

## آشنایی با XML

سرواژه XML از عبارت EXtensible Markup Language گرفته شده است. Markup یا ساختبندی (تعیین ساختار)، به فرآیند افزودن کدهایی به اطلاعات برای تعریف ساختار اطلاعات و یا تعیین شیوه تظاهر اطلاعات، اطلاق می‌شود. مثالی از این نوع زبان، زبان ساختبندی HTML است که نمونه ای از آن را در زیر مشاهده می‌کنید:

```
<p>it is a <b>test</b>paragraph.</p>
```

p = paragraph

b = bold

کد فوق را می‌توان به منظور خوانایی بیشتر به صورت زیر بازنویسی کرد. این شیوه به صورت معمول توسط کدنویسان استفاده می‌شود:

```
<p>  
it is a  
<b>test</b>  
paragraph.  
</p>
```

این نوع ساختبندی در بیشتر نرم‌افزارهای ویرایشگر متنی (مثل Microsoft Word) جدید نیز کاربرد دارد. اما این نرم‌افزارها کدهای مربوط به ساختبندی را به صورت پنهان استفاده می‌کنند و مثلاً به جای افزودن کدهای مربوط به رنگی کردن متون، متن را رنگی نمایش می‌دهند. این نوع نمایش را نمایش WYSIWYG می‌گویند که مخفف عبارت What You See Is What You Get است.

برای رعایت یکنواختی و جهانی سازی در شیوه استفاده از کدهای ساختنبندی، قواعد خاصی بنیانگذاری شد و این امر در نهایت منجر به پیدایش زبان ساختنبندی عمومی استاندارد (SGML یا Standard Generalized Markup Language) گردید. البته این زبان بیشتر به عنوان ابزاری برای توصیف سایر زبانهای ساختنبندی استفاده می شود.

یکی از کاربردهای شناخته شده این ابزار، زبان HTML یا HyperText Markup Language می باشد. یکی از مشکلات اصلی زبانهای SGML محدودیت به کدهای از پیش تعریف شده (مثل <p> یا <b>) بود. برای غلبه بر این مشکل، زبان XML متولد شد. برای مثال شما وقتی یک صفحه HTML را مشاهده می کنید، ممکن است عنوان یک مقاله را در بالای صفحه مشاهده کنید. اما هیچ کدی در متن سند HTML وجود ندارد که به شما بگوید "این عبارت عنوان یک مقاله است". شاید شما بتوانید از شیوه نوشتن مطالب یا پیش فرض ذهنی خود، عنوان یک مقاله، نام سازمان یا افراد را تشخیص دهید، اما نرم افزارهای رایانه ای نمی توانند این کار را به راحتی انجام دهند. پس زبانی برای تعریف اسناد دیجیتال مورد نیاز است. زبان مورد نظر باید قابل گسترش (extendible) باشد تا بتواند هر نوع کد مورد نظر برای توصیف اسناد را به آن اضافه کرد. به همین دلیل گروهی از اتحادیه W3C شروع به توسعه XML به عنوان زیر مجموعه ای از زبان SGML کردند. XML زبانی برای تعریف و تغییر شکل اطلاعات است. XML برخلاف HTML که برای نمایش اطلاعات در یک صفحه وب توسعه یافته است، بیشتر بر روی توصیف اطلاعات متمرکز شده است. به این ترتیب می توان XML را به یک ظرف دارای بخش بندی تشبیه کرد که هر بخش محتویات ویژه ای دارد. این امر از آمیختن داده ها با هم جلوگیری کرده و قابلیت دسترسی و استفاده از هر داده را به صورت مجزا ایجاد می کند. مثالی از یک قطعه کد XML را می توانید در زیر مشاهده کنید:

```
<titles>  
  <title>Study of Nations E-Readiness</title>  
</titles>  
<publication_date media_type="online">  
  <month>07</month>  
  <year>2004</year>
```

</publication\_date>

<publisher>

<publisher\_name>EPIST Co.</publisher\_name>

<publisher\_place>Tehran</publisher\_place>

</publisher>

XML از مجموعه کاراکترهای اروپایی و آسیایی حمایت می کند و به همین دلیل می تواند به عنوان یک قالب جهانی برای تعریف و تبادل اطلاعات استفاده شود.

## چرا از XML استفاده می کنیم؟

XML الزاما کار ناشران و تولیدکنندگان اطلاعات را آسانتر نمی کند. این ابزار نیازمند فهم، دانش و دوراندیشی است و رنج مستندسازی و توصیف داده ها را هموار نمی کند. طراحی یک XML ضعیف می تواند در هنگام نیاز به تغییرات پیش بینی نشده، ناشران را دچار دردسر کند. اما دلایل زیر برای استفاده از XML وجود دارند:

## سادگی

XML قوانین نسبتا ساده ای برای قالب بندی داده ها دارد. این امر به افزایش سرعت کار برنامه نویسان و تهیه کنندگان اسناد کمک می کند. مجموعه قوانین سختگیرانه XML باعث می شود خوانایی اسناد برای انسان ها و ماشین ها بیشتر شود.

