

<http://moodle.openfmi.net/course/view.php?id=665>

<http://tinyurl.com/xml-is-2013>

XML валидиране с DTD

XML програмиране (ИС)
зимен семестър, 2013/2014 г.

Василен Дончев, ФМИ, СУ
vdonchev@uni-sofia.bg
30.10.2013

Съдържание

- Валидиране
- DTD
- Включване на DTD
- DOCTYPE
- DTD декларации (елемент, атрибут, entity)
- Работа с <oXygen/> XML Editor

Валидиране

- проверка, дали структурата и елементите на XML документ спазват описани условия
- XML документ, който е добре конструиран и следва точно определена структура и описание
- технологии за валидиране:
 - DTD (Document Type Definition)
 - XML Schemas
 - RELAX NG
 - Schematron

DTD

- описва **структурата на документа (шаблон за маркиране)**:
 - **имена** на елементи и техни атрибути;
 - **типовете** на елементите и техните атрибути;
 - **йерархията** в документа;
 - използвани **entities** в документа
- **нестриктно валидиране** (не достатъчно строго)
- недостатъци
 - различен от XML синтаксис;
 - не може да се дефинира точен брой на повторение на елементи;
 - фиксиран модел;
 - не поддържа области на данни;
 - **не поддържа достатъчно типове от данни (числа, дати и т.н.), регулярни изрази (RegEx)**
- но:
 - по-стегнат и кратък синтаксис (в сравнение с XML Schema)

Счита за остаряла технология (препоръчват се XML Schemas)

Пример за DTD

```
<!ELEMENT movies (movie+)>
<!ELEMENT movie (title, studio, producer, stars)>
<!ATTLIST movie
  in_color (Yes|No) #IMPLIED
  length CDATA #IMPLIED
  year CDATA #REQUIRED>
<!ELEMENT studio (title,address)>
<!ATTLIST studio presc NMTOKEN #REQUIRED>
<!ELEMENT producer (name,address,net_worth)>
<!ATTLIST producer cert NMTOKEN #REQUIRED>
<!ELEMENT stars (star+)>
<!ELEMENT net_worth (#PCDATA)>
<!ELEMENT star (name,address)>
<!ATTLIST star
  birth_date NMTOKEN #REQUIRED
  gender (M|F) #REQUIRED>
<!ELEMENT title (#PCDATA)>
<!ELEMENT address (#PCDATA)>
<!ELEMENT name (#PCDATA)>
```

movies.dtd

- дефинира ограничения за употребата на тагове и атрибути
- частично ограничава текстовото съдържание

movies.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE movies SYSTEM "movies.dtd">
<movies>
  <movie year="1995" length="106" in_color="Yes">
    <title>The Usual Suspects</title>
    <studio presc="5">
      <title>MGM</title>
      <address>MGM Boulevard</address>
    </studio>
    <producer cert="199">
      <name>Merv Griffin</name>
      <address>Riot Rd.</address>
      <net_worth>112000000</net_worth>
    </producer>
    <stars>
      <star gender="M" birth_date="1949-05-05">
        <name>Jack Nicholson</name>
        <address>X path</address>
      </star>
      <star gender="F" birth_date="1948-12-05">
        <name>Sandra Bullock</name>
        <address>X path</address>
      </star>
    </stars>
  </movie>
</movies>
```

Включване на DTD

- в самия XML документ (Internal DTD) – в DOCTYPE дефиниция

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE name [
  <!ELEMENT name (first, middle, last)>
  <!ELEMENT first (#PCDATA)>
  <!ELEMENT middle (#PCDATA)>
  <!ELEMENT last (#PCDATA)>
]>
<name>
  <first>John</first>
  <middle>Fitzgerald Johansen</middle>
  <last>Doe</last>
</name>
```

- указване на външна дефиниция (External DTD)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE movies SYSTEM "movies.dtd">
<movies>
  <movie year="1995" length="106" in_color="Yes">
    <!-- *** -->
  </movie>
</movies>
```

- смесено (комбинация от двата подхода)

DOCTYPE

