

SEMESTRÁLNÍ PRÁCE

předmět AD7B36XML

Ozdravné pobytu dětí

**ČVUT FEL
obor STM - Softwarové inženýrství, kombinované studium
4. semestr**

Zpracovala: 
Radoslava Jandová (jandora1)
V Praze dne 10. května 2011

Obsah:

Úvod	3 str.
1. Úkol - Základní pravidla tvorby XML dokumentů	3 str.
1.1. Zadání	
1.2. Zpracování - <i>Obr. 1: Well-Formed Document</i>	
2. Úkol - Kontrola validity XML dokumentu	5 str.
2.1. Zadání	
2.2. Zpracování - <i>Obr. 2: Validace XML</i>	
3. Úkol - Kontrola validity pomocí jazyka XML Schema	6 str.
3.1. Zadání	
3.2. Zpracování - <i>Obr. 3: Validace XML dokumentu pomocí nástroje NetBeans IDE</i> <i>Obr. 4: Validace XML dokumentu pomocí validátoru W3C</i> <i>Obr. 5: Warnings označené validací</i>	
4. Úkol - Dotaz v jazyce XPath	8 str.
4.1. Zadání	
4.2. Zpracování - <i>Obr. 6 - 15: Výsledky dotazů 1 - 10</i>	
5. Úkol - Transformace pomocí XSLT	12 str.
5.1. Zadání	
5.2. Zpracování - <i>Obr. 16: Výstup transformace do HTML</i> <i>Obr. 17: Výstup transformace do XHTML</i>	
6. Úkol - XQuery, případně DOM/SAX	14 str.
6.1. Zadání	
6.2. Zpracování - <i>Obr. 17 - 20: Výstupy dotazů XQuery</i>	
Závěr	17 str.
Zdroje	17 str.

Úvod

Téma referátu: Ozdravné pobytu dětí

Popis: Organizátor zajišťuje ozdravné pobytu pro děti. Tyto pobytu se pořádají v několika lokalitách. Pobytu jsou rozděleny do turnusů. Každý turnus se koná v určené lokalitě a má přesně daný termín. Turnusy jsou rozděleny do skupin. Každá skupina je určena unikátním číslem a je vedena jedním vedoucím. Vedoucí je určen svým jménem a příjmením. V každé skupině je maximálně 10 dětí. Do skupin jsou rozděleny děti podle věku a pohlaví. U každého dítěte je uvedeno jméno, příjmení, pohlaví a rok narození.

Použitý SW: NetBeans IDE, verze 6.9.1. – program byl využitý k tvorbě základních dokumentů referátu a k validaci.

1. Úkol - Základní pravidla tvorby XML dokumentů

Odevzdáno: 24. února 2011

Obsah ZIP souboru: referat.xml, wellFormed.jpg

1.1. Zadání: viz <http://ocw.cvut.cz/moodle/mod/assignment/view.php?id=8744>

Zde odevzdáte 1. kapitolu Vašeho referátu jako textový soubor.

Smyslem této kapitoly je procvíčit základní pravidla tvorby dobré vytvořených (well-formed) XML dokumentů, testování správnosti (well-formedness), procvíčení základních pojmu XML - element, atribut, character data, entita apod.

Do 1. kapitoly svého referátu umístěte řešení pro Vaše téma (které si zvolíte).

Pokyny (Šablona pro 1. kapitolu):

- ```
=====
1. Vytvořte XML dokument popisující základní základní objekty Vaší aplikace.
2. Validujte tento dokument pomocí některého validátoru (řádkové nástroje xmllint, xmlwf - obojí je v Solariu nainstalováno).
3. Umístěte element dokumentu z úkolu 1 do nějakého jmenného prostoru.
4. Rozšiřte Vaši definici o text pomocí značek HTML -> dokument mixující elementy ze dvou jmenných prostorů.
```

Příklad na jmenné prostory

```
=====
<root>

<b:b xmlns:b="urn:ns:nsB" />
<c xmlns:d="urn:ns:nsD" d:d="attvalue" />
</root>
```

