

# XML Syntax Part 2

<https://www.w3schools.com/xml/>

# XML Elements are Extensible

XML elements can be extended to carry more information.

Look at the following XML example:

```
<note>
  <to>Tove</to>
  <from>Jani</from>
  <body>Don't forget me this weekend!</body>
</note>
```

Should the application break or crash?

No. The application should still be able to find the <to>, <from>, and <body> elements in the XML document and produce the same output.

This is one of the beauties of XML. It can be extended without breaking applications.

Let's imagine that we created an application that extracted the <to>, <from>, and <body> elements from the XML document to produce this output:

## MESSAGE

**To:** Tove

**From:** Jani

Don't forget me this weekend!

Imagine that the author of the XML document added some extra information to it:

```
<note>
  <date>2008-01-10</date>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

# Entity References

Some characters have a special meaning in XML.

If you place a character like "<" inside an XML element, it will generate an error because it is interpreted as the start of a new element.

This will generate an XML error:

```
<message>salary < 1000</message>
```

To avoid this error, replace the "<" character with an **entity reference**:

```
<message>salary &lt; 1000</message>
```

## five predefined XML entity references (w. special meaning)

- &lt;                      standing for <
- &gt;                      standing for >
- &amp;                    standing for &
- &apos;                   standing for ‘ apostrophe
- &quot;                    standing for” quotation mark

# Exercise with entity references.

The image displays an XML editor interface with three main sections: a code editor, a rendered preview, and a tree view.

**Code Editor:** The XML code is as follows:

```
1 <?xml version="1.0" encoding="euc-kr" standalone="yes"
  ?>
2 <?xml-stylesheet type="text/css" href
  ="memo1_Simplified2.css"?>
3 <MEMO>
4   <TO> To: &quot; Eugene &quot;</TO>
5   <xml> xml &lt; contents </xml>
6   <xml />
7   <memo> memo contents &gt; </memo>
8   <FROM> From: Hong, Gil-Dong &apos; </FROM>
9   <CONTENTS> Can we dine out at &amp; 1 PM?</CONTENTS>
10 </MEMO>
```

Four blue arrows point to specific parts of the code: the first to the encoding attribute, the second to the `&lt;` entity reference, the third to the `&gt;` entity reference, and the fourth to the `&apos;` entity reference.

**Rendered Preview:** The preview shows a mobile interface with a header "File" and "URL", a checkbox for "Auto Update", a green "Tree View" button, and an advertisement for "Ads by Google" with buttons "Stop seeing this ad" and "Why this ad?". A blue arrow points to the "Stop seeing this ad" button.

**Tree View:** The tree view shows the XML structure:

- memo ..
  - to To: " Eugene "
  - xml xml < contents
    - xml
      - memo memo contents >
        - from From: Hong, Gil-Dong '
        - contents Can we dine out at & 1 PM?

# Exercise with entity references.

 **Code Beautify**

JSON Formatter | Hex Color Codes | HMAC Generator | My Ip | Search | Recent Links | More

---

## XML Viewer★

XML Input

sample

```
1 <?xml version="1.0" encoding="euc-kr" standalone="yes"?>
2 <MEMO test="abc:">
3   <xml> To: Eugene &lt; </xml>
4   <FR.OM> From: Hong, Gil-Dong &gt; </FR.OM>
5   <CONTENTS> Can we meet &amp; &apos; tomorrow? &quot;
6   </CONTENTS>
7 </MEMO>
8
9
10
```

Load Url

Browse

Tree View

Ads by Google

Stop seeing this ad

Result : XML Tree View

memo ..

@test: abc:

xml To: Eugene <

fr.om From: Hong, Gil-Dong >

contents Can we meet & tomorrow? "



```
graph TD
    memo[memo ..] --> test["@test: abc:"]
    memo --> xml["xml To: Eugene <"]
    memo --> fr["fr.om From: Hong, Gil-Dong >"]
    memo --> contents["contents Can we meet & tomorrow? \""]
```

# Comments in XML

The syntax for writing comments in XML is similar to that of HTML:

```
<!-- This is a comment -->
```

Two dashes in the middle of a comment are not allowed:

```
<!-- This is an invalid -- comment -->
```

# Element Attributes

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

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



**<person gender="female">**

**<person gender='female'>**

**<gangster name='George "Shotgun" Ziegler'>**

**<gangster name="George &quot; Shotgun&quot;  
Ziegler"**

# XML Elements vs. Attributes

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

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

no rules about when to use attributes or  
when to use elements in XML

# Avoid XML Attributes?

w3school

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>
```

attribute values

## XML Viewer☆

### XML Input

[sample](#)  

```
1 <?xml version="1.0" encoding="euc-kr" standalone="yes"?>
2 <MEMO test="abc: <>">
3   <xml> To: Eugene &lt; </xml>
4   <FR.OM> From: Hong, Gil-Dong &gt; </FR.OM>
5   <CONTENTS> Can we meet &amp; &apos; tomorrow? &quot;
6   </CONTENTS>
7 </MEMO>
8
9
10
```

Load Url

Browse

Tree View

← Ads by Google

Stop seeing this ad

Why this ad? ⓘ

### Result : XML Tree View

```
html ..
├── body ..
│   └── parsererror ..
│       @style: display: block; whi
│       solid #c77; padding: 0 1em
│       background-color: #fdd; cc
│       h3 This page contains the
│       div error on line 2 at colu
│       allowed in attributes value
```

# XML Viewer

## XML Input

[sample](#)  

```
1 <?xml version="1.0" encoding="euc-kr" standalone
  ="yes"?>
2 <?xml-stylesheet type="text/css" href="memo1.css"?>
3 <xml gender = 'a &apos;' temp='b &quot;|'>
4   <TO> To: Eugene</TO>
5   <FROM> From: Hong, Gil-Dong</FROM>
6   <CONTENTS> Can we dine out at 1 PM?</CONTENTS>
7 </xml>
```

Load Url

Browse

Tree View

Beautify / Format

## Result : XML Tree View

```
- xml ..
  @gender: a '
  @temp: b "
  - to To: Eugene
  - from From: Hong, Gil-Dong
  - contents Can we dine out at 1
```

```
<myMessage from='Eugene' to='Tom'>
  <message>Welcome to XML!</message>
  <hello>Welcome to XML!</hello>
  <blessing>Welcome to XML!</blessing>

</myMessage>
```

# XML Viewer☆

## XML Input

 sample  

```
1 <?xml version = "1.0"?>
2 <!DOCTYPE myMessage [
3 <!ELEMENT myMessage ( message | hello | blessing )?>
4 <!ELEMENT message ( #PCDATA )>
5 <!ELEMENT hello ( #PCDATA )>
6 <!ELEMENT blessing ( #PCDATA )>
7 ]>
8 <myMessage from='Eugene' to='Tom'>
9   <message>Welcome to XML!</message>
10  <hello>Welcome to XML!</hello>
11  <blessing>Welcome to XML!</blessing>
12
13 </myMessage>
14
```

Load Url

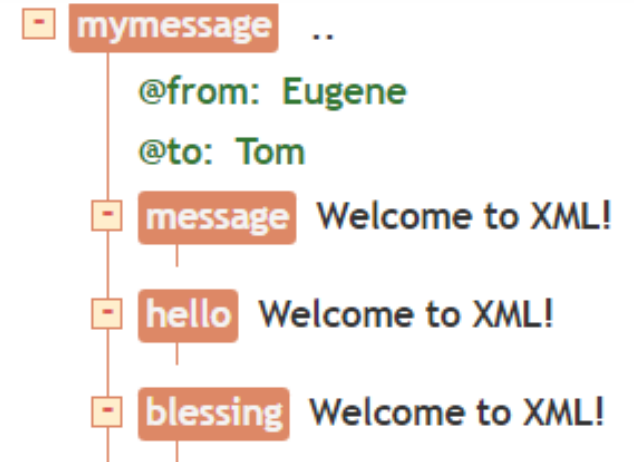
Browse

Tree View

Beautify / Format

Minify

## Result : XML Tree View



# XML Viewer☆

Sav

## XML Input

 sample  

```
1 <?xml version="1.0" encoding="euc-kr" standalone
  ="yes"?>
2 <?xml-stylesheet type="text/css" href="memo1.css"?>
3 <MEMO date = '"Sep. 8 2020"'>
4   <TO> To: Eugene</TO>
5   <FROM> From: Hong, Gil-Dong</FROM>
6   <CONTENTS> Can we dine out at 1 PM?</CONTENTS>
7 </MEMO>
8
```

