

ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ІВАНА ФРАНКА

Факультет прикладної математики та інформатики

Бази даних та інформаційні системи

ЛАБОРАТОРНА РОБОТА №15

Тема: «XSLT — мова перетворення XML-документів»

Виконав:

Ст. Лук'янчук Денис

Група ПМІ-33

2025

Тема: «XSLT — мова перетворення XML-документів».

Мета роботи: Ознайомлення з синтаксисом мови XSLT та перетворенням фрагмента XML документа з використанням XSLT.

Завдання лабораторної роботи:

- Відобразити вибрані за певною умовою (значенням елементів, атрибутів, агрегованим значенням) дані у вигляді таблиці та списку з використанням різних шрифтів, виділень колонок чи рядків, сортування.
- Створити шаблон перетворення XSLT фрагмента свого XML документа з обов'язковим використанням `<value-of>`, `<for-each>`, `<sort>`, `<if>/<choose>`, `<number>`.
- Використати довільний XSLT processor, наприклад для перевірки коректності роботи створеного шаблона перетворення й отримання, власне, коду перетворення. Зберегти код у текстовий файл з розширенням html і відкрити його браузером для демонстрації результату перетворення

Теоретичний матеріал:

XSLT — це мова, яка використовується для опису перетворень XML-документів. Вона дозволяє взяти XML як вхід, пройтись по його елементах і створити новий документ (наприклад HTML-сторінку).

Основні поняття

- **XML-джерело** — документ, який містить дані (наприклад, список CD-дисків).
- **XSLT-таблиця стилів (XSL)** — правила, які описують, як перетворювати ці дані.
- **XSLT-процесор** — програма, яка виконує це перетворення (у браузері або на сайті).

2. Основні конструкції

- `<xsl:value-of>` — виводить текстове значення вибраного вузла.
- `<xsl:for-each>` — виконує ітерацію по множині елементів.
- `<xsl:sort>` — сортує вибірку за певним елементом.
- `<xsl:if>` та `<xsl:choose>` — дозволяють задавати умови виведення.
- `<xsl:number>` — нумерація елементів або позицій у послідовності.

Хід роботи

Для виконання роботи використано мій xml файл:

```
▼<shop>
  ▼<customer>
    <id>1</id>
    <name>Customer_1</name>
    <email>user1@example.com</email>
    ▼<orders>
      <orders/>
    </orders>
  </customer>
  ▼<customer>
    <id>2</id>
    <name>Customer_2</name>
    <email>user2@example.com</email>
    ▼<orders>
      <orders/>
    </orders>
  </customer>
  ▼<customer>
    <id>3</id>
    <name>Customer_3</name>
    <email>user3@example.com</email>
    ▼<orders>
      ▼<order>
        <order_id>16</order_id>
        <date>2025-02-17T23:49:18.683291</date>
        <status>New</status>
        <quantity>4</quantity>
        ▼<part>
          <serial>SN000064</serial>
          <part_name>Part_64</part_name>
          <price>306.80</price>
        </part>
      </order>
    </orders>
  </customer>
  ▼<customer>
    <id>4</id>
    <name>Customer_4</name>
    <email>user4@example.com</email>
    ▼<orders>
      <orders/>
    </orders>
  </customer>
```

Створено XML-документ:

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <xsl:stylesheet version="1.0"
3   |   xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
4
5  <xsl:output method="html" encoding="UTF-8" indent="yes"/>
6
7  <xsl:template match="/">
8   <html>
9    <head>
10   <meta charset="UTF-8"/>
11   <title>Customer Orders Report</title>
12   <style>
13    body { font-family: Arial, sans-serif; background-color: #f8f8f8; }
14    h2 { color: #004d99; text-align: center; }
15    table { border-collapse: collapse; width: 90%; margin: 20px auto; }
16    th, td { border: 1px solid #777; padding: 8px; text-align: center; }
17    th { background-color: #d1e7ff; }
18    tr:nth-child(even) { background-color: #f2f2f2; }
19    .highprice { background-color: #ffd699; font-weight: bold; }
20    .cancelled { color: red; font-style: italic; }
21    ul { width: 80%; margin: 0 auto; }
22   </style>
23  </head>
24
25  <body>
26   <h2>Звіт про замовлення клієнтів</h2>
27
28  <table>
29   <tr>
30    <th>№</th>
31    <th>Ім'я клієнта</th>
32    <th>Номер замовлення</th>
33    <th>Статус</th>
34    <th>Ціна</th>
35   </tr>
36
37  <xsl:for-each select="//order[status='Completed']">
38   <xsl:sort select=".../../name"/>
39   <tr>
40    <td><xsl:value-of select="position()"/></td>
41    <td><xsl:value-of select=".../../name"/></td>
42    <td><xsl:value-of select="order_id"/></td>
43    <td><xsl:value-of select="status"/></td>
44
45  <xsl:choose>
46   <xsl:when test="part/price > 500">
47    <td class="highprice"><xsl:value-of select="part/price"/></td>
48   </xsl:when>
49   <xsl:otherwise>
50    <td><xsl:value-of select="part/price"/></td>
51   </xsl:otherwise>
52  </xsl:choose>
53  </tr>
54 </xsl:for-each>
55 </table>
56
57  <h2>Скасовані замовлення</h2>
58 <ul>
59 <xsl:for-each select="//order[status='Cancelled']">
60  <xsl:sort select=".../../name"/>
61  <li class="cancelled">
62   <xsl:value-of select="position()"/>.
63   <xsl:value-of select=".../../name"/> -
64   <xsl:value-of select="part/part_name"/>
65   (<xsl:value-of select="part/price"/> грн)
66  </li>
67 </xsl:for-each>
68 </ul>
69
70  </body>
71 </html>
72 </xsl:template>
73
74 </xsl:stylesheet>
```

Згенеровано HTML файл:

Option 1: Copy-paste your XML document here

Copy-paste your XML here

Option 2: Or upload your XML file

Огляд... БД.xml

File encoding

UTF-8

Option 1: Copy-paste your XSL document here (Optional if XSD referred in XML using schemaLocation)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

  <xsl:output method="html" encoding="UTF-8" indent="yes"/>
```