### 1.2. Zpracování:

**krok 1. Vytvořte XML dokument popisující základní základní objekty Vaší aplikace.**

V programu NetBeans IDE jsem vytvořila základní schéma dokumentu XML „referat.xml“ takto:

```
<pobytu>
 <turnus id=""> = turnus pobytu (může jich být 1 - n)
 <lokalita></lokalita> = lokalita, kde se turnus koná
 <termin><termin/> = termín konání turnusu

 <skupina id=""> = skupiny v rámci turnusu (může jich být 1 - n)
 <cisloSkupiny>< cisloSkupiny/> = unikátní číslo skupiny

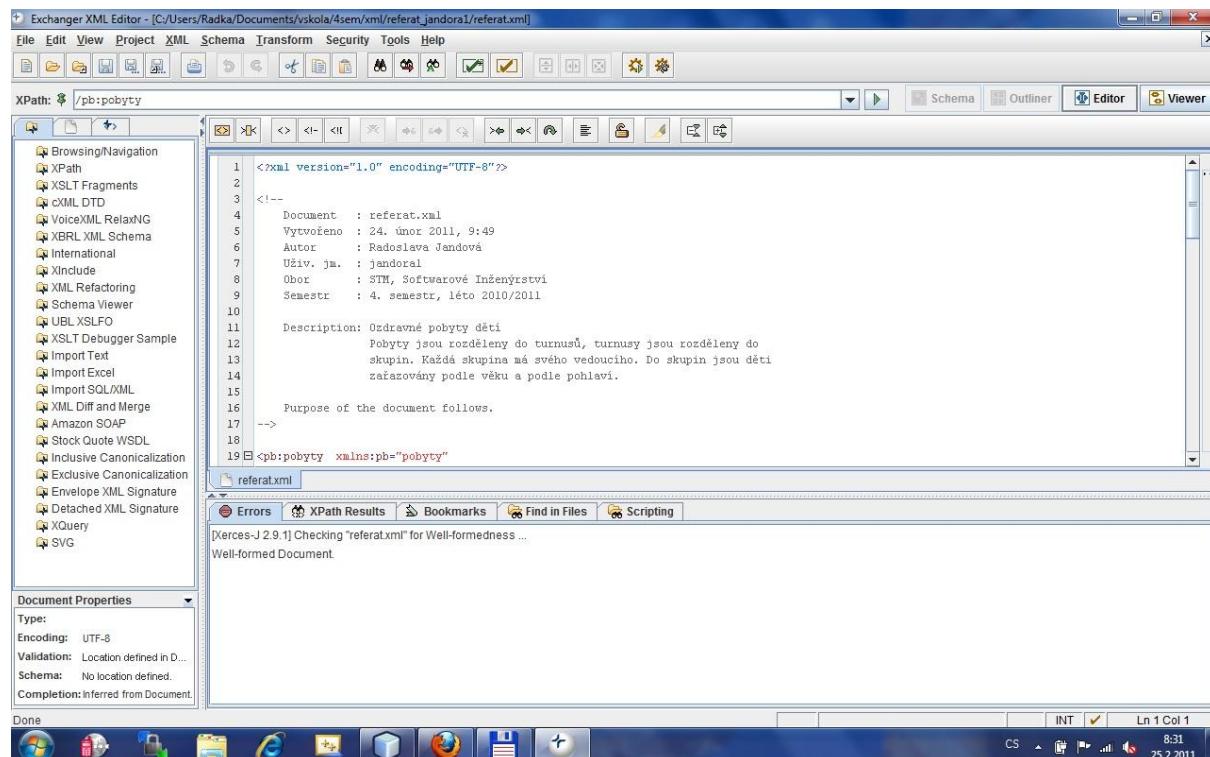
 <vedouci> = vedoucí skupiny, jeden vede jednu skupinu
 <jmenoVedouciho>< jmenoVedouciho/>
 <jmenoVedouciho>< jmenoVedouciho/>
 <vedouci/>

 <deti> = děti zařazené do dané skupiny
```

```
<dite id=""> = identifikace dítěte (může jich být 0 - n)
 <jmenoD><jmenoD/>
 <prijmeniD><prijmeniD/>
 <pohlaviD><pohlaviD/>
 <rokNarozeni><rokNarozeni/>
</dite>
<deti/>
<skupina/>
<turnus/>
</pobyty>
```

**krok 2. Validujte tento dokument pomocí některého validátoru**

Dokument XML jsem validovala v programu Exchanger XML Editor, verze 3.3.1.



Obr. 1: Well-Formed Document

**krok 3. Umístěte element dokumentu z úkolu 1 do nějakého jmenného prostoru.**

Do dokumentu byl přidán následující jmenný prostor:

```
<pb:pobyty xmlns:pb="pobyty"> ... </pobyty>
```

**krok 4. Rozšiřte Vaši definici o text pomocí značek HTML -> dokument mixující elementy ze dvou jmenných prostorů.**

Do dokumentu XML byla vložena následující značka HTML:

```
<pb:pobyty xmlns:pb="pobyty"
 xmlns:xhtml="http://www.w3.org/1999/xhtml"> ... </pobyty>
```

## 2. Úkol - Kontrola validity XML dokumentu

Odevzdáno: 1. března 2011

Obsah ZIP souboru: referat.pdf, referat.xml, referat.dtd, valideaceXML.jpg

2.1. Zadání: viz <http://ocw.cvut.cz/moodle/mod/assignment/view.php?id=8746>

Vytvořte 2. kapitolu Vašeho referátu dokumentující definici DTD pro dokumenty z Vašeho tématu. Vytvořte definici DTD pro Vaše téma, dále vytvořte příklad dokumentu z Vašeho tématu. Zkontrolujte pomocí validátora, zda je XML dokument validní vzhledem k DTD. Opravujte XML dokument, dokud není validní. Do 2.kapitoly vložte XML document, DTD a popis způsobu, jak jste jej validovali.

