

Санкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики

Факультет инфокоммуникационных технологий

Направление подготовки 11.03.02

Лабораторная работа №5

«Работа с XML-схемами»

Выполнил:

Швалов Даниил Андреевич

Группа: К33211

Проверила:

Марченко Елена Вадимовна

Санкт-Петербург

2024

1. Введение

Цель работы: создание XML-схем для проверки различных XML-документов.

2. Ход работы

Упражнение №1. Создание простой XML-схемы

В данном упражнении необходимо создать XML-схему для lessons.xml из прошлых лабораторных работ.

На рисунке 1 представлена получившаяся XML-схема. В ней представлены в те же элементы и их поля, что были в DTD-схеме. Для полей элемента *lesson* использовались такие типы данных как *string* и *positiveInteger*.

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="lessons">
    <xs:complexType>
      <xs:sequence minOccurs="1" maxOccurs="unbounded">
        <xs:element name="lesson">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="CourseName" type="xs:string"/>
              <xs:element name="ProfessorSurname" type="xs:string"/>
              <xs:element name="Semester" type="xs:positiveInteger"/>
              <xs:element name="AcademicHours" type="xs:positiveInteger"/>
              <xs:element name="ControlType" type="xs:string"/>
              <xs:element name="Format" type="xs:string"/>
            </xs:sequence>
            <xs:attribute name="id" use="required" type="xs:ID"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Рисунок 1 – XML-схема для документа lessons.xml

Упражнение №2. Работа с простыми типами данных XML-схемы

В данном упражнении необходимо создать XML-схему для resume.xml, созданного в прошлых лабораторных работах. В данной XML-схеме необходимо, чтобы:

- номер телефона проверялся с помощью регулярного выражения и был вида ###-##-## (т.е. последовательность «три символа дефис два символа дефис два символа»);
- дата рождения могла быть не раньше 1 января 1947 года и не позже 1 января 1992 года;
- элемент «Семейное положение» ограничивался значениями «женат», «не женат», «замужем», «не замужем»;
- элемент «Образование» ограничивался значениями «высшее», «среднее»;
- возраст ограничивался значениями от 20 до 65 лет.

Согласно упражнению, для номера телефона, даты рождения, семейного положения, образования и возраста были созданы собственные типы, проверяющие корректность данных (рисунок 2). Для номера телефона используется регулярное выражение, которое проверяет, что номер соответствует формату «###-##-##». Для даты рождения используется стандартный тип *date* с ограничениями по минимальному и максимальному значению. Для семейного положения и образования используется тип-перечисление. Для возраста был создан тип, который принимает неотрицательное число и проверяет, что оно находится в заданном диапазоне.

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="MoneyType">
    <xs:simpleContent>
      <xs:extension base="xs:decimal">
        <xs:attribute name="currency" use="required" type="xs:string"/>
      </xs:extension>
    </xs:simpleContent>
```

```

</xs:complexType>
<xs:simpleType name="PhoneType">
  <xs:restriction base="xs:string">
    <xs:pattern value="\d{3}\-\d{2}\-\d{2}" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="BirthDateType">
  <xs:restriction base="xs:date">
    <xs:minInclusive value="1947-01-01" />
    <xs:maxInclusive value="1992-01-01" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="AgeType">
  <xs:restriction base="xs:nonNegativeInteger">
    <xs:minInclusive value="20" />
    <xs:maxInclusive value="65" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="FamilyStatusType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="женат" />
    <xs:enumeration value="не женат" />
    <xs:enumeration value="замужем" />
    <xs:enumeration value="не замужем" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="EducationType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="высшее" />
    <xs:enumeration value="среднее" />
  </xs:restriction>
</xs:simpleType>
<xs:element name="resume">
  <xs:complexType>
    <xs:sequence minOccurs="1" maxOccurs="unbounded">
      <xs:element name="candidate">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="Name" type="xs:string" />
            <xs:element name="LastName" type="xs:string" />

```

```

<xs:element name="PatronymicName" type="xs:string"/>
<xs:element name="Age" type="AgeType"/>
<xs:element name="WorkPlace" type="xs:string"/>
<xs:element name="Post" type="xs:string"/>
<xs:element name="DateOfBirth" type="BirthDateType"/>
<xs:element name="Education" type="EducationType"/>
<xs:element name="Address" type="xs:string"/>
<xs:element name="Phone" type="PhoneType"/>
<xs:element name="FamilyStatus" type="FamilyStatusType"/>
<xs:element name="DesiredSalary" type="MoneyType"/>
</xs:sequence>
<xs:attribute name="id" use="required" type="xs:ID"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

Рисунок 2 – XML-схема для документа resume.xml

Упражнение №3. Работа с XML-схемой

В данном упражнении необходимо создать XML-схему для документа, созданного в третьем упражнении прошлых лабораторных работ.

В XML-схеме, представленной на рисунке 3, были созданы отдельные типы данных для ISBN, языка, возрастного ограничения. Так, например, для ISBN производится проверка соответствия введенного значения формату «###-#-##-#####-#» с помощью регулярного выражения. Для проверки корректности языка и возрастного ограничения используется перечисление. Также были созданы типы для всех остальных элементов, которые раньше были представлены в DTD-схеме.

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="isbn">
    <xs:restriction base="xs:string">
      <xs:pattern value="\d{3}\-\d{1}\-\d{2}\-\d{6}\-\d{1}"/>
    </xs:restriction>
  </xs:simpleType>

```

```

    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="language">
  <xs:restriction base="xs:string">
    <xs:enumeration value="RU"/>
    <xs:enumeration value="EN"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="age_limit">
  <xs:restriction base="xs:nonNegativeInteger">
    <xs:enumeration value="0"/>
    <xs:enumeration value="6"/>
    <xs:enumeration value="12"/>
    <xs:enumeration value="16"/>
    <xs:enumeration value="18"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="authors">
  <xs:sequence minOccurs="1" maxOccurs="unbounded">
    <xs:element name="author" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="genres">
  <xs:sequence minOccurs="1" maxOccurs="unbounded">
    <xs:element name="genre" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="book">
  <xs:sequence>
    <xs:element name="title" type="xs:string"/>
    <xs:element name="summary" type="xs:string"/>
    <xs:element name="authors" type="authors"/>
    <xs:element name="genres" type="genres"/>
    <xs:element name="pages" type="xs:positiveInteger"/>
    <xs:element name="language" type="language"/>
    <xs:element name="publisher" type="xs:string"/>
    <xs:element name="publication_year" type="xs:nonNegativeInteger"/>
    <xs:element name="isbn" type="isbn"/>
    <xs:element name="age_limit" type="age_limit"/>
  </xs:sequence>

```

```
    <xs:attribute name="id" use="required" type="xs:ID"/>
</xs:complexType>
<xs:element name="books">
  <xs:complexType>
    <xs:sequence minOccurs="1" maxOccurs="unbounded">
      <xs:element name="book" type="book"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```

Рисунок 3 – XML-схема для документа books.xml

3. Вывод

В ходе выполнения лабораторной работы были созданы XML-схемы для проверки корректности XML-документов.