

MODUL PEMOGRAMAN WEB I

Oleh:

CHALIFA CHAZAR

Modul 10

XML - eXtensible Markup Language

Tujuan:

Mahasiswa mengenal fungsi XML dan komponen XML sebagai pelengkap element HTML dalam membangun website statis.

Pustaka:

- » **HTML dan XML Edisi 2: Betha Sidik Ir dan Husni I. Pohan, 2002**
- » **Buku Pintar Webmaster: Adhi Prasetio, 2015**
- » **w3schools.com**

Pada modul kali ini, kita akan membahas mengenai XML. XML memiliki peran penting dalam perkembangan teknologi informasi. Sebelum membahas lebih lanjut mengenai XML, Anda perlu mengetahui dasar-dasar dari HTML dan Javascript.

1. XML

Berikut ini adalah beberapa penjelasan mengenai XML yang dikutip dari w3schools.com antara lain sebagai berikut.

- » XML merupakan singkatan dari *eXtensible Markup Language*
- » XML merupakan bahasa markup seperti HTML
- » XML dirancang untuk menyimpan data dan pertukaran data
- » XML didesign untuk mendeskripsinya dirinya sendiri
- » XML direkomendasikan oleh W3C

Seperti yang telah dijelaskan diatas, XML dirancang untuk mendeskripsikan sebuah data. XML merupakan sebuah alat yang bersifat independent untuk membawa sebuah informasi.

Masih banyak orang kesulitan untuk membedakan kegunaan dan fungsi dari HTML dan XML. XML dan HTML dirancang dengan tujuan yang berbeda. berikut ini adalah beberapa perbedaan antara HTML dan XML, antara lain:

- » XML dirancang untuk membawa informasi sebuah data (fokus pada data apa itu sendiri)
- » HTML dirancang untuk menampilkan data (fokus pada bagaimana data ditampilkan)
- » XML tidak memiliki tag yang ditetapkan
- » HTML memiliki tag-tag yang telah ditetapkan yang memiliki fungsi tertentu

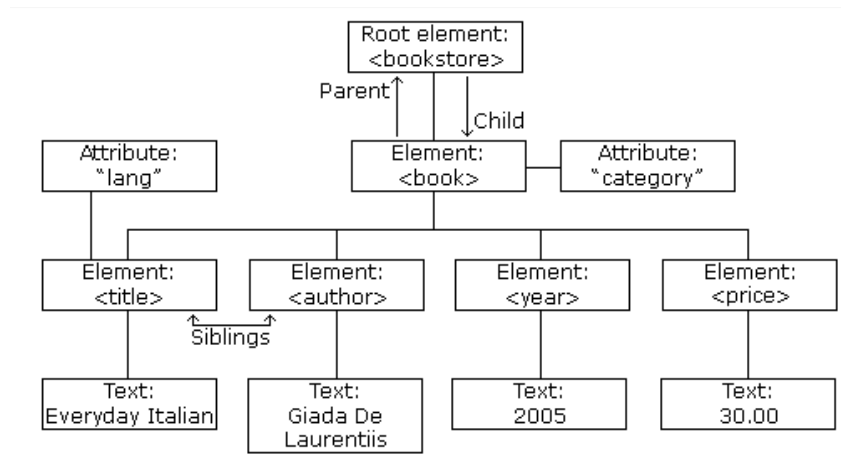
Dapat diambil sebuah kesimpulan bahwa XML bukanlah pengganti HTML. XML memungkinkan pihak developer untuk mendefinisikan tag-nya dan struktur dokumennya tersendiri. HTML adalah tentang menampilkan informasi, sedangkan XML adalah pembawa informasi. Maka dapat dikatakan bahwa XML melengkapi HTML.

2. XML Structure

Dokumen XML berberbentuk seperti pohon (*tree structure*), dimulai dari akar (*roots*) hingga daun (*leaves*). Setiap dokument XML digambarkan dalam bentuk elemen pohon, yang dimulai dari elemen akar dampai dengan elemen daun. Semua elemen dapat memiliki sub-elemen.

```
<root>
  <child>
    <subchild> ... </subchild>
  </child>
</root>
```

Gambar di bawah ini adalah sebuah bentuk dari struktur pohon XML.