Load Url

Browse

Tree View

Beautify / Format

Minify

## Result : XML Tree View

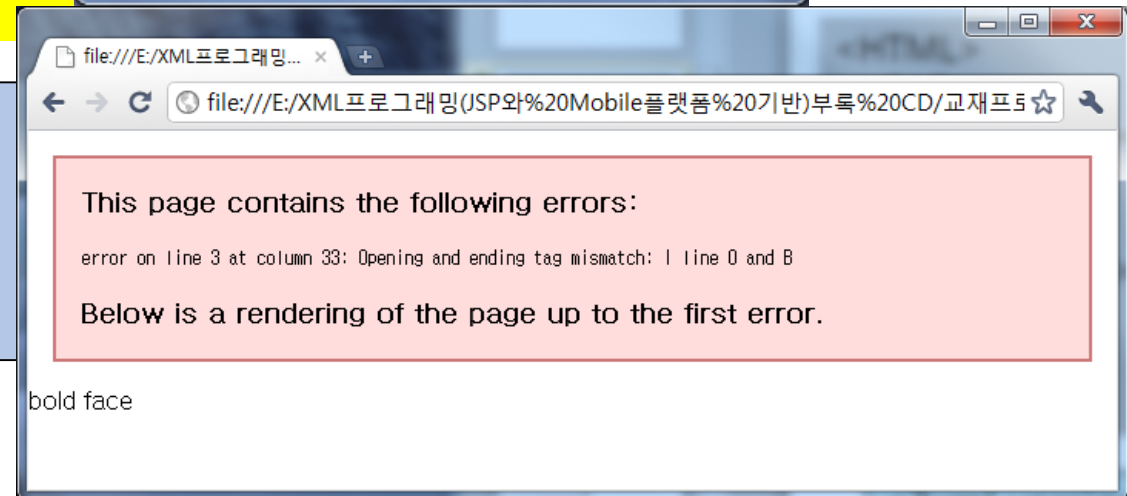
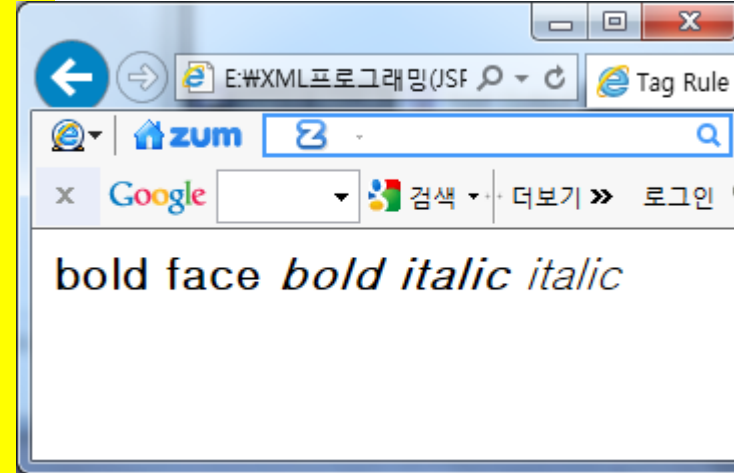


# tag overlapping

Type the source for HTML and XML, then apply them to the browser.

```
<HTML>
<HEAD>
<TITLE>Tag Rule</TITLE>
</HEAD>
<BODY>
<B>bold face <I>bold italic </B>
italic</I>
</BODY>
</HTML>
```

```
<?xml version="1.0" encoding="euc-kr" ?>
<tag>
<B>bold face <I>bold italic </B>italic </I>
</tag>
```





**This page contains the following errors:**

error on line 4 at column 8: Encoding error

**Below is a rendering of the page up to the error**

## Well-Formed of XML Syntax

```
<?xml version="1.0" ?>
<?xml-stylesheet type="text/css" href="memo1.css"?>
<MEMO>
  <TO> To: Eugene 유진 </tO>
  <FROM> From: Hong, Gil-Dong</FROM>
  <CONTENTS> Can we meet tomorrow?</CONTENTS>
</MEMO>
```

## Case-Sensitive Element Name

Try to use legal XML element names and test it on your browser.

<https://codebeautify.org/htmlviewer>

**Code Beautify** JSON Formatter XML Formatter Hex Color Codes Calculators JSON

# HTML Viewer

Sample ↻ 📁 💾 🗑️ 📄 ↗

```
1 <HTML>
2 <HEAD>
3 <TITLE>Tag Rule</TITLE>
4 </HEAD>
5 <BODY>
6 <B>bold face <I>bold italic </B> italic</I>
7 </BODY>
8 </HTML>
9
```

File URL

☒ Auto Update 2 Space ▾

Beautify HTML

Run / View

Output

**bold face** *bold italic italic*

Updated View

## HTML Viewer★

HTML Input sample 🗑️ ↗

```
1 <HTML>
2 <HEAD>
3 <TITLE>Tag Rule</TITLE>
4 </HEAD>
5 <BODY>
6 <B>bold face <I>bold italic </B> italic</I>
7 </BODY>
8 </HTML>
```