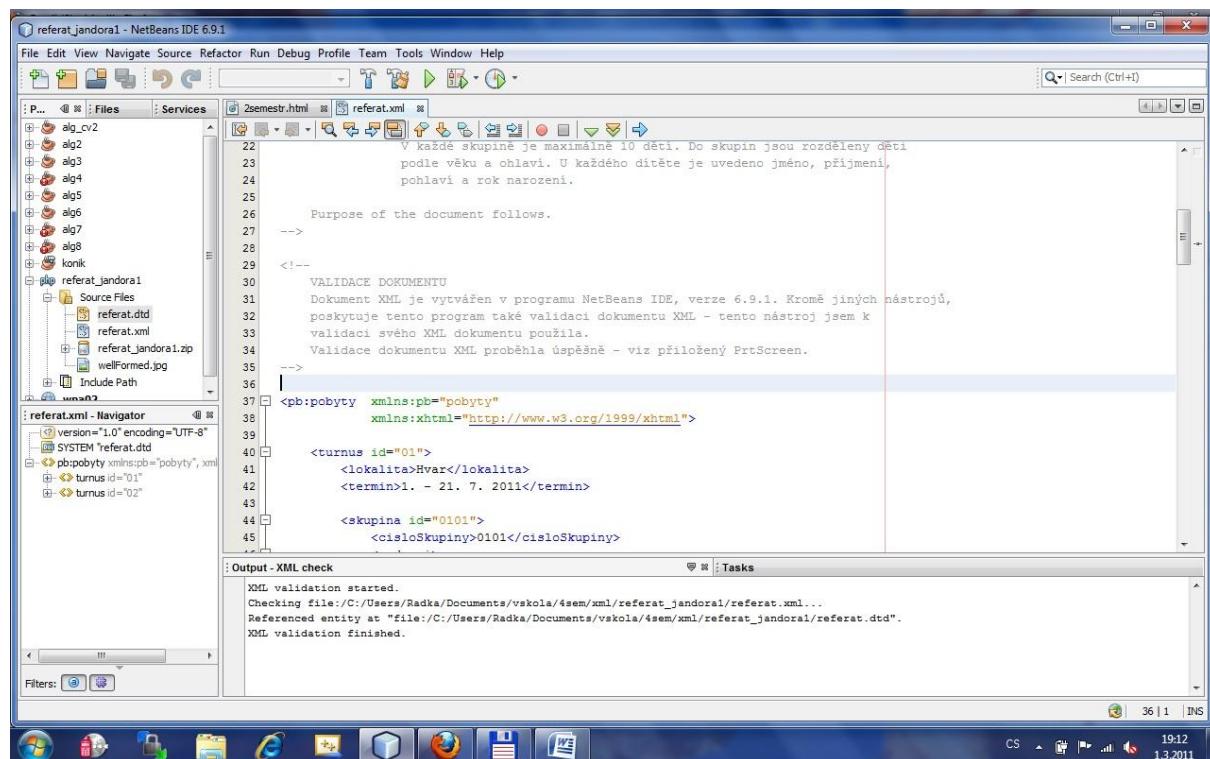
Pokud si chcete téma podrobněji vyzkoušet, prostudujte si doplňující materiály.

### 2.2. Zpracování:

Pro dokument „referat.xml“ jsem vytvořila DTD – soubor s názvem „referat.dtd“ a provedla jsem propojení se souborem „referat.xml“:

```
<!DOCTYPE pb:pobyty SYSTEM "referat.dtd">
```

Následně jsem soubor „referat.xml“ validovala, a to po mocí nástroje „Validate XML“, který poskytuje přímo pro program NetBeans IDE. Na základě výsledku validace jsem opravila chyby, dokud validace dokumentu nebyla úspěšná.



Obr. 2: Validace XML dokumentu

### 3. Úkol - Kontrola validity pomocí jazyka XML Schema

Odevzdáno: 11. 3. 2011

Obsah ZIP souboru: referat.pdf, referat.xml, referat.xsd, uk3validaceNb.jpg, uk3validaceW3C.jpg, warnings.jpg

**3.1. Zadání:** viz <http://ocw.cvut.cz/moodle/mod/assignment/view.php?id=8748>

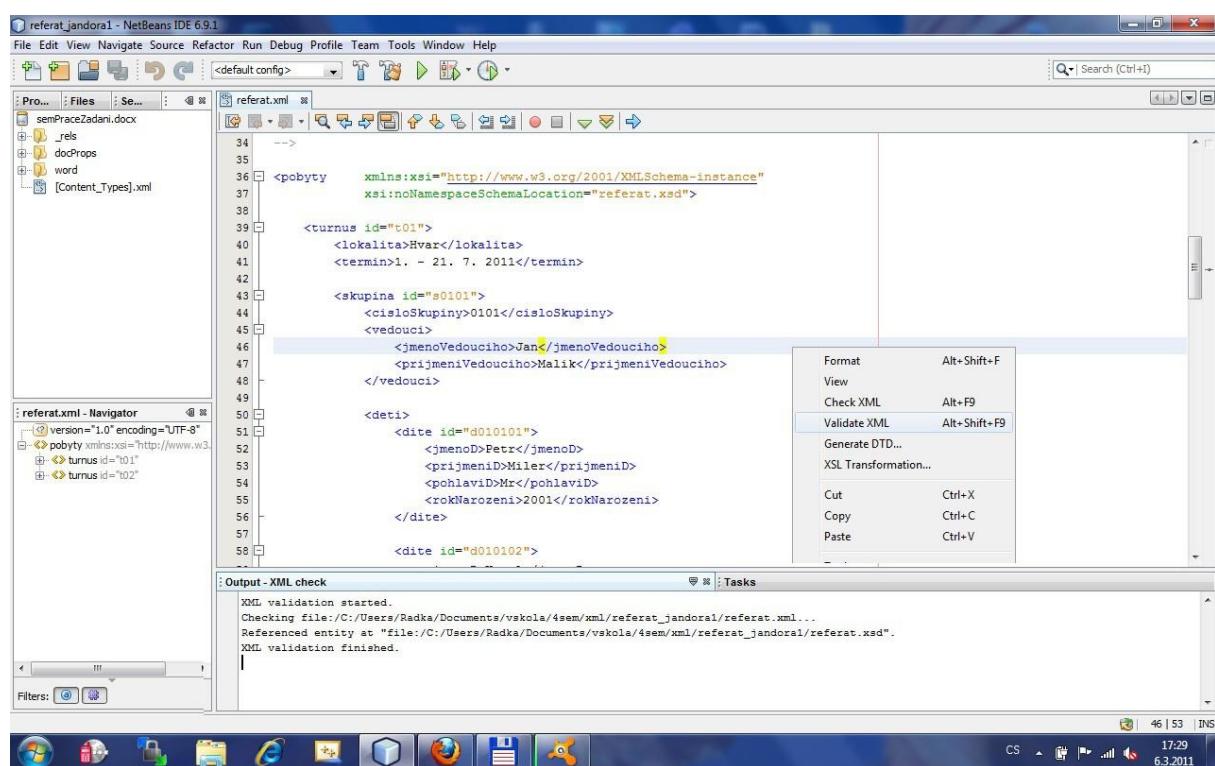
Vytvořte XML schéma v jazyce XML Schema pro XML dokumenty se strukturou danou vaším XML schématem a zkuste ho validovat. Opravujte ho, dokud není validní.

