# ЛАБОРАТОРНАЯ **РАБОТА №9**

## XML: ОСНОВЫ СОЗДАНИЯ

## И ОБРАБОТКИ СТРУКТУРИРОВАННЫХ ДАННЫХ

Цель работы: освоить принципы создания, валидации и обработки XMLдокументов, развить навыки анализа данных и работы с инструментами для обработки структурированных форматов

1. Создание XML-документа:

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

<library xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

         xsi:noNamespaceSchemaLocation="library.xsd">

    <book id="b001">

        <title>Война и мир</title>

        <author>Лев Толстой</author>

        <year>1869</year>

        <genre>Роман</genre>

        <price>25.99</price>

    </book>

    <book id="b002">

        <title>1984</title>

        <author>Джордж Оруэлл</author>

        <year>1949</year>

        <genre>Антиутопия</genre>

        <price>19.50</price>

    </book>

    <book id="b003">

        <title>Мастер и Маргарита</title>

        <author>Михаил Булгаков</author>

        <year>1967</year>

        <genre>Фантастика</genre>

        <price>22.75</price>

    </book>

</library>

2. Создание XSD-схемы:

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

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

    <xs:element name="library">

        <xs:complexType>

            <xs:sequence>

                <xs:element name="book" maxOccurs="unbounded">

                    <xs:complexType>

                        <xs:sequence>

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

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

                            <xs:element name="year" type="xs:integer"/>

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

                            <xs:element name="price" type="xs:decimal"/>

                        </xs:sequence>

                        <xs:attribute name="id" type="xs:ID" use="required"/>

                    </xs:complexType>

                </xs:element>

            </xs:sequence>

        </xs:complexType>

    </xs:element>

</xs:schema>

3. Валидация XML-документа и Обработка XML-документа:

import xml.etree.ElementTree as ET

from datetime import datetime

def validate\_xml(xml\_file, xsd\_file):

    """Валидация XML по XSD схеме (требуется установка lxml)"""

    try:

        from lxml import etree

        xmlschema = etree.XMLSchema(etree.parse(xsd\_file))

        xml\_doc = etree.parse(xml\_file)

        if xmlschema.validate(xml\_doc):

            print("XML документ валиден по XSD схеме.")

            return True

        else:

            print("XML документ НЕ валиден по XSD схеме:")

            print(xmlschema.error\_log)

            return False

    except ImportError:

        print("Для валидации требуется установить библиотеку lxml")

        return False

    except Exception as e:

        print(f"Ошибка при валидации: {e}")

        return False

def process\_library(xml\_file):

    """Обработка XML файла библиотеки"""

    try:

        tree = ET.parse(xml\_file)

        root = tree.getroot()

        # Вывод всех книг

        print("\nСписок всех книг в библиотеке:")

        total\_price = 0

        book\_count = 0

        for book in root.findall('book'):

            book\_id = book.get('id')

            title = book.find('title').text

            author = book.find('author').text

            year = book.find('year').text

            genre = book.find('genre').text

            price = float(book.find('price').text)

            print(f"\nID: {book\_id}")

            print(f"Название: {title}")

            print(f"Автор: {author}")

            print(f"Год: {year}")

            print(f"Жанр: {genre}")

            print(f"Цена: {price:.2f}")

            total\_price += price

            book\_count += 1

        # Вычисление средней цены

        if book\_count > 0:

            avg\_price = total\_price / book\_count

            print(f"\nСредняя цена книги: {avg\_price:.2f}")

        # Фильтрация книг

        filter\_genre = input("\nВведите жанр для фильтрации (или нажмите Enter чтобы пропустить): ")

        if filter\_genre:

            filtered\_books = [b for b in root.findall('book')

                            if b.find('genre').text.lower() == filter\_genre.lower()]

            print(f"\nНайдено {len(filtered\_books)} книг в жанре '{filter\_genre}':")

            for book in filtered\_books:

                print(f"- {book.find('title').text} ({book.find('year').text})")

        filter\_year = input("\nВведите год для фильтрации (или нажмите Enter чтобы пропустить): ")

        if filter\_year:

            try:

                year = int(filter\_year)

                filtered\_books = [b for b in root.findall('book')

                                if int(b.find('year').text) == year]

                print(f"\nНайдено {len(filtered\_books)} книг, опубликованных в {year} году:")

                for book in filtered\_books:

                    print(f"- {book.find('title').text} ({book.find('genre').text})")

            except ValueError:

                print("Ошибка: введите корректный год")

    except ET.ParseError as e:

        print(f"Ошибка при разборе XML: {e}")

    except FileNotFoundError:

        print(f"Файл {xml\_file} не найден")

    except Exception as e:

        print(f"Неизвестная ошибка: {e}")

if \_\_name\_\_ == "\_\_main\_\_":

    xml\_file = "library.xml"

    xsd\_file = "library.xsd"

    # Валидация XML

    validate\_xml(xml\_file, xsd\_file)

    # Обработка XML

    process\_library(xml\_file)

