

تعریف نوعهای پیچیده

به عنصری که می‌تواند از عناصر دیگر و ویژگیها تشکیل شود نوع پیچیده می‌گویند. از آنجا که بیشتر اسناد XML عناصری دارند که از عناصر دیگر تشکیل شده‌اند، ایجاد نوعهای پیچیده در الگو کار بعیدی به نظر نمی‌رسد. چهار عنصر نوع پیچیده داریم که عبارتند از:

عناصر «فقط عنصر» که از عناصر یا ویژگیها تشکیل شده‌اند و متنی ندارند (شکل ۱-۷). عناصر «خالی» عنصر و متنی ندارند ولی شاید از ویژگیهایی تشکیل شوند (شکل ۲-۷). عناصر «مختلط» که شامل ترکیبی از عناصر، ویژگیها و یا متن می‌باشند. بیشتر، ترکیب عناصر و متن به کار می‌رود (شکل ۳-۷). عناصر «متنی» که فقط شامل متن هستند و شاید ویژگیهایی نیز داشته باشند (شکل ۴-۷).



```
code.html
<subspecies>
  <name language="English">Amur</name>
  <name language="Latin">P.f. altaica</name>
  <region>Far East Russia</region>
  <population year="1999">445</population>
</subspecies>
```

شکل ۱-۷. عنصر subspecies از عناصر دیگر تشکیل می‌شود ولی متنی ندارد. هرچند که این عنصر بدون ویژگی است، می‌تواند جزء عناصر «فقط عنصر» باشد.

```
code.html
<source sectionid="101" newspaperid="21"/>
<picture filename="tiger.jpg" x="200" y="197"/>
```

شکل ۲-۷. عناصر source و picture عناصر «خالی» هستند زیرا محتوایی ندارند. عناصر این مثال ویژگیهایی دارند که این امر برای تمام عناصر خالی الزامی نیست.

```
code.html
<description length="short">The <name
language="English">tiger</name> (panthera
tigris), largest of all cats, is one of the biggest and
most fearsome predators in the
world.</description>
```

شکل ۳-۷. عنصر description شامل متن و عنصر name و ویژگی است. بنابراین به دلیل داشتن محتویات گوناگون عنصر «مختلط» محسوب می‌گردد.

```
code.html
<name language="Latin">panthera tigris</name>
```

شکل ۴-۷. عنصر name فقط شامل متن و یک ویژگی است؛ بنابراین یک عنصر «فقط متنی» می‌باشد.

```
code.hsd
<xsd:complexType name="endspeciesType">
  <xsd:sequence>
    <xsd:element name="animal"
      type="animalType"/>
  </xsd:sequence>
</xsd:complexType>
```

شکل ۵-۷. در اینجا تعریف comlexType (که برای تعریف عنصر endangered_species به کار می‌رود) دیده می‌شود که از عنصر animal با نوع animalType تعریف شده تشکیل می‌گردد. نوعهای پیچیده می‌توانند شامل عناصر نوع پیچیده دیگر باشند.

تعریف عناصر فقط عنصری

به عنصر نوع پیچیده‌ای که می‌تواند از عناصر و ویژگیها تشکیل شود ولی متنی نداشته باشد عنصر فقط عنصری می‌گویند.

برای ایجاد نوع پیچیده‌ای که فقط شامل عناصر است :

- ۱- عبارت `<xsd:complexType>` را تایپ کنید.
- ۲- سپس عبارت `name="label"` را تایپ نمایید. label مشخص کننده نوع پیچیده است نه نام عنصر.
- ۳- برای تکمیل برچسب علامت `>` را تایپ کنید.
- ۴- به منظور تعیین عناصر تشکیل دهنده نوع پیچیده، یک دنباله (صفحه ۹۵)، گزینه (صفحه ۹۶) یا گروه نامنظم (صفحه ۹۷) اعلان کنید یا به یک گروه نامگذاری شده (صفحه ۹۹) اشاره نمایید.
- ۵- سپس ویژگیها (صفحه ۱۰۷) یا گروه ویژگیها (صفحه ۱۱۱) را که در عناصر این نوع ظاهر خواهند شد اعلان کرده یا به آنها اشاره کنید.
- ۶- سپس `</xsd:complexType>` را تایپ کنید.

نکته‌ها

- ◀ پس از ایجاد نوع پیچیده باید عنصر یا عناصر تشکیل دهنده آن را اعلان کنید (صفحه ۱۰۶).
- ◀ عناصر تشکیل دهنده یک نوع پیچیده باید بخشی از یک دنباله، گزینه، گروه نامنظم یا گروه نامگذاری شده باشند.

عناصری که در یک دنباله ظاهر می‌گردند

اگر عنصر نوع پیچیده‌ای از عناصر دیگر تشکیل می‌شود باید برای عناصر آن دنباله‌ای تعریف کرد.

برای تعریف دنباله‌ای از عناصر :

- ۱- عبارت `<xsd:sequence>` را تایپ کنید.
- ۲- اگر بخواهید می‌توانید تعداد دفعات ظاهر شدن دنباله عناصر را با تنظیم ویژگیهای `minOccurs` و `maxOccurs` تعیین کنید (صفحه ۱۰۱).
- ۳- برای تکمیل کار، علامت `>` را تایپ کنید.
- ۴- هر یک از اجزای دنباله را اعلان کنید (صفحه ۷۶ و ۱۰۶) یا ارجاع دهید (صفحه ۱۰۰).
- ۵- عبارت `<xsd:sequence>` را تایپ کنید.

نکته‌ها

- ◀ یک دنباله که ترتیب ظاهر شدن عناصر تشکیل دهنده را در سند XML تعیین می‌نماید تعریف کنید.
- ◀ یک دنباله می‌تواند شامل دنباله‌ها، گزینه‌ها (صفحه ۹۶) یا مراجع گروههای نامگذاری شده (صفحه ۹۹) باشد.
- ◀ یک دنباله می‌تواند شامل تعریف نوع پیچیده (صفحه ۹۴)، دنباله‌های دیگر یا مجموعه‌ای از گزینه‌ها (صفحه ۹۶) یا تعاریف گروههای نامگذاری شده (صفحه ۹۸) باشد.
- ◀ عنصر `xsd:sequence` معادل علامت کاما (,) در DTDها است.
- ◀ یک دنباله می‌تواند فقط شامل یک عنصر باشد.

```
code.xsd
<xsd:complexType name="animalType">
  <xsd:sequence>
    <xsd:element ref="name" minOccurs="2"/>
    <xsd:element name="threats"
      type="threatsType"/>
    <xsd:element name="weight" type="xsd:string"/>
    <xsd:element name="length" type="xsd:string"/>
    <xsd:element name="source"
      type="sourceType"/>
    <xsd:element name="picture"
      type="pictureType"/>
    <xsd:element name="subspecies"
      type="subspeciesType"/>
  </xsd:sequence>
</xsd:complexType>
```

شکل ۶-۷. هر عنصر تعریف شده با نوع `animalType` به ترتیب شامل عناصر `threats`، `name`، `weight`، `length`، `source`، `picture` و `subspecies` است.

