

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ»

Институт информатики, математики и робототехники  
Кафедра математического и компьютерного моделирования

**Лабораторная работа №6: Работа с форматом XML в Python**

**ОБУЧАЮЩЕГОСЯ**  
4 курса группы ПИ-4ИВТ221Б

Санникова Михаила Александровича

Уровень высшего образования:	высшее образование – бакалавриат
Направление подготовки (специальность)	<u>09.03.03 “Прикладная информатика”</u>
Направленность (профиль) программы	<u>Информационные и вычислительные технологии</u>
Дата выполнения	20.10.2025

## Постановка задачи:

Создать файл library.xml

Написать функцию для чтения файла

Добавить в файл library.xml:

- Новую книгу: id="3", title="Машинное обучение", author="Сидорова С.С.", year="2023", genre="Искусственный интеллект"
- Обновите год издания книги с id=2 на 2021
- Добавьте атрибут is\_available="true" для всех книг

Написать функцию, которая проверяет:

- Наличие обязательных элементов у каждой книги (title , author , year, genre)
- Корректность типов данных (year - число, id - число)
- Уникальность идентификаторов книг

Написать функцию, которая реализует поиск книг по различным критериям (по автору, по жанру, по году издания)

## Практическая часть:

Листинг кода с комментариями

```
import xml.etree.ElementTree as ET
import os
from functools import wraps

def main():
    filename = 'library.xml'

    # Задание 1: Создание XML-файла
    create_initial_xml(filename)

    # Задание 2: Чтение XML-файла
    print("=" * 60)
    print("ЧТЕНИЕ XML-ФАЙЛА")
    print("=" * 60)
    books_data = read_xml_file(filename)

    # Задание 3: Модификация XML-файла
    print("\n" + "=" * 60)
    print("МОДИФИКАЦИЯ XML-ФАЙЛА")
    print("=" * 60)
    modified_data = modify_xml_file(filename)

    # Задание 4: Валидация данных
    print("\n" + "=" * 60)
    print("ВАЛИДАЦИЯ ДАННЫХ")
    print("=" * 60)
    if is_data_valid(modified_data):
        print("✓ Все данные прошли валидацию успешно!")
    else:
        print("✗ Данные содержат ошибки!")

    # Чтение обновленного файла
    print("\n" + "=" * 60)
    print("ОБНОВЛЕННЫЕ ДАННЫЕ")
    print("=" * 60)
    updated_books = read_xml_file(filename)

    # Задание 5: Поиск данных
    print("\n" + "=" * 60)
    print("ПОИСК ДАННЫХ")
    print("=" * 60)

    # Поиск по автору
    author_books = search_books(updated_books, author="Петров П.П.")
    print(f"Книги автора 'Петров П.П.': {len(author_books)}")
    for book in author_books:
        print(f" - {book['title']} ({book['year']} год)")

    # Поиск по жанру
    genre_books = search_books(updated_books, genre="Программирование")
    print(f"\nКниги жанра 'Программирование': {len(genre_books)}")
    for book in genre_books:
        print(f" - {book['title']} - {book['author']}")
```

```

# Поиск по году
year_books = search_books(updated_books, min_year=2021)
print(f"\nКниги изданные с 2021 года: {len(year_books)}")
for book in year_books:
    print(f" - {book['title']} ({book['year']} год)")

def create_initial_xml(filename):
    """Задание 1: Создание начального XML-файла"""
    # Создание корневого элемента
    library = ET.Element('library')

    # Создание первой книги
    book1 = ET.SubElement(library, 'book')
    book1.set('id', '1')

    title1 = ET.SubElement(book1, 'title')
    title1.text = 'Python для начинающих'

    author1 = ET.SubElement(book1, 'author')
    author1.text = 'Иванов И.И.'

    year1 = ET.SubElement(book1, 'year')
    year1.text = '2022'

    genre1 = ET.SubElement(book1, 'genre')
    genre1.text = 'Программирование'

    # Создание второй книги
    book2 = ET.SubElement(library, 'book')
    book2.set('id', '2')

    title2 = ET.SubElement(book2, 'title')
    title2.text = 'Алгоритмы и структуры данных'

    author2 = ET.SubElement(book2, 'author')
    author2.text = 'Петров П.П.'

    year2 = ET.SubElement(book2, 'year')
    year2.text = '2020'

    genre2 = ET.SubElement(book2, 'genre')
    genre2.text = 'Компьютерные науки'

    # Создание XML-дерева и запись в файл
    tree = ET.ElementTree(library)
    tree.write(filename, encoding='utf-8', xml_declaration=True)

    print(f"✓ Создан файл {filename}")

def check_file_exists(func):
    @wraps(func)
    def wrapper(file_path):
        if not os.path.exists(file_path):
            print(f"Не существует файла с заданным путем: {file_path}")
            return None

