

Lecture 1: 使用檔名：myXmlCSS_0010.xml & myXmlDSO_0010.HTML

一、先 edit 一個 myXmlCss_0010.xml 檔。

```
<?xml version="1.0" encoding="Big5" standalone="no"?>
<!-- my first Xml file -->
<?xml-stylesheet type="text/css" href="myXmlCSS_0010.css" ?>

<inventory>
  <book>
    <title>通訊原理與應用</title>
    <author>藍國桐</author>
    <page>288 頁</page>
    <price>NT$ 350</price>
  </book>
```

```
<book>
  <title>行動通訊系統</title>
  <author>李大明</author>
  <page>450 頁</page>
  <price>NT$ 400</price>
</book>

<book>
  <title>網路程式設計</title>
  <author>王李吉</author>
  <page>550 頁</page>
  <price>NT$ 450</price>
</book>
</inventory>
```

二、再 edit 一個 myXmlDSO_0010.html 檔。

Data binding is the technique that you'll learn for displaying an XML document from within a conventional HTML page. In data binding, you (1) Link an XML document to an HTML page, and then (2) Bind standard HTML elements, such as SPANs or TABLEs, to individual XML elements or attributes.

myXmlDSO_0010.html 檔 如下：

```
<!-- myFile: myXmlDSO_0010.Html -->

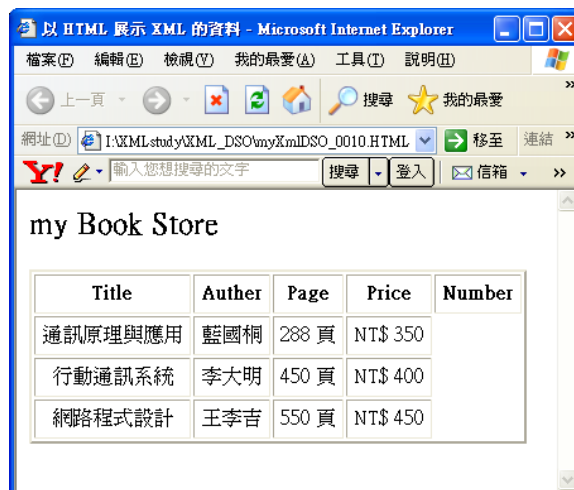
<HTML>
  <Head>
    <Title> 以 HTML 展示 XML 的資料
  </Head>
  <Body>
    <XML ID="myBookStore" SRC="myXmlCss_0010.xml"></XML>
```

載入 myXmlCss_0010.xml 檔 作為本 myXmlDSO_0010.html 檔的資料來源(data source)。
稱之 Data Island。

```
<H2> my Book Store </H2>
<Table DATASRC="#myBookStore" border="2" cellpadding="5">
  <THEAD>
    <TH> Title </TH>
    <TH> Auther </TH>
    <TH> Page </TH>
    <TH> Price </TH>
    <TH> Number </TH>
  </THEAD>
  <TR align="center">
    <TD> <Span DATAFLD="title"></Span> </TD>
    <TD> <Span DATAFLD="author"></Span> </TD>
    <TD> <Span DATAFLD="page"></Span> </TD>
    <TD> <Span DATAFLD="price"></Span> </TD>
    <TD> <Span DATAFLD="num"></Span> </TD>
  </TR>
</Table>
```

Binding HTML elements to XML elements or attributes。

用 IE 6.0 點選 myXmlDSO_0010.html 後：



說明一、 To display an XML document in an HTML page, you must link the document to the page. You can program a Web page using scripts that employ the same underlying programming object as **data binding** (namely, the *Data Source Object*, or **DSO**). The easiest way to do this with Microsoft Internet Explorer is to include in the page an HTML element named XML, which is also known as a **data island**.

說明二、When Internet Explorer opens the HTML page, its XML processor reads and parses the XML document. Internet Explorer also creates a programming object known as a **Data Source Object (DSO)**, which stores, or caches, the XML data and provides access to this data. The DSO stores the XML data as a record set- that is, as a collection of records and their fields.