ایجاد مجموعه‌ای از گزینه‌ها

گاهی لازم می‌شود عنصری از یک عنصر، گروهی از عناصر یا چیزهای دیگر تشکیل شود. این کار با ایجاد یک گزینه امکانپذیر می‌گردد.

برای ارائه یک گزینه :

۱- عبارت `<xsd:choice>` را تایپ کنید.
 ۲- با ویژگیهای `minOccurs` و `maxOccurs` می‌توانید تعداد دفعات ظاهر شدن مجموعه گزینه‌ها را تنظیم نمایید (صفحه ۱۰۱).

۳- علامت `>` را تایپ کنید.

۴- عناصر تشکیل دهنده مجموعه گزینه‌ها را اعلان کنید (صفحه ۷۶ و ۱۰۶) و یا ارجاع دهید (صفحه ۱۰۰).

۵- عبارت `</xsd:choice>` را تایپ کنید.

نکته‌ها

◀ مقدار پیش فرض ویژگیهای `minOccurs` و `maxOccurs` برابر یک است. در این صورت عناصر مجموعه گزینه‌ها فقط یک بار می‌توانند در سند XML ظاهر گردند. اگر مقدار ویژگی `maxOccurs` بزرگ‌تر از یک باشد تعداد دفعات ظهور عناصر بیشتر خواهد شد. همچنین عبارت `maxOccurs="unbounded"` معادل علامت * در مجموعه گزینه‌های DTD است (صفحه ۴۷).

◀ از طرفی مجموعه گزینه‌ها می‌تواند شامل دنباله‌های تودرتو، مجموعه گزینه‌های دیگر یا ارجاع به گروه‌های نامگذاری شده (صفحه ۹۹) باشد.

◀ مجموعه گزینه‌ها می‌تواند شامل تعریف نوع پیچیده (صفحه ۹۴)، دنباله، مجموعه گزینه‌های دیگر یا تعاریف گروه‌های نامگذاری شده (صفحه ۹۸) باشد.

◀ عنصر `xsd:choice` معادل خط عمودی در DTDها است (صفحه ۴۷).

```
code.xsd
<xsd:complexType name="animalType">
  ...
  <xsd:choice>
    <xsd:element name="subspecies"
      type="subspeciesType"/>

    <xsd:sequence>
      <xsd:element name="region"
        type="xsd:string"/>
      <xsd:element name="population"
        type="popType"/></xsd:sequence>
  </xsd:choice> ...
```

شکل ۷-۷ . برخی از گونه‌های در معرض خطر (endangered species). زیرگونه (subspecies) ندارند. در چنین مواقعی تهیه فهرستی از زیرگونه‌ها مفید نیست. بلکه بهتر است فهرستی از منطقه (region) و جمعیت (population) گونه‌ها تهیه کرد.

```
code.xml
<animal><name language="English">
  Tiger</name>
  ...
  <subspecies>
    <name language="English">Amur</name>
    <name language="Latin">P.t. altaica</name>
    <region>Far East Russia</region>
    <population year="1999">445</population>
  </subspecies>
  <subspecies>
    <name language="English">Balian</name>
  ...
</animal>
```

```
code.xml
<animal><name language="English">Great River
  Otter</name>
  ...
  <region>Peru, Northern Argentina</region>
  <population year="2000">5000</population>
</animal>
```

شکل ۷-۸ . عنصرهای animal در هر دو مثال معتبرند. در اولی اختصاص هر تعداد عنصر subspecies مجاز است و در دومی به جای subspecies دنباله‌ای از عناصر region و population داریم.

ایجاد عناصر نامرتب

اگر عنصری شامل عناصر دیگری باشد که رعایت ترتیب در آنها مهم نیست می‌توان عناصر داخلی را به صورت گروه `all` معرفی نمود.

برای ایجاد ترتیبهای گوناگون در عناصر:

- ۱- برای آغاز گروه عناصر نامرتب عبارت `<xsd:all>` را تایپ کنید.
- ۲- اگر بخواهید می‌توانید به کمک تنظیم ویژگیهای `minOccurs` و `maxOccurs` تعداد دفعات ظاهر شدن گروه نامرتب را تعیین نمایید (صفحه ۱۰۱).
- ۳- علامت `<` را تایپ کنید.
- ۴- براساس مطالب صفحات ۷۶، ۱۰۶ و ۱۰۰ می‌توانید عناصر گروه را اعلان کنید یا ارجاع دهید.
- ۵- برای تکمیل گروه، `<xsd:all>` را تایپ کنید.

نکته‌ها

- ◀ اعضای گروه `all` می‌توانند با توجه به مقادیر `minOccurs` یا `maxOccurs` یک یا هیچ دفعه با هر ترتیبی ظاهر شوند.
- ◀ مقادیر قابل قبول ویژگیهای `minOccurs` و `maxOccurs` صفر و یک است.
- ◀ در یک گروه `all` اعلانها یا ارجاعهای عناصر، به صورت جداگانه تعریف می‌شوند گروهی. درضمن هیچ عنصری نمی‌تواند بیش از یک بار ظاهر شود.
- ◀ گروه `all` تنها می‌تواند شامل تعریف نوع پیچیده (صفحه ۹۴) یا گروه نامگذاری شده (صفحه ۹۸) باشد.

```
code.xsd
<xsd:complexType name="subspeciesType">
  <xsd:all>
    <xsd:element ref="name"/>
    <xsd:element name="region"
      type="xsd:string" minOccurs="0"/>
    <xsd:element name="population">
      <xsd:complexType base="xsd:integer"
        derivedBy="extension">
        <xsd:attribute name="year"
          type="xsd:year"/>
      </xsd:complexType></xsd:element>
  </xsd:all>
</xsd:complexType>
```

شکل ۹-۷. در گروه `all` عناصر `region` و `population` با هر ترتیبی می‌توانند در عناصر `subspeciesType` ظاهر شوند. همچنین عنصر `region` اختیاری است؛ زیرا برای ویژگی `minOccurs` مربوط به آن عدد صفر در نظر گرفته شده است (صفحه ۱۰۱). حذف این عنصر مشکلی به وجود نمی‌آورد.

```
code.xml
<subspecies>
  <name language="English">Bengal</name>
  <population year="1999">3159</population>
  <region>India</region>
</subspecies>
<subspecies>
  <population year="1998">1227</population>
  <name language="English">Amoy</region>
</subspecies>
```

شکل ۱۰-۷. عناصر `subspecies` در هر دو مثال معتبرند؛ زیرا `region` اختیاری است و ترتیب عناصر اهمیتی ندارد.