```
<!DOCTYPE root_name External-ID [ declarations ]>
```

име на root
елемента

указател към файл от вида:

Internal DTD (не е задължително)

или

```
SYSTEM "URI"  
PUBLIC "Public ID" "URI"
```

- SYSTEM – указва път (URI) до външен файл с DTD дефиниция

```
<!DOCTYPE name SYSTEM "file:///c:/name.dtd">  
<!DOCTYPE name SYSTEM "http://wiley.com/hr/name.dtd">  
<!DOCTYPE name SYSTEM "name.dtd">
```

- PUBLIC – указват запис от каталог

```
<!DOCTYPE name PUBLIC "-//Beginning XML//DTD Name Example//EN" "name.dtd">  
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

Formal Public Identifier (FPI) – от ISO 9070
със структура:

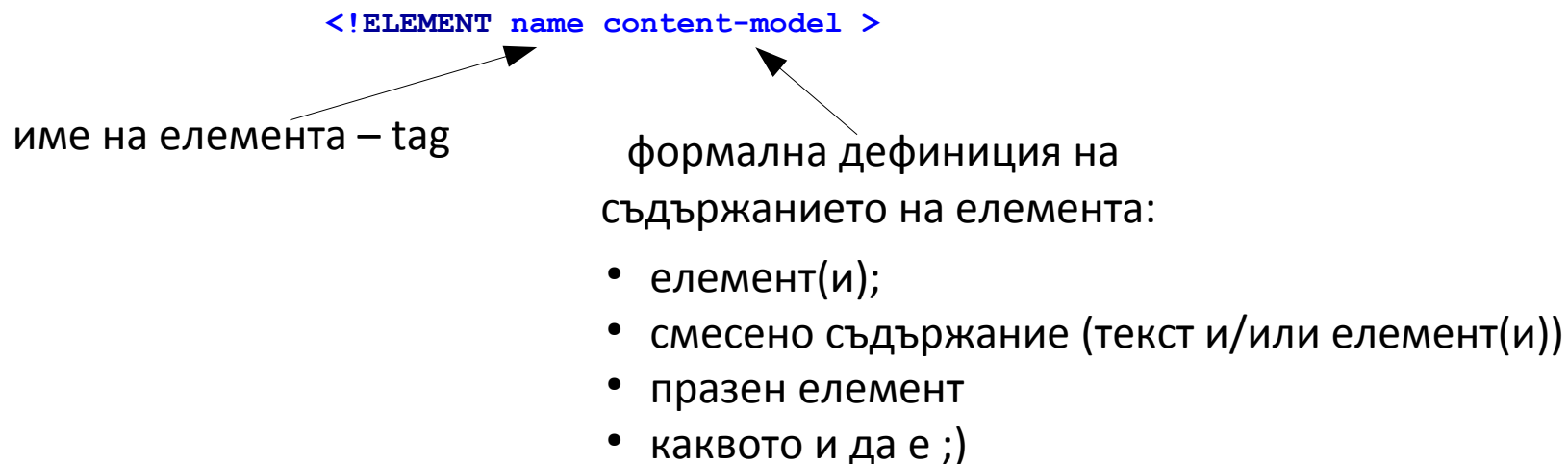
```
-//Owner//Class Description//Language//Version
```

URI до копие (не е задължително)

DTD декларации

<code><!ELEMENT ... ></code>	елемент
<code><!ATTLIST ... ></code>	атрибут
<code><!ENTITY ... ></code>	entity
<code><!NOTATION ... ></code>	нотация
<code><![INCLUDE[...]]></code>	условни секции
<code><![IGNORE[...]]></code>	

Елемент (1)



Елемент (2)

- елемент(и)

`<!ELEMENT movies (movie) >`

movies съдържа **точно 1** елемент на име movie

`<movies><movie /></movies>`

`<!ELEMENT movie (title, studio, producer, stars)>`

movie съдържа **точно по 1** от елементите title, studio, producer и stars **в тази последователност**

