

Lab 02

DTDs

Phần I - Thực hiện trong 120 phút

1.1 Mục tiêu

- ✓ Hiểu biết khái niệm DTD.
- ✓ Sử dụng DTD để định nghĩa cấu trúc tài liệu XML.
- ✓ Tạo một tin DTD mới để khai báo DTD.
- ✓ Khai báo Phần tử, Thuộc tính và Thực thể.
- ✓ Tạo tài liệu XML có ràng buộc với DTD.
- ✓ Kiểm tra tính hợp lệ của tài liệu XML.

1.2 Thực hiện

Ôn tập kiến thức

DTD - Example

```
1  <?xml version="1.0"?>
2  <!DOCTYPE name [
3    <!ELEMENT name (first, middle, last)>
4    <!ELEMENT first (#PCDATA)>
5    <!ELEMENT middle (#PCDATA)>
6    <!ELEMENT last (#PCDATA)>
7  ]>
8  <name>
9    <first>Joseph</first>
10   <middle>John</middle>
11   <last>Fawcett</last>
12 </name>
```

1. DTD là gì?

Tài liệu XML sử dụng các cặp thẻ tự định nghĩa nên do đó có thể sẽ bị sai sót trong quá trình viết ví dụ như sai tên thẻ, sai thuộc tính, giá trị lưu trữ không phù hợp quy tắc. Do vậy XML đưa ra khái niệm tài liệu DTD để định nghĩa dữ liệu trong tài liệu XML. DTD xác định những thẻ nào xuất hiện trong tài liệu XML, trong thẻ có những thẻ thành viên nào...

2. Sử dụng DTD như nào

Có 2 cách sử dụng DTD, viết trực tiếp trên file XML và viết ở file bên ngoài rồi gọi vào xml

XML document with an internal DTD

```
<?xml version="1.0"?>
<!DOCTYPE note [
  <!ELEMENT note (to,from,heading,body)>
  <!ELEMENT to (#PCDATA)>
  <!ELEMENT from (#PCDATA)>
  <!ELEMENT heading (#PCDATA)>
  <!ELEMENT body (#PCDATA)>
]>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend</body>
</note>
```

Viết trực tiếp trên file xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE note SYSTEM "Note.dtd">
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

Gọi từ file bên ngoài

DTD ràng buộc tài liệu XML bởi:

- Elements
- Attributes
- Entities
- PCDATA
- CDATA

1. Element

Dùng để định nghĩa các thẻ XML được sử dụng

```
<!ELEMENT note (to,from,heading,body)>  
<!ELEMENT to (#PCDATA)>  
<!ELEMENT from (#PCDATA)>  
<!ELEMENT heading (#PCDATA)>  
<!ELEMENT body (#PCDATA)>
```

```
<!ELEMENT element-name (#PCDATA)>
```

Example:

```
<!ELEMENT from (#PCDATA)>
```

Element với dữ liệu PCDATA

```
<!ELEMENT element-name ANY>
```

Example:

```
<!ELEMENT note ANY>
```

Element với dữ liệu ANY

```
<!ELEMENT element-name (child1)>  
or  
<!ELEMENT element-name (child1,child2,...)>  
  
Example:  
  
<!ELEMENT note (to,from,heading,body)>
```

Element với các element con khác (lưu ý là có tuần tự)

```
<!ELEMENT element-name (child-name+)>  
  
Example:  
  
<!ELEMENT note (message+)>
```

Element có thể xác định: +(tối thiểu có 1), *(tối thiểu 0), ? (có 0 hoặc 1)

2. Attributes

Thuộc tính được định nghĩa như sau:

```
<!ATTLIST element-name attribute-name attribute-type attribute-value>  
  
DTD example:  
  
<!ATTLIST payment type CDATA "check">  
  
XML example:  
  
<payment type="check" />
```

Các giá trị có thể chọn mặc định cho thuộc tính:

DTD:

```
<!ATTLIST payment type (check|cash) "cash">
```

XML example:

```
<payment type="check" />
```

or

```
<payment type="cash" />
```

3. Entity

Entity định nghĩa những ký tự viết tắt của một cụm từ.

DTD Example:

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

```
<!ENTITY copyright "Copyright W3Schools.">
```