Pozn.: Využijte validátor <http://www.w3.org/2001/03/webdata/xsv> k testu, zda Vámi vytvořené XML schéma je validní XML Schema.

#### 3.2. Zpracování:

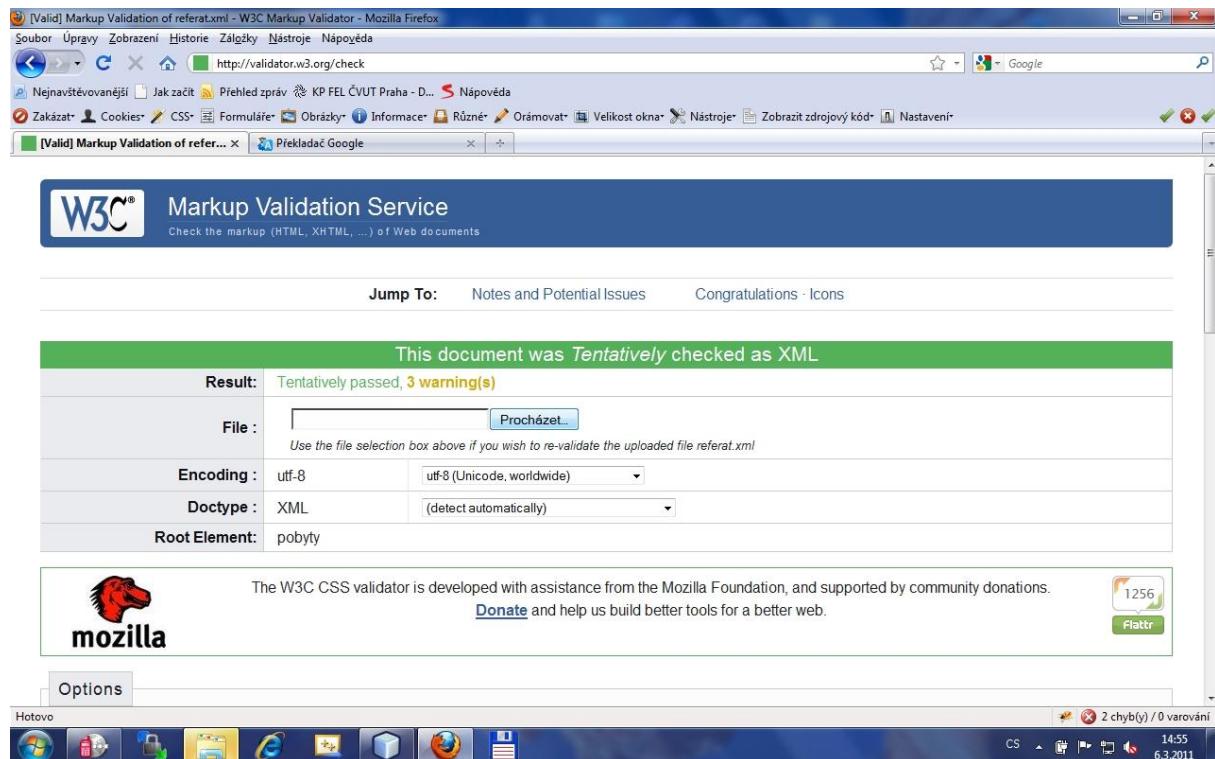
K semestrální práci jsem přidala další dokument - „referat.xsd“, ve kterém jsem vytvořila XML schéma odpovídající původně vytvořenému XML dokumentu (viz Úkol 1.).

K první validaci jsem použila validátor v rámci programu NetBeans IDE. První validace odhalila chybné propojení dokumentů „referat.xml“ a „referat.xsd“. Po opravě proběhla validace úspěšně.