`<!ELEMENT location (address | gps) >`

location съдържа **или** address, **или** gps **точно 1** път

`<!ELEMENT location (address | (lat, lng)) >`

location съдържа или address (1 път), или lat и lng (по 1 път в тази последователност)

`<!ELEMENT movie (title, studio?, producer*, star+)>`

movie **точно 1** път елемента title,
0 или 1 път елемента studio,
0 или повече пъти – producer и
1 или повече пъти – star
в тази последователност

- скоби – за групиране;
- запетаи – за изброяване и задаване на наредба;
- права черта – за задаване на избор (или);
- индикатори за брой срещане:
 - ? – 0 или 1;
 - * – 0 или повече;
 - + – 1 или повече

Елемент (3)

- смесено съдържание (текст и/или елементи)

`<!ELEMENT title (#PCDATA)>` ← title съдържа **само текст**

`<!ELEMENT description (#PCDATA | title | detail)*>` ← description съдържа или текст, или елемента title, или елемента detail

Трябва да се започне с #PCDATA !!!

При това изброяване е задължително !!!

- PCDATA = Parsed Character Data
- Текстът в PCDATA ще се третира като markup, т.е. може да има entities
- Символите **&**, **<** и **>** трябва да се заместват с entity-та им!

Елемент (4)

- празен елемент

`<!ELEMENT country EMPTY>`

← country е **празен елемент**

`<country />`

- каквото и да е ;)

`<!ELEMENT description ANY>`



description може да съдържа
текст (PCDATA)
и/или
кои да е елементи, дефинирани в DTD

Атрибут (1)

Повтарят се за всеки атрибут на елемента (не е задължително)

`<!ATTLIST element name declvalue default >`

име на елемента (tag),
за който е атрибутът

име на атрибута

тип или изброяване на
стойности

стойност по
подразбиране и/или
ключова дума

Атрибут (2)

Типове стойности:

- CDATA** —► низ
- NMTOKEN** —► низ без white space или празен низ (1 дума)
- NMTOKENS** —► списък от NMTOKEN, отделени с интервал
- ID** —► уникален идентификатор (трябва да започва с буква или _)
- IDREF** —► указател към уникален идентификатор в документа
- IDREFS** —► множество указатели към уникален идентификатор в документа, разделени с интервал
- ENTITY** —► име на entity в документа
- ENTITIES** —► множество имена на entity в документа
- (Yes|No|Да|Не)** —► изброяване на допустими стойности

Атрибут (3)

Стойност по подразбиране и/или ключова дума:

- "Yes" —▶ стойност по подразбиране (в кавички)
- #FIXED "1.0" —▶ единствена стойност, която е позволена за атрибута, е 1.0
- #REQUIRED —▶ задължително трябва да се зададе стойност (не може да се укаже стойност по подразбиране)
- #IMPLIED —▶ в останалите случаи