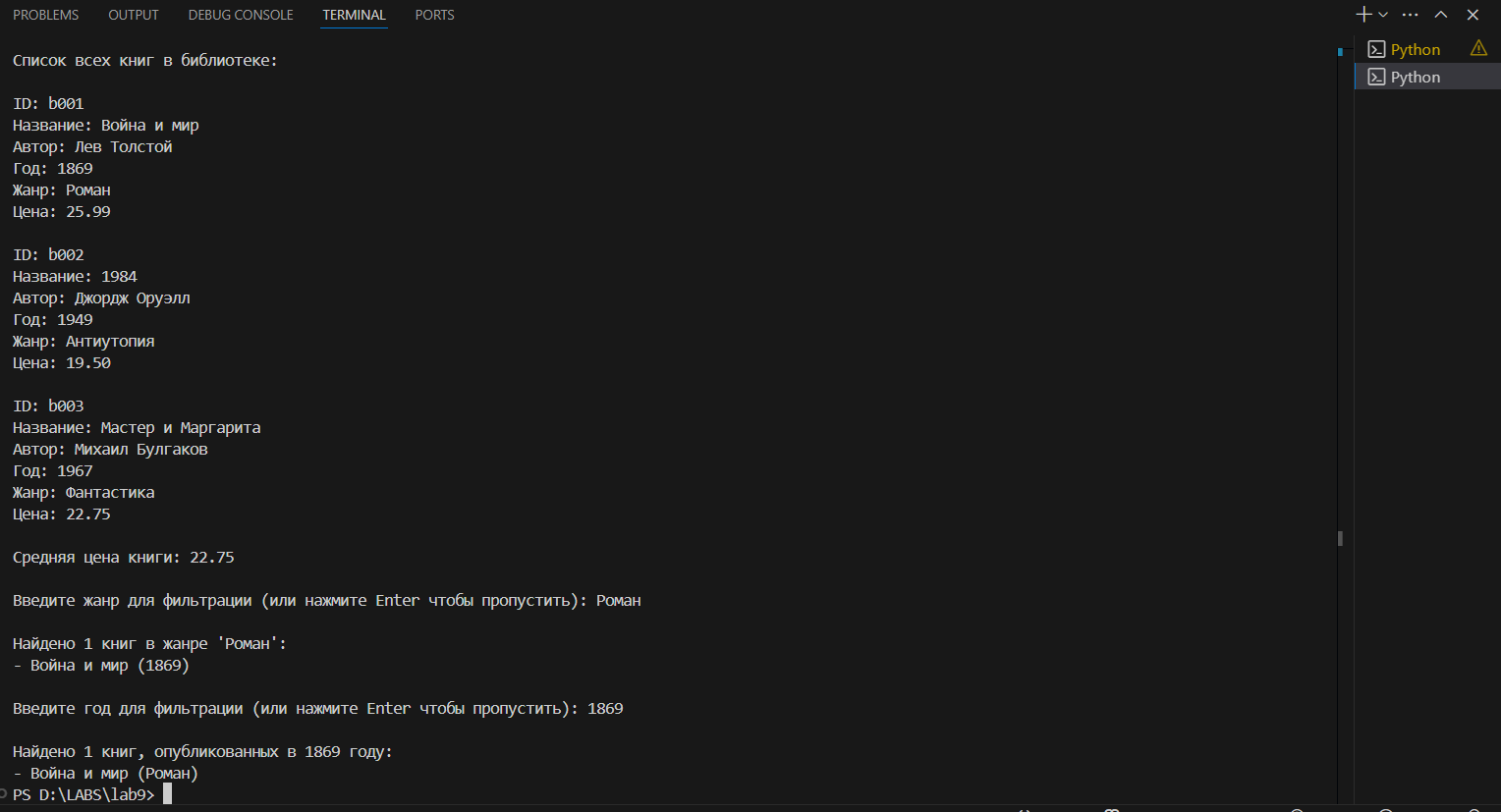


Рисунок 1 – Пример вывода программы

4. Исследование и сравнение

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

<rss version="2.0">

    <channel>

        <title>BBC News - World</title>

        <link>https://www.bbc.co.uk/news/world</link>

        <description>The latest stories from the World section of the BBC News website.</description>

        <language>en-gb</language>

        <lastBuildDate>Mon, 21 Feb 2022 12:00:00 GMT</lastBuildDate>

        <item>

            <title>Ukraine crisis: Russia orders troops into separatist regions</title>

            <link>https://www.bbc.co.uk/news/world-europe-60468237</link>

            <description>President Putin orders forces into two rebel-held areas of eastern Ukraine, after recognising them as independent.</description>

            <pubDate>Mon, 21 Feb 2022 11:45:00 GMT</pubDate>

            <guid isPermaLink="false">https://www.bbc.co.uk/news/world-europe-60468237</guid>

        </item>

        <item>

            <title>Canada protests: Police clear Ottawa bridge as crackdown continues</title>

            <link>https://www.bbc.co.uk/news/world-us-canada-60468238</link>

            <description>Police in Canada make dozens of arrests as they continue efforts to end protests against Covid restrictions.</description>

            <pubDate>Mon, 21 Feb 2022 10:30:00 GMT</pubDate>

            <guid isPermaLink="false">https://www.bbc.co.uk/news/world-us-canada-60468238</guid>

        </item>

    </channel>

</rss>

Rss новостная лента от BBC News.

<rss> - основной корневой элемент с указанием версии.

<channel> - содержит метаданные(название, ссылка, описание) и список новостных элементов

<item> - содержит заголовок, ссылку, описание и дату публикации

<guide> с атрибутом isPermaLink служит уникальным идентификатором