說明三、You assign to the SRC attributes the URL of the file containing the XML data.

```
<XML  
  ID="myBookStore"  
  SRC="myXmlCss_0010.xml"  
</XML>
```

You can use a fully URL type name, as in

```
<XML  
  ID="myBookStore"  
  SRC="http://www.ntou.edu.tw/CS/AI/XML/myXmlCss_0010.xml"  
</XML>
```

說明四、When you bind an HTML element to an XML element, the DSO automatically supplies the value of the XML element and handles all the details. The DSO also lets you directly access and manipulate the stored record set through a collection of methods, properties, and events. **Methods** are functions that you can call from the page to access or modify the record set. **Properties** are current feature settings that you can read and sometimes modify from the page. And **events** are occurrences that you can handle from a script function that you include in the page.

說明五、In the page, the identifier that you assigned to the **ID attributes** in the data island represents the DSO. (In the example, this identifier is **myBookStore**.)

說明六、In this example, the page's TABLE element is bound to the entire XML document by assigning to the element's **DATASRC** attribute and **ID** of the data island, prefaced with a pound sign (#):

```
<Table DATASRC="#myBookStore" border="2" cellpadding="5">
```

The table is defined with a standard heading (the **THEAD** element), and one row

(the TR element). Each cell in that row (that is, each TD element) contains a **SPAN** element that is bound to one of the fields in the XML document so that it will display the contents of that field. The SPAN element is bound to the XML field by assigning the field's name to the element's DATAFLD attribute.

```
<TD> <Span DATAFLD="author"></Span> </TD>
```

說明二、Here's how the data binding works: Even though the TABLE element defines only a single row, when the browser displays the table, it repeats the row element for each record in the XML document. You might wonder why the cell (TD) elements aren't bound directly to the XML fields. The reason is that the TD element is not one of the bindable HTML elements.

範例一、(myXmlDSO_0010.html & myXmlCSS_0010.xml)。

(Demo)

說明八、資料(xml)上的空缺造成空白(html)。

Lecture 2: 使用檔名：myXmlDSO_0051.HTML & myXmlCSS_0050.xml

(不使用 Microsoft 的 Data Island 技術，以一般 Document Object Model (DOM) 方式載入 XML 檔)

一、使用 myXmlDSO_0050.HTML 展現 myXmlCSS_0050.xml 檔。

(Demo)

(資料和 myXmlCSS_0010.xml 類似) (Html 是採用 Data Island 方式形成 DSO 物件)

二、編輯一個 myXmlDSO_0051.HTML 檔。

```
<!-- myFile: myXmlHtml_0051.Html -->

<HTML>
  <Head>
    <Title> 以 HTML 展示 XML 的資料</Title>
    <Script Language="JavaScript" FOR="window" EVENT="onload">
      var myBookStore;
      myBookStore=new ActiveXObject("MSXML.DOMDocument");
      myBookStore.async=false;
      myBookStore.load("myXmlCss_0050.xml");
    </Script>
  </Head>
  <Body>
    <H2> my Book Store </H2>
    <Table DATASRC="#myBookStore" border="2" cellpadding="5">
      <THEAD>
        <TH> Title </TH>
        <TH> Auther </TH>
        <TH> Page </TH>
        <TH> Price </TH>
        <TH> Number </TH>
      </THEAD>
      <TR align="center">
        <TD> <Span DATAFLD="title"></Span> </TD>
        <TD> <Span DATAFLD="author"></Span> </TD>
        <TD> <Span DATAFLD="page"></Span> </TD>
        <TD> <Span DATAFLD="price"></Span> </TD>
      </TR>
    </Table>
  </Body>
</HTML>
```

三、(demo: myXmlDSO_0051.HTML) (無法用 DOM 的載入方式形成 DSO 物件)

Lecture 3: 使用檔名：myXmlDSO_0052.HTML & myXmlCSS_0050.xml

(以一般 Document Object Model (DOM) 方式 載入 XML 檔，必須自行 code Script)

一、使用 myXmlCSS_0050.xml 檔。

(show it)