Entity

- General

```
<!ENTITY copyright "2013 ФМИ">
```

В XML:

```
<title>The Usual Suspects &copyright;</title>
```

- Parameter

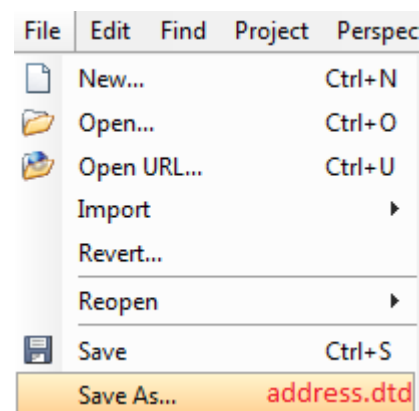
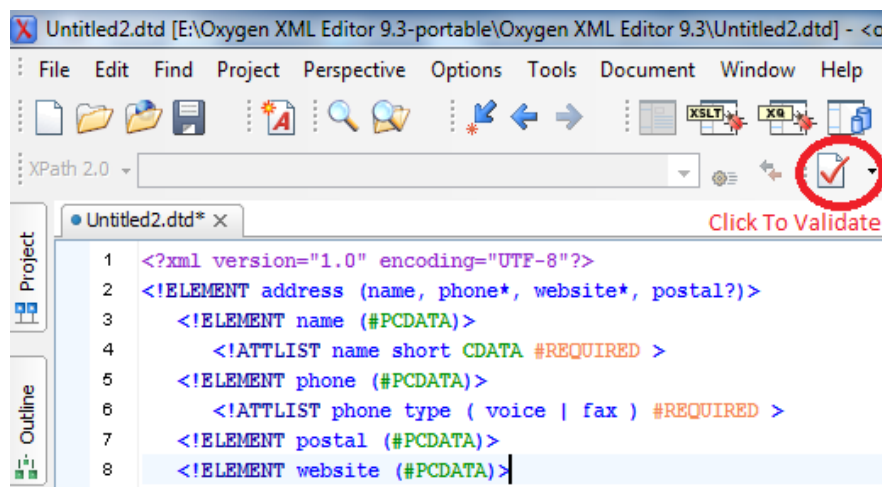
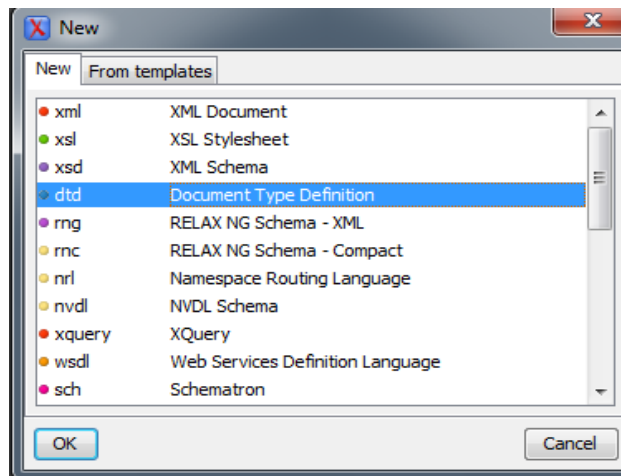
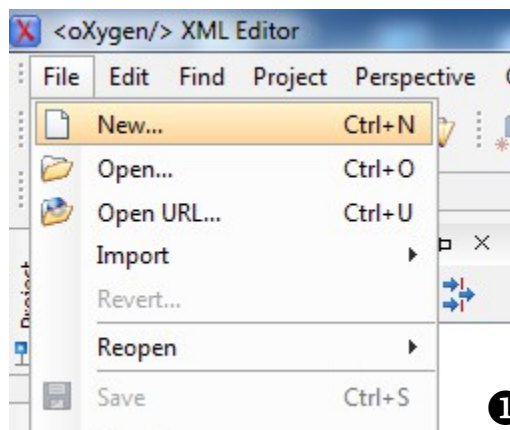
```
<!ENTITY % param "(paragraph | list)">  
<!ELEMENT section (%param;)*>
```

ВМЕСТО

```
<!ELEMENT section (paragraph | list)*>
```

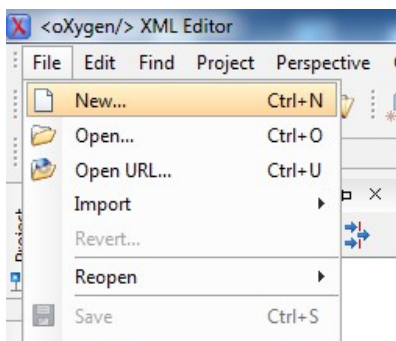

Работа с <oXygen/> XML Editor (1)

- Създаване на DTD

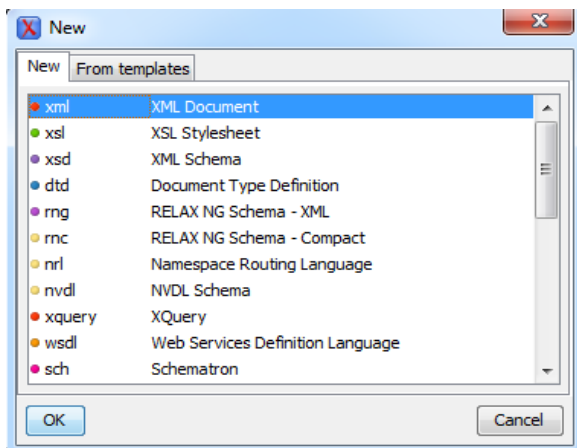


Работа с <Oxygen/> XML Editor (2)