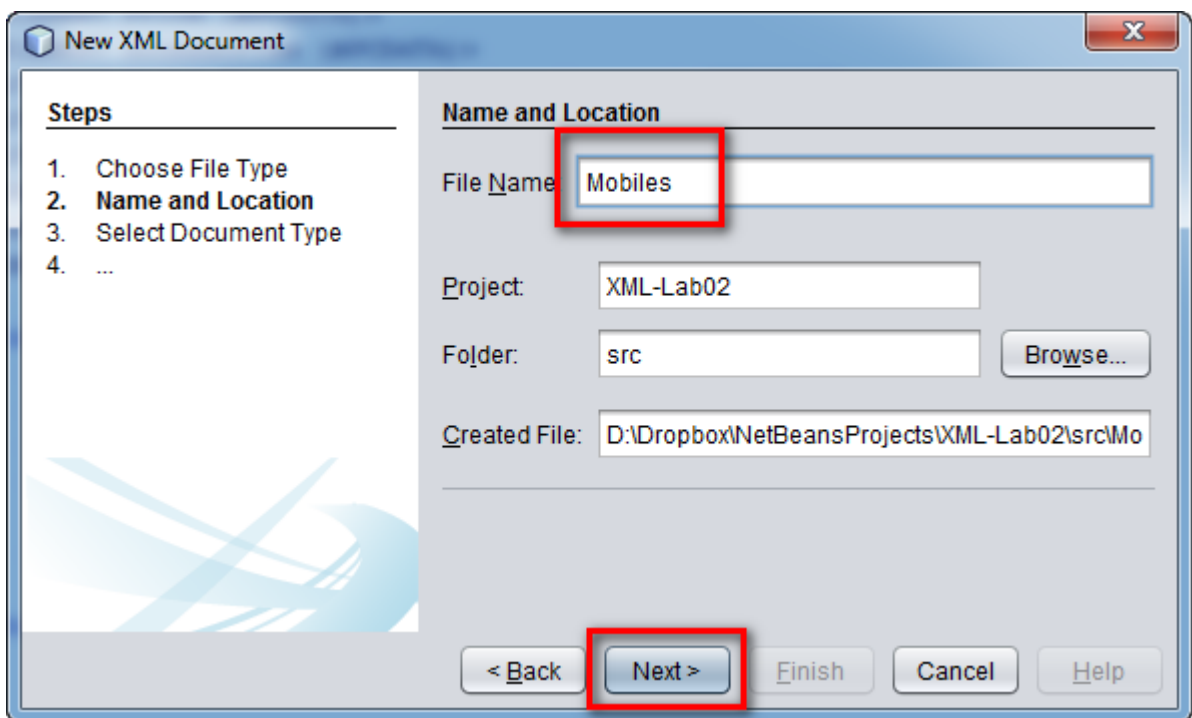
XML example:

```
<author>&writer;&copyright;</author>
```

BÀI THỰC HÀNH

Bài thực hành 0: Tạo một tập tin Mobiles.xml, viết định nghĩa DTD trong file.

Bước 1: Tạo mới một file XML



Bước 2: định nghĩa DTD trong file xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Mobiles
[
  <!ELEMENT Mobiles (Mobile*)>
  <!ELEMENT Mobiles (Company, Model, Price, Accessories)>
  <!ELEMENT Company (#PCDATA)>
  <!ELEMENT Model (#PCDATA)>
  <!ELEMENT Price (#PCDATA)>
  <!ELEMENT Accessories (#PCDATA)>
  <!ATTLIST Model Type (cam | web | pc | mouse) #REQUIRED>
  <!ATTLIST Model Code CDATA "Camera">
  <!ATTLIST Model Fac CDATA "Camera">
  <!ENTITY HP "Head Phones">
  <!ENTITY CH "Charger">
  <!ENTITY SK "Starter's Kit">
]>
<Mobiles>

</Mobiles>
```

Bước 3:Viết xml theo định nghĩa DTD trên

```

<Mobiles>
  <Mobile>
    <Company >&HP;</Company>
    <Model Type="web" Code=""></Model>
    <Price ></Price>
    <Accessories></Accessories>
  </Mobile>
</Mobiles>

```

Ctrl + space > hiển thị gợi ý cho Type

```

<Mobiles>
  <Mobile>
    <Company >&HP;</Company>
    <Model Type="web" |></Model>
    <Price ></Price>
    <Accessories></Accessories>
  </Mobile>
</Mobiles>

```

Hiển thị gợi ý cho thuộc tính của Model

```

<Mobiles>
  <Mobile>
    <Company >&HP;</Company>
    <Model Type="web"></Model>
    <Price>$120</Price>
    <Accessories>In Hand</Accessories>
  </Mobile>
  <Mobile>
    <Company >&CH;</Company>
    <Model Type="cam"></Model>
    <Price>$60</Price>
    <Accessories>Free online</Accessories>
  </Mobile>
  <Mobile>
    <Company >&SK;</Company>
    <Model Type="mouse"></Model>
    <Price>$9</Price>
    <Accessories>Dinanosauros</Accessories>
  </Mobile>
</Mobiles>

