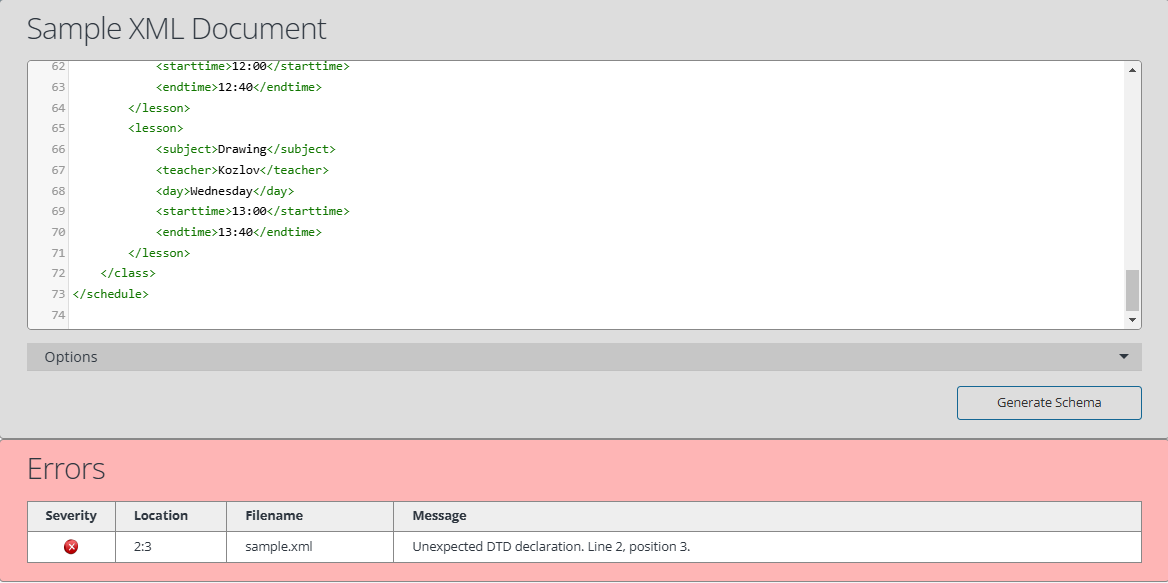
# Практическая работа №3: работа с XML-схемами

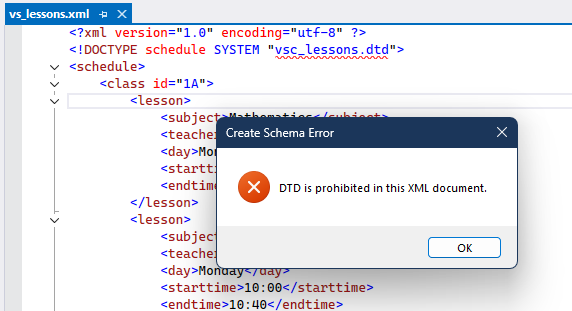
Упражнение №1**: создание простой XML-схемы**

* Откройте документ **lessons.xml**.
* Удалите из документа указание на .dtd файл.
* Реализуйте XML-схему для этого документа и сохраните схему в файле **lessons\_XML.xml**.
* Подключите созданную XML-схему к документу **lessons.xml** и проверьте соответствие документа **lessons.xml** созданной XML-схеме.

Online XML to XSD Converter



VS



ну ладно, удалим entity



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

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

<!-- schema -->

</xs:schema>

attributeFormDefault="unqualified"

elementFormDefault="qualified"

xs:element

<schedule><xs:complexType>

<xs:sequence>

<class>

<lesson>

subject

teacher

day

starttime

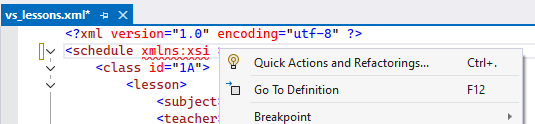
endtime

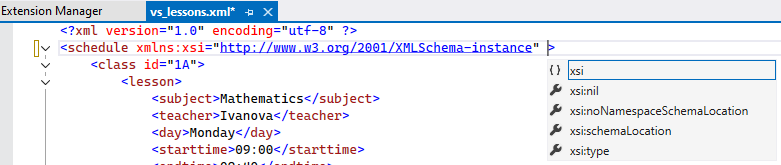
maxOccurs="unbounded"

