# Практическая работа №4: Введение в XSLT и работа с XPath выражениями

Упражнение №1**: работа с XPath выражениями**

1. В Notepad++ создайте документ с именем **less\_template\_2.xsl.**
2. Напишите в корневом шаблоне только что созданного документа XPath-выражение, выводящее на экран второе занятие из описанных в документе **lesson.xml.**
3. Напишите в корневом шаблоне только что созданного документа XPath-выражение, выводящее на экран предпоследнее занятие из описанных в документе **lesson.xml.**
4. Напишите в корневом шаблоне только что созданного документа XPath-выражение, выводящее на экран общее количество занятий, описанных в документе **lesson.xml.**
5. Напишите в корневом шаблоне только что созданного документа XPath-выражение, выводящее на экран дату начала третьего курса из курсов, описанных в документе **lesson.xml.**
6. Проверьте корректность применения таблиц, сохраните изменения в файле **les\_template\_2.xsl.**

как вариант можно ВАЛИДИРОВАТЬ Xpath ONLINE

применяются ко всем элементам «lesson»

применяются ко элементу class[@id='1B']

функция count применяется ко всем элементам «lesson»

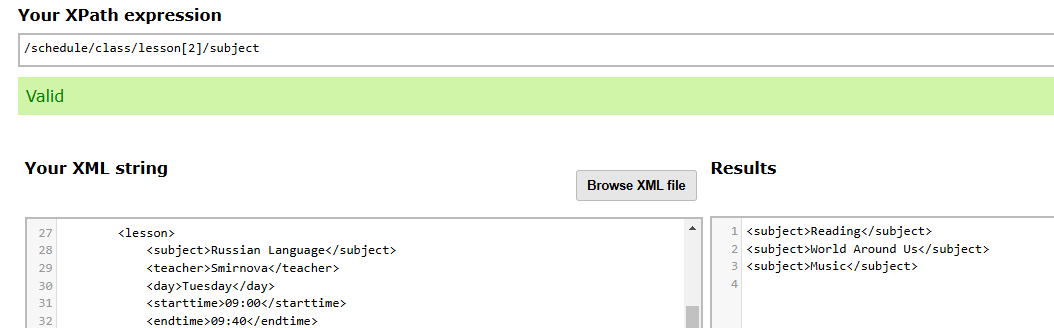
сделали XSLT и далее делаем трансформацию несколькими вариантами в разных средах