نامگذاری گروهها

```

code.hsd
<xsd:schema>
  <xsd:group name="physical_traits">
    <xsd:sequence>
      <xsd:element name="weight"
        type="xsd:string"/>
      <xsd:element name="length"
        type="xsd:string"/>
      <xsd:element name="gestation"
        type="xsd:timeDuration"/>
      <xsd:element name="distinguishing"
        type="xsd:string"/>
    </xsd:sequence>
  </xsd:group>
  ...

```

شکل ۱۱-۷. یک گروه، فهرستی از عناصر مرتبط را که در یک یا چند عنصر با یکدیگر مورد استفاده قرار می‌گیرند تعریف می‌کند.

اگر مجموعه‌ای از عناصر در قسمت‌های گوناگون سند XML ظاهر گردند می‌توان آن عناصر را به صورت یک گروه در نظر گرفت تا فراخوانی آنها آسانتر گردد.

برای تعریف نام گروه :

- ۱- عبارت `<xsd:group` را تایپ کنید.
- ۲- سپس `name="label"` را تایپ نمایید. منظور از label واژه‌ای است که این گروه را معرفی می‌کند.
- ۳- علامت `>` را تایپ کنید.
- ۴- دنباله (صفحه ۹۵)، مجموعه گزینه‌ها (صفحه ۹۶) یا گروه نامرتب (صفحه ۹۷) تشکیل دهنده گروه را اعلان نمایید.
- ۵- برای تکمیل تعریف گروه عبارت `</xsd:group>` را تایپ کنید.

نکته‌ها

- ◀ یک گروه معادل پارامتر موجودیت در DTDهاست (صفحه ۶۰).
- ◀ در حالی که گروه در بالای الگو (و پایین `xsd:schema`) تعریف می‌شود امکان ارجاع به آن در تمام قسمت‌ها وجود دارد (صفحه ۹۹).
- ◀ در XML Schema به دنباله‌ها، مجموعه گزینه‌ها، گروه‌های نامرتب و گروه‌های نامگذاری شده، گروه‌ها یا گروه‌های مدل می‌گویند. در این کتاب به مجموعه اجزایی که با عنصر `xsd:group` تعریف می‌شوند و نامی دارند گروه‌های نامگذاری شده می‌گوییم.

ارجاع به گروه نامگذاری شده

پس از ایجاد یک گروه می‌توانید از گروههای دیگر و یا در تعریف نوعهای پیچیده به آن ارجاع کنید.

برای ارجاع به یک گروه :

۱- در بخشی از الگو، جایی که می‌خواهید عناصر گروه ظاهر شوند عبارت `<xsd:group>` را تایپ کنید.

۲- سپس `ref="label"` را تایپ نمایید. `label` واژه‌ای است که برای مشخص کردن گروه، هنگام ایجاد آن (مرحله ۲ صفحه قبل) به کار می‌برید.

۳- برای تکمیل ارجاع، علامتهای `</>` را تایپ کنید.

نکته

می‌توان به یک گروه در اعلان نوع پیچیده (صفحه ۹۴)، یک دنباله (صفحه ۹۵)، مجموعه‌ای از گزینه‌ها (صفحه ۹۶)، یک گروه نامرتب (صفحه ۹۷) یا گروههای نامگذاری شده دیگر ارجاع داد.

```
code.xsd
<xsd:element name="animal">
<xsd:complexType>
  <xsd:sequence>
    <xsd:element ref="name"/>
    <xsd:group ref="physical_traits"/>
    <xsd:element name="subspecies"
type="subspeciesType"/>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>

<xsd:element name="individual">
<xsd:complexType>
  <xsd:group ref="physical_traits"/>
  <xsd:attribute name="birthdate"
type="xsd:date"/>
  <xsd:attribute name="nickname"
type="xsd:string"/>
</xsd:complexType>
</xsd:element>
```

شکل ۷-۱۲ . عناصر `animal` و `individual` شامل فهرست عناصر گروه `physical_traits` می‌باشند (شکل ۷-۱۱). همچنین از عناصر و ویژگیهای جداگانه دیگری نیز تشکیل شده‌اند.

```
code.xml
<animal><name language="English">Tiger
</name><weight>500
pounds</weight><length>3 yards from nose
to tail</length>...<subspecies>...</animal>

<individual birthdate="1999-06-10"
nickname="Zoe"><weight>268
pounds</weight><length>2.5
yards</length>...</individual>
```

شکل ۷-۱۳ . عناصر می‌توانند به روشهای گوناگون در محتویات عناصر گوناگون به کار روند.

ارجاع به عناصر تعریف شده

عناصر نوعهای ساده و پیچیده که به صورت سراسری اعلان می‌گردند (داخل عنصر xsd:schema) می‌توانند در سند XML فراخوانی شده یا ارجاع داده شوند.

برای ارجاع به عنصر سراسری :

۱- در دنباله (صفحه ۹۵)، مجموعه گزینه‌ها (صفحه ۹۶)، گروه نامرتب (صفحه ۹۷) یا تعریف گروه نامگذاری شده (صفحه ۹۸) که عناصر ظاهر می‌شوند، عبارت `<xsd:element` را تایپ کنید.

۲- سپس `ref="label"` را تایپ کنید. `label` نام عنصر سراسری است.

۳- تعداد دفعات ظاهر شدن عنصر را در این محل تعیین نمایید (صفحه ۱۰۱).

۴- برای تکمیل ارجاع عنصر، علامتهای `>/` را تایپ کنید.

نکته‌ها

- ◀ به اعلان عنصر فقط در تعاریف دنباله‌ها، مجموعه گزینه‌ها، گروههای نامرتب و گروههای نامگذاری شده می‌توان ارجاع داد.
- ◀ به عنصر سراسری در هر قسمت از سند XML می‌توان ارجاع داد. هر ارجاع باید شامل مقدار دقیق `minOccurs` و `maxOccurs` باشد.
- ◀ به عناصر محلی در اجزای تعریفی که ظاهر می‌شوند به صورت خودکار ارجاع داده می‌شود. به این عناصر در قسمتهای دیگر نمی‌توان رجوع کرد.
- ◀ در صفحه ۷۱ درباره اعلانهای محلی و سراسری صحبت شده است.

```
code.xsd
<xsd:element name="name" type="nameType"/>
<xsd:complexType name="animalType">
  <xsd:sequence>
    <xsd:element ref="name" minOccurs="2"/>
    <xsd:element name="threats"
      type="threatsType"/>
    ...
  </xsd:complexType>
<xsd:complexType name="subspeciesType">
  <xsd:sequence>
    <xsd:element ref="name" minOccurs="1"/>
    <xsd:element name="region"
      type="xsd:string"/>
    ...
  </xsd:complexType>
```

شکل ۱۴-۷ . عنصر `name` در خط اول به صورت سراسری اعلان شده و می‌تواند در هر جای تعریف نوع پیچیده به آن رجوع کرد.

```
code.xml
<animal>
  <name language="English">Tiger</name>
  <name language="Latin">Panthera tigris</name>
  <threats><threat>poachers</threat>
  ...
  <subspecies>
    <name language="English">Amur</name>
    ...
  </subspecies>
</animal>
```

شکل ۱۵-۷ . براساس الگوی شکل ۱۴-۷، عنصر `name` باید حداقل دو بار در عنصر `animal` ظاهر شود؛ ولی هنگامی که شامل عنصر `subspecies` باشد کافی است یک بار ظاهر گردد.