xs:attribute

type="xs:string" пусть будет для всех

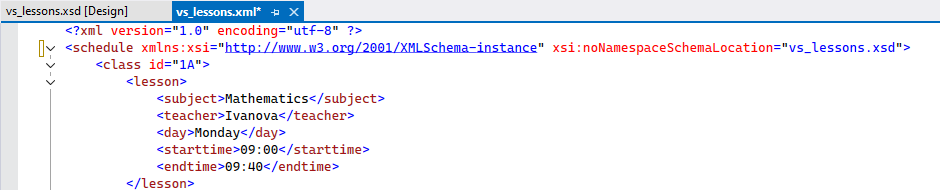
как подключить??? 🤬

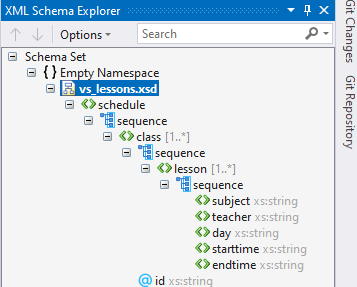


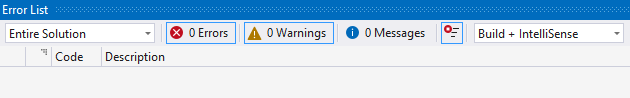


xsi:noNamespaceSchemaLocation

ну вот и дизайнер







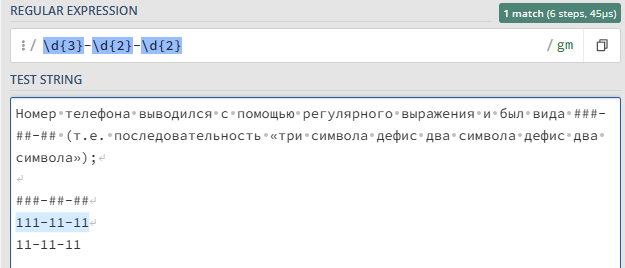
все работает

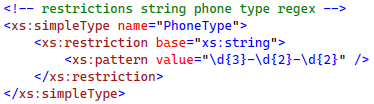
Упражнение №2**: работа с простыми типами данных XML-схемы**

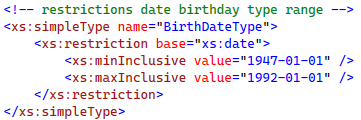
1. Откройте документ **resume.xml**.
2. Реализуйте XML-схему для данного документа и сохраните схему в файле **resume\_XML.xml.**
3. Подключите созданную XML-схему к документу **resume.xml** и проверьте его валидность.



1. Расширьте следующие простые типы данных в текущем в xml-документе так, чтобы:
   1. Номер телефона выводился с помощью регулярного выражения и был вида ###-##-## (т.е. последовательность *«три символа дефис два символа дефис два символа»*);
   2. Дата рождения могла быть не раньше 1 января 1947 года и не позже 1 января 1992 года;
   3. Элемент «Семейное положение» ограничивался значениями «женат», «не женат», «замужем», «не замужем»;
   4. Элемент «Образование» органичивался значениями «высшее», «среднее».
   5. На базе простого типа данных «integer» создайте пользовательский тип данных «AgeType», ограничив его значениями от 20 до 65 лет и используйте этот тип для определения типа данных элемента с тэгом «Age».

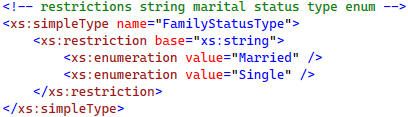


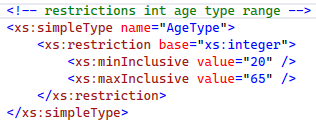




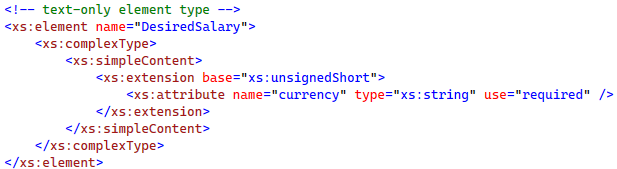
«женат», «не женат», «замужем», «не замужем»

сорри, только два



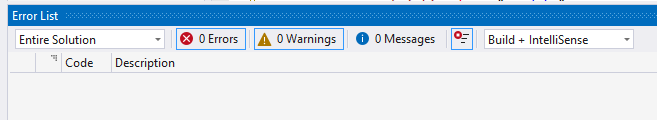


составной тип only-text

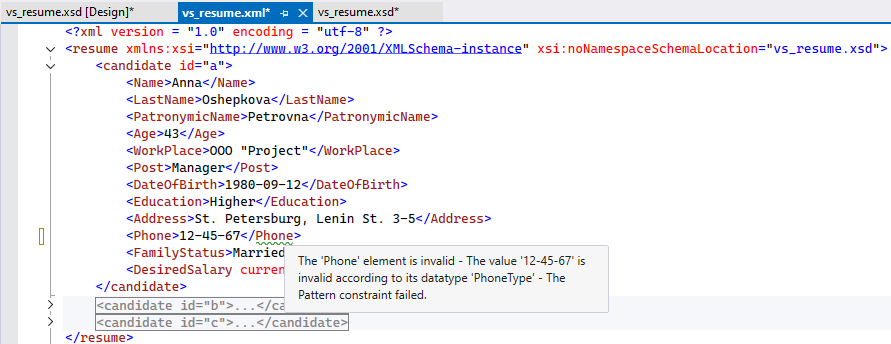


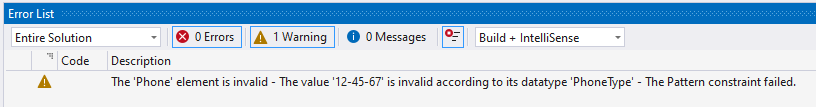
1. Проверьте корректность работы схемы данных, сохраните изменения.