二、編輯 一個 myXmlDSO_0052.HTML 檔。

```
<!-- myFile: myXmlHtml_0052.Html -->
```

```
<HTML>
```

```
<Head>
```

```
<Title> 以 HTML 展示 XML 的資料</Title>
```

```
<Script Language="JavaScript" FOR="window" EVENT="onload">
```

```
var myBookStore;  
myBookStore=new ActiveXObject("MSXML.DOMDocument");  
myBookStore.async=false;  
if (!myBookStore.load("myXmlCss_0050.xml")) {  
    window.alert("load error!!");  
}
```

```
myTitle.innerHTML=myBookStore.childNodes(3).childNodes(0).childNodes(0).text;  
myAuthor.innerHTML=myBookStore.childNodes(3).childNodes(0).childNodes(1).text;  
myPage.innerHTML=myBookStore.childNodes(3).childNodes(0).childNodes(2).text;  
myPrice.innerHTML=myBookStore.childNodes(3).childNodes(0).childNodes(3).text;  
myNum.innerHTML=myBookStore.childNodes(3).childNodes(0).childNodes(4).text;
```

```
</Script>
```

```
</Head>
```

```
<Body>
```

```
<H2> my Book Store </H2>
```

```
<Table border="2" cellpadding="5">
```

```
<THEAD>
```

```
<TH> Title </TH>
```

```
<TH> Auther </TH>
```

```
<TH> Page </TH>
```

```
<TH> Price </TH>
```

```
<TH> Number </TH>
```

```
</THEAD>
<TR align="center">
  <TD> <Span ID="myTitle"></Span> </TD>
  <TD> <Span ID="myAuthor"></Span> </TD>
  <TD> <Span ID="myPage"></Span> </TD>
  <TD> <Span ID="myPrice"></Span> </TD>
  <TD> <Span ID="myNum"></Span> </TD>
</TR>
</Table>

</Body>
</HTML>
```

三、(demo: myXmlDSO_0052.HTML)



說明一、因為沒有在網頁上形成 Data Island，Table 無法顯示(xml)上的所有資料，除非自行 code 上具有遮圍功能的 JavaScript code。

Lecture 4: 使用檔名：myXmlDSO_0061.HTML & myXmlCSS_0061.xml

(其他可以册來作爲 Data Binding 的 HTML elements)

一、先看一下 myXmlDSO_0060.HTML 及 myXmlCSS_0060.xml 檔。

(show it and demo)

二、再更新一下 xml 的資料成爲 myXmlCSS_0061.xml 檔。

```
<?xml version="1.0" encoding="Big5" ?>
<!-- my Xml file myXmlCSS_0061.xml -->
<?xml-stylesheet type="text/css" href="myXmlCSS_0030.css" ?>

<inventory >
  <book>
    <title>通訊原理與應用</title>
    <author>藍國桐 </author>
    <page>288 頁</page>
    <price>NT$ 350</price>
    <num>0001 </num>
    <publish>全華圖書公司</publish>
    <webAddress>http://tw.yahoo.com</webAddress>
  </book>

  <book>
    <title>行動通訊系統</title>
    <author>藍國桐</author>
    <page>450 頁</page>
    <price>NT$ 400</price>
    <num>0002</num>
    <publish>奇摩圖書公司</publish>
    <webAddress>http://tw.yahoo.com</webAddress>
  </book>

  .....
```

三、編輯一個 myXmlDSO_0061.HTML 檔。

```
<!-- myFile: myXmlHtml_0061.Html -->

<HTML>
  <Head>
    <Title> 以 HTML 展示 XML 的資料</Title>
  </Head>
  <Body>
    <XML ID="myBookStore" SRC="myXmlCss_0061.xml"></XML>

    <H2> my Book Store </H2>
```



```
<Table DATASRC="#myBookStore" border="2" cellpadding="5">
  <THEAD>
    <TH> Title </TH>
    <TH> Auther </TH>
    <TH> Page </TH>
    <TH> Price </TH>
    <TH> Number </TH>
    <TH> Publish Inc. </TH>
  </THEAD>
  <TR align="center">
    <TD> <Span DATAFLD="title"></Span> </TD>
    <TD> <Span DATAFLD="author"></Span> </TD>
    <TD> <Span DATAFLD="page"></Span> </TD>
    <TD> <Span DATAFLD="price"></Span> </TD>
    <TD> <Span DATAFLD="num"></Span> </TD>
    <TD> <A DATAFLD="webAddress"> <Span DATAFLD="publish"></Span> </A> </TD>
  </TR>
</Table>

</Body>
</HTML>
```

四、(demo: myXmlDSO_0061.HTML)



說明一、有一些 non-table HTML elements 可以 binding 到 XML field node。These can be individual HTML elements used for single-record data binding, or they can be HTML elements contained an HTML table that's bound to an XML record node.