вариант 1

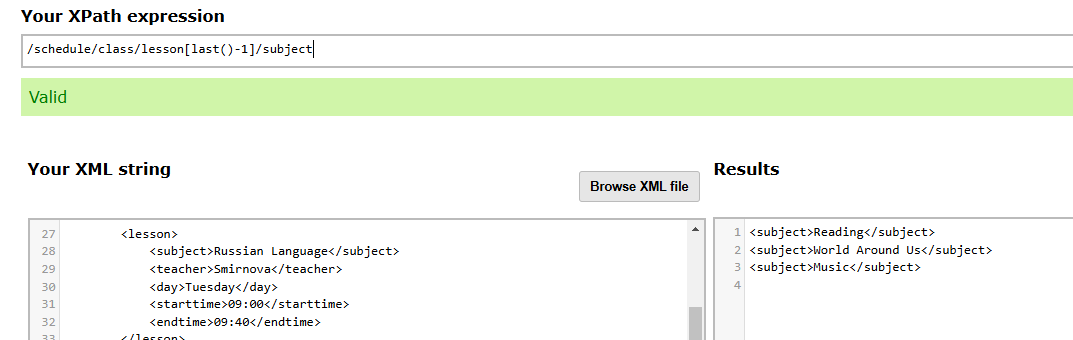
Online validator/tester Xpath

<https://extendsclass.com/xpath-tester.html>

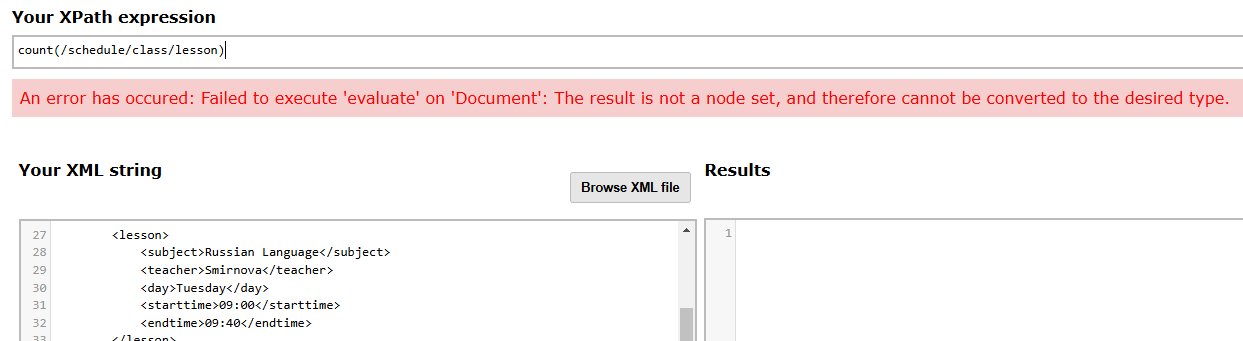
select="/schedule/class/lesson[2]/subject"



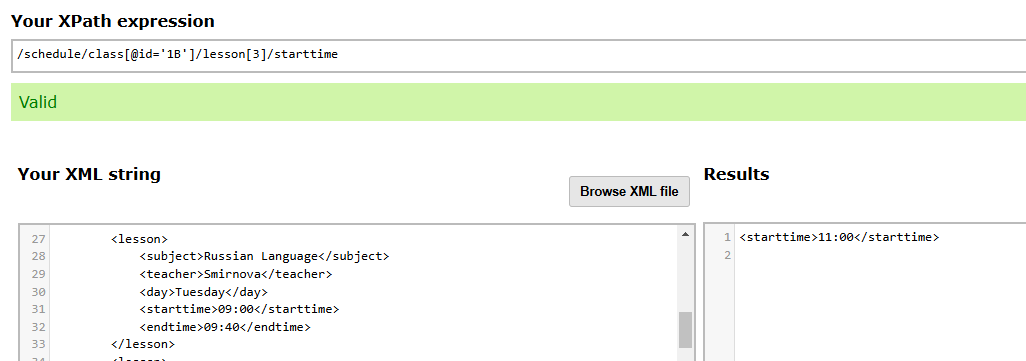
select="/schedule/class/lesson[last()-1]/subject"



select="count(/schedule/class/lesson)"