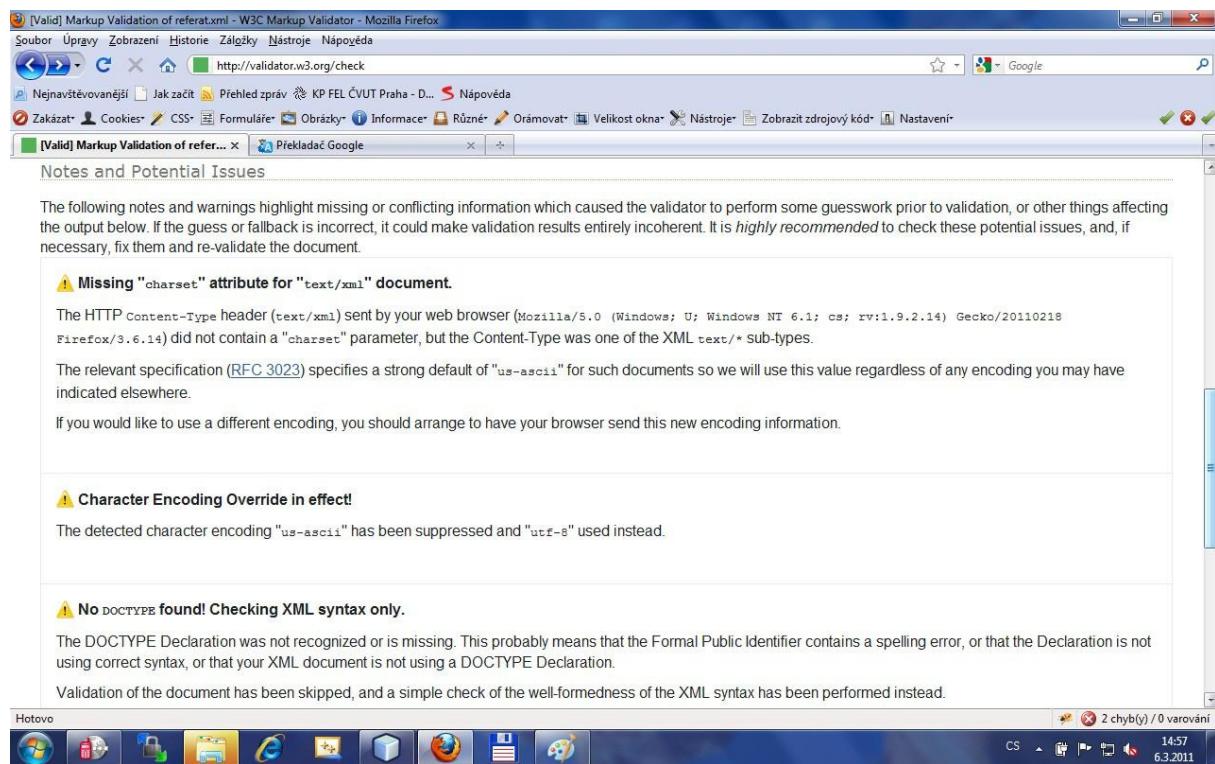
Obr. 3: Validace XML pomocí nástroje NetBeans IDE

Následně jsem dokument validovala ještě pomocí validátoru W3C, který předepisuje zadání úkolu č. 3. Validace proběhla s výsledkem „Tentatively passed“ – viz PrintScreen.



Obr. 4: Validace XML dokumentu pomocí validátoru W3C

Validace označila 3 „warnings“, které jsem ale již nedokázala odstranit.



Obr. 5: „Warnings“ označené validací.

#### 4. Úkol - Dotaz v jazyce XPath

Odevzdáno: 10. 4. 2011

Obsah ZIP souboru: referat.pdf, referat.xml, xxx

**4.1. Zadání:** viz <http://ocw.cvut.cz/moodle/mod/assignment/view.php?id=8749>

Vytvořte dotazy v jazyce XPath, které se týkají obsahu Vašeho dokumentu. Demonstrujte správný výsledek pomocí nějakého nástroje.

##### 4.2. Zpracování:

**Dotaz 1:** Vypiš všechny turnusy.

/pobyty/turnus  
nebo  
//turnus

The screenshot shows the XPath tool interface with the following details:

- Toolbar:** Buttons for XPATH 1.0 and XPATH 2.0, Allow complete XPath, and various search and export options.
- Input Bar:** The query `1 //turnus` is entered.
- Result Tree View:** A tree structure showing the results of the query. The root node is labeled `1`. It contains two child nodes, both labeled `turnus`, with IDs `t01` and `t02` respectively. The tree view has a depth indicator on the left.
- Status Bar:** Shows "Items: 2".
- Bottom Navigation:** Tabs for Messages, XPath (selected), XSL Outline, Find in Files, Find in Schemas, Find in XBRL, and Charts.

Obr. 6: Výsledek dotazu 1

**Dotaz 2:** Vypiš všechny skupiny bez ohledu na turnus.

/pobyty//skupina  
nebo  
//skupina

The screenshot shows the XPath tool interface with the following details:

- Toolbar:** Buttons for XPATH 1.0 and XPATH 2.0, Allow complete XPath, and various search and export options.
- Input Bar:** The query `1 //skupina` is entered.
- Result Tree View:** A tree structure showing the results of the query. The root node is labeled `1`. It contains four child nodes, all labeled `skupina`, with IDs `s0101`, `s0102`, `s0201`, and `s0202` respectively. The tree view has a depth indicator on the left.
- Status Bar:** Shows "Items: 4".
- Bottom Navigation:** Tabs for Messages, XPath (selected), XSL Outline, Find in Files, Find in Schemas, Find in XBRL, and Charts.

Obr. 7: Výsledek dotazu 2

*Dotaz 3:* Vypiš všechny skupiny z turnusu t02.

```
/pobyty/turnus[@id='t02']/skupina
nebo
/turnus[@id='t02']/skupina
```

The screenshot shows the XPath tool interface. The top bar has tabs for 'PATH 1.0' and 'PATH 2.0', with 'Allow complete XPath' checked. The 'Where:' dropdown is set to 'Current file'. The main area displays the XPATH expression '/pobyty/turnus[@id='t02']/skupina'. Below it, the results pane shows two items under path step 2, both labeled 'skupina' with IDs 's0201' and 's0202'. The left sidebar shows the full path from root to step 2. The bottom navigation bar includes tabs for 'Messages', 'XPath' (which is selected), 'XSL Outline', 'Find in Files', 'Find in Schemas', 'Find in XBRL', and 'Charts'.