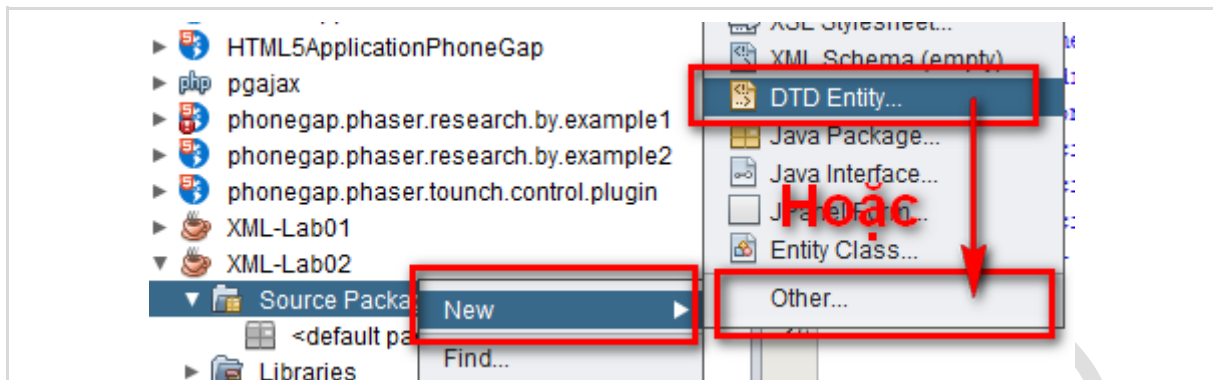
```

Bài thực hành 1: Tạo một tập tin DTD mới để khai báo DTD.

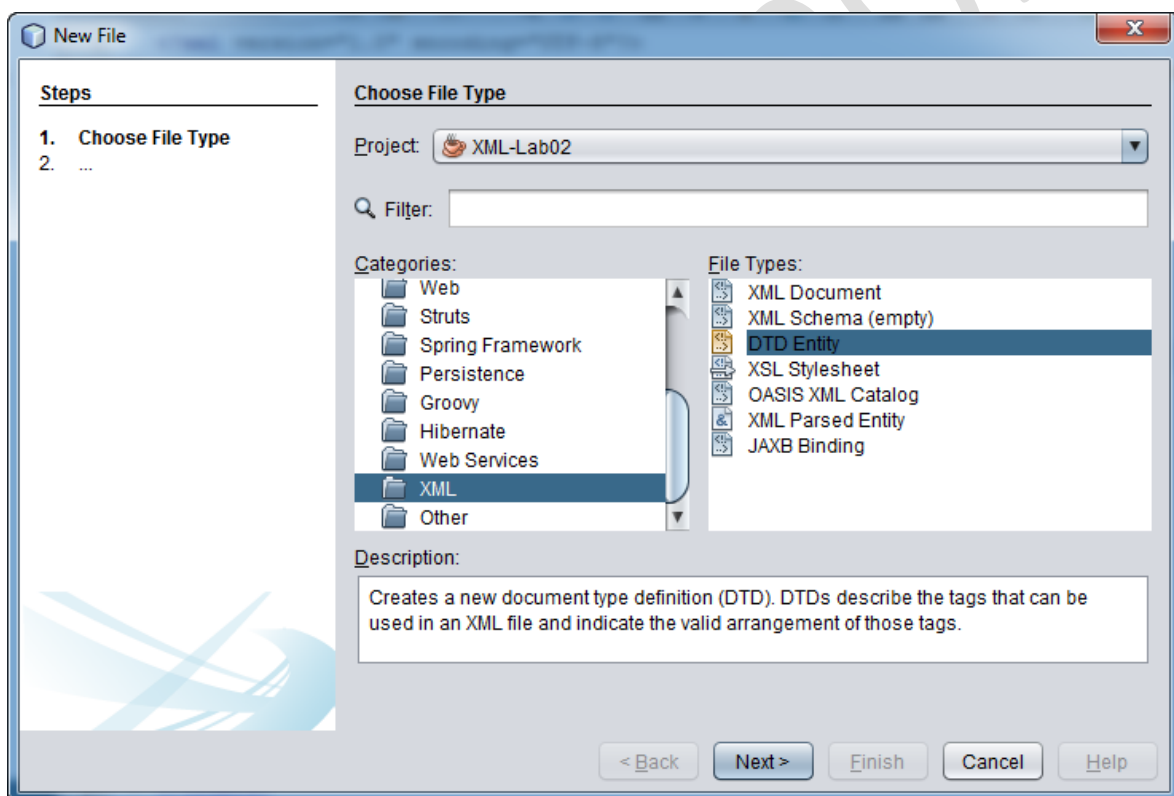
Bước 1: Tạo mới một file DTD Entity

- Chọn menu File => Chọn New File.
- Chọn **XML** trong hộp **Categories**

- Chọn **DTD Entity** trong hộp **File Type**
- Bấm **Next**



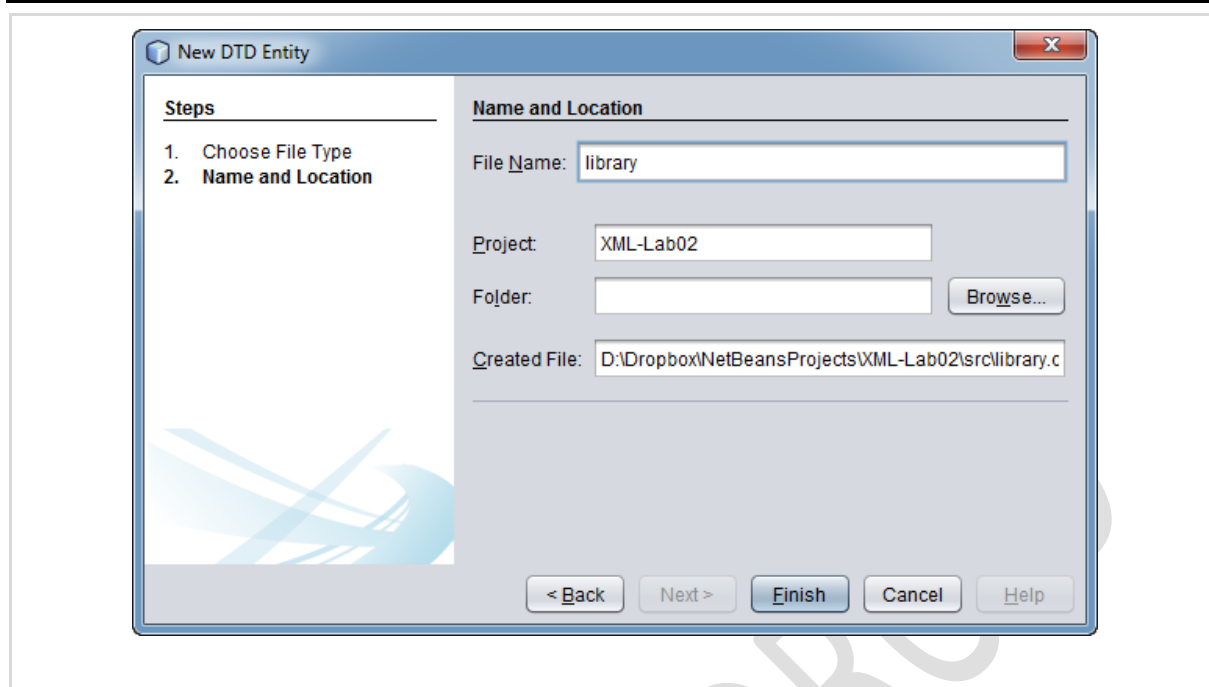
Tạo mới project > chọn New



...hoặc chọn Other > XML > DTD Entity

Bước 2: Đặt tên cho file DTD

- Đặt tên cho tập tin DTD
- Bấm Finish



Bài thực hành 2: Khai báo Phần tử, Thuộc tính và Thực thể trong file SaleDetails.dtd.

Bước 1: Khai báo phần tử gốc có tên SaleDetails có chứa phần tử con là Sale. Phần tử con Sale xuất hiện một hoặc nhiều lần bên trong phần tử SaleDetails.