## بسط پذیری

بسط پذیری XML به توسعه دهندگان امکان می دهد با ساخت مجموعه برجسب های بسط پذیری که می توانند به نحو کارآمدی برای کاربردهای مختلف استفاده شوند، الگوی DTD ویژه خودشان را بسازند. XML می تواند توسط چند استاندارد دیگر که قابلیت سبک (style)، پیونددهی (linking) و تعیین مرجع (referencing) را به مجموعه خواص ذاتی آن می افزایند، گسترش یابد.

## قابلیت عملکرد بینابینی

XML می تواند در پلاتفورم های مختلفی استفاده شود و توسط ابزارهای متنوعی تفسیر گردد. به خاطر ثبات ساختار سند XML، مفسرهای XML می توانند با هزینه ای نسبتاً کم با هر زبان برنامه نویسی ساخته شوند.

## دسترسی آسان و باز بودن

استانداردهای XML کاملاً باز متن و رایگان بوده و از طریق وب در دسترس هستند.

## قواعد اسناد XML کدامند؟

اسناد XML متنی و قابل خواندن برای انسان هستند. اسناد XML با استفاده از هر نوع ویرایشگر متنی و برخی مرورگرهای وب قابل مشاهده هستند. قواعد XML چندان پیچیده نیستند. اسناد XML با یک مقدمه (prolog) آغاز می شوند که الزامی نیست. یک مقدمه ساده در زیر نشان داده شده است:

```
<?xml version="1.0">
```

در صورت وجود بخش مقدمه، این بخش حتماً باید شماره نسخه مشخصات XML (XML specification) که از آن برای ساخت XML استفاده شده است بیان کند. از این بخش همچنین برای مقاصدی مانند تعیین خصوصیات مجموعه کاراکترهای سند XML یا اطلاعات جانبی آن استفاده می شود.

اسناد XML از مجموعه ای از عناصر (element) تشکیل می شوند. هر سند باید یک و تنها یک عنصر ریشه داشته باشد. تمام عناصر دیگر به عنوان شاخه ای از ریشه در نظر گرفته می شوند و این ساختار درختی (والد و فرزندی) می تواند بسط پیدا کند. در نمونه کد XML زیر عنصر journal به عنوان عنصر ریشه در نظر گرفته شده است.

```
<?xml version="1.0" encoding="utf-8"?>  
<journal>  
  <language>fa</language>
```

```
<title>The Blood Journal</title>  
<journal_id type= "isnet">jrn000101</journal_id >  
<journal_id type= "issn">>1487-897-980</journal_id >  
<pubdate>  
  <type>gregorian</type>  
  <year>2006</year>  
  <month>11</month>  
  <day>4</day>  
</pubdate>  
<volume>4</volume>  
<number>9</number>  
</journal>
```

عناصر به صورت برجسب‌های آغاز (<element>) و پایان (</element>) نوشته می‌شوند. تمام عناصر باید از نظر داشتن برجسب‌های باز و بسته و نیز ترتیب باز یا بسته شدن عناصر متعادل باشند. داده‌ها (مانند نام نشریه) در بین برجسب‌های باز و بسته عناصر نوشته می‌شوند. اگر عنصری حاوی داده نباشد (خالی باشد) می‌توان آن را فقط به صورت برجسب پایانی نمایش داد.

```
<journal>  
  <language>fa</language>  
  <title>The Blood Journal</title>  
  </journal_id>  
</journal>
```

به این ترتیب می‌توان XML را به یک ظرف دارای بخش‌بندی تشبیه کرد که هر بخش محتویات ویژه‌ای دارد. این امر از آمیختن داده‌ها با هم جلوگیری کرده و قابلیت دسترسی و استفاده از هر داده را به صورت مجزا ایجاد می‌کند. عناصر درون یک سند XML بر سه نوع ساده، حاوی عنصر یا مرکب هستند. عناصر ساده تنها حاوی داده‌های متنی می‌باشند. عناصر حاوی عنصر تنها حاوی عناصر دیگر هستند. عناصر مرکب شامل داده‌های متنی و نیز عناصر فرزند دیگری می‌شوند که خود حاوی داده‌های دیگرند.

عناصر journal\_id در مثال بالا دارای یک صفت با نام type نیز می‌باشد. صفات حاوی اطلاعاتی درباره یک عنصر هستند. مقادیر وارد شده برای صفات باید در بین دو علامت نقل قول قرار داده شوند.

```
<journal_id type= "isnet">jrn000101</journal_id >  
<journal_id type= "issn">>1487-897-980</journal_id >
```

در مثال فوق صفت type را می‌توان به صورت عناصر فرزند بازنویسی کرد:

<journal\_id >  
<isnet> jrn000101</isnet>  
<issn>1487-897-980</issn>  
</journal\_id >

درباره این موضوع که کدام خصوصیات به صورت صفت و کدام خصوصیات به صورت عنصر فرزند نوشته شوند، نظرات مختلفی وجود دارد. در بیشتر مواقع عناصری که از اهمیت زیادی در تفسیر اسناد XML برخوردار نیستند به صورت صفت نوشته می شوند.

---

ارسال نظرات و تبادل دیدگاه:

علیرضا نعمتی - yektaweb@gmail.com

www.yektaweb.com