

ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ІВАНА ФРАНКА

Факультет прикладної математики та інформатики

**Бази даних та інформаційні системи**

**ЛАБОРАТОРНА РОБОТА №12**

**Тема: «DTD-схема XML-документа»**

Виконав:

Ст. Лук'янчук Денис

Група ПМІ-33

2025

**Тема:** «DTD-схема XML-документа».

**Мета роботи:** Практичне ознайомлення з конструкціями DTD-схеми XML документа, її створенням і використанням.

### Завдання лабораторної роботи:

1. Опрацювати теоретичний матеріал щодо DTD-схем і правил формування XML-документів.
2. Створити DTD-схему для XML-документа `<shop>`.
3. Використати онлайн-валідатор для перевірки коректності та валідності XML-документа з DTD.
4. Оформити звіт з кодом XML та DTD, а також результатами перевірки.

### Теоретичний матеріал

- XML-документ є правильно сформованим (well-formed), якщо він відповідає всім синтаксичним правилам XML.
- XML-документ є коректним (valid), якщо він правильно сформований і відповідає обмеженням, визначенім у DTD, XML Schema або RELAX NG.
- DTD (Document Type Definition) визначає структуру XML-документа: порядок елементів, обов'язкові та необов'язкові елементи, порожні елементи та текстовий вміст (#PCDATA).
- За допомогою DTD можна також визначати атрибути елементів і обмежувати допустимі значення.

### Хід роботи

1. Взял готовий XML-документ `<shop>` із минулої лабораторної роботи клієнта.

```
Formatted XML
1 <shop>
2   <customer>
3     <id>85</id>
4     <name>Customer_85</name>
5     <email>user85@example.com</email>
6     <orders>
7       <order>
8         <order_id>1</order_id>
9         <date>2024-12-17T19:17:52.265148</date>
10        <status>Completed</status>
11        <quantity>21</quantity>
12        <part>
13          <serial>SN000035</serial>
14          <part_name>Part_35</part_name>
15          <price>81.72</price>
16        </part>
17      </order>
18    <order>
19      <order_id>99</order_id>
20      <date>2024-12-21T08:35:52.862473</date>
21      <status>Cancelled</status>
22      <quantity>9</quantity>
```

```

23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42 </shop>

```

2. Створив вбудовану DTD-схему для документа.

3. Документ перевірив через онлайн-валідатор TRUUGO.

```

1  <!DOCTYPE shop [
2    <!ELEMENT shop (customer+)>
3    <!ELEMENT customer (id, name, email, orders)>
4    <!ELEMENT id (#PCDATA)>
5    <!ELEMENT name (#PCDATA)>
6    <!ELEMENT email (#PCDATA)>
7    <!ELEMENT orders (order+)>
8    <!ELEMENT order (order_id, date, status, quantity, part)>
9    <!ELEMENT order_id (#PCDATA)>
10   <!ELEMENT date (#PCDATA)>
11   <!ELEMENT status (#PCDATA)>
12   <!ELEMENT quantity (#PCDATA)>
13   <!ELEMENT part (serial, part_name, price)>
14   <!ELEMENT serial (#PCDATA)>
15   <!ELEMENT part_name (#PCDATA)>
16   <!ELEMENT price (#PCDATA)>
17 ]>
18 <shop>

```

| ▼ Validation result  |         |
|--|---------|
| Syntax wellformed  | PASSED  |
| DTD validation   | PASSED  |
| XSD validation   | OMITTED |
| No schema reference provided using either xsi:schemaLocation or<br>xsi:noNamespaceSchemaLocation attribute.  |         |
| Want to cover various integrity and conditional requirements as well?<br>Check <a href="#">video tutorials</a> to setup test profiles for professional use cases.<br><a href="#">Sign up now »</a> |         |

## Висновок:

У лабораторній роботі я ознайомився з принципами створення DTD-схем XML-документів. Було реалізовано структуру XML-документа `<shop>` та визначено всі елементи і порядок їх появи. Використання вбудованої DTD-схеми дозволило перевірити коректність і валідність документа за допомогою онлайн-валідатора. Робота дозволила закріпити практичні навички визначення структури XML-документів та їх перевірки на відповідність DTD.

## **Відповіді на контрольні питання**

### **1. Що дозволяє визначити схема XML документа?**

Схема XML документа (DTD або XML Schema) дозволяє визначити:

- Структуру документа (які елементи можуть бути присутні, їх порядок та вкладеність).
- Обов'язкові та необов'язкові елементи.
- Типи даних для елементів (у випадку XML Schema).
- Обмеження на значення елементів та атрибутів.
- Коректність документа при перевірці валідності (validity).

### **2. Як описується складний елемент у DTD схемі?**

Складний елемент — це елемент, який містить інші елементи всередині, а не просто текст. В DTD він описується через перелік дочірніх елементів у дужках:

*<!ELEMENT order (order\_id, date, status, quantity, part)>*

Тут `<order>` містить обов'язкові елементи `<order_id>`, `<date>`, `<status>`, `<quantity>` і `<part>`.

### **2. Як описується простий елемент текстового типу в DTD схемі?**

Простий текстовий елемент (що містить тільки текст) позначається як #PCDATA:

*<!ELEMENT name (#PCDATA)>*

#PCDATA означає, що всередині цього елемента може бути тільки текст, а інші елементи вкладені бути не можуть.

### **4. Які спеціальні символи використовуються в DTD схемі для вказання обмежень для елементів?**

- ? – елемент необов'язковий (може з'являтися 0 або 1 раз).
- \* – елемент може з'являтися нуль або більше разів.
- + – елемент повинен з'являтися один або більше разів.
- , – використовується для послідовності елементів.
- | – використовується для вибору одного з декількох елементів.
- () – групування елементів для комбінацій чи опцій.

## **5. Як описуються атрибути елемента?**

Атрибути описуються за допомогою декларації ATTLIST:

```
<!ATTLIST city state CDATA #REQUIRED>
```

Тут:

- city – елемент, для якого визначається атрибут.
- state – ім'я атрибута.
- CDATA – тип значення (текстові дані).
- #REQUIRED – атрибут обов'язковий.  
Можливі також значення #IMPLIED (необов'язковий) або #FIXED "value" (фіксоване значення).

## **6. Яка є рекомендація стосовно використання атрибутів у XML документі?**

- Атрибути краще використовувати для метаданих (додаткова інформація про елемент, але не самі дані).
- Основні дані слід зберігати як дочірні елементи, оскільки вони більш гнучкі, дозволяють вкладеність і краще розширяються.

## **7. Що означає в DTD схемі поняття “сутність”?**

Сутність (entity) – це ярлик або змінна, яка замінюється певним вмістом під час обробки XML.

- Внутрішня сутність визначає текст у самому DTD:

```
<!ENTITY writer "Donald Duck.">
```

- Зовнішня сутність посилається на зовнішній файл чи ресурс:

```
<!ENTITY writer SYSTEM "https://example.com/writer.txt">
```

Сутності використовуються для повторного використання однакових даних або посилань у XML-документі.