

Theory: XML

🕒 15 minutes 0 / 5 problems solved

Skip this topic

Start practicing

2704 users solved this topic. Latest completion was 7 minutes ago.

XML (eXtensible Markup Language) is a text-based format for storing and exchanging structured data on the Internet. In this format, the data is presented as documents with a clear and flexible structure. An XML document can be stored on the computer with the `.xml` extension that is often used to keep configuration files of programs.

At this moment, XML is one of the most popular formats around the world, used both by small startups and huge companies. XML is especially valued for being very expressive for people and easy for machine processing.

§1. Tags and elements

Each XML document consists of **tags** and **elements**.

A **tag** is a string with an assigned meaning like a book, a person or something of the sort. It is interesting that XML does not provide tags at all, but it gives developers an opportunity to invent tags independently.

An **element** is a building block of an XML structure: it may contain text, tags, other elements and attributes.

Here is an example of an XML document that describes a book with a title and one author:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <book>
3   <title>The Three-Body Problem</title>
4   <author>Liu Cixin</author>
5 </book>
```

This document has three tags enclosed in angle brackets: `<book>`, `<title>` and `<author>`.

By **element** we understand the combination of a starting tag with the corresponding ending tag together with their content. Elements set the structure of a document since they can be nested in each other.

Our document has the following elements:

- `<book>...</book>` contain two other elements;
- `<title>The Three-Body Problem</title>`
- `<author>Liu Cixin</author>`

All elements should have a closing tag (a similar tag, but with a slash `/` in front) or just end with a slash (`/>`).

The first line in the XML document is called a **prologue**:

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

It specifies the version of the XML standard (usually 1.0) and defines the encoding (here it's UTF-8). The prologue is optional, but if it's there, it must come first in the document.

Note that a prologue does not have a closing tag!

§2. Child elements

Each XML document always has a single element called **root**. This element can contain other elements called **child elements** which in their turn can have their own **children**.

Current topic:

[XML](#) ...

Topic is required for:

[XML Layouts](#) ...

[XML in Python](#) ...

[BeautifulSoup](#) ...

Table of contents:

[↑ XML](#)

[§1. Tags and elements](#)

[§2. Child elements](#)

[§3. Attributes](#)

[§4. Pros and cons of XML](#)

[Feedback & Comments](#)

The following XML document represents books contained in a library:

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <library>
3    <book>
4      <title>The Three-Body Problem</title>
5      <author>Liu Cixin</author>
6    </book>
7    <book>
8      <title>Modern Operating Systems</title>
9      <author>Andrew S. Tanenbaum</author>
10  </book>
11 </library>
```

Here, the root element `<library>` has two children `<book>` elements, while the elements `<title>` and `<author>` are the children of `<book>`. So, XML documents represent hierarchical structures that are often used in programming.

§3. Attributes

XML elements can possess **attributes** that provide additional information about the element.

The value of the attribute is always set in either double or single quotes. For example, the name of a picture can be written as follows:

```
1  <picture name="The Black Square"/>
```

If the value of the attribute contains double quotes, you need to use single quotes. For example:

```
1  <picture name='"Sunset at Sea", Ivan Aivazovsky' />
```

Sometimes, you can also see quotes replaced by special entity symbols (`"`):

```
1  <picture name="&quot;Sunflowers&quot;; Vincent van Gogh"/>
```

An element can contain more than one attribute:

```
1  <picture name='Sunset at Sea' author='Ivan Aivazovsky' />
```

The following XML document presents an art gallery:

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <gallery>
3    <picture name='Sunset at Sea' author='Ivan Aivazovsky' />
4    <picture name='The Black Square' author='Kazimir Malevich' />
5    <picture name='Sunflowers' author='Vincent van Gogh' />
6  </gallery>
```

As you can see, in some cases, attributes can replace child elements. There is no consensus about what's better to use. It usually depends on the data you are trying to model, your tools for XML processing and, of course, the people you work with.

Note that an element can have both attributes and child elements together.

§4. Pros and cons of XML

XML has won popularity due to its apparent advantages:

- it can be easily understood by machines and people alike;
- the format is based on international standards;

- it has a well-defined structure which facilitates the search and extraction of information;
- modern programming languages have libraries for processing XML documents automatically.

At the same time, XML has an important disadvantage. Its redundant syntax causes higher storage and transportation cost. It is especially important when we need to store or transfer a large amount of data.

Now that you have learned the basic principles of XML, it is time to put your skills to practice!

 Report a typo

267 users liked this theory. 1 didn't like it. What about you?



Start practicing

[Comments \(11\)](#)[Hints \(0\)](#)[Useful links \(0\)](#)[Show discussion](#)