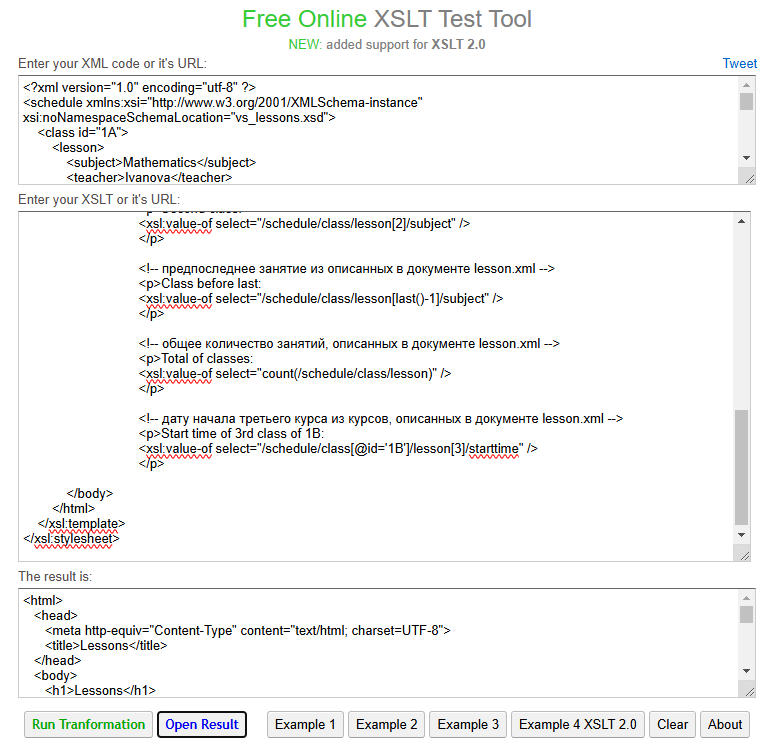


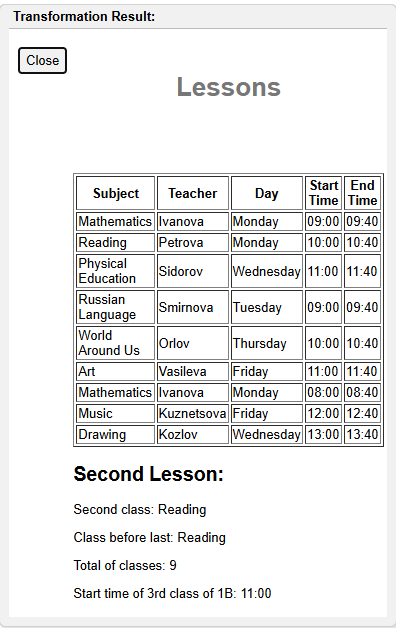
select="/schedule/class[@id='1B']/lesson[3]/starttime"



Online XSLT tester

<https://xslttest.appspot.com/>





xpath exp работает и трансформация тоже

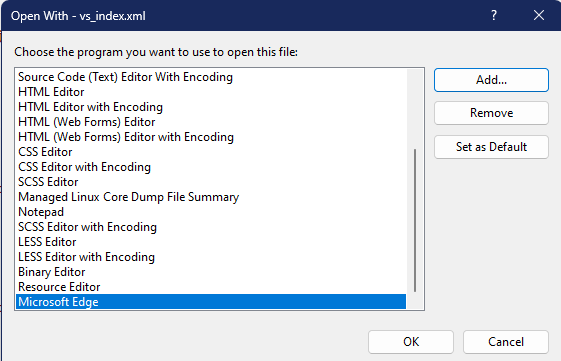
вариант 2

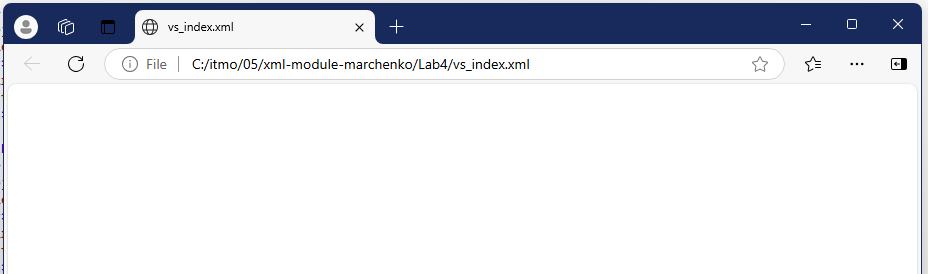
Visual Studio

здесь все то же самое, только файлы называются «vs\_имя файла»

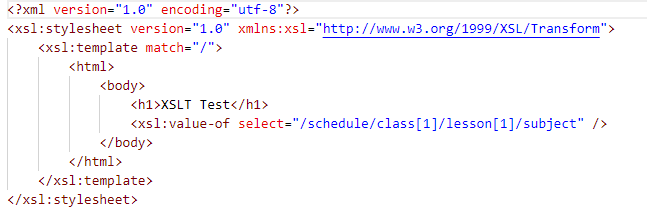
пытаемся сделать и увидеть трансформацию автоматически