کنترل تعداد دفعات ظاهر شدن

تعداد دفعات ظاهر شدن یک عنصر، دنباله، مجموعه گزینه‌ها، گروههای نامرتب و نامگذاری شده قابل کنترل می‌باشند.

برای تعیین حداقل تعداد ظاهر شدن یک عنصر یا گروه:

در برچسب شروع عنصر یا گروه عبارت "minOccurs="n" را تایپ کنید. منظور از n مقدار حداقل تعداد دفعات مجاز ظاهر شدن عنصر یا گروه در سند است.

برای تعیین حداکثر تعداد ظاهر شدن یک عنصر یا گروه:

در برچسب شروع عنصر یا گروه عبارت "maxOccurs="n" را تایپ کنید. منظور از n مقدار حداکثر تعداد دفعات مجاز ظاهر شدن عنصر یا گروه در سند است.

نکته‌ها

- ◀ ویژگی minOccurs می‌تواند مقادیر صحیح غیر منفی {۰, ۱, ۲, ...} داشته باشد.
- ◀ ویژگی maxOccurs می‌تواند هر عدد صحیح مثبتی باشد. همچنین می‌توان از واژه unbounded استفاده کرد تا محدودیتی برای تعداد دفعات ظاهر شدن عنصر وجود نداشته باشد.
- ◀ مقدار پیش فرض برای minOccurs و maxOccurs عدد یک است.
- ◀ ویژگیهای minOccurs و maxOccurs نمی‌توانند برای عناصر سراسری (که در پایین عنصر xsd:schema اعلان می‌شوند) به کار روند. آنها تنها برای عناصر محلی قابل استفاده هستند و می‌توانند به عناصر سراسری ارجاع داده شوند.
- ◀ این ویژگیها می‌توانند در xsd:choice، xsd:sequence، xsd:all، و ارجاع به گروههای نامگذاری شده استفاده گردند.

```
code.xsd
<xsd:element name="threat" type="xsd:string"
minOccurs="2" maxOccurs="5"/>
<xsd:element name="population"
type="xsd:integer"/>
```

شکل ۱۶-۷. ویژگیهای minOccurs و maxOccurs تعداد دفعات ظاهر شدن یک عنصر را کنترل می‌کنند.

```
code.xml
<threat>poachers</threat>
<threat>habitat destruction</threat>
<threat>trade in tiger bones for traditional Chinese
medicine (TCM)</threat>
<population>28</population>
```

شکل ۱۷-۷. عنصر threat می‌تواند حداقل ۲ و حداکثر ۵ بار استفاده شود. چون در اینجا ۳ بار ظاهر شده، معتبر می‌باشد. هرگاه برای عنصری نامی از ویژگیهای minOccurs و maxOccurs برده نشود (مانند عنصر population) مقدار پیش فرض (یعنی یک) برای آنها در نظر گرفته می‌شود و عنصر مربوطه باید یک بار در سند XML ظاهر گردد.

```
code.xsd
<xsd:choice minOccurs="0"
maxOccurs="unbounded">
<xsd:element name="sister_name"
type="xsd:string"/>
<xsd:element name="brother_name"
type="xsd:string"/>
</xsd:choice>
```

شکل ۱۸-۷. ویژگیهای minOccurs و maxOccurs را می‌توان برای دنباله‌ها، مجموعه گزینه‌ها، گروههای نامرتب یا ارجاع به گروههای نامگذاری شده به کار برد. در این مثال امکان ظاهر شدن گزینه‌های گروه به هر تعداد وجود دارد. یعنی عناصر sister_name و brother_name می‌توانند هیچ بار یا بیشتر نمایش داده شوند. مانند اضافه کردن علامت * به گزینه در یک DTD (صفحه ۹۶).

تعریف عناصر متنی

اگر به نوع ساده‌ای نیاز دارید که از متن خاصی تشکیل شده و شامل ویژگی‌هایی نیز می‌باشد می‌توانید از یک نوع پیچیده فقط متنی استفاده نمایید.

برای تعریف نوع پیچیده فقط متنی :

- ۱- عبارت `<xsd:complexType>` را تایپ کنید.
- ۲- سپس `name="label"` را تایپ کنید. `label` معرف نوع پیچیده است نه نام عنصری که با استفاده از تعریف نوع پیچیده در اعلانهای چند عنصر به کار می‌برید.
- ۳- برای تکمیل برجسب شروع علامت `>` را تایپ کنید.
- ۴- `<xsd:simpleContent>` را تایپ نمایید.
- ۵- اگر می‌خواهید برای نوع ساده محدودیتهایی به وجود آورید عبارت `<xsd:restriction>` را تایپ کنید.

اگر می‌خواهید نوع ساده را گسترش دهید `<xsd:extension>` را تایپ نمایید.

۶- سپس `>base="foundation"` را تایپ کنید. `foundation` تعریف نوع ساده‌ای است که نوع جدیدی روی آن ایجاد خواهید کرد.

۷- اگر دستورالعمل ۵ را اجرا کرده‌اید محدودیتهایی را که باعث محدود شدن تعریف نوع پیچیده می‌گردد اعلان نمایید (صفحه‌های ۹۰-۸۳).

۸- ویژگیها (صفحه ۱۰۸) یا گروه ویژگی‌هایی (صفحه ۱۱۱) را که در عناصر این نوع، ظاهر خواهند شد اعلان کنید.

۹- با توجه به عملی که در شماره ۵ انجام داده‌اید یکی از عبارتهای `<xsd:restriction>` یا `<xsd:extension>` را تایپ نمایید.

۱۰- عبارت `</xsd:simpleContent>` را تایپ کنید.

۱۱- عبارت `</xsd:complexType>` را برای تکمیل اعلان تایپ کنید.

```
code.xsd
<xsd:complexType name="popType">
  <xsd:simpleContent>
  <xsd:extension base="xsd:integer">
  <xsd:attribute name="year" type="xsd:year"/>
  </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```

شکل ۱۹-۷. عنصر `simpleContent` نشان می‌دهد که عناصر تعریف شده با تعریف این نوع پیچیده شامل متن خاصی (بر اساس نوع ساده) می‌باشند؛ ولی از عنصر دیگری تشکیل نمی‌گردند و در ضمن شاید ویژگی‌هایی نیز داشته باشند.

```
code.xsd
<xsd:element name="population"
  type="popType"/>
```

شکل ۲۰-۷. همیشه باید عنصر مورد استفاده تعریف نوع پیچیده را اعلان نمود (صفحه ۱۰۶).

