

Бази даних та інформаційні системи

ЛАБОРАТОРНА РОБОТА №4

DTD схема XML документа

Виконав:

Ст. Заречанський

Олексій

Група ПМІ-33

Оцінка

Прийняв:

ас. Жировецький В.В.

Тема

Вивчення поняття DTD схема XML документа.

Мета роботи

Ознайомлення з конструкціями DTD схеми XML документа, її створенням і використанням.

Завдання

Розробити базу даних для web представлення системи управління документами. Система підтримує інформацію про документи, які проходять через неї. Крім власне документа важливими є інформація про його створення, історія змін, користувачі, які вносили зміни, та інформація про оффлайнове представлення документа (паперовий, CD, flash). Користувачі можуть створювати і змінювати документи і асоціювати їх з оффлайновим представленням, а також шукати документи за користувачами, які змінювали їх, станом чи представленням.

Звіт

1. Використаю XML документ з минулої лабораторної роботи:

The screenshot displays a SQL Query Editor interface. The top bar has tabs for 'Query Editor' and 'Query History'. The main area contains a SQL query with line numbers 1 through 14. The query uses nested subqueries and XML functions like `xmlforest` and `xmlagg` to create an XML document from database records. Below the query editor, there are tabs for 'Data Output', 'Explain', 'Notifications', and 'Messages'. The 'Data Output' tab is active, showing a table with two columns: 'xmlelement' and 'xml'. The first row of data shows the root element '`<users><user><e...`'.

```
1 with result as (  
2     select (xmlelement(name user, (  
3         xmlforest(customer.email, customer.surname, (  
4             xmlagg(  
5                 xmlelement(name document, (  
6                     xmlforest(document.name, document.creation_date, document.content))))  
7             as docs)))) as users  
8     from document  
9     join customer on customer.email = document.owner  
10    where document.owner = customer.email  
11    group by customer.email)  
12  
13 select xmlelement (name users, xmlagg(result.users))  
14 from result
```

	xmlelement	xml
1		<users><user><e...

З цієї квері з минулої лабораторної роботи ми отримали XML документ, який містить всіх користувачів, які є власниками документів в базі даних. Формується XML документ де вказані дані користувача та список всіх документів власниками яких вони є, з даними про ці документи.

Результат виконання запиту:

```
<?xml version="1.0" encoding="UTF-8"?>
<users>
  <user>
    <email>anotheremail228@mail.ua</email>
    <surname>Chungus</surname>
    <docs>
      <document>
        <name>Third document</name>
        <creation_date>2021-03-01</creation_date>
        <content>Another user</content>
      </document>
      <document>
        <name>Empty document</name>
        <creation_date>2021-03-01</creation_date>
        <content>empty</content>
      </document>
    </docs>
  </user>
  <user>
    <email>anotheremail@gmail.com</email>
    <surname>Floppa</surname>
    <docs>
      <document>
        <name>Last document</name>
        <creation_date>2021-03-01</creation_date>
        <content>Not empty</content>
      </document>
    </docs>
  </user>
  <user>
    <email>last@mail.ua</email>
    <surname>Shreker</surname>
    <docs>
      <document>
        <name>Second document</name>
        <creation_date>2021-03-01</creation_date>
        <content>Some more content</content>
      </document>
      <document>
        <name>First document</name>
        <creation_date>2021-03-01</creation_date>
        <content>Some content12</content>
      </document>
    </docs>
  </user>
</users>
```

2. Створюємо DTD схему на основі цих XML документу та провалідуємо її разом з XML документом:

```
XML Editor ▼
1 <?xml version="1.0" encoding="utf-8"?>
2 <!DOCTYPE users [
3   <!ELEMENT users (user*)>
4   <!ELEMENT user (email, surname, docs)>
5   <!ELEMENT docs (document+)>
6   <!ELEMENT document (name, creation_date, content)>
7   <!ELEMENT email (#PCDATA)>
8   <!ELEMENT surname (#PCDATA)>
9   <!ELEMENT name (#PCDATA)>
10  <!ELEMENT creation_date (#PCDATA)>
11  <!ELEMENT content (#PCDATA)>
12 ]>
13 <users>
14   <user>
15     <email>anotheremail228@mail.ua</email>
16     <surname>Chungus</surname>
17     <docs>
18       <document>
19         <name>Third document</name>
20         <creation_date>2021-03-01</creation_date>
```

XML Editor ▼

1 <?xml version="1.0" encoding="utf-8"?>
2 <!DOCTYPE users [
3 <!ELEMENT users (user*)>
4 <!ELEMENT user (email, surname, docs)>
5 <!ELEMENT docs (document+)>
6 <!ELEMENT document (name, creation_date, co
7 <!ELEMENT email (#PCDATA)>
8 <!ELEMENT surname (#PCDATA)>
9 <!ELEMENT name (#PCDATA)>
10 <!ELEMENT creation_date (#PCDATA)>
11 <!ELEMENT content (#PCDATA)>
12]>
13 <users>
14 <user>
15 <email>anotheremail228@mail.ua</email>
16 <surname>Chungus</surname>
17 <docs>
18 <document>
19 <name>Third document</name>
20 <creation_date>2021-03-01</creation_date>

▼ Validation result

Syntax wellformed PASSED

DTD validation PASSED

XSD validation OMITTED

No schema reference provided using either xsi:schemaLocation or
xsi:noNamespaceSchemaLocation attribute.

Cover format, integrity and conditional restrictions as well? Check [video tutorials](#)
on how to create test profiles and share your test reports ([examples](#)) with ease.

[Create free account »](#)

UPLOAD... LOAD URL FOLLOW US ON LINKEDIN BEAUTIFY | MINIFY VALIDATE XML

- Наша схема включає елемент users, який має мати елементи user, мінімум 0 (символ *).
- User має мати електронну пошту, прізвище, документи обов'язково в такому порядку.
- Docs має мати 1 або більше (символ +) документів (якщо юзер не є власником документів, то його і не буде в цьому XML документі, бо тут лише власники документів).
- Document має назву, дату створення та вміст, всі поля обов'язкові та мають бути в такому порядку.
- Поля електронної пошти, прізвища, назви документу, дати створення та вмісту містять текст, а не внутрішні XML елементи.