Solution Explorer > Open With > Web Browser





не работает даже с тестовым шаблоном

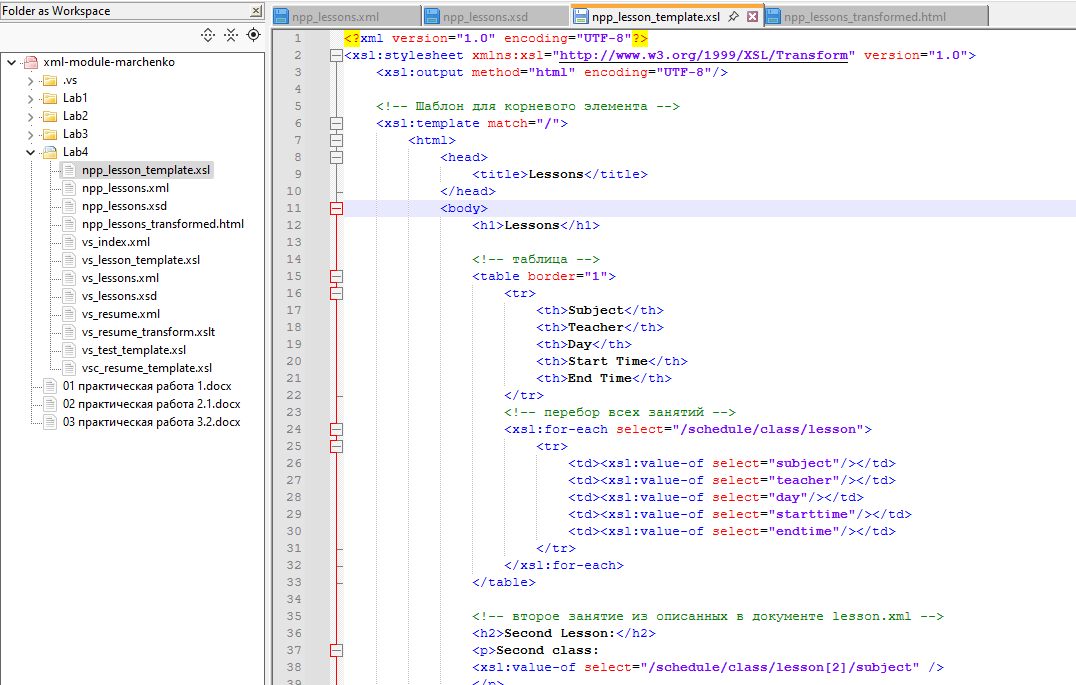


убрал файлы «vs\_имя файла» из проекта

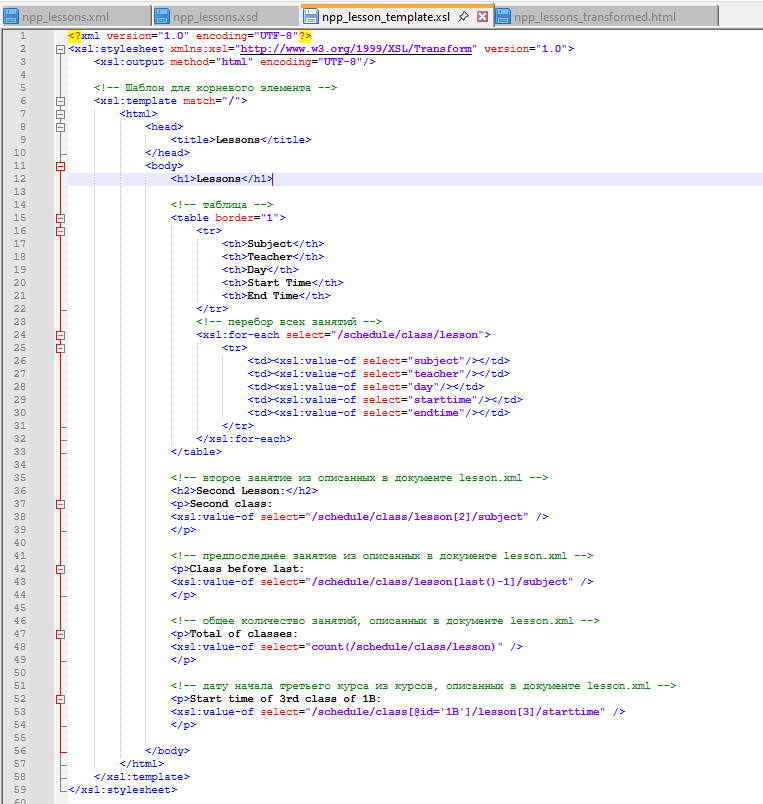
вариант 3

Npp

добавляем файлы данных и схемы в проект