```
code.xml
<population year="1999">445</population>
```

شکل ۲۱-۷. عنصر `population` با توجه به تعریف نوع پیچیده شکل ۱۹-۷ معتبر است؛ زیرا شامل یک عدد صحیح و ویژگی `year` می‌باشد.

```
code.xml
<population year="1999">Less than
500</population>
```

شکل ۲۲-۷. عنصر `population` غیر معتبر است؛ زیرا عبارت "Less than 500" یک عدد صحیح نیست. در حالی که نوع ساده‌ای که نوع پیچیده بر اساس آن ایجاد شده به صورت عدد صحیح می‌باشد.

تعریف عناصر خالی

به عناصری که از ویژگیها تشکیل شده ولی بین برچسب شروع و پایان محتویاتی ندارند عناصر خالی می‌گویند.

به منظور تعریف نوعهای پیچیده برای عناصر خالی:

- ۱- `<xsd:complexType` را تایپ کنید.
- ۲- سپس `<name="label">` را تایپ نمایید. label نوع پیچیده را مشخص می‌کند و نام عنصری که نوع پیچیده در اعلانهای چند عنصر به کار می‌برد نیست.
- ۳- `<xsd:complexContent` را تایپ کنید.
- ۴- `<xsd:extension base="xsd:anyType"/>` را تایپ نمایید. در نتیجه مشخص می‌شود که نوع پیچیده براساس نوع خاصی نیست و محتویاتی نخواهد داشت.
- ۵- ویژگیهای عناصر این نوع را در صورت وجود اعلان نمایید (صفحه ۱۰۸).
- ۶- `</xsd:complexContent>` را تایپ کنید.
- ۷- `</xsd:complexType>` را برای تکمیل اعلان تایپ نمایید.

نکته

برای به‌دست آوردن اطلاعات بیشتر درباره تعریف نوعهای پیچیده به صفحه‌های ۹۴ و ۱۰۷ مراجعه کنید.

```
code.xsd
<xsd:complexType name="sourceType">
  <xsd:complexContent>
    <xsd:extension base="xsd:anyType"/>
    <xsd:attribute name="sectionid"
      type="xsd:integer"/>
    <xsd:attribute name="newspaperid"
      type="xsd:integer"/>
  </xsd:extension>
</xsd:complexContent>
</xsd:complexType>
```

شکل ۲۳-۷. عنصر `complexContent` هنگامی که به اعلان ویژگیها نیاز دارید به کار می‌رود؛ ولی از عنصر دیگری تشکیل شده است.

```
code.xsd
<xsd:element name="source"
  type="sourceType"/>
```

شکل ۲۴-۷. عنصر را در محلی که باید ظاهر شود اعلان کنید.

```
code.html
<source sectionid="101" newspaperid="21"/>
```

شکل ۲۵-۷. پس از اعلان عنصر `source` می‌توان آن را در سند XML به کار برید.

```
code.xsd
<xsd:complexType name="paragraph"
  mixed="true">
  <xsd:sequence>
    <xsd:element name="name"
      type="nameType">
    </xsd:sequence>
    <xsd:attribute name="length"
      type="xsd:string"/>
  </xsd:complexType>
```

شکل ۲۶-۷. تعریف paragraph شامل یک عنصر و یک ویژگی است. ویژگی mixed="true" امکان وجود هر نوع متنی را فراهم می‌نماید.

```
code.xsd
<xsd:element name="description"
  type="paragraph"/>
```

شکل ۲۷-۷. باید عنصر را در محلی که قرار است ظاهر شود اعلان نمایید.

```
code.xml
<description length="short">The <name
  language="English">tiger</name>
  (panthera tigris), largest of all cats, is one of the
  biggest and most fearsome predators in the
  world.</description>
```

شکل ۲۸-۷. عنصر description شامل متن دلخواه (های لایت شده) و عناصر (های لایت و پررنگ) می‌باشد.

تعریف عناصر با محتویات ترکیبی

اگر یک نوع پیچیده، محتویات مختلط دارد باید این موضوع را هنگام اعلان نوع پیچیده اعلان نمود. یک نوع پیچیده مانند یک عنصر تعریف می‌شود.

برای ایجاد نوعهای پیچیده با محتویات مختلط :

- ۱- `<xsd:complexType` را تایپ کنید.
- ۲- سپس عبارت `name="label"` را تایپ کنید. `label` مشخصه نوع پیچیده است. چون `label` نام عنصر نیست می‌توان تعریف این نوع پیچیده را در اعلان چند عنصر به کار برد.
- ۳- برای اینکه عنصر بتواند شامل عناصر، ویژگیها و یا حتی متن باشد عبارت `mixed="true"` را تایپ نمایید.
- ۴- برای تکمیل برجسب شروع، علامت `>` را تایپ کنید.
- ۵- یک دنباله (صفحه ۹۵)، گزینه (صفحه ۹۶)، گروه نامرتب (صفحه ۹۷) یا ارجاع به یک گروه نامگذاری شده را (که می‌توان شامل هر یک از موارد مذکور باشد) اعلان کنید. در این صورت مشخص می‌گردد که نوع پیچیده شامل چه عناصری می‌تواند باشد.
- ۶- سپس ویژگیها (صفحه ۱۰۸) یا گروه ویژگیهایی (صفحه ۱۱۱) را که ممکن است در عناصر این نوع، در صورت وجود، ظاهر شوند اعلان کنید یا ارجاع دهید.
- ۷- با تایپ عبارت `</xsd:complextype` اعلان را تکمیل نمایید.

نکته

محتویات مختلط عناصر برای توصیف اطلاعاتی که بخشی از آن مبتنی بر متن است بسیار مناسب می‌باشد.

تعریف نوعهای پیچیده بر اساس

نوعهای پیچیده موجود

نوعهای پیچیده را می‌توان بر اساس نوعهای پیچیده موجود ایجاد کرد. نوع پیچیده جدید با تمام اطلاعات نوع موجود آغاز می‌گردد و سپس جزئیاتی به آن اضافه یا از آن حذف می‌گردد.

برای ایجاد نوعهای پیچیده بر اساس نوعهای موجود:

- ۱- برای آغاز تعریف نوع پیچیده جدید عبارت `<xsd:complexType>` را تایپ کنید.
- ۲- `"label">` name را تایپ نمایید. label مشخصه نوع پیچیده‌ای است که در حال ایجاد آن هستید.
- ۳- `<xsd:complexContent>` را تایپ نمایید.
- ۴- برای اضافه کردن اجزای نوع پیچیده جدید به نوع پیچیده موجود عبارت `<xsd:extension>` را تایپ کنید. یا اگر اجزای نوع پیچیده جدید محدودتر از نوع پیچیده موجود است عبارت `<xsd:restriction>` را تایپ نمایید.
- ۵- `"existing">` base را تایپ کنید. existing نام نوعی است که نوع جدید بر اساس آن ایجاد خواهد شد.
- ۶- علامت `>` را تایپ کنید.
- ۷- دنباله‌ها، گزینه‌ها یا ارجاع به گروههای نامگذاری شده‌ای را که ممکن است بخشی از نوع جدید باشند، اعلان نمایید.
- ۸- ویژگیهای نوع جدید را اعلان کنید یا ارجاع دهید.
- ۹- بر چسب پایان مناسب با شماره ۴ را تایپ کنید.
- ۱۰- `</xsd:complexContent>` را تایپ نمایید.
- ۱۱- با تایپ `</xsd:complexType>` تعریف نوع پیچیده تکمیل می‌گردد.

