilder;

import javax.xml.parsers.DocumentBuilderFactory;

import javax.xml.parsers.ParserConfigurationException;

import javax.xml.xpath.XPath;

import javax.xml.xpath.XPathFactory;

import javax.xml.xpath.XPathConstants;

import javax.xml.xpath.XPathExpression;

import javax.xml.xpath.XPathExpressionException;

import java.io.IOException;

import java.util.ArrayList;

import java.util.List;

import org.w3c.dom.Document;

import org.w3c.dom.NodeList;

public class Main {

public static void main(String[] args) {

DocumentBuilderFactory builderFactory = DocumentBuilderFactory.newInstance();

builderFactory.setNamespaceAware(true);

DocumentBuilder builder;

try {

builder = builderFactory.newDocumentBuilder();

Document doc = builder.parse("lab6.xml");

XPathFactory pathFactory = XPathFactory.newInstance();

XPath xpath = pathFactory.newXPath();

//Книга, у которой автор Оруэлл, Великобритания

List<String> author = getTitle(doc, xpath, "Оруэлл", "Великобритания");

System.out.println("Название книги: " + author.get(0));

//Нахождение номера телефона читателя по атрибуту номер билета

List<String> telephone = getPhone(doc, xpath, "173133");

System.out.println("Телефон читателя: " + telephone.get(0));

} catch (ParserConfigurationException | IOException | SAXException e) {

e.printStackTrace();

}

}

public static List<String> getTitle(Document doc, XPath xpath, String name\_author, String country\_author) {

List<String> list = new ArrayList<>();

try {

XPathExpression xExp = xpath.compile(String.format(

"//books/book[author[name\_author='" + name\_author + "' and country\_author = '" + country\_author + "']]/title/text()"));

NodeList books = (NodeList) xExp.evaluate(doc, XPathConstants.NODESET);

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

list.add(books.item(i).getTextContent());

}

} catch (XPathExpressionException e) {

e.printStackTrace();

}

return list;

}

public static List<String> getPhone(Document doc, XPath xpath, String number\_ticket) {

List<String> list = new ArrayList<>();

try {

XPathExpression xExp = xpath.compile(String.format(

"//readers/reader[@number\_ticket='" + number\_ticket + "']/telephone/text()"));

NodeList readers = (NodeList) xExp.evaluate(doc, XPathConstants.NODESET);

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

list.add(readers.item(i).getTextContent());

}

} catch (XPathExpressionException e) {

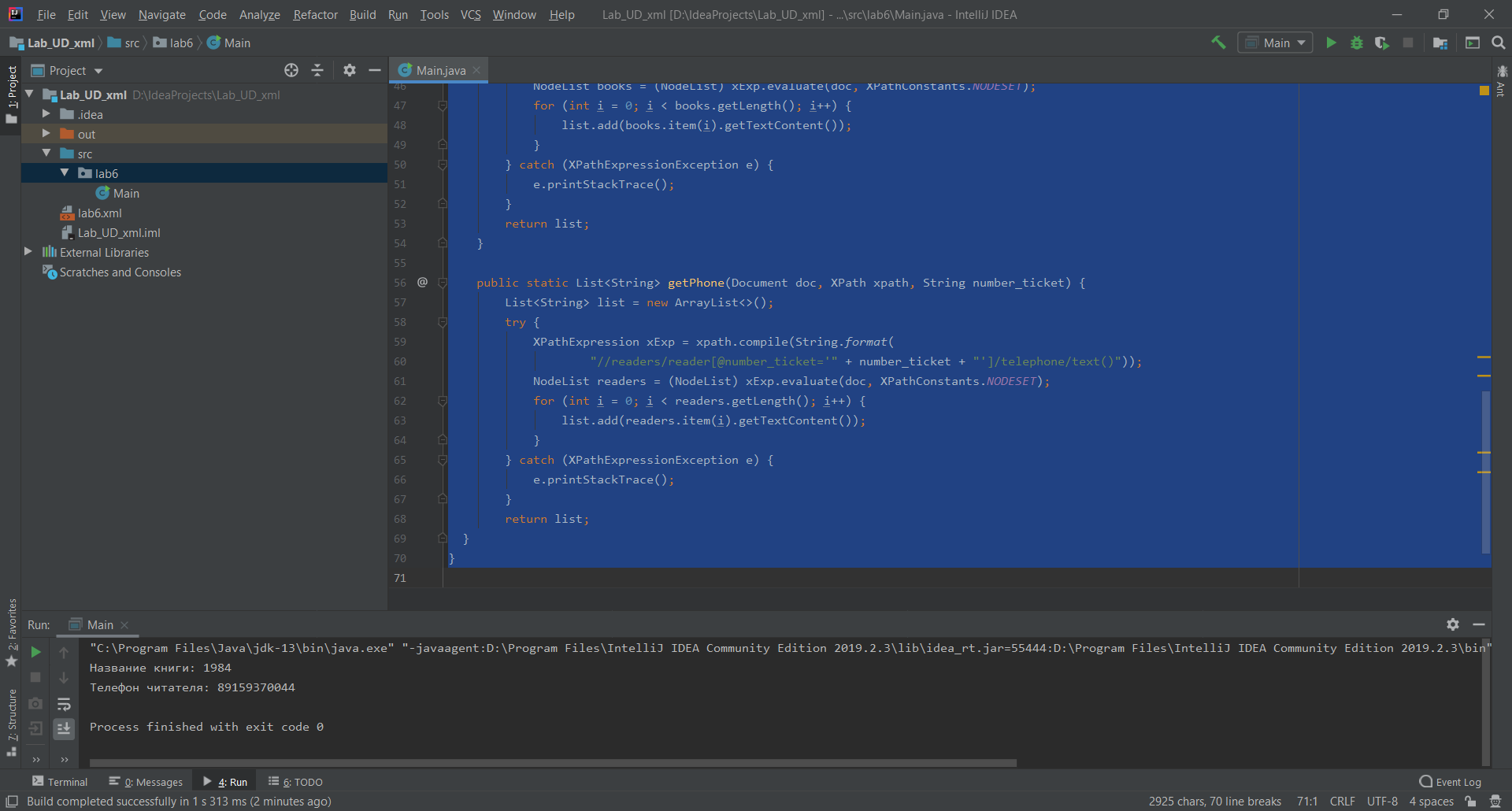
e.printStackTrace();

}

return list;

}

}

**Скриншот:**

**Вывод:**

В процессе выполнения лабораторной работы были изучены основы написания валидных xml-документов, а также получены базовые навыки в работе с языком запросов XPath.