```
<!ELEMENT SaleDetails (Sale+)>
```

Bước 2: Khai báo phần tử Sale và các phần tử con chứa trong nó lần lượt là:

- **BillNumber:** Phần tử lưu trữ ID duy nhất của hóa đơn bán hàng.
- **Date:** Phần tử lưu trữ ngày hóa đơn bán được tạo.
- **Branch:** Lưu thông tin chi nhánh bán
- **Item:** phần tử lưu chi tiết các mặt hàng gồm tên, mã, số serial, giá, giảm giá và số lượng. Mỗi hóa đơn có ít nhất một mặt hàng (sử dụng kí hiệu +).
- **Payment:** Phần tử lưu thông tin chi tiết về tên thẻ, tên khách hàng, số tài khoản và tổng cộng

```
<!ELEMENT Sale (BillNumber, Date, Branch, Item+, Payment)>
```

Bước 3: Khai báo phần tử BillNumber, Date, Branch và chỉ ra kiểu dữ liệu PCDATA (parsable character data).

```
<!--ELEMENT BillNumber (#PCDATA)>
<!--ELEMENT Date (#PCDATA)>
<!--ELEMENT Branch (#PCDATA)>
```

Bước 4: Khai báo phần tử Item và các phần tử con chứa trong nó lần lượt là:

- **Number:** Phần tử lưu trữ số serial của mặt hàng.
- **Name:** Phần tử lưu trữ tên của mặt hàng.
- **Code:** Lưu trữ mã duy nhất của mỗi mặt hàng.
- **Quantity:** Phần tử lưu trữ số lượng mua.
- **Price:** Phần tử lưu trữ giá bán.
- **Discount:** Phần tử lưu trữ giảm giá
- **Total:** Phần tử lưu trữ tổng tiền phải trả.

```
<!--ELEMENT Item (Number, Name, Code, Quantity, Price, Discount, Total)>
<!--ELEMENT Number (#PCDATA)>
<!--ELEMENT Name (#PCDATA)>
<!--ELEMENT Code (#PCDATA)>
<!--ELEMENT Quantity (#PCDATA)>
<!--ELEMENT Price (#PCDATA)>
<!--ELEMENT Discount (#PCDATA)>
<!--ELEMENT Total (#PCDATA)>
```

Bước 5: Khai báo phần tử Payment và các phần tử con chứa trong nó lần lượt là:

- **CardName:** Phần tử lưu trữ tên của thẻ đang dùng.
- **CustomerName:** Phần tử lưu trữ tên khách hàng.
- **CardNumber:** Phần tử lưu trữ số ID duy nhất của thẻ.
- **AccountNumber:** Lưu trữ số tài khoản.
- **GrandTotal:** Phần tử lưu trữ tổng tiền phải trả.

```
<!--ELEMENT Payment (CardName, CustomerName, CardNumber,
AccountNumber,GrandTotal)>
```

```
<!ELEMENT CardName (#PCDATA)>
<!ELEMENT CustomerName (#PCDATA)>
<!ELEMENT CardNumber (#PCDATA)>
<!ELEMENT AccountNumber (#PCDATA)>
<!ELEMENT GrandTotal (#PCDATA)>
```

Bước 6: Khai báo thuộc tính (Attribute):

- Thuộc tính **Category** cho phần tử **Name**. Thuộc tính chỉ nhận giá trị cố định (fixed) là **Apparel** và **Footwear** và **Apparel** được đặt làm giá trị mặc định
- Thuộc tính **Mode** cho phần tử **Payment**. Thuộc tính chỉ nhận giá trị cố định (fixed) là **Card** và **Cash**. **Card** được đặt làm giá trị mặc định
- Thuộc tính **Type** cho phần tử **CardName**. Thuộc tính chỉ nhận giá trị cố định (fixed) là **Credit** và **Debit**. **Credit** được đặt làm giá trị mặc định

```
<!ATTLIST Name Category (Apparel|Footwear) "Apparel">
<!ATTLIST Payment Mode (Card|Cash) "Card">
<!ATTLIST CardName Type (Credit|Debit) "Credit">
```

Bước 7: Khai báo thực thể (Entity):