создаем файл таблицы трансформации с правилами



он должен быть валидным XML и валидным по XSD и внутри HTML тоже должен быть валидный

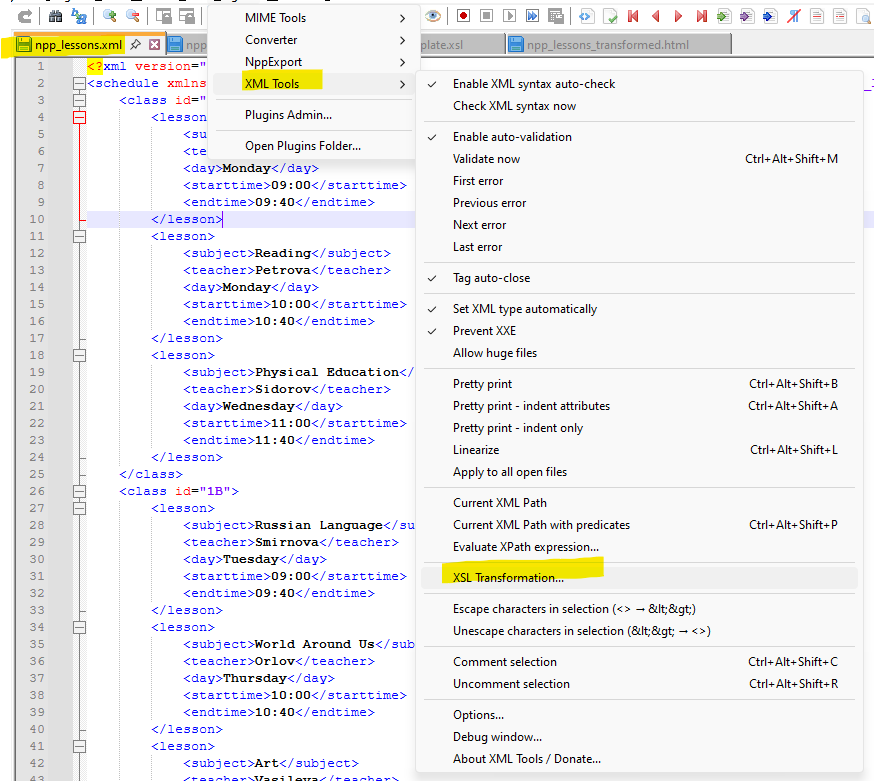
Online XML Validator (Well formed)

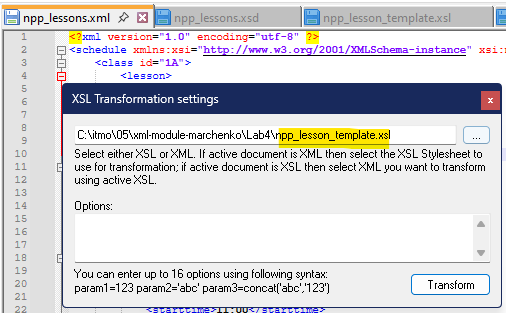
<https://www.liquid-technologies.com/online-xml-validator>