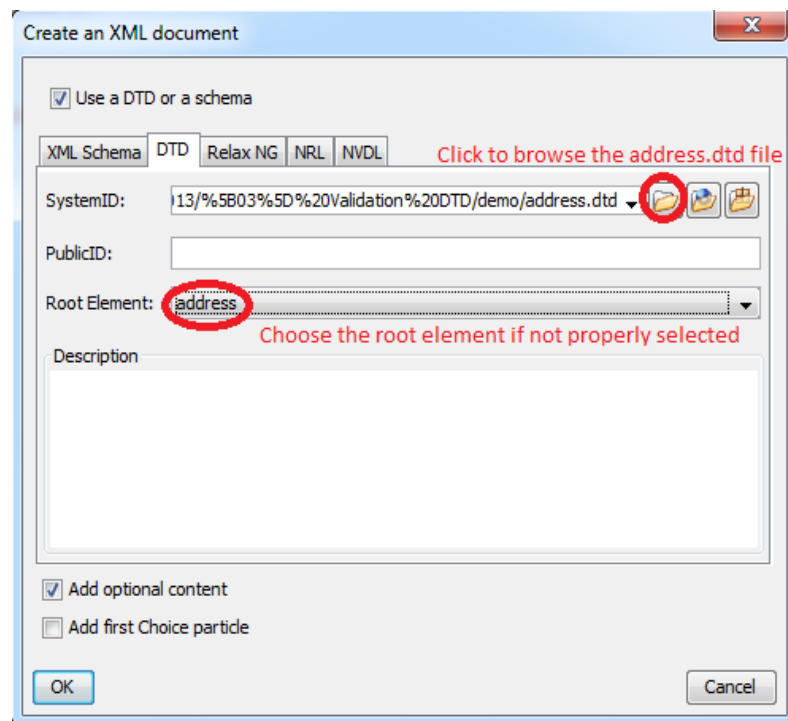
- Създаване на XML по DTD



1



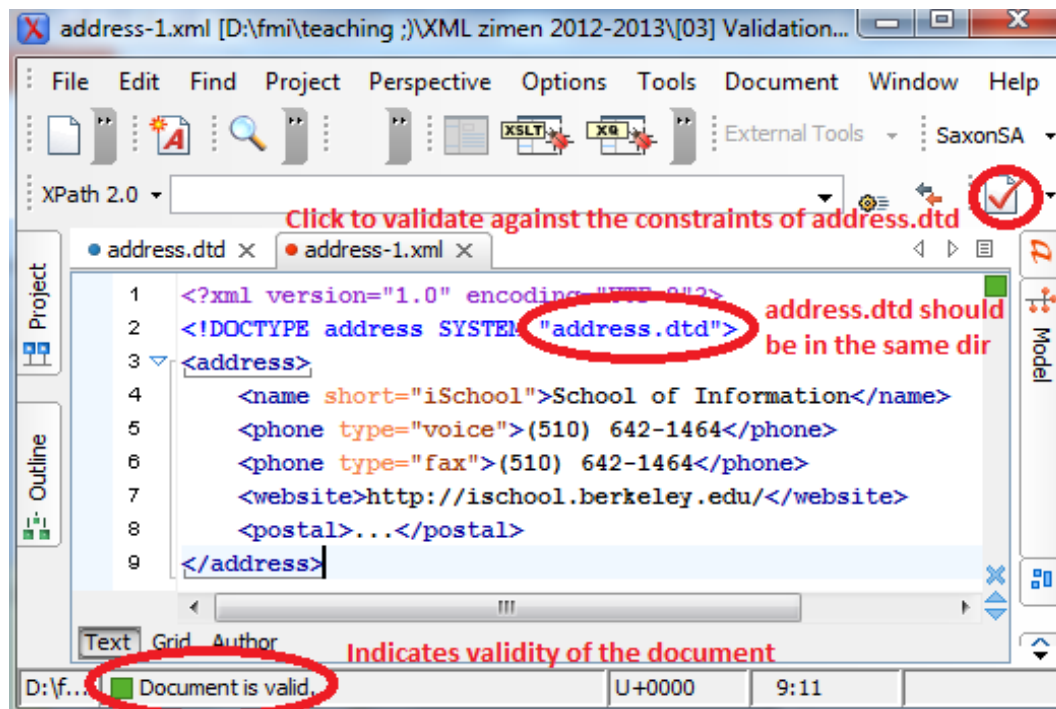
2



3

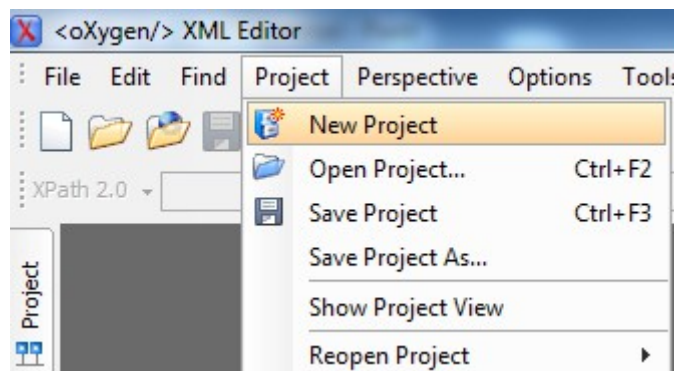
Работа с <Oxygen/> XML Editor (3)