```
<!ENTITY VH "Van Huesen">
<!ENTITY AS "Allen Solly">
<!ENTITY PA "Park Avenue">
<!ENTITY RD "Reid and Taylor">
<!ENTITY LS "Levi Strauss">
<!ENTITY TH "Tommy Hilfiger">
<!ENTITY MC "Monte Carlo">
<!ENTITY MC "Marco Ricci">
<!ENTITY IC "ICICI">
```

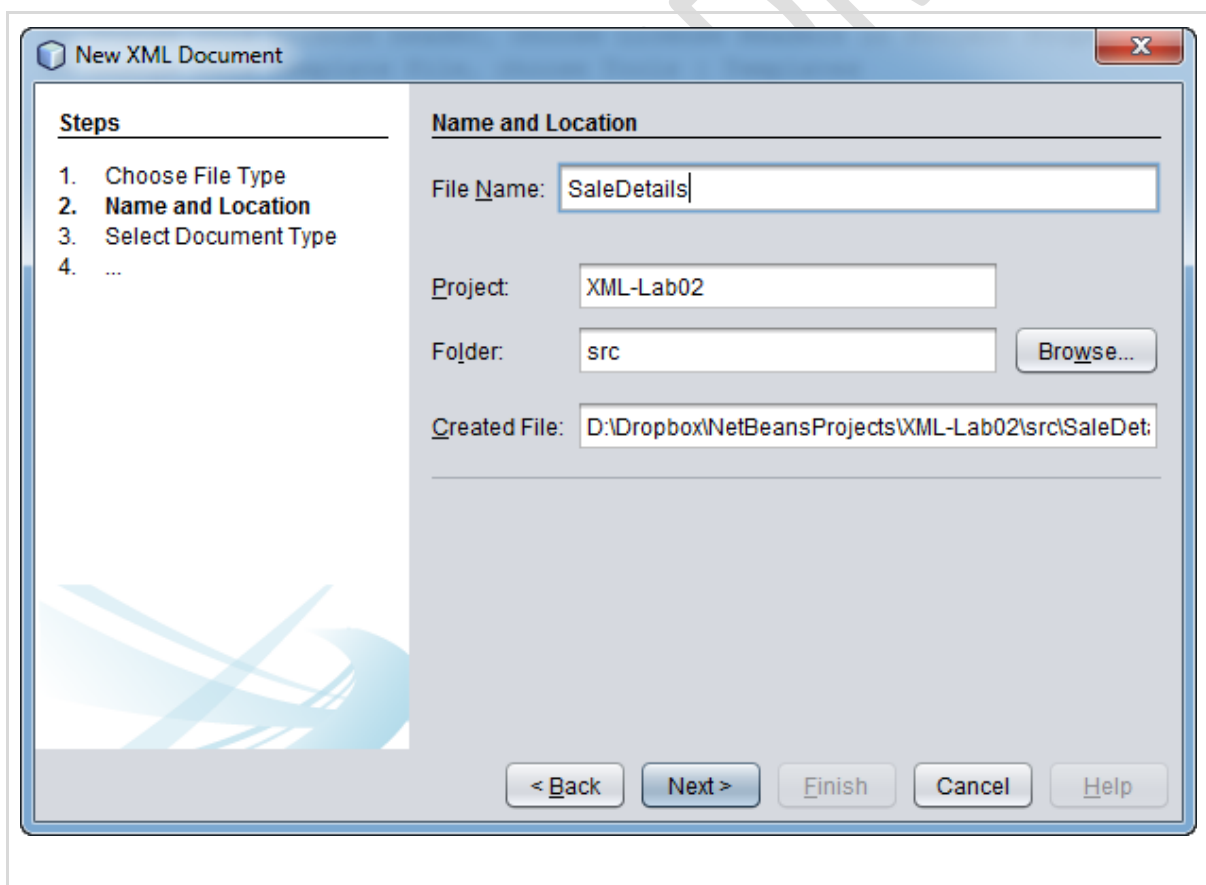
```
<!ENTITY HD "HDFC">  
<!ENTITY VS "VISA">  
<!ENTITY AX "AMERICAN EXPRESS">
```

Bài thực hành 3: Tạo tài liệu XML có ràng buộc với DTD.

Bước 1: Tạo một tập tin tài liệu XML với tên SaleDetails.xml để lưu thông tin chi tiết các bệnh nhân của bệnh viện Brooklyn.

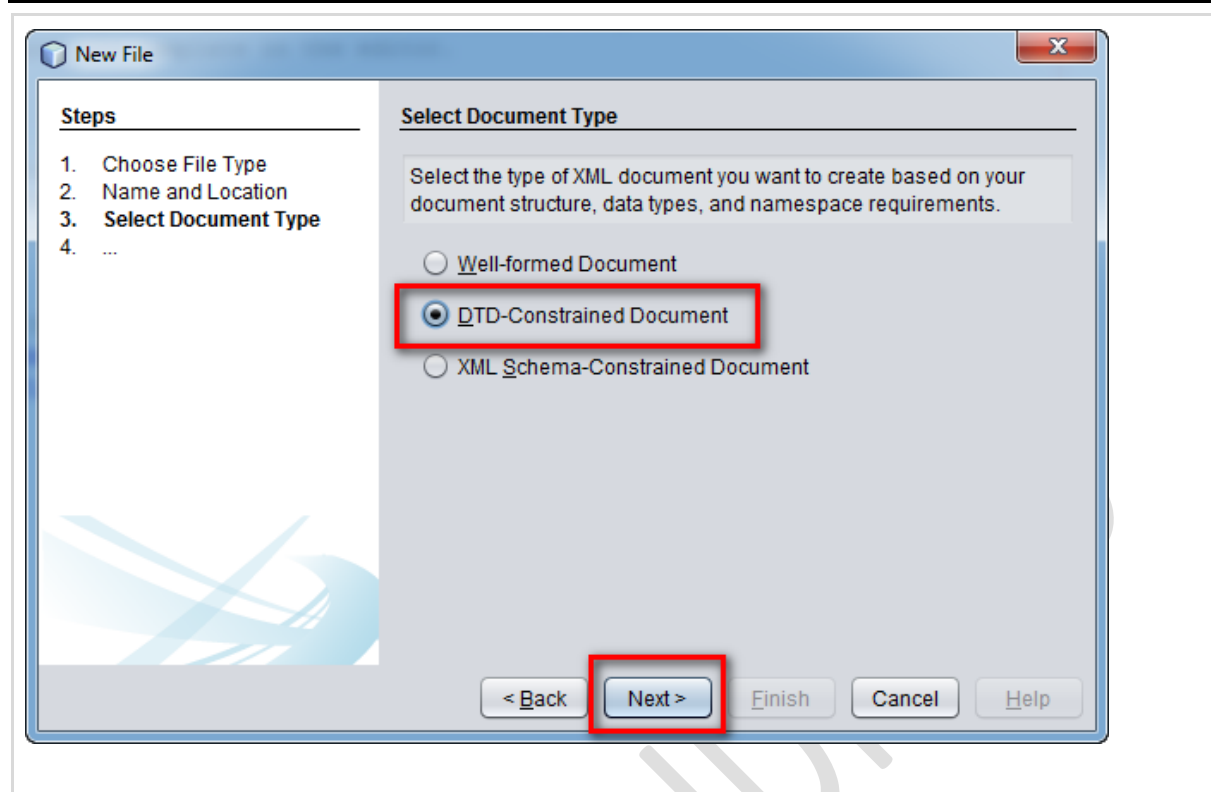
Tạo một file XML mới

- Chọn menu File => Chọn New File.