```

```

    return func(file_path)

return wrapper

@check_file_exists
def read_xml_file(filename):
    """Задание 2: Чтение XML-файла"""
    try:
        tree = ET.parse(filename)
        root = tree.getroot()

        books = []
        print("Книги в библиотеке:")
        print("-" * 50)

        for book in root.findall('book'):
            book_id = book.get('id')
            title = book.find('title').text
            author = book.find('author').text
            year = int(book.find('year').text)
            genre = book.find('genre').text
            is_available = book.get('is_available', 'Не указан')

            book_data = {
                'id': book_id,
                'title': title,
                'author': author,
                'year': year,
                'genre': genre,
                'is_available': is_available
            }
            books.append(book_data)

            print(f"ID: {book_id}, Название: {title}")
            print(f"Автор: {author}, Год: {year}")
            print(f"Жанр: {genre}, Доступна: {is_available}")
            print("-" * 30)

        print(f"\nОбщее количество книг: {len(books)}")

        if books:
            newest_book = max(books, key=lambda x: x['year'])
            print(f"Самая новая книга: {newest_book['title']} ({newest_book['year']} год)")

        return books

    except Exception as e:
        print(f"Ошибка при чтении файла: {e}")
        return None

def modify_xml_file(filename):
    """Задание 3: Модификация XML-файла"""
    try:
        tree = ET.parse(filename)
        root = tree.getroot()

```

```

# Добавление новой книги
new_book = ET.SubElement(root, 'book')
new_book.set('id', '3')
new_book.set('is_available', 'true')

title = ET.SubElement(new_book, 'title')
title.text = 'Машинное обучение'

author = ET.SubElement(new_book, 'author')
author.text = 'Сидорова С.С.'

year = ET.SubElement(new_book, 'year')
year.text = '2023'

genre = ET.SubElement(new_book, 'genre')
genre.text = 'Искусственный интеллект'

print("✓ Добавлена новая книга: Машинное обучение")

# Обновление года издания книги с id=2
for book in root.findall('book'):
    if book.get('id') == '2':
        year_element = book.find('year')
        year_element.text = '2021'
        print("✓ Обновлен год издания книги 'Алгоритмы и структуры данных' на 2021")

# Добавление атрибута is_available="true" для всех книг
for book in root.findall('book'):
    if book.get('is_available') is None:
        book.set('is_available', 'true')

print("✓ Добавлен атрибут is_available для всех книг")

# Запись обновленных данных
tree.write(filename, encoding='utf-8', xml_declaration=True)
print("✓ Файл успешно обновлен!")

# Возвращаем обновленные данные
return read_xml_file(filename)

except Exception as e:
    print(f"Ошибка при модификации файла: {e}")
    return None

def is_data_valid(books):
    """Задание 4: Валидация данных"""
    if not books:
        return False

    errors = []
    book_ids = set()

    for book in books:
        # Проверка обязательных элементов