```
code.xsd
<xsd:complexType name="characteristicsType">
  <xsd:sequence>
    <xsd:element name="weight"
      type="xsd:string"/>
    <xsd:element name="length"
      type="xsd:string"/>
    <xsd:attribute name="kind"
      type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
```

شکل ۲۹-۷. تعریف characteristicsType به دنباله عناصر weight و length و ویژگی kind نیاز دارد.

```
code.xsd
<xsd:complexType name="birthType">
  <xsd:complexContent>
    <xsd:extension base="characteristicsType">
      <xsd:sequence>
        <xsd:element name="mother"
          type="xsd:string"/>
        <xsd:element name="birthdate"
          type="xsd:date"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

شکل ۳۰-۷. نوع جدید birthType فقط عناصر اضافه را نشان می‌دهد.

```
code.xsd
<xsd:element name="birth_characteristics"
  type="birthType"/>
```

شکل ۳۱-۷. همیشه باید عنصری را که از تعریف نوع پیچیده استفاده می‌کند اعلان نمود (صفحه ۱۰۶).

```
code.html
<birth_characteristics kind="normal">
  <weight>2-3 pounds</weight>
  <length>18-24 inches</length>
  <mother>Danai</mother>
  <birthdate>1999-06-10</birthdate>
</birth_characteristics>
```

شکل ۳۲-۷. عنصر birth_characteristics باید شامل اجزایی از characteristicsType (عناصر weight و length و ویژگی kind) و اجزای جدید (mother و birthdate) باشد.

اعلان عنصر نوع پیچیده

پس از تعریف یک نوع پیچیده، می‌توان آن را به عنصری که در سند XML به کار خواهد رفت نسبت داد.

برای اعلان عنصر نوع پیچیده :

- ۱- برای شروع اعلان عنصر، عبارت `<xsd:element` را تایپ کنید (شکل ۳۳-۷).
- ۲- `type="lable"` را تایپ کنید. label مشخصه‌ای است که برای تعریف نوع پیچیده به کار برده‌اید (مرحله ۲ صفحه ۹۴).
- ۳- علامتهای `</>` را برای تکمیل اعلان عنصر تایپ نمایید.

نکته‌ها

◀ عناصر را می‌توان به صورت *سراسری* (در سطوح اولیه یک الگو، پایین عنصر `xsd:schema`) یا *محلی*، در قسمتهای گوناگون اعلان نمود. این قسمتها عبارتند از: در یک تعریف نوع پیچیده (صفحه ۹۴)، یک دنباله (صفحه ۹۵)، مجموعه‌ای از گزینه‌ها (صفحه ۹۶)، یک گروه نامرتب (صفحه ۹۷) یا تعریف گروه نامگذاری شده (صفحه ۹۸).

◀ اگر عنصری را به صورت محلی اعلان نمایید (صفحه ۷۱) می‌توانید تعداد صفحه‌های ظهور عنصر را توسط ویژگیهای `maxOccurs`، `minOccurs` کنترل کنید (صفحه ۱۰۱).

```
code.xsd
<xsd:complexType name="characteristicsType">
  <xsd:sequence>
    <xsd:element name="weight"
      type="xsd:string"/>
    <xsd:element name="length"
      type="xsd:string"/>
    <xsd:attribute name="kind"
      type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
```

شکل ۳۳-۷. در این مثال تعریف نوع پیچیده‌ای که در شکل ۲۹-۷ وجود داشت مشاهده می‌شود.

```
code.xsd
<xsd:element name="characteristics"
  type="characteristicsType"/>
<xsd:complexType name="animalType"/>
  <xsd:sequence>
    <xsd:element name="name" type="nameType"
      minOccurs="2"/>
    <xsd:element ref="characteristics"
      minOccurs="1"/>
    ...
```

شکل ۳۴-۷. عنصر را می‌توان به صورت *سراسری* در بالای سند الگو اعلان کرد. بنابراین برای استفاده از این عنصر می‌توانید در هر قسمت از الگو به آن ارجاع دهید.

```
code.xsd
<xsd:complexType name="animalType"/>
  <xsd:sequence>
    <xsd:element name="name" type="nameType"
      minOccurs="2"/>
    <xsd:element name="characteristics"
      type="characteristicsType" minOccurs="1"/>
    ...
```

شکل ۳۵-۷. با اعلان محلی یک عنصر در اجزای دیگر (مانند تعریف یک نوع پیچیده یا گروه نامگذاری شده) به صورت خودکار به آن عنصر اشاره خواهد شد.

عناصری با نوعهای پیچیده بی نام

اگر به استفاده مجدد از نوع پیچیده‌ای نیاز ندارید بهتر است در قسمت اعلان یک عنصر، نوع پیچیده بدون نامی ایجاد نمایید.

برای اعلان عنصری با نوع پیچیده بی نام:

- ۱- عبارت `<xsd:element>` را تایپ کنید.
- ۲- `<xsd:element name="label">` را تایپ نمایید. منظور از label نام عنصری است که اعلان کرده‌اید و در سند XML به صورت `<label>` ظاهر خواهد شد.

۳- با تایپ `<xsd:complexType>` نوع پیچیده بی نام را اعلان کنید.

۴- برای تعیین عناصر تشکیل دهنده نوع پیچیده، یک دنباله (صفحه ۹۵)، گزینه (صفحه ۹۶)، گروه نامرتب (صفحه ۹۷) یا ارجاعی به یک گروه نامگذاری شده که می‌تواند شامل هر یک از موارد مذکور باشد (صفحه ۹۹) اعلان نمایید.

۵- سپس ویژگیها (صفحه ۱۵۸) یا گروه ویژگیهایی (صفحه ۱۱۱) را که در عناصر این نوع ظاهر خواهند شد، اعلان کنید یا ارجاع دهید.

۶- برای تکمیل تعریف نوع پیچیده بی نام عبارت `<xsd:complexType>` را تایپ کنید.

۷- با تایپ عبارت `</xsd:element>` اعلان عنصر نوع پیچیده را کامل نمایید.