- Chọn XML trong hộp Categories
- Chọn XML Document trong hộp File Type
- Bấm Next



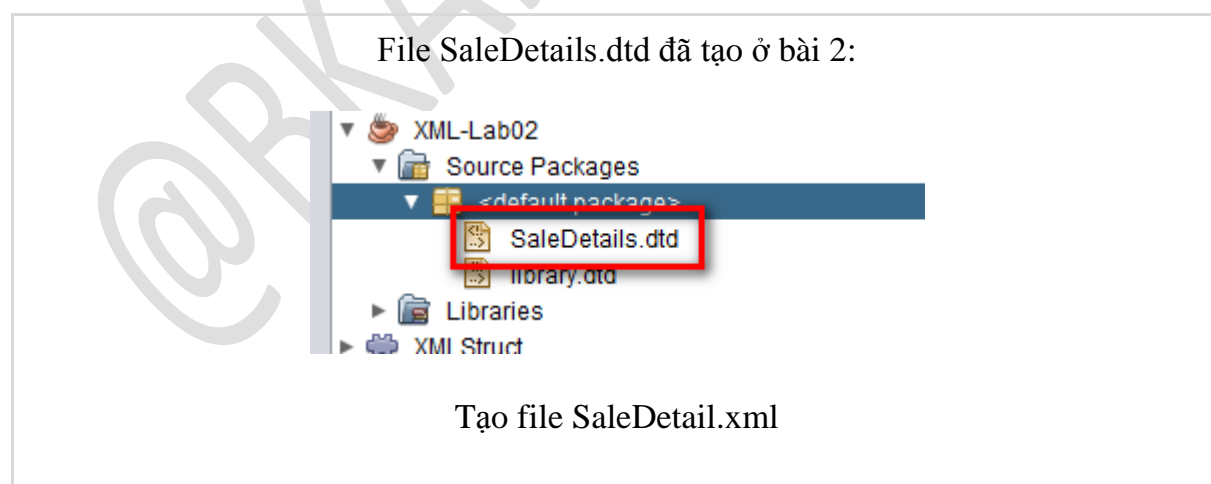
Bước 2: Chọn kiểu file tài liệu

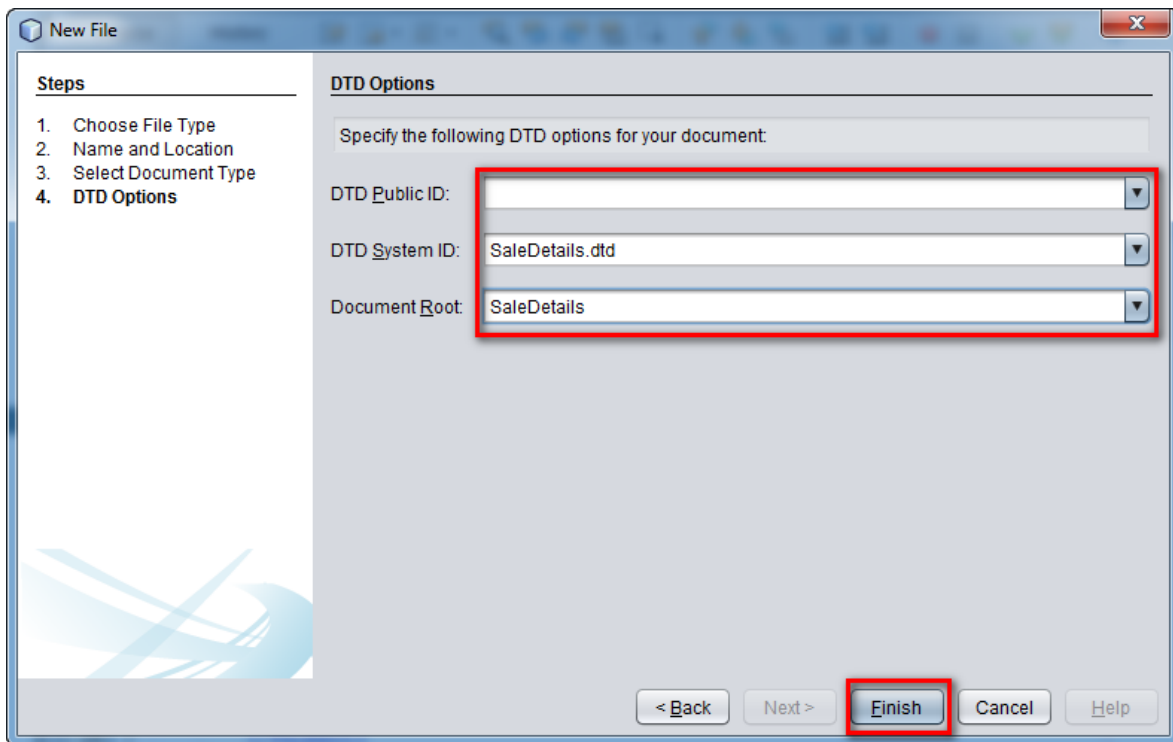
- Đánh dấu chọn “DTD-Constained Document”
- Bấm Next



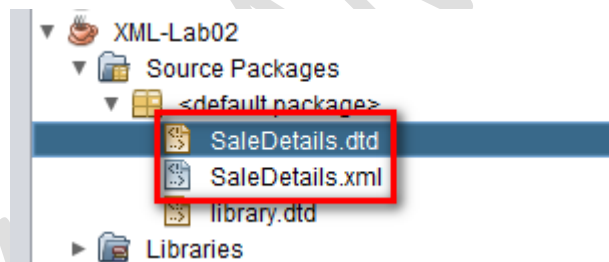
Bước 3: Chỉ ra vị trí file DTD dùng cho tài liệu XML SystemDetails.xml

- Xóa giá trị mặc định trong hộp Combo “DTD Public ID”
- Nhập vào vị trí file DTD trong hộp Combo “DTD System ID”
- Khai báo phần tử gốc là SystemDetails
- Bấm Next





Sau khi tạo file xml thành công:



Code:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<!--
```

