

Что такое XML?

XML, в переводе с англ. *eXtensible Markup Language* — расширяемый язык разметки. Используется для хранения и передачи данных.

Многие компьютерные системы содержат данные в несовместимых форматах. Обмен данными между несовместимыми системами (или обновленными системами) — трудоемкая задача для веб-разработчиков. Необходимо преобразовывать большие объемы данных, а несовместимые данные часто теряются.

XML хранит данные в текстовом формате. Это обеспечивает независимый от программного и аппаратного обеспечения способ хранения, транспортировки и обмена данными.

Этот формат рекомендован Консорциумом Всемирной паутины (W3C), поэтому он часто используется для передачи данных по API. В SOAP API это вообще единственно возможный формат входных и выходных данных!

Как устроен XML?

XML документы формируют древовидную структуру, которая начинается с "корневого" элемента и разветвляется на "дочерние" элементы.

XML документы используют простой и самоописательный синтаксис:

Пример XML документа:

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Напоминание</heading>
  <body>Не забудь обо мне в эти выходные!</body>
</note>
```

XML-пролог

Эта строка называется **прологом** XML :

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

Пролог XML является необязательным. Если он существует, он должен стоять первым в документе.

Документы XML могут содержать международные символы, такие как норвежский øæå или французский êèé.

Во избежание ошибок следует указать используемую кодировку или сохранить XML-файлы в формате UTF-8.

UTF-8 — это кодировка символов по умолчанию для XML-документов.

Теги

В XML каждый элемент должен быть заключен в теги. Тег — это некий текст, обернутый в угловые скобки: **<tag>**

Текст внутри угловых скобок — название тега. Тега всегда два:

- Открывающий (начальный) — текст внутри угловых скобок
<tag>
- Закрывающий (конечный) — тот же текст, но добавляется символ «/»
</tag>

Открывающий и закрывающий теги должны быть написаны с одинаковым регистром.

Теги XML чувствительны к регистру. Тег <Note> отличается от тега <note>.

В XML запрещено опускать закрывающий тег. Все элементы **должны** иметь закрывающий тег:

```
<p>This is a paragraph.</p>  
<br />
```

Примечание. Пролог XML не имеет закрывающего тега! Это не ошибка. Пролог не является частью документа XML.

Корневой элемент

XML документ должен содержать корневой элемент. Этот элемент является "родительским" для всех других элементов.

Все элементы в XML документе формируют иерархическое дерево. Это дерево начинается с корневого элемента и разветвляется на более низкие уровни элементов.

Все элементы могут иметь подэлементы (дочерние элементы):

Любой XML-документе должен содержать один **корневой элемент**, являющийся **родительским** для всех остальных элементов. Это тег, с которого документ начинается, и которым заканчивается.

```
<root>
  <child>
    <subchild>.....</subchild>
  </child>
</root>
```

Значение элемента

Значение элемента XML — это все, от (включая) начального тега элемента до (включая) конечного тега элемента.

```
<price>29.99</price>
```

Элемент может содержать:

- текст
- атрибуты
- другие элементы
- или смесь вышеперечисленного

Пустые XML-элементы

Элемент без содержимого называется пустым.

В XML вы можете указать пустой элемент следующим образом:

```
<element></element>
```

Вы также можете использовать так называемый самозакрывающийся тег:

```
<element />
```

Эти две формы дают идентичные результаты в программном обеспечении XML.

Пустые элементы могут иметь атрибуты.

Элементы XML должны быть правильно вложены

В HTML вы можете увидеть неправильно вложенные элементы:

```
<b><i>This text is bold and italic</b></i>
```

В XML все элементы **должны** быть правильно вложены друг в друга:

```
<b><i>This text is bold and italic</i></b>
```

В приведенном выше примере «Правильно вложенный» означает, что, поскольку элемент `<i>` открыт внутри элемента ``, он должен быть закрыт внутри элемента ``.

Правила написания имен XML

XML элементы должны следовать следующим правилам написания имен:

- Имена могут содержать буквы, числа и другие символы
- Имена не могут начинаться с цифры или символа пунктуации
- Имена не могут начинаться с сочетания "xml" (или XML, или Xml и т.п.)
- Имена не могут содержать пробельные символы

В качестве имен можно использовать любые слова. Нет зарезервированных слов.

Хорошая практика составления имен.

Старайтесь придумать описательные имена: `<first_name>`, `<last_name>`.

Имена следует составлять короткие и простые, вроде: `<book_title>`; а не: `<the_title_of_the_book>`.

Избегайте символ "-". Если вы напишете нечто вроде "first-name", то некоторые приложения могут решить, что вы вычитаете имя "name" из имени "first".

Избегайте символ ".". Если вы напишете нечто вроде "first.name", то некоторые приложения могут решить, что "name" это свойство объекта "first".

Избегайте символ ":". Двоеточие зарезервировано для механизма пространства имен.

Нелатинские символы, вроде, *éðá* вполне легальны в XML, однако, если некое приложение их не поддерживает, то возникнут проблемы.

Стили написания имен.

Для XML элементов не существует какого-либо определенного стиля написания имен. Тем не менее, вот несколько наиболее часто используемых стилей:

Стиль	Пример	Описание
Нижний регистр	<code><firstname></code>	Все буквы в нижнем регистре
Верхний регистр	<code><FIRSTNAME></code>	Все буквы в верхнем регистре
С символом подчеркивания	<code><first_name></code>	Слова разделяются символом подчеркивания
В стиле Pascal	<code><FirstName></code>	Первые буквы всех слов в верхнем регистре
"Верблюжий горб"	<code><firstName></code>	Первые буквы всех слов за исключением первого в верхнем регистре

Если вы выбрали какой-либо стиль написания имен, то следует последовательно придерживаться его!

Атрибуты элемента

У элемента могут быть атрибуты — один или несколько. Атрибуты предоставляют дополнительную информацию об элементе. Их мы указываем внутри отрывающегося тега после названия тега через пробел в виде

название_атрибута = «значение атрибута»

Например:

```
<query attr1="value 1">Виктор Иван</query>
```

```
<query attr1="value 1" attr2="value 2">Виктор Иван</query>
```

Значения атрибутов всегда должны быть заключены в кавычки. Можно использовать как одинарные, так и двойные кавычки.

Для пола человека элемент `<person>` можно записать так:

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

или вот так:

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

Если само значение атрибута содержит двойные кавычки, вы можете использовать одинарные кавычки, как в этом примере:

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

или вы можете использовать символы сущностей:

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

Взгляните на эти два примера:

Пример 1

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

Пример 2

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

В первом примере пол указан в атрибуте. Во втором, пол записан, как элемент. Оба примера предоставляют одну и ту же информацию.

Нет правил, регулирующих то, когда использовать атрибуты, а когда элементы.

При использовании атрибутов возникают некоторые проблемы:

- атрибуты не могут содержать множественные значения (элементы могут)
- атрибуты не могут содержать древовидные структуры (элементы могут)
- атрибуты сложно расширять (для будущих изменений)

Атрибуты нелегко читать и обслуживать. Используйте элементы для данных, а атрибуты для информации, не относящейся к данным.

Правила синтаксиса XML

Ссылки на объекты

Некоторые символы имеют особое значение в XML.

Если вы поместите внутри XML-элемента такой символ, как "<", это вызовет ошибку, поскольку синтаксический анализатор интерпретирует его как начало нового элемента.

Это вызовет ошибку XML:

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

Чтобы избежать этой ошибки, замените символ «<» на его сущность:

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

В XML есть 5 предопределенных сущностей:

Сущность	Символ	Значение
<	<	меньше, чем
>	>	больше, чем
&	&	амперсанд
'	'	апостроф
"	"	кавычки

Замечание: Только символы "<" и "&" строго запрещены в XML. Символ ">" допустим, но лучше его всегда заменять на сущность.

Комментарии в XML

Синтаксис комментариев в XML такой же, как и в HTML.

`<!-- Это комментарий -->`

Использование двух символов тире в середине комментария не допустимо.

Неверно:

`<!-- Это -- комментарий -->`

Странно, но так можно:

`<!-- Это - - комментарий -->`

В XML пробелы сохраняются

В HTML несколько последовательных пробельных символов усекаются до одного. В XML документе все пробельные символы сохраняются.