Option 2: Or upload your XSL document

Огляд... Файл не выбрано.

File encoding

UTF-8

Transform XML

Transform XML to new window

-Transformed Document-

```
<html>
  <head>
    <meta charset="UTF-8"/>
    <title>Customer Orders Report</title>
    <style>
      body { font-family: Arial, sans-serif; background-color: #f8f8f8; }
      h2 { color: #004d99; text-align: center; }
      table { border-collapse: collapse; width: 90%; margin: 20px auto; }
      th, td { border: 1px solid #777; padding: 8px; text-align: center; }
      th { background-color: #d1e7ff; }
      tr:nth-child(even) { background-color: #f2f2f2; }
```

Copy

Save

Відтворено HTML файл:

```
1 <html>
2   <head>
3     <meta charset="UTF-8"/>
4     <title>Customer Orders Report</title>
5     <style>
6       body { font-family: Arial, sans-serif; background-color: #f8f8f8; }
7       h2 { color: #004d99; text-align: center; }
8       table { border-collapse: collapse; width: 90%; margin: 20px auto; }
9       th, td { border: 1px solid #777; padding: 8px; text-align: center; }
10      th { background-color: #d1e7ff; }
11      tr:nth-child(even) { background-color: #f2f2f2; }
12      .highprice { background-color: #ffd699; font-weight: bold; }
13      .cancelled { color: red; font-style: italic; }
14      ul { width: 80%; margin: 0 auto; }
15    </style>
16  </head>
17  <body>
18    <h2>Звіт про замовлення клієнтів</h2>
19    <table>
20      <tr>
21        <th>№</th>
22        <th>Ім'я клієнта</th>
23        <th>Номер замовлення</th>
24        <th>Статус</th>
25        <th>Ціна</th>
26      </tr>
27      <tr>
28        <td>1</td>
29        <td>Customer_10</td>
30        <td>58</td>
31        <td>Completed</td>
32        <td class="highprice">673.47</td>
33      </tr>
34      <tr>
35        <td>2</td>
36        <td>Customer_14</td>
37        <td>70</td>
38        <td>Completed</td>
39        <td>453.44</td>
40      </tr>
41      <tr>
42        <td>3</td>
43        <td>Customer_17</td>
44        <td>38</td>
45        <td>Completed</td>
46        <td>225.82</td>
47      </tr>
48      <tr>
49        <td>4</td>
50        <td>Customer_20</td>
51        <td>14</td>
52        <td>Completed</td>
53        <td class="highprice">710.65</td>
54      </tr>
```

Таблиця і список з результатами прочитаний браузером HTML:

Звіт про замовлення клієнтів				
№	Ім'я клієнта	Номер замовлення	Статус	Ціна
1	Customer_10	58	Completed	673.47
2	Customer_14	70	Completed	453.44
3	Customer_17	38	Completed	225.82
4	Customer_20	14	Completed	710.65
5	Customer_32	54	Completed	914.26
6	Customer_39	78	Completed	101.61
7	Customer_42	10	Completed	675.90
8	Customer_43	62	Completed	851.39
9	Customer_45	86	Completed	186.79
10	Customer_52	66	Completed	826.98
11	Customer_57	50	Completed	640.53
12	Customer_59	34	Completed	110.01
13	Customer_59	22	Completed	509.85
14	Customer_6	82	Completed	182.23
15	Customer_7	26	Completed	166.18
16	Customer_70	94	Completed	757.81
17	Customer_74	18	Completed	406.79
18	Customer_74	46	Completed	349.45
19	Customer_74	30	Completed	927.99
20	Customer_8	42	Completed	661.17
21	Customer_9	74	Completed	253.75
22	Customer_91	98	Completed	99.13
23	Customer_92	90	Completed	14.64
24	Customer_93	6	Completed	774.58
25	Customer_96	2	Completed	451.65