To change this license header, choose License Headers in Project Properties.

To change this template file, choose Tools | Templates and open the template in the editor.

```
-->
```

```
<!DOCTYPE SaleDetails SYSTEM 'SaleDetails.dtd'>
```

```
<SaleDetails>
```

```
</SaleDetails>
```

Bước 4: code file SaleDetail.xml.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Create by MinhVT FC ver 6.2016
-->
<!DOCTYPE SaleDetails SYSTEM 'SaleDetails.dtd'>
< SaleDetails >
  <Sale>
    <BillNumber>4568</BillNumber>
    <Date>23/02/2007</Date>
    <Branch>Hampshire</Branch>
    <Item>
      <Number>1</Number>
      <Name Category="Apparel">&LS; Jeans</Name>
      <Code>2323</Code>
      <Quantity>1</Quantity>
      <Price>200</Price>
      <Discount>10</Discount>
      <Total>180</Total>
    </Item>
    <Payment Mode="Card">
      <CardName Type="Credit">&IC;</CardName>
      <CustomerName>Allan</CustomerName>
      <CardNumber>99044993</CardNumber>
      <AccountNumber>77688906</AccountNumber>
      <GrandTotal>180</GrandTotal>
    </Payment>
  </Sale>
  <Sale>
    <BillNumber>4576</BillNumber>
    <Date>23/02/2007</Date>
    <Branch>Hampshire</Branch>
    <Item>
      <Number>1</Number>
      <Name Category="Footwear">&MC; Leather Shoes</Name>
      <Code>1312</Code>
      <Quantity>3</Quantity>
      <Price>1099</Price>
      <Discount>0</Discount>
      <Total>1099</Total>
    </Item>
    <Payment Mode="Cash">
      <CardName Type="Credit"></CardName>
      <CustomerName></CustomerName>
      <CardNumber></CardNumber>
      <AccountNumber></AccountNumber>
    </Payment>
  </Sale>
</SaleDetails>
```

```
<GrandTotal>1099</GrandTotal>
</Payment>
</Sale>
</SaleDetails>
```

Bước 5: xem file xml trên trình duyệt. (Lưu ý: file.dtd bên ngoài nhiều trình duyệt không hỗ trợ lấy thuộc tính ENTITY, nếu viết trực tiếp trong XML thì hoạt động bình thường)

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<!--
  Create by MinhVT FC ver 6.2016
-->
<!-- <!DOCTYPE SaleDetails SYSTEM 'SaleDetails.dtd' --> -->
▼<SaleDetails>
  ▼<Sale>
    <BillNumber>4568</BillNumber>
    <Date>23/02/2007</Date>
    <Branch>Hampshire</Branch>
    ▼<Item>
      <Number>1</Number>
      <Name Category="Apparel">Levi Strauss Jeans</Name>
      <Code>2323</Code>
      <Quantity>1</Quantity>
      <Price>200</Price>
      <Discount>10</Discount>
      <Total>180</Total>
    </Item>
    ▼<Payment Mode="Card">
      <CardName Type="Credit">ICICI</CardName>
      <CustomerName>Allan</CustomerName>
      <CardNumber>99044993</CardNumber>
      <AccountNumber>77688906</AccountNumber>
      <GrandTotal>180</GrandTotal>
    </Payment>
  </Sale>
  ▼<Sale>
    <BillNumber>4576</BillNumber>
    <Date>23/02/2007</Date>
    <Branch>Hampshire</Branch>
    ▼<Item>
      <Number>1</Number>
      <Name Category="Footwear">Monte Carlo Leather Shoes</Name>
      <Code>1312</Code>
      <Quantity>3</Quantity>
      <Price>1099</Price>
      <Discount>0</Discount>
      <Total>1099</Total>
    </Item>
  </Payment Mode="Cash">
```

Bài thực hành 4: Kiểm tra tính hợp lệ của tài liệu XML.

