



XML Attributes

[< Previous](#)[Next >](#)

XML elements can have attributes, just like HTML.

Attributes are designed to contain data related to a specific element.

XML Attributes Must be Quoted

Attribute values must always be quoted. Either single or double quotes can be used.

For a person's gender, the `<person>` element can be written like this:

```
<person gender="female">
```

or like this:

```
<person gender='female'>
```

If the attribute value itself contains double quotes you can use single quotes, like in this example:

Dark mode



```
<gangster name="George Shotgun Ziegler">
```

or you can use character entities:

```
<gangster name="George &quot;Shotgun&quot; Ziegler">
```

XML Elements vs. Attributes

Take a look at these two examples:

```
<person gender="female">
  <firstname>Anna</firstname>
  <lastname>Smith</lastname>
</person>
```

```
<person>
  <gender>female</gender>
  <firstname>Anna</firstname>
  <lastname>Smith</lastname>
</person>
```

In the first example, gender is an attribute. In the last example, gender is an element. Both examples provide the same information.

There are no rules about when to use attributes or when to use elements in XML.



The following three XML documents contain exactly the same information:

A date attribute is used in the first example:

```
<note date="2008-01-10">  
  <to>Tove</to>  
  <from>Jani</from>  
</note>
```

A <date> element is used in the second example:

```
<note>  
  <date>2008-01-10</date>  
  <to>Tove</to>  
  <from>Jani</from>  
</note>
```

An expanded <date> element is used in the third example: (THIS IS MY FAVORITE):

```
<note>  
  <date>  
    <year>2008</year>  
    <month>01</month>  
    <day>10</day>  
  </date>  
  <to>Tove</to>  
  <from>Jani</from>  
</note>
```



Some things to consider when using attributes are:

- attributes cannot contain multiple values (elements can)
- attributes cannot contain tree structures (elements can)
- attributes are not easily expandable (for future changes)

Don't end up like this:

```
<note day="10" month="01" year="2008"  
to="Tove" from="Jani" heading="Reminder"  
body="Don't forget me this weekend!">  
</note>
```

XML Attributes for Metadata

Sometimes ID references are assigned to elements. These IDs can be used to identify XML elements in much the same way as the id attribute in HTML. This example demonstrates this:

```
<messages>  
  <note id="501">  
    <to>Tove</to>  
    <from>Jani</from>  
    <heading>Reminder</heading>  
    <body>Don't forget me this weekend!</body>  
  </note>  
  <note id="502">  
    <to>Jani</to>  
    <from>Tove</from>  
    <heading>Re: Reminder</heading>  
    <body>I will not</body>  
  </note>  
</messages>
```



What I'm trying to say here is that metadata (data about data) should be stored as attributes, and the data itself should be stored as elements.

[< Previous](#)[Next >](#)

Start
DEVELOPING
Without Installations

Try Now



BACKEND
Support Included



COLOR PICKER



Get certified
by completing
a XML
course today!

Dark mode

[HTML](#)[CSS](#)[Get started](#)[Report Error](#)[Spaces](#)[Upgrade](#)[Newsletter](#)[Get Certified](#)[Dark mode](#)

[HTML](#)[CSS](#)

- [HTML Tutorial](#)
- [CSS Tutorial](#)
- [JavaScript Tutorial](#)
- [How To Tutorial](#)
- [SQL Tutorial](#)
- [Python Tutorial](#)
- [W3.CSS Tutorial](#)
- [Bootstrap Tutorial](#)
- [PHP Tutorial](#)
- [Java Tutorial](#)
- [C++ Tutorial](#)
- [jQuery Tutorial](#)

Top References

- [HTML Reference](#)
- [CSS Reference](#)
- [JavaScript Reference](#)
- [SQL Reference](#)
- [Python Reference](#)
- [W3.CSS Reference](#)
- [Bootstrap Reference](#)
- [PHP Reference](#)
- [HTML Colors](#)
- [Java Reference](#)
- [Angular Reference](#)
- [jQuery Reference](#)

Top Examples

- [HTML Examples](#)
- [CSS Examples](#)
- [JavaScript Examples](#)
- [How To Examples](#)
- [SQL Examples](#)
- [Python Examples](#)
- [W3.CSS Examples](#)
- [Bootstrap Examples](#)
- [PHP Examples](#)
- [Java Examples](#)
- [XML Examples](#)
- [jQuery Examples](#)

Get Certified

- [HTML Certificate](#)
- [CSS Certificate](#)
- [JavaScript Certificate](#)
- [Front End Certificate](#)
- [SQL Certificate](#)
- [Python Certificate](#)
- [PHP Certificate](#)
- [jQuery Certificate](#)

[Dark mode](#)

[HTML](#)[CSS](#)

C# Certificate
XML Certificate

[FORUM](#) | [ABOUT](#)

W3Schools is optimized for learning and training. Examples might be simplified to improve reading and learning. Tutorials, references, and examples are constantly reviewed to avoid errors, but we cannot warrant full correctness of all content. While using W3Schools, you agree to have read and accepted our terms of use, cookie and privacy policy.

Copyright 1999-2023 by Refsnes Data. All Rights Reserved.
W3Schools is Powered by W3.CSS.