例如：

HTML element	Bound property (bound 到 這個 HTML element 的那 個 Attribute 上面)	
A	<i>href</i>	
APPLET		
BUTTON		
DIV		
FRAME		
IFRAME	<i>src</i>	
IMG	<i>src</i>	
INPUT TYPE =BUTTON		
INPUT TYPE =CHECKBOX	<i>checked</i>	
INPUT TYPE =HIDDEN	<i>value</i>	
INPUT TYPE =PASSWORD	<i>value</i>	
INPUT TYPE =RADIO	<i>checked</i>	
INPUT TYPE =TEXT	<i>value</i>	
LABEL		
LEGEND		
MARQUEE		
SELECT		
SPAN	<i>innerText</i> and <i>innerHTML</i>	
TEXTAREA		

Lecture 5: 使用檔名：myXmlDSO_0070.HTML & myXmlCSS_0060.xml

(Using paging to display the records on group at one time)

一、更新一下 myXmlDSO_0070.html 檔。

```
<!-- myFile: myXmlHtml_0070.Html -->

<HTML>
  <Head>
    <Title> 以 HTML 展示 XML 的資料</Title>
  </Head>
  <Body>
    <XML ID="myBookStore" SRC="myXmlCss_0060.xml"></XML>

    <H2> my Book Store </H2>
    <input type="button" value="PreviousPage" onclick="myTable.previousPage()">
    <input type="button" value="NextPage" onclick="myTable.nextPage()">

    <Table ID="myTable" DATASRC="#myBookStore" DATAPAGESIZE="4" border="2" cellpadding="5">
    <THEAD>
      <TH> Title </TH>
      <TH> Auther </TH>
      <TH> Page </TH>
      <TH> Price </TH>
      <TH> Number </TH>
      <TH> Publish Inc. </TH>
    </THEAD>
    <TR align="center">
      <TD> <Span DATAFLD="title"></Span> </TD>
      <TD> <Span DATAFLD="author"></Span> </TD>
      <TD> <Span DATAFLD="page"></Span> </TD>
      <TD> <Span DATAFLD="price"></Span> </TD>
      <TD> <Span DATAFLD="num"></Span> </TD>
      <TD> <Span DATAFLD="publish"></Span> </TD>
    </TR>
  </Table>

  </Body>
</HTML>
```

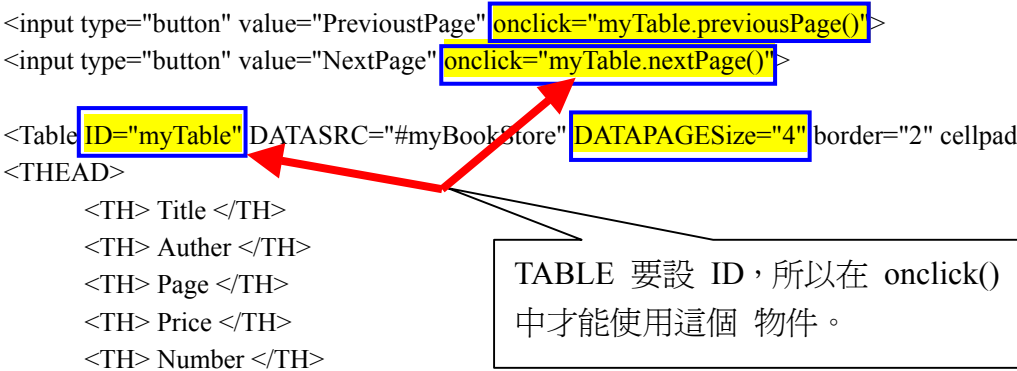


TABLE 要設 ID，所以在 onclick() 中才能使用這個 物件。

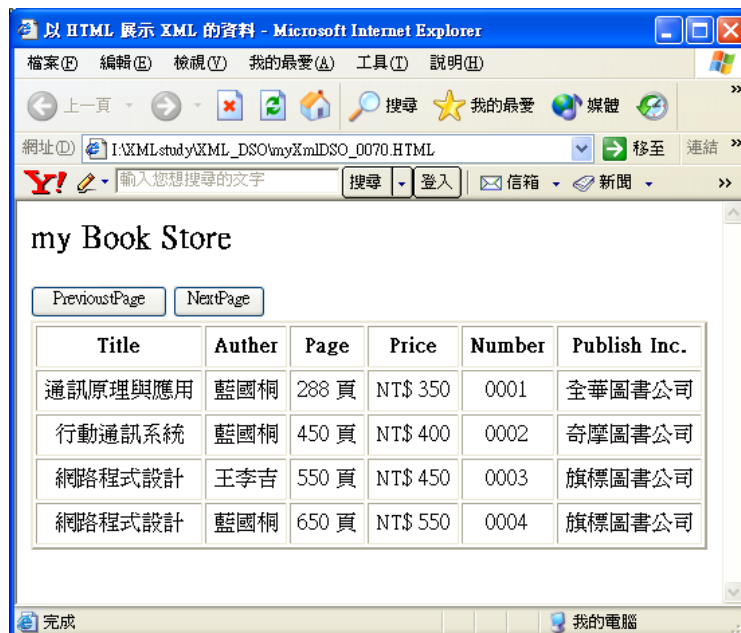
說明一、Set the **DATAPAGESIZE** attribute of the bound TABLE element to the maximum number of records that you want to display at one time.