- Валидиране на XML по DTD

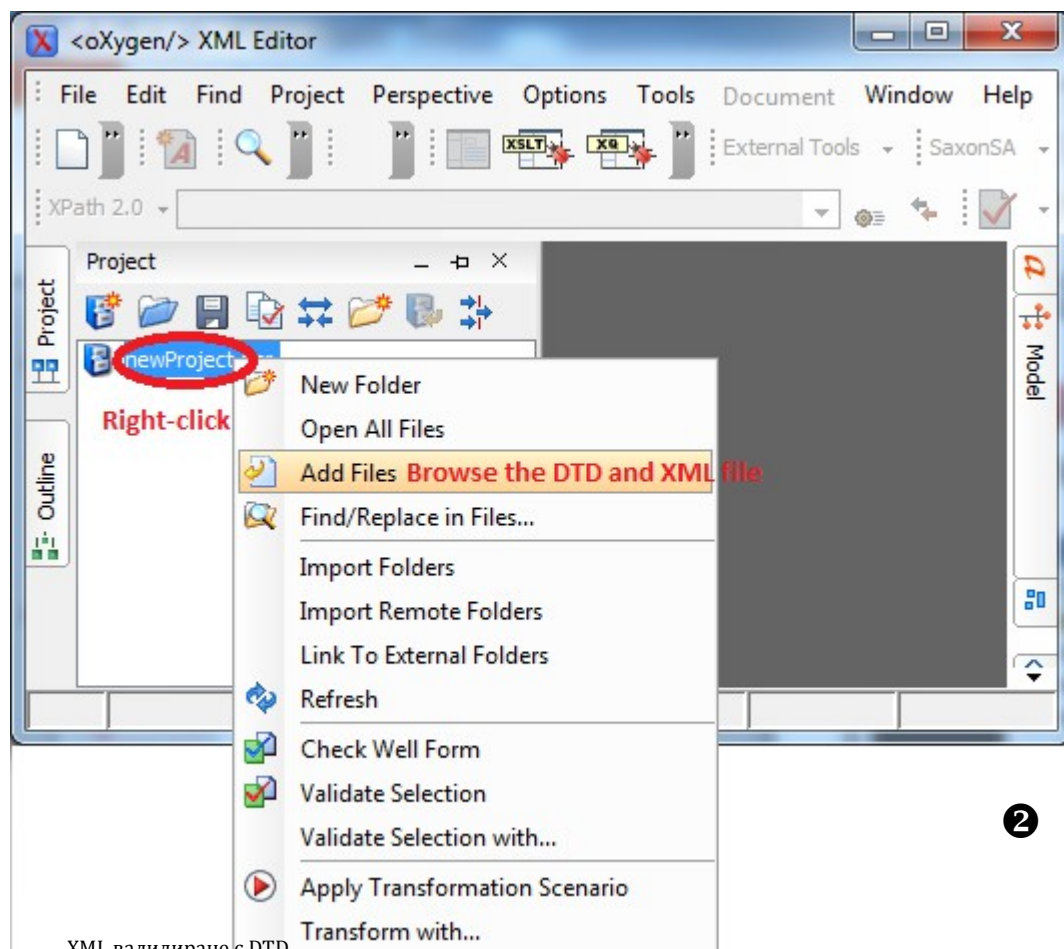


Работа с <oXygen/> XML Editor (4)

- Работа с проект



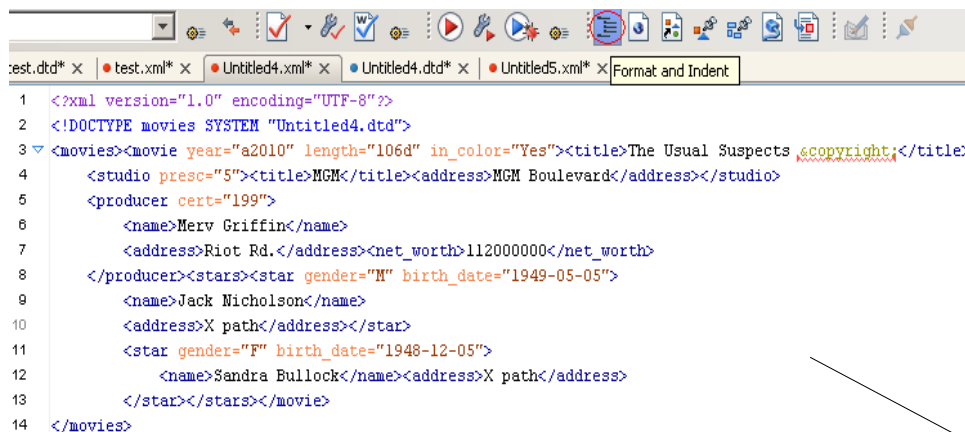