Dari struktur di atas dapat dituliskan dalam bentuk dokumen XML sebagai berikut.

```
<?xml version="1.0" encoding="UTF-8"?>
<bookstore>
  <book category="Cooking">
    <title lang="En">Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>
</bookstore>
```

3. XML Syntax

Untuk menulis sebuah sintaks pada dokumen XML adalah hal sangat mudah. Berikut ini adalah aturan dalam menuliskan sebuah sintaks XML antara lain.

- » Dokumen XML harus memiliki elemen akar (*root element*)
- » Dokumen XML akan diawali dengan sebuah prolog yang menandakan dokumen tersebut adalah dokumen XML
- » Setiap elemen XML harus memiliki tag penutup
- » XML bersifat case sensitive
- » Penulisan XML yang bersarang harus sesuai dengan format yang benar
- » Nilai dari sebuah atribut harus berada diantara tanda kutip dua (") atau kutip satu (')
- » XML memiliki karakter khusus
- » Penulisan komentar dalam XML sama seperti pada HTML
- » XML memperhitungkan jumlah spasi

4. XML Element

Elemen XML adalah semua bagian dimulai dari tag elemen awal sampai dengan tag elemen akhir.

```
<title>Harry Potter</title>
```

Sebuah elemen XML dapat diperluas untuk membawa informasi yang lebih banyak. Elemen XML dapat berisi teks, atribut, elemen XML lain, dan mungkin juga ketiga gabungannya. Perhatikan sintaks XML di bawah ini.

```
<?xml version="1.0" encoding="UTF-8"?>
<bookstore>
  <book category="Novel">
    <title>Harry Potter</title>
    <author>J.K Rowling</author>
    <year>2005</year>
    <price>299.999</price>
  </book>
</bookstore>
```

Berikut ini penjelasan dari sintaks di atas.

1. <bookstore> dan <book> memiliki suatu kontent elemen, karena mengandung unsur-unsur lain
2. <title>, <author>, <year>, dan <price> mengandung element yang berisi teks
3. <book> memiliki sebuah atribut yaitu category="novel"

Sebuah elemen XML juga dapat tidak memiliki nilai atau berupa elemen kosong. Berikut ini adalah bentuk penulisan elemen kosong pada XML.

```
<title></title>
```

Atau dapat juga dituliskan sebagai berikut.

```
<title />
```

XML tidak memiliki tag elemen khusus, dalam hal ini pihak developer dapat mendeskripsikan sendiri nama dari elemen tersebut. Akan tetapi penamaan elemen XML memiliki beberapa aturan yang perlu diperhatikan diantaranya.

- » Nama elemen bersifat case sensitive
- » Nama elemen harus diawali dengan huruf atau underscore (_)
- » Nama elemen tidak boleh mengandung spasi
- » Nama elemen sebaiknya singkat dan sederhana

5. XML Attributes

XML juga dapat memiliki atribut seperti halnya XML. Atribut menyediakan informasi yang bukan merupakan bagian dari data. Nilai suatu atribut berada diantara tanda kutip dua (") atau kutip satu (').

```
<book category="Novel">Harry Potter</book>
```

Berikut ini adalah perbedaan antara XML Elemen dan XML Atribut

<pre><?xml version="1.0" encoding="UTF-8"?> <bookstore> <book> <category>Novel</category> <title>Harry Potter</title> <author>J.K Rowling</author> <year>2005</year> <price>299.999</price> </book> </bookstore></pre>	<pre><?xml version="1.0" encoding="UTF-8"?> <bookstore> <book category="Novel"> <title>Harry Potter</title> <author>J.K Rowling</author> <year>2005</year> <price>299.999</price> </book> </bookstore></pre>
--	--

Atribut memiliki beberapa keterbatasan, antar lain:

- » Atribut tidak dapat berisi beberapa nilai
- » Atribut tidak dapat berisi struktur pohon
- » Atribut tidak mudah diperluas
- » Atribut sulit dibaca dan dipelihara

5.1 XML Atribut untuk Metadata

Terkadang ID dapat mereferensikan elemen. ID dapat digunakan untuk mengidentifikasi XML dalam banyak cara yang sama seperti atribut ID pada HTML. Berikut ini contoh penulisan XML atribut untuk metadata.

```
<?xml version="1.0" encoding="UTF-8"?>
<bookstore>
  <book id="01">
    <category>Novel</category>
    <title>Harry Potter</title>
    <author>J.K Rowling</author>
```

```
<year>2005</year>
<price>299.999</price>
</book>
<book id="02">
<category>Diklat</category>
<title>Matematika Dasar</title>
<author>Rinaldi</author>
<year>2003</year>
<price>59.000</price>
</book>
</bookstore>
```

ID pada atribut diatas adalah untuk mengidentifikasi buku yang berbeda (bukan bagian dari buku itu sendiri). Metadata harus disimpan sebagai atribut, dan data itu sendiri disimpan sebagai elemen.