Obr. 8: Výsledek dotazu 3

*Dotaz 4:* Vypiš z pobytů příjmení všech dětí, které jsou zařazeny do 2 skupiny a mají pořadové číslo 1.

```
/pobyty/turnus/skupina[2]/deti/dite[1]/prijmeniD
nebo
// skupina[2]//dite[1]/prijmeniD
```

The screenshot shows the XPath tool interface. The top bar has tabs for 'PATH 1.0' and 'PATH 2.0', with 'Allow complete XPath' checked. The 'Where:' dropdown is set to 'Current file'. The main area displays the XPATH expression '/pobyty/turnus/skupina[2]/deti/dite[1]/prijmeniD'. Below it, the results pane shows two items under path step 2, both labeled 'prijmeniD' with values 'Muchova' and 'Pavelka'. The left sidebar shows the full path from root to step 2. The bottom navigation bar includes tabs for 'Messages', 'XPath' (which is selected), 'XSL Outline', 'Find in Files', 'Find in Schemas', 'Find in XBRL', and 'Charts'.

Obr. 9: Výsledek dotazu 4

*Dotaz 5:* Vypiš příjmení všech dětí z turnusu t01.

```
/pobyty/turnus[@id='t01']/deti/dite/prijmeniD
nebo
/turnus[@id='t01']//prijmeniD
```

The screenshot shows the XPath tool interface with the following details:

- Toolbar:** Buttons for PATH 1.0, PATH 2.0, Allow complete XPath, and various file operations.
- Query Bar:** The query entered is `1 /pobyty/turnus[@id='t01']/*/deti/dite/prijmeniD`.
- Result Tree View:** A tree view showing the structure of the XML. The root node has four children labeled `prijmeniD`, each with a value: Miler, Janda, Muchova, and Jirsová.
- Status Bar:** Shows "Items: 4".
- Bottom Navigation:** Buttons for Messages, XPath (which is selected), XSL Outline, Find in Files, Find in Schemas, Find in XBRL, and Charts.

Obr. 10: Výsledek dotazu 5

*Dotaz 6:* Vypiš jména a příjmení všech vedoucích z lokality Hvar a id dětí v jejich skupinách.

```
/*[lokalita='Hvar']/*/*/*
```

The screenshot shows the XPath tool interface with the following details:

- Toolbar:** Buttons for PATH 1.0, PATH 2.0, Allow complete XPath, and various file operations.
- Query Bar:** The query entered is `1 //*[lokalita='Hvar']/*/*/*`.
- Result Tree View:** A tree view showing the structure of the XML. The root node has eight children, all labeled `jmenoVedouciho` or `prijmeniVedouciho`, with values Jan, Malik, id="d010101", id="d010102", Alena, Kadavá, id="d010201", and id="d010202".
- Status Bar:** Shows "Items: 8".
- Bottom Navigation:** Buttons for Messages, XPath (which is selected), XSL Outline, Find in Files, Find in Schemas, Find in XBRL, and Charts.

Obr. 11: Výsledek dotazu 6

*Dotaz 7:* Vypiš všechny chlapce.

```
/* [pohlaviD="Mr"]
nebo
/* [pohlaviD!="Mrs"]
```

The screenshot shows the XPath tool interface with the following details:

- Toolbar:** Buttons for XPATH 1.0 and XPATH 2.0, Allow complete XPath, and Where dropdown set to Current file.
- Query Bar:** The query entered is `1 //*[pohlaviD="Mr"]`.
- Result Tree View:** A tree view showing four items under node 1, all labeled "dite". The items have IDs: id="d010101", id="d010102", id="d020201", and id="d020202".
- Status Bar:** Shows "Items: 4".
- Bottom Navigation:** Tabs for Messages, XPath (selected), XSL Outline, Find in Files, Find in Schemas, Find in XBRL, and Charts.

Obr. 12: Výsledek dotazu 7

*Dotaz 8:* Vypiš vše o dětech z lokality Makarska.

```
/*[lokalita="Makarska"]//deti/dite/*
```

The screenshot shows the XPath tool interface with the following details:

- Toolbar:** Buttons for XPATH 1.0 and XPATH 2.0, Allow complete XPath, and Where dropdown set to Current file.
- Query Bar:** The query entered is `1 /*[lokalita="Makarska"]//deti/dite/*`.
- Result Tree View:** A tree view showing 16 items under node 1, all labeled "dite". The items have attributes: jmenoD, prijmeniD, pohlaviD, and rokNarozeni. The values for these attributes are: Petra, Novotná, Mrs, 1996; Ivana, Vokounová, Mrs, 1996; Martin, Pavelka, Mr, 2002; Jan, Žindulka, Mr, 2002.
- Status Bar:** Shows "Items: 16".
- Bottom Navigation:** Tabs for Messages, XPath (selected), XSL Outline, Find in Files, Find in Schemas, Find in XBRL, and Charts.

Obr. 13: Výsledek dotazu 8

*Dotaz 9:* Vypiš všechny vedoucí.  
 //vedouci/descendant::\*

| Path                  | Value     |
|-----------------------|-----------|
| (1) jmenoVedouciho    | Jan       |
| (2) prijmeniVedouciho | Malik     |
| (3) jmenoVedouciho    | Alena     |
| (4) prijmeniVedouciho | Kadavá    |
| (5) jmenoVedouciho    | Pavla     |
| (6) prijmeniVedouciho | Koukalová |
| (7) jmenoVedouciho    | Karel     |
| (8) prijmeniVedouciho | Havlena   |

Obr. 14: Výsledek dotazu 9

*Dotaz 10:* Vypiš všechny „rodiče“ elementu děti.  
 //deti/parent::\*

| Path        | Attribute  |
|-------------|------------|
| (1) skupina | id="s0101" |
| (2) skupina | id="s0102" |
| (3) skupina | id="s0201" |
| (4) skupina | id="s0202" |

Obr. 15: Výsledek dotazu 10

## 5. Úkol – Transformace pomocí XSLT

Odevzdáno: do 28. 4. 2011

Obsah ZIP souboru: referat\_jandora1.pdf

referatHTML.xml + referatHTML.xls + referatHTML.png,  
 referatXHTML.xml + referatXHTML.xls + referatXHTML.png  
 referatXML.xml

**5.1. Zadání:** viz <http://ocw.cvut.cz/moodle/mod/assignment/view.php?id=8750>

Vytvořte a vyzkoušejte nejméně 3 transformace Vašeho dokumentu do XHTML, HTML a jiného XML. Výsledky a jejich dokumentaci vložte do kapitoly č. 5.