1



2

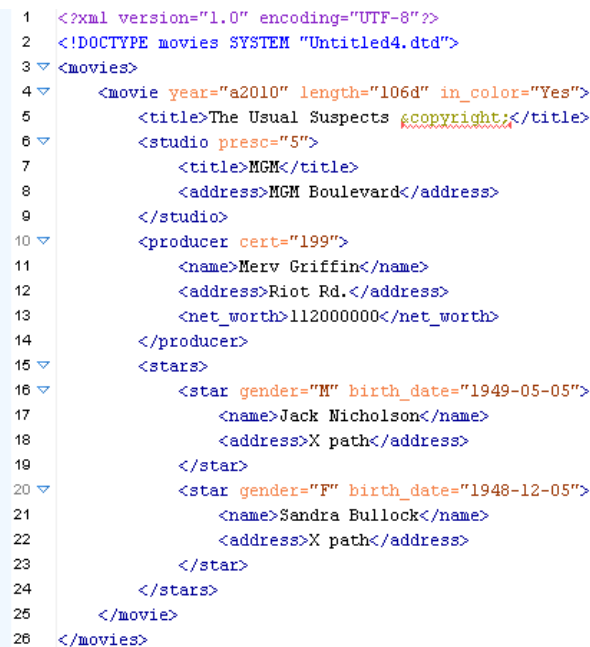
Работа с <Oxygen/> XML Editor (5)

- Форматиране и поставяне на табулации (индентирание :))



