Cheatography

XML 1.0 Cheat Sheet

by nqramjets via cheatography.com/20223/cs/3057/

Common

XML stands for eXtensible Markup Language.

XML documents are structured, tagged documents composed of elements, attributes, processing instructions, namespace declarations, comments, and text.

Elements

`<root> </root>`

`<root />`

`<root> child data</root>`

There can be exactly one root element of an XML document. (The document element)

Elements have a name and may also have children. Child data may be text, elements, attributes, or other types of XML structures.

Element names are case-sensitive

Elements must be correctly nested:

`<outer><inner>Content</inner></outer>`✔

`<outer><inner>Content</outer>`⇄`</inner>`✖

Element names must start with a letter or underscore. The following characters may contain letters, digits, periods, hyphens, underscores, or colons.

Attributes

`key='value'`

`key="value"`

Elements can be annotated with attributes. Attributes cannot exist in a serialized document without a parent element.

Attributes are name/value pairs that are separated by equals sign, with a value surrounded by matching single or double quotes.

The attributes for an element are serialized within the start tag of an element.

`<User login="J.Smith" enrolled="2014-12-14" />`

`<User enrolled="2014-12-14">J.Smith</User>`

Attribute names follow the same rules as element names.

Namespaces

`<prefix:localname xmlns:prefix="URI" />`

`<tagname xmlns="URI" />`

Namespaces resolve ambiguity when tag names collide.

Namespace declarations appear in an element start tag and are typically mapped to shorter prefix.

Namespace declarations are scoped to the element they are defined in, and all descendants thereof.

All elements in an XML document are QNames (*qualified names*); A QName is a local name with an optional prefix.

Both the prefix and the local name are NCNames (*No-colon names*).

The namespace of an element with a given prefix is the namespace specified by the in-scope namespace declaration for that prefix. (*May be overridden*)

The namespace of un-prefixed elements is the default namespace.

Attribute names are QNames; unprefixed attributes are in no namespace, even if a default namespace is in scope.

Comments

`<!-- Commented text -->`

Comments can contain spaces, text, elements, and line breaks, and can therefore span multiple lines of XML

You may not use a double hyphen (`--`) within a comment itself; therefore, you cannot nest comments

Character References

&#DecimalUnicodeValue;

&#xHexadecimalUnicodeValue;

Character references can only be used for element and attribute content

Processing-Instructions

Predefined Entities

There are five predefined entities used in the XML markup which must be escaped in order to appear in XML content

&amp;	&
&lt;	<
&gt;	>
&quot;	"
&apos;	'

You **may not** use (<) or (&) anywhere in your XML document, except to begin a tag or an entity, respectively

CDATA

`<![CDATA[ Almost any content her`
`]]>`

Use a `CDATA` section to include content that you do not want the XML parser to interpret

CDATA stands for (unparsed) character data, as opposed to PCDATA (parsed character data)

You may not nest CDATA sections

CDATA sections may be used anywhere withing the root element of an XML document

If you need to include the string `]]>` in the CDATA text, and you are **not** closing the CDATA section, the closing > must be written as &gt;

`<?xml versio n="1.0"?>`
`<?t arget` **`data`**`?>`
Only processing instructions and comments may appear before the document lement
The processing instruction target `xml` is reserved
No processing instruction may start with the string `xml` or any recapitalization thereof