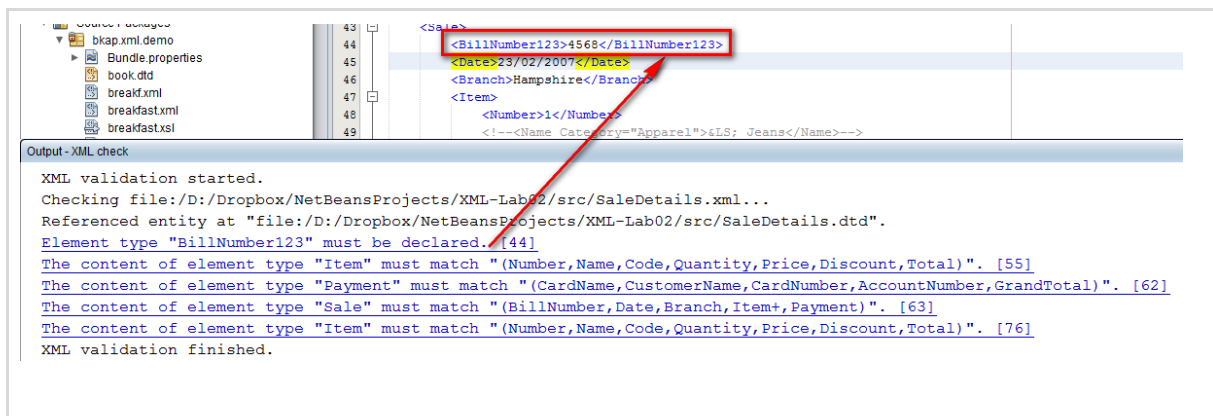
Kiểm tra Well-formed

XML checking started.
 Checking file:/D:/Dropbox/NetBeansProjects/XML-Lab02/src/SaleDetails.xml...
 Referenced entity at "file:/D:/Dropbox/NetBeansProjects/XML-Lab02/src/SaleDetails.dtd".
 The end-tag for element type "Discount" must end with a ">" delimiter. [53]
 XML checking finished.

Bấm chuột phải => Chọn Validate XML

Xem kết quả trên của sổ Output.

XML checking started.
 Checking file:/D:/Dropbox/NetBeansProjects/XML-Lab02/src/SaleDetails.xml...
 Referenced entity at "file:/D:/Dropbox/NetBeansProjects/XML-Lab02/src/SaleDetails.dtd".
 The end-tag for element type "Discount" must end with a ">" delimiter. [53]
 XML checking finished.



Phần II - Bài tập tự làm

Bài 1: Tạo file DTD (**Orders.dtd**) để Valid tài liệu XML với cấu trúc được mô tả như sau:

Tài liệu cần lưu trữ một danh sách các đơn đặt hàng (Orders). Mỗi đơn đặt hàng (Order) có các thông tin (OrderNo, OrderDate, CustomerName, Detail) trong đó OrderNo là thuộc tính số đặt hàng của đơn hàng đó (Order), Detail là thông tin chi tiết về các mặt hàng bao gồm các thuộc tính ItemNo, ItemName, Quantity, Price.

Constraints : Các thông tin của đơn đặt hàng bắt buộc phải nhập, OrderNo phải là duy nhất, trong danh sách đơn đặt hàng có thể có 0 hoặc nhiều đơn đặt hàng, trong 1 đơn đặt hàng (OrderNo, OrderDate, CustomerName) chỉ xuất hiện 1 lần nhưng có thể có 1 hoặc nhiều chi tiết đặt hàng, Các thuộc tính Quantity mặc định là 1, Price mặc định là 0 .

Tạo file XML (**Orders.xml**) và link **Orders.dtd** để valid dữ liệu của nó. Nhập 5 đơn đặt hàng demo

Bài 2: Tạo file dtd theo như mô tả sau và viết file xml theo định nghĩa dtd này.

Valid XML & DTDs: Exercises

1. Construct a sample XML document which is valid according to the external DTD given below. This describes information an Estate Agent might wish to store about a house. Your document should be sufficiently complex to test your implementation of the DTD (e.g., it should contain information on at least *two* houses).

```
<!ELEMENT houses (house*)>
<!ELEMENT house (room+)>
<!ELEMENT room (dimensions, windows, furniture*)>
<!ELEMENT dimensions (width, length)>
<!ELEMENT windows (window*)>
<!ELEMENT window (elevation, width, height)>
<!ELEMENT furniture (#PCDATA)>
<!ELEMENT width (#PCDATA)>
<!ELEMENT length (#PCDATA)>
<!ELEMENT height (#PCDATA)>
<!ELEMENT elevation EMPTY>
<!ATTLIST room name ID #REQUIRED
              colour IDREF #IMPLIED>
<!ATTLIST elevation facing (north | south | east | west) #REQUIRED>
```