The screenshot shows the Oxygen XML Editor interface with a menu bar and a toolbar. The 'Format and Indent' menu is open. The XML code in the editor is as follows:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE movies SYSTEM "Untitled4.dtd">
3 <movies><movie year="a2010" length="106d" in_color="Yes"><title>The Usual Suspects <copyright;</title>
4   <studio presc="5"><title>MGM</title><address>MGM Boulevard</address></studio>
5   <producer cert="199">
6     <name>Merv Griffin</name>
7     <address>Riot Rd.</address><net_worth>112000000</net_worth>
8   </producer><stars><star gender="M" birth_date="1949-05-05">
9     <name>Jack Nicholson</name>
10    <address>X path</address></star>
11    <star gender="F" birth_date="1948-12-05">
12      <name>Sandra Bullock</name><address>X path</address>
13    </star></stars></movie>
14 </movies>
```



The screenshot shows the same XML code after formatting with tabs. An arrow points from the unformatted code to this formatted version. The code is as follows:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE movies SYSTEM "Untitled4.dtd">
3 <movies>
4   <movie year="a2010" length="106d" in_color="Yes">
5     <title>The Usual Suspects <copyright;</title>
6     <studio presc="5">
7       <title>MGM</title>
8       <address>MGM Boulevard</address>
9     </studio>
10    <producer cert="199">
11      <name>Merv Griffin</name>
12      <address>Riot Rd.</address>
13      <net_worth>112000000</net_worth>
14    </producer>
15    <stars>
16      <star gender="M" birth_date="1949-05-05">
17        <name>Jack Nicholson</name>
18        <address>X path</address>
19      </star>
20      <star gender="F" birth_date="1948-12-05">
21        <name>Sandra Bullock</name>
22        <address>X path</address>
23      </star>
24    </stars>
25  </movie>
26 </movies>
```

Работа с <Oxygen/> XML Editor (6)

- Reverse Engineering – създаване на DTD за даден XML документ

The screenshot displays the Oxygen XML Editor interface. The main window shows an XML document with the following content:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE movies SYSTEM "Untitled4.dtd">
3 <movies>
4   <movie year="a2010" length="106d" in_color="Yes">
5     <title>The Usual Suspects <copyright/></title>
6     <studio presc="5">
7       <title>MGM</title>
8       <address>MGM Boulevard</address>
9     </studio>
10    <producer cert="199">
11      <name>Merv Griffin</name>
12      <address>Riot Rd.</address>
13      <net_worth>112000000</net_worth>
14    </producer>
15    <stars>
16      <star gender="M" birth_date="1949-05-05">
17        <name>Jack Nicholson</name>
18        <address>X path</address>
19      </star>
20      <star gender="F" birth_date="1948-12-05">
21        <name>Sandra Bullock</name>
22        <address>X path</address>
23      </star>
24    </stars>
25  </movie>
26 </movies>
27
```

The 'Convert to...' button in the top toolbar is circled in red, with an arrow pointing to the 'Trang Converter' dialog box. In the dialog, the 'XML 1.0 DTD' option is selected and circled in red. The 'Convert' button at the bottom is also circled in red. The 'Open in editor' checkbox is checked.

XML валидиране с DTD

Използвана литература

- Fawcett J., L. Quin, D. Ayers – Beginning XML, 5th Edition – John Wiley & Sons, Inc., 2012 – CHAPTER 4: Document Type Definitions
- Бончев Б. – XML валидиране чрез Document Type Definition (DTD) – Лекции по Маркърп езици за семантичен Уеб /XML/, СУ, ФМИ, СИ, 2013
- Wilde E. – XML Foundations Fall 2008 – UC Berkeley School of Information – on-line @ <http://dret.net/lectures/xml-fall08/dtd> – Document Type Definition (DTD)
- Цветков Н. – Лекции към упражнения по XML програмиране, зимен семестър 2011 – ФМИ, СУ – on-line @ <http://tiny.cc/dqfwmw> – Упражнение 03
- XML Syntax Quick Reference
- Георгиева Ю. – Средства за дефиниране на XML документи – DTD – Лекции по XML и Web базирани технологии, ТУ, ПКТ