說明二、Assign a unique identifier to the **ID** attribute of the TABLE element.

說明三、To navigate through the records, call the TABLE element methods listed in the following table.

TABLE element method	Effect	example
firstPage		myTable.firstPage()
previousPage		myTable.previousPage()
nextPage		myTable.nextPage()
lastPage		myTable.lastPage()

二、(demo: myXmlDSO_0070.HTML)




Lecture 6: 使用檔名：myXmlDSO_0080.HTML & myXmlCSS_0080.xml

(Using nested Table to display a hierarchical record set)

一、將 myXmlCSS 0060.xml 更新成 myXmlCSS 0080.xml 檔。

```
<?xml version="1.0" encoding="Big5" ?>  
<!-- my Xml file myXmlCSS_0080.xml -->  
<?xml-stylesheet type="text/css" href="myXmlCSS_0080.css" ?>
```

```
<inventory>  
  <book>  
    <title>通訊原理與應用</title>  
    <author>  
      <lastName>藍</lastName>  
      <firstName>國桐</firstName>  
    </author>  
    <page>288 頁</page>  
    <price>NT$ 350</price>
```



將 <author> 改
成 一個
hierarchical record

二、(demo: myXmlDSO 0080.HTML)


說明一、In a hierarchical record set, each record can contain, zero or more nested records. As a general rule, if an element contains one or more children (or attributes), or if the same element type occurs more than once within a given parent, the DSO interprets the element or elements as a **record (or set of records)**, rather than as a **field (or set of fields)**.

說明二、A **TABLE** elements is the only type of HTML element that you can bind to an XML record.

說明三、(在 demo 中) 因為 myXmlCSS_0080.xml 欄位 含有 Hierarchical Record Set 所以 Author 無法顯示出來 !!

三、(demo) 更新一下 myXmlDSO 0080.HTML 為 myXmlDSO 0082.HTML

在 TABLE 中插入 TABLE。



整份 XML 文件的 DSO

```
<Table ID="parentTable" DATASRC="#myBookStore" DATAPAGESIZE="3" border="2" cellpadding="5">  
  <THEAD>
```

```

<TH> Title </TH>
<TH> Author </TH>
<TH> Page </TH>
<TH> Price </TH>
<TH> Number </TH>
<TH> Publish Inc. </TH>
</THEAD>
<TR align="center">
  <TD> <Span DATAFLD="title"></Span> </TD>
  <!-- <TD> <Span DATAFLD="author"></Span> </TD> 抓不到 author 的資料 -->
  <!-- <TD> <Span DATAFLD="lastName"></Span> </TD> 抓不到 lastName 的資料-->
  <TD>
    <Table ID="childTable" DATASRC="#myBookStore" DATAFLD="author">
      <TR>
        <TD> <Span DATAFLD="firstName"></Span>&nbsp;&nbsp;&nbsp;</TD>
        <TD> <Span DATAFLD="lastName"></Span> </TD>
      </TR>
    </Table>
  </TD>
  <TD> <Span DATAFLD="page"></Span> </TD>
  <TD> <Span DATAFLD="price"></Span> </TD>
  <TD> <Span DATAFLD="num"></Span> </TD>
  <TD> <Span DATAFLD="publish"></Span> </TD>
</TR>
</Table>

```

<author> 所形成的 hierarchy record

造一個新的 TABLE 來 Display 一個 hierarchy record。

四、更新一下 myXmlCSS 0060.xml 為 myXmlCSS 0081.xml

```

<?xml version="1.0" encoding="Big5" ?>
<!-- my Xml file myXmlCSS_0081.xml -->
<?xml-stylesheet type="text/css"
href="myXmlCSS_0081.css" ?>

<inventory>
  <category>
    <catName>非電腦類</catName>
    <book>
      <title>通訊原理與應用</title>
      <author>藍國桐 </author>
      <page>288 頁</page>
      <price>NT$ 350</price>
      <num>0001 </num>
      <publish>全華圖書公司</publish>
    </book>
    <book>
      .....
    </book>
  </category>