VS

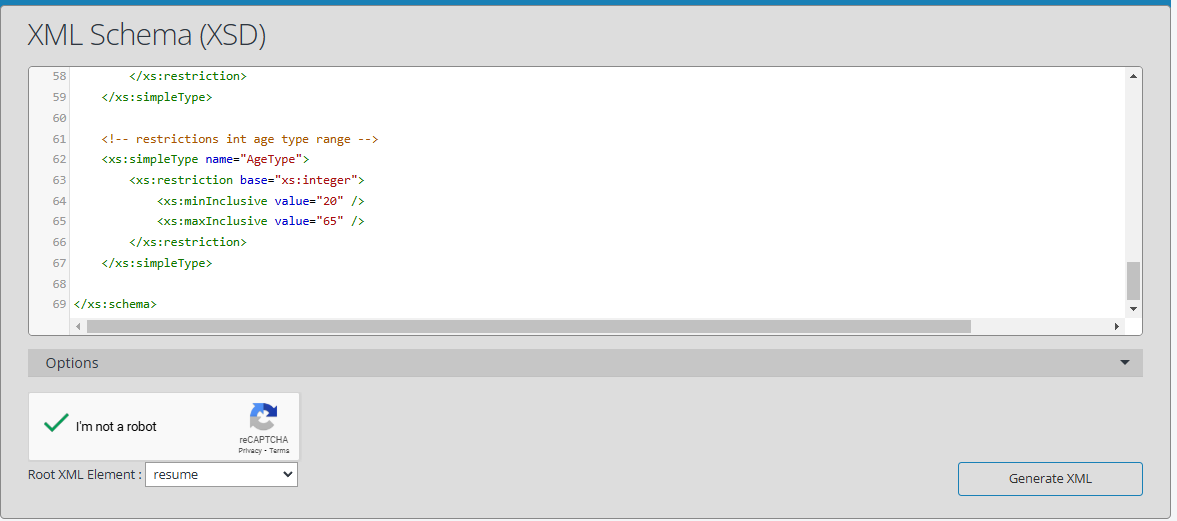


поломать

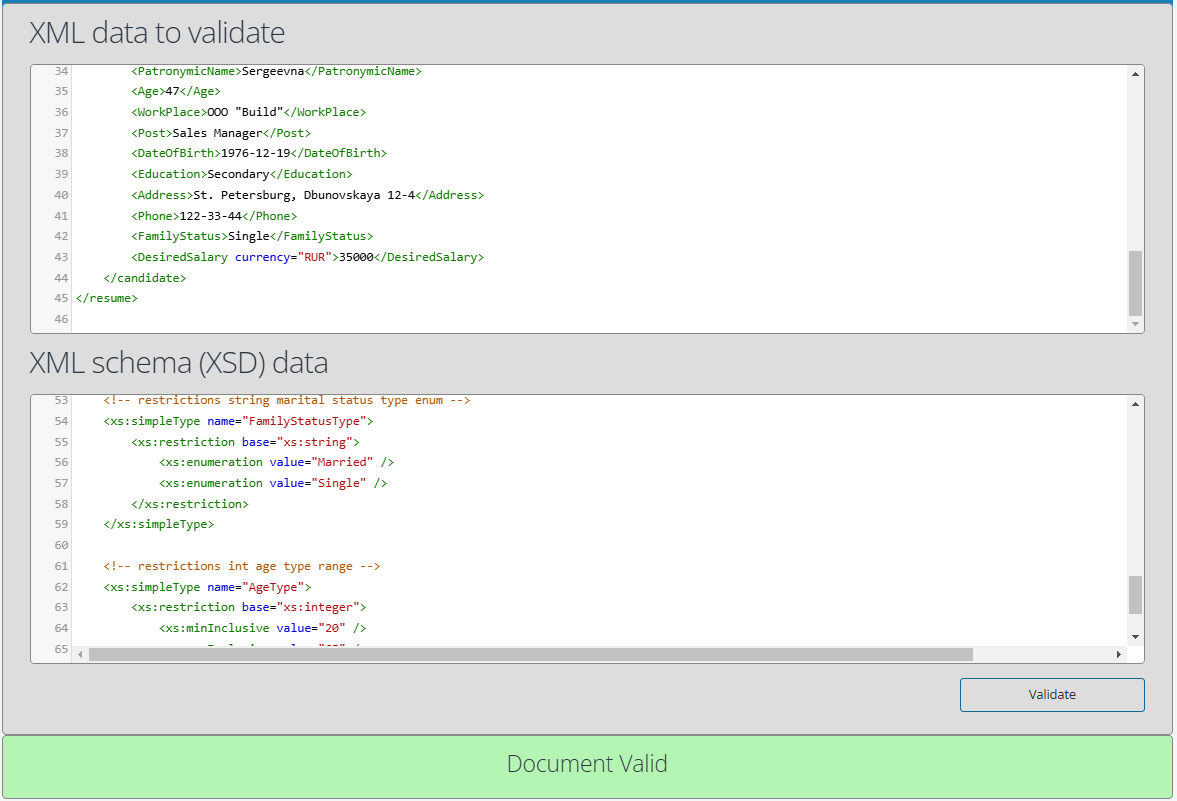




Online







*Упражнение №3****: работа с комплексными типами данных XML-схемы\****

1. Скопируйте данные из документа **resume.xml** в новый файл и сохраните его под именем **resume\_complex.xml.**
2. Измените новый документ таким образом, чтобы он сдержал отельные данные о резюме людей с высшим образованием и отдельные данные о резюме людей со средним образованием, например:

<resume>

<HigherEducation>

<Candidate>

…

</Candidate>

</HigherEducation>

<SecondaryEducation>

…

</SecondaryEducation>

1. И у лиц с высшим образованием, и у лиц со средним образованием одни и те же элементы данных. Для того, чтобы в xml-документе их не описывать несколько раз, создайте XML-схему с произвольным