6. Membuat dan Menampilkan XML

Langkah-langkah untuk membuat sebuah dokumen XML, hampir sama seperti membuat sebuah dokumen HTML atau CSS. Perbedaannya hanya pada ekstensi dokumennya. Untuk dokumen XML menggunakan ekstensi .XML atau .xml. Editor yang digunakan juga sama seperti editor yang digunakan untuk membuat dokumen HTML.

LATIHAN!

Salin sintaks dokumen XML pada contoh sebelumnya <bookstore>. Simpan dengan nama `catalog_book.xml`. Selanjutnya coba buka dokumen XML tersebut pada browser.

Dokumen XML dapat ditampilkan hampir di semua browser. Tampilan dokumen XML adalah berbentuk sebuah kumpulan data. Berikut ini contoh dari hasil tampilan XML yang berisi data buku.

```
- <bookstore>
  - <book>
    <category>Novel</category>
    <title>Harry Potter</title>
    <author>J.K Rowling</author>
    <year>2005</year>
    <price>299.999</price>
  </book>
  - <book>
    <category>Diklat</category>
    <title>Matematika Dasar</title>
    <author>Rinaldi</author>
    <year>2003</year>
    <price>59.000</price>
  </book>
</bookstore>
```

Tampilan XML akan membentuk struktur pohon (*tree structure*). Tombol minus (-) atau plus (+) dapat di-klik untuk meng-expand atau meng-collapse setiap elemen XML.

Dokumen XML tidak membawa informasi bagaimana data ditampilkan. Bagaimana jika kita ingin memperindah bentuk tampilan sebuah dokumen XML?

7. XML dan CSS

CSS dapat digunakan untuk memberikan style pada dokumen XML. Anda hanya perlu membuat sebuah dokumen CSS (eksternal CSS) yang berisi style-style dan selanjutnya dokumen CSS akan dihubungkan dengan XML, dengan menggunakan perintah sebagai berikut.

```
<?xml-stylesheet type="text/css" href="# ">
```

Baris perintah tersebut dituliskan dibawah deskripsi sebuah dokumen XML. Tanda pagar (#) akan diganti isinya menjadi nama dokumen CSS yang telah Anda buat.

Untuk melihat bagaimana cara menampilkan XML dengan menggunakan CSS, silahkan download **latihan-xml-css.zip** di web <http://edu.script.id>.

8. XSLT

XSLT adalah singkatan dari *eXtensible Stylesheet Language Transformations*. XSLT adalah bahasa style sheet yang dianjurkan untuk dokumen XML. XSLT jauh lebih canggih dibandingkan dengan CSS. Dengan XSLT, Anda dapat menambahkan atau menghapus elemen dan atribut dari/ke output file. XSLT memungkinkan Anda untuk mengatur ulang elemen, menyembunyikan elemen, menampilkan elemen, dan banyak lagi lainnya.

Cara menampilkan XSLT adalah dengan mentransformasi XML ke HTML sebelum ditampilkan di browser. Dan Anda membutuhkan koneksi ke <http://www.w3.org/TR/WD-xsl>.

XSLT menggunakan XPath untuk menemukan lokasi informasi di dalam dokumen XML.

Untuk melihat bagaimana cara menampilkan XML dengan menggunakan XSLT, silahkan download **latihan-xml-xslt.zip** di web <http://edu.script.id>.

LATIHAN!

1. Buat XML berisi data mahasiswa (nama, nim, jurusan), dan tampilkan dengan menggunakan CSS
2. Buat XML berisi data jadwal kuliah Anda (matakuliah, dosen, ruang, jam), dan tampilkan dalam bentuk table dengan menggunakan XSLT