```

```

<category>
  <catName>電腦類</catName>
  <book>
    <title>網路程式設計</title>
    <author>王李吉</author>
    <page>550 頁</page>
    <price>NT$ 450</price>
    <num>0003</num>
    <publish>旗標圖書公司</publish>
  </book>
  .....
</category>

```

</inventory>

形成 hierarch record

五、(demo: myXmlDSO_0081.HTML)

說明一、(在 demo 中) 因為 myXmlCss_0081.xml 的 document element 本身就是一個 Hierarchical Record Set 所以無法單單只設 DATASRC 就可以顯示出來 !!

說明二、在 myXmlDSO_0080.HTML 及 myXmlCSS_0080.xml 檔中

HTML file:

<HTML>

```
<Table ID="myTable" DATASRC="#myBookStore">
  <TD> <Span DATAFLD="title"></Span> </TD>
  <TD> <Span DATAFLD="author"></Span> </TD>
</Table>
```

表示這個 Table 連到 myBookStore 所代表的 xml 資料檔案，其所以要顯示的資料為 Document Element 下的所有 Record node。

在某個 Record node 下的 Field node 名稱為 "title" 的 text 資料內容。

在某個 Record node 下的 Field node 名稱為 "author" 的 text 資料內容。
在本 Table 中看到五個 Field node (title, page, price, num, publish) 和一個 Record node (author)，所以無法顯示 (author) 這個 Record node。

</HTML>

XML file:

```
<inventory>
  <book>
    <title>通訊原理與應用</title>
    <author>
      <lastName>藍</lastName>
      <firstName>國桐</firstName>
    </author>
    <page>288 頁</page>
    <price>NT$ 350</price>
    <num>0001 </num>
    <publish>全華圖書公司</publish>
  </book>
</inventory>
```

Document Element. Record node. Field node

Document Element. Record node. Record node

說明三、在 myXmlDSO_0081.HTML 及 myXmlCSS_0081.xml 檔中

HTML file:

<HTML>

.....

<Table ID="myTable" DATASRC="#myBookStore">

<TD> </TD>

<TD> </TD>

</Table>

</HTML>

表示這個 Table 連到 myBookStore 所代表的 xml 資料檔案，其所要顯示的資料為 Document Element 下的所有 Record node。

在某個 Record node 下的 Field node 名稱為 "author" 的 text 資料內容。不幸地，在本 Table 中只能看到一個 Field node (catName) 和一個 Record node (book)，所以無法顯示。

XML file:

<inventory>

<category>

<catName>非電腦類</catName>

<book>

<title>通訊原理與應用</title>

<author>

<lastName>藍</lastName>

<firstName>國桐</firstName>

</author>

<page>288 頁</page>

<price>NT\$ 350</price>

<num>0001 </num>

<publish>全華圖書公司</publish>

</book>

</inventory>

Document Element. Record node. Field node

Document Element. Record node. Record node

六、(demo) 更新一下 myXmlDSO_0081.HTML 為 myXmlDSO_0083.HTML

說明一、在 myXmlDSO_0083.HTML 及 myXmlCSS_0081.xml 檔中

HTML file:

<HTML>

表示這個 Table 連到 myBookStore 所代表的 xml 資料檔案，其所要顯示的資料為 Document Element 下的 Record node 下的 Record node (名叫 book)。

<Table ID="myTable" DATASRC="#myBookStore" DATAFLD="book">

<TD>

<TD>

在某個 Record node (book) 下的 Field node 名稱為 "author" 的 text 資料內容。

</Table>

</HTML>

XML file:

<inventory>

<category>

<catName>非電腦類</catName>

<book>

<title>通訊原理與應用</title>

<author>藍國桐</author>

<page>288 頁</page>

<price>NT\$ 350</price>

<num>0001 </num>

<publish>全華圖書公司</publish>

</book>

</inventory>

Document Element. Record node. Field node

Document Element. Record node. Record node (book)

myXmlDSO_0083.HTML 的 Table 目前是 bound 到這個 Record node.

七、(demo) 再看一下 myXmlDSO_0084.HTML

Q：是不是只要設定 ... DATASRC="#myBookStore" DATAFLD="book" ... 它就會向下找，一直找到 "book" 這個 Field Node 為止 ??