### 5.2. Zpracování:

#### Část A – transformace do HTML

Transformace do HTML je uvedena v souboru „referatHTML.xsl“ a PrintScreen výstupu v „referatHTML.png“.

V této části jsem uvedla přehled turnusů (včetně lokality a termínu konání) a skupin v těchto turnusech (včetně čísla skupiny a vedoucího).

**Výstupní kód pro HTML transformaci:**

```
<xsl:output method="html" encoding="utf-8" indent="yes" doctype-public="-//W3C//DTD HTML 4.01//EN"/>
```

**PrintScreen výstupu:**



**Transformace XML do HTML.**

Úloha 5. - část A: transformace původního XML souboru "referat.xml" do HTML.  
Z tématu "Ozdravné pobyt" je v této části uveden

## Přehled turnusů a skupin.

---

---

### Turnus t01

**Lokalita:** Hvar

**Termín:** 1. - 21. 7. 2011

---

**Číslo skupiny:** 0101

Vedoucí skupiny: Jan Malik

**Číslo skupiny:** 0102

Vedoucí skupiny: Alena Kadavá

---

### Turnus t02

**Lokalita:** Makarska

**Termín:** 10. - 31. 7. 2011

---

**Číslo skupiny:** 0201

Vedoucí skupiny: Pavla Koukalová

Hotovo



Obr. 16: Výstup transformace do HTML

**Část B – transformace do XHTML**

Transformace do XHTML je uvedena v souboru „referatXHTML.xsl“ a PrintScreen výstupu v „referatXHTML.png“.

V této části jsem uvedla přehled skupin (včetně čísla skupiny a vedoucího) a dětí do nich zařazených (včetně všech údajů, které jsou o nich uvedeny).

**Výstupní kód pro XHTML transformaci:**

```
<xsl:output method="html" encoding="utf-8" indent="yes" doctype-public="-//W3C//DTD HTML 4.01//EN"/>
```

**PrintScreen výstupu:**



**Transformace XML do XHTML.**

Úloha 5. - část B: transformace původního XML souboru "referat.xml" do XHTML.  
Z tématu "Ozdravné pobytu" je v této části uveden

## Přehled skupin a dětí do nich zařazených.

### Skupina s0101

Číslo skupiny: 0101  
Vedoucí skupiny: JanMalík

#### Děti

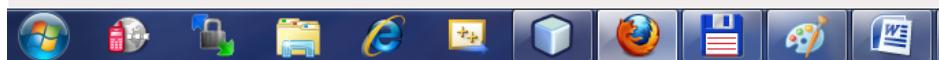
##### ID dítěte Příjmení Jméno Pohlaví Rok narození

|         |       |       |    |      |
|---------|-------|-------|----|------|
| d010101 | Miler | Petr  | Mr | 2001 |
| d010102 | Janda | Karel | Mr | 2001 |

### Skupina s0102

Číslo skupiny: 0102  
Vedoucí skupiny: AlenaKadavá

Hotovo



Obr. 17: Výstup transformace do XHTML

### Část C – transformace do jiného XML

Transformace do jiného XML je uvedena v souboru „referatXML.xsl“. V této části jsem transformovala původní XML na seznam účastníků pobytů.

## 6. Úkol - XQuery, případně DOM/SAX

Odevzdáno: do 12. 5. 2011

**Obsah ZIP souboru:** referat\_jandora1.pdf, referat.xml,  
dotazy = q01.query, q02.query, q03.query,  
výstupy = out01.xml, out02.xml, out03.xml,

**6.1. Zadání:** viz <http://ocw.cvut.cz/moodle/mod/assignment/view.php?id=8752>

Podklady : [SAX - ContentHandler](#), [DOM - Document](#)

Podmínky pro splnění DÚ:

- Vyzkoušejte si alespoň 3 dotazy v jazyce XQuery, které se týkají Vaší problematiky a Vašich dokumentů. V řešení uveďte zdrojové XML dokumenty, dotazy v XQuery a protokol o výsledcích dotazů získaných zvoleným nástrojem.

Pokud chcete získat body navíc (až 15 bodů), můžete vyřešit následující úkoly:

- Prostřednictvím rozhraní DOM nebo SAX implementovat vyhodnocení jednoho XQuery dotazu nad daty z předchozích úloh.
- Prostřednictvím rozhraní SAX spočítat zvolenou charakteristiku XML dokumentu (např. hloubku, počet elementů, průměrnou délku názvů elementů, počet atributů apod.) nebo XSD (např. počet složených datových typů, počet elementů se smíšeným obsahem, počet lokálně a globálně definovaných jednoduchých datových typů apod.). Každý si vymyslí vlastní téma.

- Prostřednictvím rozhraní DOM provést vybranou transformaci XML dokumentu nebo XSD, která změní jeho strukturu (např. obrátit pořadí elementů, převést atributy na elementy, převést elementy s jednoduchým datovým typem na atributy, převést všechny lokálně definované jednoduché/složené datové typy na globálně definované nebo obráceně apod.) Každý si vymyslí vlastní téma.