نکته

◀ تنها تفاوت بین نوعهای پیچیده بی نام و نامگذاری شده این است که نوع پیچیده نامگذاری شده را می‌توان برای اعلان عناصر گوناگون دلخواه و بر اساس نوعهای پیچیده دیگر به دفعات مورد استفاده قرار داد، در حالی که نوع پیچیده بی نام فقط عنصر داخل خود را تعریف می‌کند.

```
code.xsd
<xsd:element name="characteristics">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="weight"
        type="xsd:string"/>
      <xsd:element name="length"
        type="xsd:string"/>
      <xsd:attribute name="kind"
        type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

شکل ۳۶-۷. این بار نوع characteristics به صورت نوع بی نام تعریف شده است. حالا فقط می‌توان تعریف این نوع پیچیده را برای عنصری که شامل characteristics است به کار برد. عنصر نیز به دو صورت سراسری و محلی قابل تعریف می‌باشد.

```
code.xml
<characteristics kind="physical">
  <weight>500 pounds</weight>
  <length>3 yards from nose to tail</length>
</characteristics>
```

شکل ۳۷-۷. این یک مثال معتبر است. نوع پیچیده نامگذاری شده و بی نام، یکسان به نظر می‌رسند.

اعلان ویژگیها

```
code.xsd
<xsd:element name="source">
  <xsd:complexType>
  <xsd:complexContent>
  <xsd:extension base="xsd:anyType">
  <xsd:attribute name="sectionid"
    type="xsd:string"/>
  <xsd:attribute name="newspaperid"
    type="xsd:string"/>
  </xsd:extension>
  </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

شکل ۳۸-۷. اعلان یک ویژگی شبیه اعلان عنصر نوع ساده است.

```
code.xml
<source sectionid="101" newspaperid="21"/>
```

شکل ۳۹-۷. ویژگیها همیشه در برچسب شروع محتویات عنصر ظاهر می گردند.

با توجه به اینکه یک ویژگی از عناصر و ویژگیهای دیگر تشکیل نمی گردد، همیشه یک نوع ساده محسوب می شود. یک ویژگی همیشه در عنصر یک نوع پیچیده ظاهر می شود. در این فصل درباره ایجاد ویژگیها بیش از پیش صحبت خواهیم کرد.

برای اعلان یک ویژگی :

- ۱- عبارت `<xsd:attribute>` را در قسمت تعریف نوع پیچیده تایپ کنید.
- ۲- سپس عبارت `name="label"` را تایپ نمایید. `label` نامی است که برای ویژگی سند XML به کار خواهید برد.
- ۳- سپس `type="simple"` را تایپ کنید. منظور از `simple` نوع ساده ای است که ویژگی به آن تعلق دارد. برای به دست آوردن اطلاعات بیشتر درباره نوعهای ساده به صفحه ۷۶ مراجعه نمایید.
- همچنین می توانید عبارت `ref="label"` را تایپ کنید. `label` مشخصه تعریف یک ویژگی است که در مراحل ۱ و ۲ به صورت سراسری اعلان نموده اید.
- ۴- برای تکمیل برچسب شروع علامت `>` را تایپ نمایید.
- ۵- هرگونه محدودیت و طریقی را که می خواهید اضافه کنید (صفحه های ۸۳ تا ۹۰).
- ۶- عبارت `</xsd:attribute>` را تایپ کنید.

نکته ها

- ◀ اگر قصد اضافه کردن طرح و محدودیتی را ندارید به جای انجام مراحل ۴ تا ۶، علامتهای `>/` را تایپ کنید.
- ◀ برای ویژگیها، نوعهای ساده درونی بیشتری وجود دارد که در آدرس <http://www.w3.org/TR/xmlschema-2> قابل دسترسی هستند.
- ◀ امکان تعریف یک ویژگی نوع ساده بی نام نیز وجود دارد.
- ◀ محل اعلان یک ویژگی در نوع پیچیده ای که به آن تعلق دارد پس از اعلان تمام اجزای نوع پیچیده می باشد.

تعیین لزوم وجود ویژگی

تعریف ویژگی اختیاری است. یعنی بودن یا نبودن ویژگی هیچ مشکلی در سند XML به وجود نمی‌آورد. از طرفی می‌توانید لزوم ظهور یا عدم ظهور ویژگی تعریف شده‌ای را تعیین کنید تا هنگام بررسی اعتبار سند XML در نظر گرفته شود.

برای تعیین لزوم ظهور یک ویژگی:

- ۱- هنگام تعریف یک ویژگی عبارت `use="required"` را تایپ کنید.
- ۲- سپس `required` را تایپ نمایید. در نتیجه برای معتبر بودن سند، ظهور ویژگی الزامی می‌گردد.
- ۳- با تایپ عبارت `value="must"` می‌توانید تنها مقدار قابل قبول ویژگی (`must`) را تعیین نمایید.

برای تعیین عدم لزوم ظهور یک ویژگی:

هنگام تعریف ویژگی عبارت `use="prohibited"` را تایپ کنید تا شرط اعتبار سند عدم ظهور ویژگی گردد.

نکته

تایپ عبارت `use="optional"` باعث می‌شود تا ظهور ویژگی اختیاری باشد. از آنجایی که این مقدار پیش فرض است تایپ آن ضرورتی ندارد و تنها باعث شلوغی کار می‌گردد.

```
code.xsd
<xsd:element name="source">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="xsd:anyType">
        <xsd:attribute name="sectionid"
          type="xsd:string" use="required"/>
        <xsd:attribute name="newspaperid"
          type="xsd:string"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

شکل ۷-۴۰. چون ظهور ویژگی اختیاری است الزام ظهور آن باید تعیین شود.

```
code.xml
<source sectionid="101"/>
```

```
code.xml
<source sectionid="101" newspaperid="21"/>
```

شکل ۷-۴۱. هر دو عنصر `source` معتبر هستند؛ زیرا تنها ظهور ویژگی `sectionid` اجباری است.

تعریف دوباره محتویات ویژگی

برای تعریف دوباره محتویات یک ویژگی، به دو روش می‌توان از یک الگو استفاده کرد. یک راه، نوشتن محتویات ویژگی هنگام ظاهر شدن در سند XML است. راه دیگر، در نظر گرفتن مقدار اولیه برای ویژگی است و این مقدار به محل ظهور ویژگی در سند بستگی ندارد. به مقدار روش اول مقدار ثابت و در روش دوم مقدار پیش فرض می‌گویند.

برای تعیین مقدار ثابت ویژگی:

- ۱- در برچسب ویژگی عبارت `use="fixed"` را تایپ کنید.
- ۲- سپس عبارت `value="content"` را تایپ نمایید. منظور از `content` مقدار ویژگی است که هنگام ظهور در سند معتبر XML خواهد داشت.

برای تنظیم مقدار اولیه یک ویژگی:

- ۱- در برچسب شروع ویژگی عبارت `use="default"` را تایپ نمایید.
- ۲- سپس عبارت `value="content"` را تایپ کنید. `content` مقدار ویژگی است که در صورت ظاهر شدن ویژگی در سند، باز هم برای آن در نظر گرفته خواهد شد.

نکته‌ها

- ◀ مقادیر ثابت ویژگیها تنها در صورتی برای آنها در نظر گرفته می‌شوند که ویژگی مربوطه در سند ظاهر گردد. در غیر این صورت مقداری برای آنها وجود نخواهد داشت.
- ◀ مقادیر پیش فرض در صورت عدم ظهور ویژگی در سند باز هم برای آنها در نظر گرفته می‌شوند.
- ◀ اگر یک ویژگی مقدار پیش فرض داشته باشد و در سند XML ظاهر گردد امکان تعیین مقادیر ثابت برای آن وجود دارد و به مقدار پیش فرض خود محدود نمی‌گردد.