```

```

required_fields = ['title', 'author', 'year', 'genre']
for field in required_fields:
    if field not in book or not book[field]:
        errors.append(f"Книга ID {book['id']}: отсутствует обязательное поле '{field}'")

# Проверка уникальности идентификаторов
if book['id'] in book_ids:
    errors.append(f"Дублирующийся ID книги: {book['id']}")
book_ids.add(book['id'])

# Проверка корректности типов данных
if not book['id'].isdigit():
    errors.append(f"Книга '{book['title']}': ID должен быть числом")

if not isinstance(book['year'], int) or book['year'] <= 0:
    errors.append(f"Книга '{book['title']}': год должен быть положительным целым числом")

# Проверка атрибута is_available
if book.get('is_available') not in ['true', 'false']:
    errors.append(f"Книга '{book['title']}': неверное значение is_available")

# Вывод результатов валидации
if errors:
    print("Найдены ошибки валидации:")
    for error in errors:
        print(f"  ✗ {error}")
    return False
else:
    return True

def search_books(books, author=None, genre=None, min_year=None, max_year=None):
    """Задание 5: Поиск данных по различным критериям"""
    filtered_books = []

    for book in books:
        # Проверка критерия автора
        if author and book['author'] != author:
            continue

        # Проверка критерия жанра
        if genre and book['genre'] != genre:
            continue

        # Проверка критерия года
        if min_year and book['year'] < min_year:
            continue

        if max_year and book['year'] > max_year:
            continue

        filtered_books.append(book)

    return filtered_books

if __name__ == '__main__':

```

```
main()
```



## Скриншоты выполнения программы

```
✓ Создан файл library.xml
=====
ЧТЕНИЕ XML-ФАЙЛА
=====
Книги в библиотеке:
-----

ID: 1, Название: Python для начинающих
  Автор: Иванов И.И., Год: 2022
  Жанр: Программирование, Доступна: Не указан
-----

ID: 2, Название: Алгоритмы и структуры данных
  Автор: Петров П.П., Год: 2020
  Жанр: Компьютерные науки, Доступна: Не указан
-----

Общее количество книг: 2
Самая новая книга: Python для начинающих (2022 год)

=====
МОДИФИКАЦИЯ XML-ФАЙЛА
=====
✓ Добавлена новая книга: Машинное обучение
✓ Обновлен год издания книги 'Алгоритмы и структуры данных' на 2021
✓ Добавлен атрибут is_available для всех книг
✓ Файл успешно обновлен!
Книги в библиотеке:
-----

ID: 1, Название: Python для начинающих
  Автор: Иванов И.И., Год: 2022
  Жанр: Программирование, Доступна: true
-----

ID: 2, Название: Алгоритмы и структуры данных
  Автор: Петров П.П., Год: 2021
  Жанр: Компьютерные науки, Доступна: true
-----
```

```
ID: 3, Название: Машинное обучение
  Автор: Сидорова С.С., Год: 2023
  Жанр: Искусственный интеллект, Доступна: true
-----
```

```
Общее количество книг: 3
Самая новая книга: Машинное обучение (2023 год)
```

```
=====
ВАЛИДАЦИЯ ДАННЫХ
=====
```

```
✓ Все данные прошли валидацию успешно!
```

```
=====
ОБНОВЛЕННЫЕ ДАННЫЕ
=====
```

```
Книги в библиотеке:
-----
```

```
ID: 1, Название: Python для начинающих
  Автор: Иванов И.И., Год: 2022
  Жанр: Программирование, Доступна: true
-----
```

```
ID: 2, Название: Алгоритмы и структуры данных
  Автор: Петров П.П., Год: 2021
  Жанр: Компьютерные науки, Доступна: true
-----
```

```
ID: 3, Название: Машинное обучение
  Автор: Сидорова С.С., Год: 2023
  Жанр: Искусственный интеллект, Доступна: true
-----
```

```
Общее количество книг: 3
Самая новая книга: Машинное обучение (2023 год)
```

```
=====
ПОИСК ДАННЫХ
=====
```

```
Книги автора 'Петров П.П.': 1
  - Алгоритмы и структуры данных (2021 год)
```

```
Книги жанра 'Программирование': 1
  - Python для начинающих - Иванов И.И.
```

```
Книги изданные с 2021 года: 3
  - Python для начинающих (2022 год)
  - Алгоритмы и структуры данных (2021 год)
  - Машинное обучение (2023 год)
```