## 6.2. Zpracování:

*Dotaz 1:* Vypiš všechny chlapce zařazené do pobytů a seřaď je podle příjmení.

```
<deti>
{
for $dite in (fn:doc("referat.xml")//pobyty/turnus/skupina/deti/dite)
where $dite/pohlaviD = "Mr"
order by $dite/prijmeniD

return
<dite>
 {$dite/prijmeniD}
 {$dite/jmenoD}
</dite>
}
</deti>
```

```

1 <deti>
2 <dite>
3 <prijmeniD>Janda</prijmeniD>
4 <jmenoD>Karel</jmenoD>
5 </dite>
6 <dite>
7 <prijmeniD>Miler</prijmeniD>
8 <jmenoD>Petr</jmenoD>
9 </dite>
10 <dite>
11 <prijmeniD>Pavelka</prijmeniD>
12 <jmenoD>Martin</jmenoD>
13 </dite>
14 <dite>
15 <prijmeniD>Zindulka</prijmeniD>
16 <jmenoD>Jan</jmenoD>
17 </dite>
18 </deti>
19
20

```

The screenshot shows the XQuery editor interface with the following details:  
- \*\*Text tab\*\*: Active tab.  
- \*\*Grid, Schema, WSDL, XBRL, Authentic, Browser tabs\*\*: Other tabs available.  
- \*\*Toolbar\*\*: Buttons for file operations (New, Open, Save) and a refresh button.  
- \*\*Status bar\*\*: Shows the current files: referat, q01, and out01 \*.

Obr. 18: Výstup 1. dotazu XQuery.

*Dotaz 2:* Vypiš číslo skupiny a děti zařazené do skupny, kde je vedoucí Koukalová.

```
<skupina>
{
for $skupina in (fn:doc("referat.xml")//pobyty/turnus/skupina)
where $skupina/vedouci/prijmeniVedouciho = "Koukalová"
return
<skupina>
 {$skupina/cisloSkupiny}
 {$skupina/deti/dite/prijmeniD}
 {$skupina/deti/dite/jmenOD}
```

```

 </skupina>
}
</skupina>
```

```

1 <skupina>
2 <skupina>
3 <cisloSkupiny>0201</cisloSkupiny>
4 <prijmeniD>Novotná</prijmeniD>
5 <prijmeniD>Vokounová</prijmeniD>
6 <jmenoD>Petra</jmenoD>
7 <jmenoD>Ivana</jmenoD>
8 </skupina>
9 </skupina>
10 </skupina>
11
```

Obr. 19: Výstup 2. dotazu XQuery.

*Dotaz 3:* Vypiš číslo skupiny, příjmení a jméno dětí starších než děti narozené v roce 2002.

```

<skupina>
{
for $skupina in (fn:doc("referat.xml")//pobyty/turnus/skupina)
where $skupina/deti/dite/rokNarozeni < "2002"
order by $skupina/deti/dite/prijmeniD
return
 <skupina>
 {$skupina/cisloSkupiny}
 {$skupina/deti/dite/prijmeniD}
 {$skupina/deti/dite/rokNarozeni}
 </skupina>
}
</skupina>
```

```
1 <skupina>
2 <skupina>
3 <cisloSkupiny>0101</cisloSkupiny>
4 <prijmeniD>Miler</prijmeniD>
5 <prijmeniD>Janda</prijmeniD>
6 <rokNarozeni>2001</rokNarozeni>
7 <rokNarozeni>2001</rokNarozeni>
8 </skupina>
9 <skupina>
10 <skupina>
11 <cisloSkupiny>0201</cisloSkupiny>
12 <prijmeniD>Novotná</prijmeniD>
13 <prijmeniD>Vokounová</prijmeniD>
14 <rokNarozeni>1996</rokNarozeni>
15 <rokNarozeni>1996</rokNarozeni>
16 </skupina>
17 </skupina>
18
```

Obr. 20: Výstup 3. dotazu XQuery.

## Závěr

V rámci této semestrální práce jsem měla možnost prakticky si vyzkoušet vše, co bylo obsahem jednotlivých přednášek k předmětu AD7B36XML. Vzhledem k tomu, že ve své praxi nemám možnost se s tvorbou dokumentů XML setkat, velmi jsem tuto možnost uvítala.

## Zdroje

- [1] Přednášky k předmětu AD7B36XML. Dostupné z WWW: <<http://ocw.cvut.cz/moodle>>.
- [2] MLÝNKOVÁ, I., NEČASNÝ, Michal, POKORNÝ, RICHTA, Karel, KAMIL, VOJTĚCH. *Technologie XML, Principy a aplikace v praxi*, ISBN: 978-80-247-2725-7, vydáno 2008.
- [3] SW: *Exchanger XML Editor*. Verze 3.3.1. Dostupné z WWW: <<http://www.freexmleditor.com>>.
- [4] KOSEK, Jiří. *Profesionální školení a poradenství v oblasti technologií XML, RELAX NG*. Video z cyklu Programátorské večery, rok 2005. Dostupné z WWW: <<http://www.avc-cvut.cz/avc.php?id=2309>> [cit. 2011-03-04].
- [5] SW: *Validátor*. Dostupné z WWW: <<http://www.w3.org/2001/03/webdata/xsv>> [cit. 2011-03-04].
- [6] SW: *Altova XML Spy*. Dostupné z WWW: <<http://ocw.cvut.cz/moodle>>.