```
code.xml
<xsd:attribute name="sectionid" type="xsd:string"
  use="fixed" value="101"/>
<xsd:attribute name="newspaperid"
  type="xsd:string"/>
```

شکل ۴۲-۷. مقدار ویژگی `sectionid` هنگام ظاهر شدن در سند XML باید مقدار ۱۰۱ داشته باشد. این ویژگی می‌تواند حذف گردد. این مثال از تعریف نوع پیچیده شکل ۴۰-۷ برگزیده شده است.

```
code.xml
<source sectionid="101" newspaperid="21"/>
```

شکل ۴۳-۷. عنصر `source` در مقایسه با الگوی داده شکل ۴۲-۷ معتبر است. در صورت حذف ویژگی `sectionid` نیز مشکلی پیش نخواهد آمد.

```
code.xml
<source sectionid="456" newspaperid="21"/>
```

شکل ۴۴-۷. عنصر `source` این مثال با توجه به اعلان شکل ۴۲-۷ معتبر نمی‌باشد.

```
code.xml
<xsd:attribute name="sectionid" type="xsd:string"
  use="fixed" value="101"/>
<xsd:attribute name="newspaperid" type="
  xsd:string" use="default" value="21"/>
```

شکل ۴۵-۷. اگر ویژگی `newspaperid` با مقدار پیش فرض تعریف شود این مقدار بدون توجه به ظهور یا عدم ظهور ویژگی برای آن در نظر گرفته خواهد شد. از تعریف نوع پیچیده شکل ۴۰-۷ اقتباس شده است.

```
code.xml
<source sectionid="101" newspaperid="21"/>
```

```
code.xml
<source sectionid="101" newspaperid="25"/>
```

شکل ۴۶-۷. عناصر `source` هر دو مثال معتبر هستند. مقدار پیش فرض تنها به عنوان مقدار اولیه در نظر گرفته می‌شود و مقادیر دیگر نیز قابل قبول خواهند بود.

تعریف گروه ویژگی

اگر برای چند عنصر به مجموعه‌ای از ویژگیها نیاز دارید، بهتر است یک گروه ویژگی تعریف کنید. سپس از داخل نوع پیچیده هر عنصر به آن گروه ویژگی ارجاع دهید.

برای تعریف گروه ویژگی:

- ۱- عبارت `<xsd:attributeGroupname="label">` را تایپ کنید. label مشخصه گروه ویژگی است.
- ۲- هر یک از ویژگیهای گروه را اعلان کنید یا به هر یک از آنها ارجاع دهید.
- ۳- در آخر به منظور تکمیل تعریف گروه ویژگی، عبارت `</xsd:attributeGroup>` را تایپ نمایید.

نکته‌ها

- ◀ یک گروه ویژگی باید به صورت سراسری اعلان گردد. یعنی در ابتدای الگو و پایین عنصر `xsd:schema`.
- ◀ پس از اعلان یک گروه ویژگی باید مانند توضیحات صفحه ۱۱۲ به آن ارجاع دهید.
- ◀ فقط می‌توان به ویژگیهای سراسری ارجاع نمود. یعنی آنهایی که در ابتدای الگو اعلان شده‌اند (صفحه ۷۱).
- ◀ اگر ویژگیها را داخل گروه ویژگی به صورت محلی اعلان کنید تنها برای گروه ویژگی مربوطه قابل دسترس خواهند بود. گاهی این حد از دسترسی کافی است.
- ◀ یک گروه ویژگی می‌تواند به گروههای ویژگی دیگر نیز ارجاع داده شود.

```
code.xsd
<xsd:attributeGroup name="imageAIts">
  <xsd:attribute name="filename" type="xsd:uri-reference"/>
  <xsd:attribute name="x" type="xsd:integer"/>
  <xsd:attribute name="y" type="xsd:integer"/>
</xsd:attributeGroup>
```

شکل ۴۷-۷. با نامگذاری یک گروه ویژگی، استفاده از ویژگیهای گروه در تعاریف نوعهای گوناگون آسانتر خواهد شد.

ارجاع به گروههای ویژگی

پس از تعریف یک گروه ویژگی می‌توان از هر محلی به ویژگیهای آن گروه اشاره نمود.

برای ارجاع یک گروه ویژگی :

۱- در تعریف یک نوع پیچیده، پس از اعلان تمام عناصر تشکیل دهنده آن، عبارت `<xsd:attributeGroup` را تایپ کنید.

۲- سپس عبارت `<ref="label"/>` را تایپ نمایید. label نام گروه ویژگی مورد نظر است که در مرحله ۱ صفحه ۱۱۱ به کار بردید.

نکته‌ها

◀ گروه ویژگیها مانند پارامتر موجودیتها در DTDها هستند (صفحه ۶۰) ولی البته برای ارائه ویژگیها محدودیتهایی دارند.

◀ ویژگیها (و گروه ویژگیها) همیشه در انتهای اجزا و پس از اعلان تمام عناصر اعلان می‌گردند.

```
code.xsd
<xsd:element name="picture">
  <xsd:complexType>
  <xsd:complexContent>
  <xsd:extension base="xsd:anyType">
  <xsd:attributeGroup ref="imageAtts" />
  </xsd:extension>
  </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="video">
  <xsd:complexType>
  <xsd:complexContent>
  <xsd:extension base="xsd:anyType">
  <xsd:element name="description"
    type="xsd:string" />
  <xsd:element name="running_time"
    type="xsd:timeDuration" />
  <xsd:attributeGroup ref="imageAtts" />
  <xsd:attribute name="format"
    type="xsd:string" />
  </xsd:extension>
  </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

شکل ۴۸-۷. تایپ `ref="imageAtts"` از تعریف سه ویژگی جداگانه بسیار آسان‌تر و سریعتر می‌باشد. عنصر Video یک ویژگی اضافی به نام `format` دارد که پس از گروه ویژگی اعلان شده است.

```
code.xml
<picture filename="tiger.jpg" x="200" y="197" />

<video filename="tiger.ram" x="100" y="100"
  format="realplayer">
  <description>A tiger moves gracefully through the
  dappled light of the forest.</description>
  <running_time>PT3M43S</running_time>
</video>
```

شکل ۴۹-۷. عناصر `video` و `picture` ویژگیهای `filename`، `x` و `y` دارند ولی هر یک از آنها یک بار تعریف شده‌اند.