
KickStart Tutorial XML

version 1.0

free ebooks by spiderpro



General information

The tutorial is available at SpiderPro in the following versions:

- **Online HTML** <http://www.spiderpro.com/bu/buxmlm001.html>
- **PDF** <http://www.spiderpro.com/ebooks/kickstartxml.pdf>
- **Zipped PDF** <http://www.spiderpro.com/ebooks/kickstartxml.zip>

The *KickStart Tutorial XML* is free. If somebody asked you money for it, you've been swindled!

You're allowed to distribute this ebook as long as you leave the original pdf-file intact and you don't charge anything for it.

Happy XML-ing

Jan Kampherbeek
Webmaster of SpiderPro
<http://www.spiderpro.com/>
jan@spiderpro.com

This tutorial

In this tutorial you will learn what XML is about.

You'll understand the basic XML syntax. And you will know what's needed to make XML usable.

You won't be an XML expert after following this kickstart tutorial. But you'll understand the basics of XML. And you'll be able to understand XML Documents and most of XML DTD's.

Index

General information	2
This Tutorial	3
Index	4
Why do we needXML ?	5
What is XML ?	6
The general structure of XML	7
XML Tags	8
Elements and sub elements	9
XML documents	10
XML attributes	12
Well formed XML documents	13
Valid XML documents	14
XML: The DTD	15
Presenting XML documents	18
About SpiderPro	19
Disclaimer	19

Why do we need XML?

Data-exchange

XML is used to aid the exchange of data. It makes it possible to define data in a clear way.

Both the sending and the receiving party will use XML to understand the kind of data that's been sent. By using XML everybody knows that the same interpretation of the data is used

Replacement for EDI

EDI (Electronic Data Interchange) has been for several years the way to exchange data between businesses.

EDI is expensive, it uses a dedicated communication infrastructure.

And the definitions used are far from flexible.

XML is a good replacement for EDI. It uses the Internet for the data exchange. And it's very flexible.

More possibilities

XML makes communication easy. It's a great tool for transactions between businesses.

But it has much more possibilities.

You can define other languages with XML. A good example is WML (Wireless Markup Language), the language used in WAP-communications.

WML is just an XML dialect.

What is XML ?

Simpler SGML

XML is a meta-language.

A meta-language is a language that's used to define other languages. You can use XML for instance to define a language like WML.

XML is a smaller version of SGML. It's easy to master and that's a major advantage compared to SGML which is a very complex meta-language.

XML: What it can do

With XML you can :

- Define data structures
- Make these structures platform independent
- Process XML defined data automatically
- Define your own tags

With XML you cannot

- Define how your data is shown. To show data, you need other techniques.

The general structure of XML

Define your own tags

In XML, you define your own tags.

If you need a tag <TUTORIAL> or <STOCKRATE>, that's no problem.

DTD or Schema

If you want to use a tag, you'll have to define its meaning.

This definition is stored in a **DTD** (**D**ocument **T**ype **D**efinition). You can define your own DTD or use an existing one.

Defining a DTD actually means defining a XML language.

An alternative for a DTD is Schema.

Showing the results

Often it's not necessary to display the data in a XML document. It's for instance possible to store the data in a database right away.

If you want to show the data, you can. XML itself is not capable of doing so.

But XML documents can be made visible with the aid of a language that defines the presentation.

XSL (**eX**tensible **S**tylesheet **L**anguage) is created for this purpose. But the presentation can also be defined with **CSS** (**C**ascading **S**tyle **S**heets).

XML Tags

Tags

XML tags are created like HTML tags. There's a start tag and a closing tag.

<TAG>content</TAG>

The closing tag uses a slash after the opening bracket, just like in HTML.

The text between the brackets is called an element.

Syntax

The following rules are used for using XML tags:

- Tags are case sensitive. The tag **<TRAVEL>** differs from the tags **<Travel>** and **<travel>**
- Starting tags always need a closing tag
- All tags must be nested properly
- Comments can be used like in HTML: **<!-- Comments -->**
- Between the starting tag and the end tag XML expects the content.
<amount>135</amount> is a valid tag for an element amount that has the content 135

Empty tags

Besides a starting tag and a closing tag, you can use an empty tag. An empty tag does not have a closing tag.

The syntax differs from HTML: **<TAG/>**

Elements and sub elements

Elements and children

With XML tags you define the type of data. But often data is more complex. It can consist of several parts.

To describe the element car you can define the tags `<car>mercedes</car>`. This model might look like this:

```
<car>
<brand>volvo</brand>
<type>v40</type>
<color>green</color>
</car>
```

Besides the element car three other elements are used: **brand**, **type** and **color**. Brand, type and color are sub-elements of the element car. In the XML-code the tags of the sub-elements are enclosed within the tags of the element car. Sub-elements are also called children

XML documents

The XML declaration

The first line of an XML document is the XML declaration.

It's a special kind of tag:

```
<?xml version="1.0"?>
```

The version 1.0 is the actual version of XML.

The XML declaration makes clear that we're talking XML and also which version is used.

The version identification will become important after new versions of XML are used.

The root element

All XML documents must have a root element.

All other elements in the same document are children of this root element. The root element is the top level of the structure in an XML document.

Structure of an XML page

```
<?xml version="1.0"?>
<root>
  <element>
    <sub-element>
      content
    </sub-element>
    <sub-element>
      content
    </sub-element>
  </element>
</root>
```

All elements must be nested. The level of nesting can be arbitrarily deep.

Thank You for previewing this eBook

You can read the full version of this eBook in different formats:

- HTML (Free /Available to everyone)
- PDF / TXT (Available to V.I.P. members. Free Standard members can access up to 5 PDF/TXT eBooks per month each month)
- Epub & Mobipocket (Exclusive to V.I.P. members)

To download this full book, simply select the format you desire below