Скасовані замовлення
<ul style="list-style-type: none"> • 1. Customer_100 — Part_44 (945.43 грн) • 2. Customer_11 — Part_17 (525.37 грн) • 3. Customer_12 — Part_56 (661.17 грн) • 4. Customer_24 — Part_38 (126.25 грн) • 5. Customer_24 — Part_53 (535.57 грн) • 6. Customer_25 — Part_85 (394.11 грн) • 7. Customer_30 — Part_47 (113.98 грн) • 8. Customer_30 — Part_71 (836.72 грн) • 9. Customer_41 — Part_51 (987.36 грн) • 10. Customer_45 — Part_7 (993.06 грн) • 11. Customer_52 — Part_70 (614.94 грн) • 12. Customer_58 — Part_33 (774.58 грн) • 13. Customer_60 — Part_99 (569.67 грн) • 14. Customer_61 — Part_35 (119.65 грн) • 15. Customer_62 — Part_71 (836.72 грн) • 16. Customer_71 — Part_56 (661.17 грн) • 17. Customer_71 — Part_70 (614.94 грн) • 18. Customer_76 — Part_24 (612.30 грн) • 19. Customer_83 — Part_63 (590.84 грн) • 20. Customer_85 — Part_38 (126.25 грн) • 21. Customer_88 — Part_6 (115.14 грн) • 22. Customer_90 — Part_35 (119.65 грн) • 23. Customer_91 — Part_75 (38.94 грн) • 24. Customer_94 — Part_63 (590.84 грн) • 25. Customer_94 — Part_81 (225.82 грн)

Висновок:

У результаті виконання лабораторної роботи було розроблено XSLT-шаблон для перетворення XML-документа у формат HTML. Завдяки використанню конструкцій `<xsl:value-of>`, `<xsl:for-each>`, `<xsl:sort>`, `<xsl:choose>` та `position()`, отримано динамічну таблицю з умовним форматуванням і стабільною нумерацією. Робота показала практичне застосування XSLT у відображені структурованих XML-даних у зручному для користувача вигляді.

Відповіді на контрольні питання

1. Для чого призначена мова XSLT?

Мова XSLT призначена для перетворення XML-документів у інші формати (HTML, XML, текст тощо). Вона дозволяє відокремити структуру даних від способу їх подання.

2. За допомогою якого механізму досягається перетворення XML-документу?

Перетворення XML-документу здійснюється за допомогою шаблонів (templates), які визначають, як обробляти певні елементи або вузли XML.

3. Яка структура декларації стилю?

Декларація стилю починається з:

```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

або

```
<xsl:transform version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

і містить набір шаблонів <xsl:template>...</xsl:template>.

4. Що таке шаблон перетворення і з яких частин він складається?

Шаблон перетворення — це правило, що описує, як відобразжати певний елемент XML.

Він має дві частини:

- умову вибору (match) — який елемент обробляється;
- вміст шаблону — як саме елемент буде відображенено.

5. Як отримати значення вибраного вузла та додати його до вихідного потоку перетворення?

Для цього використовується елемент:

```
<xsl:value-of select="назва_елемента"/>
```

Він вставляє текстове значення вибраного вузла у результат.

6. Який оператор використовується для вибору кожного XML-елемента певного набору вузлів?

Оператор <xsl:for-each>:

```
<xsl:for-each select="shop/customer/orders/order">
```

...

</xsl:for-each>

виконує дії для кожного елемента у вибірці.

7. Як можна відсортувати та пронумерувати результати виводу?

- Для сортування — використовується <xsl:sort select="..." /> всередині <xsl:for-each>.
- Для нумерації — <xsl:number> або функція position() для послідовних номерів.

8. Наведіть приклад встановлення умови виводу вмісту файлу XML.

```
<xsl:if test="price > 500">  
    <td class="highprice"><xsl:value-of select="price"/></td>  
</xsl:if>
```

9. Як здійснити вивід за однією з кількох умов?

За допомогою конструкції <xsl:choose>:

```
<xsl:choose>  
    <xsl:when test="status='Completed'">...</xsl:when>  
    <xsl:otherwise>...</xsl:otherwise>  
</xsl:choose>
```

10. Який елемент застосовує правило шаблону до поточного елемента або дочірніх вузлів поточного елемента?

Елемент <xsl:apply-templates>:

```
<xsl:apply-templates select="child::node()"/>
```

використовується для рекурсивного застосування шаблонів до дочірніх елементів.