именем, где реализуйте пользовательский тип данных, а затем опишите обе группы кандидатов на должность с помощью этого типа данных.
2. Проверьте корректность работы схемы данных и валидность xml-документа и сохраните изменения.

общий пользовательский тип данных

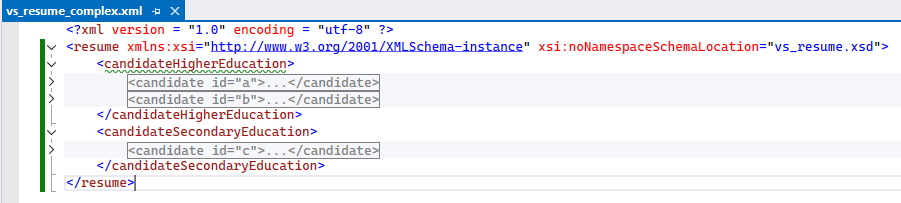
группа кандидатов с высшим образованием

candidateHigherEducation

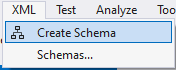
группа кандидатов со средним образованием

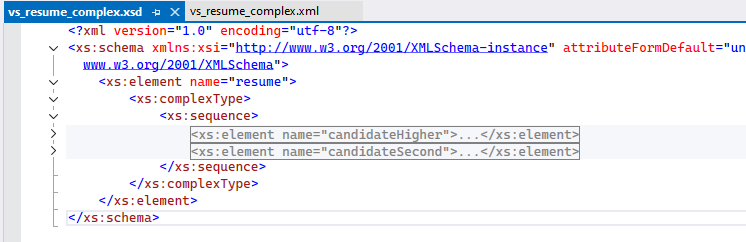
SecondaryEducation

те же данные, новая структура



новая схема

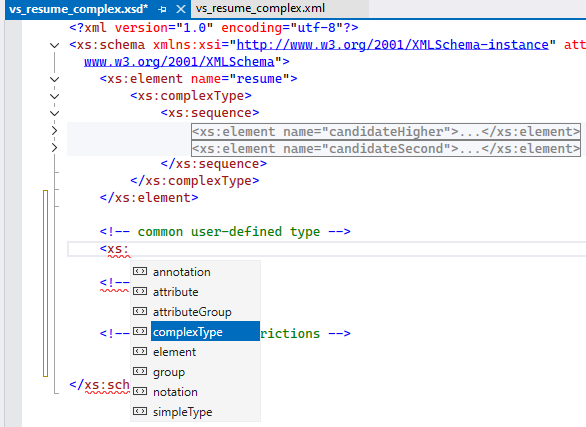






ну вот он создал и двойное описание и без ограничений и расширений

разделим схему на блоки, опишем тип, и восстановим ограничения



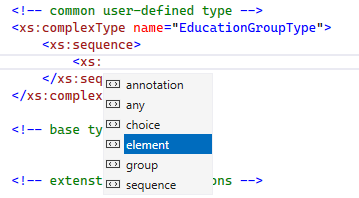


схема валидная