таблица является валидным XML файлом

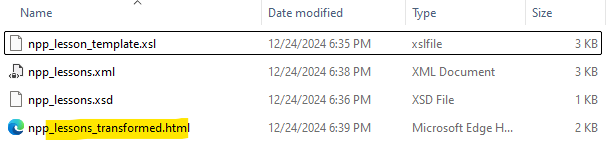
далее запускаем трансформацию





она создает HTML документ





открываем в браузере



трансформация работает

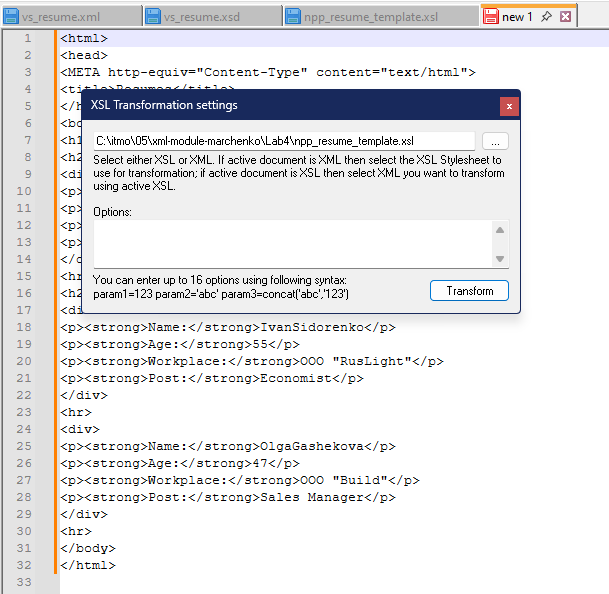
Упражнение №2**: XPath выражения**

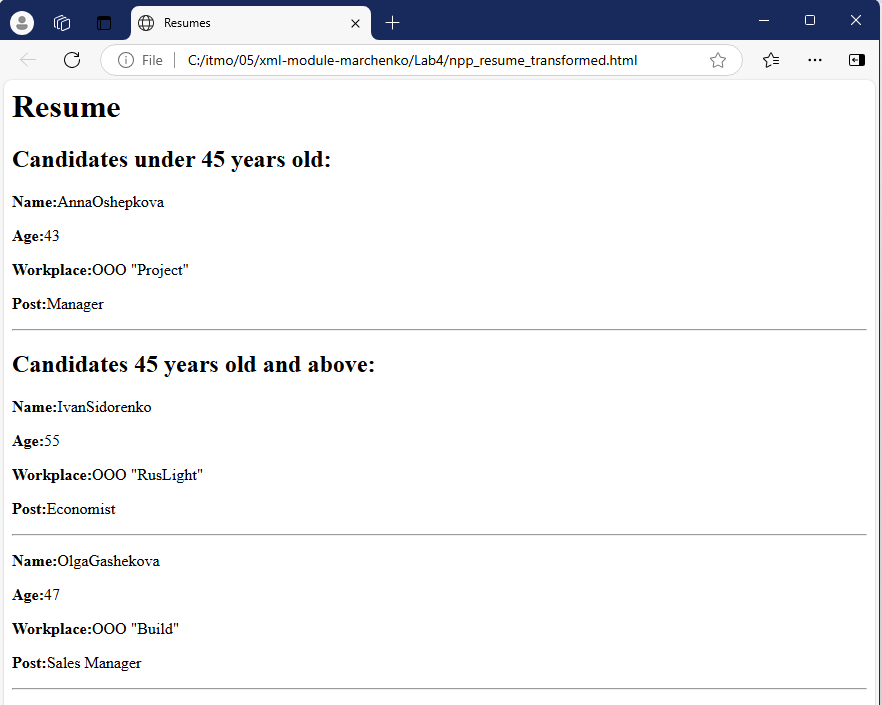
1. В Notepad++ создайте документ с именем **resume\_template.xsl.**
2. Напишите в шаблоне документе XPath-выражение, которое выводило бы на экран данные тех кандидатов, чей возраст меньше 32 лет.
3. Напишите в шаблоне документе XPath-выражение, которое выводило бы на экран данные тех кандидатов, чей возраст больше или равен 33 годам.
4. Проверьте корректность применения таблиц, сохраните изменения в файле **resume\_template.xsl.**

select="/resume/candidate[Age &lt; 32]"

select="/resume/candidate[Age &gt;= 33]"

сделали XSLT и делаем трансформацию





работает