Load Url

Browse

Default Tab Space ▾

Beautify

Minify

Run

Result : HTML Output

**bold face** *bold italic italic*

# XML Viewer☆

Save & Share

## XML Input

 [sample](#)  

```
1 <?xml version="1.0" encoding="euc-kr" ?>
2 <tag>
3 <B>bold face <I>bold italic </B>italic </I>
4 </tag>
5
```

Load Url

Browse

Tree View

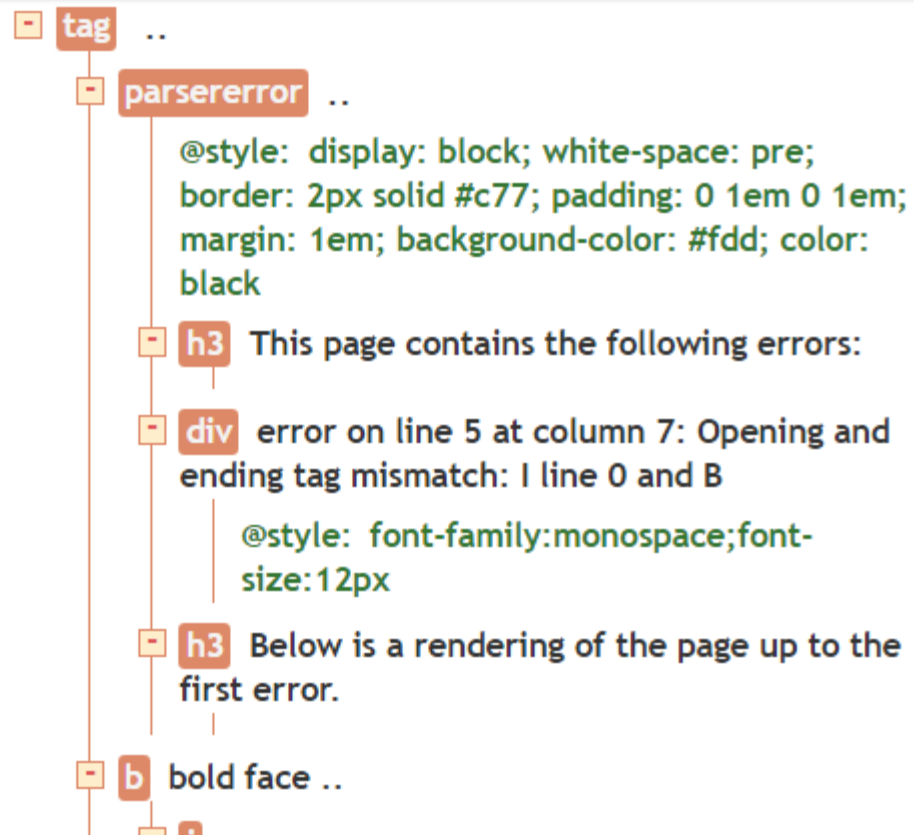
Beautify / Format

Minify

Ad closed by Google

## Result : XML Tree View



File Edit Navigate Search Project Run Design Window Help

Package Explorer ×

- MartianHello
- StudentTest
- XMLExamples
  - GolfCountryClub
  - Invoice
  - PhoneBanking
  - PublicationCatalogue
  - SpaceWarGame
  - readme.html
  - xmlIDTDTTest1.xml
  - YukiChainExample

readme \*xmlIDTDTTest1.xml ×

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE note [
3 <!ELEMENT note (to,from,heading,body)>
4 <!ELEMENT to (#PCDATA)>
5 <!ELEMENT from (#PCDATA)>
6 <!ELEMENT heading (#PCDATA)>
7 <!ELEMENT body (#PCDATA)>
8 ]>
9 <note>
10 <to1>Tove</to1>
11 <from>Jani</from>
12 <heading>Reminder</heading>
13 <body>Don't forget me this weekend</body>
14 </note>
```

Outline ×

- xml
- DOCTYPE:note
- note
  - to1 : Tove
  - from : Jani
  - heading : Reminder
  - body : Don't forget me this w

Design Source

Problems × @ Javadoc Declaration Console

2 errors, 0 warnings, 0 others

| Description   | Resource         | Path         | Location | Type            |
|---|------------------|--------------|----------|-----------------|
| ✖ Errors (2 items)  |                  |              |          |                 |
| ✖ Element type "to1" must be declared.                                    | xmlIDTDTTest1... | /XMLExamples | line 10  | Language Ser... |
| ✖ The content of element type "note" must match "(to,from,heading,body)". | xmlIDTDTTest1... | /XMLExamples | line 9   | Language Ser... |

