

XML Schema

(12주차)

학습개요

- 학습 목표

- XML Schema의 컴플렉스 타입을 설명할 수 있다.
- XML Schema의 컴플렉스 타입에 대한 확장과 제한 활용법을 설명할 수 있다.
- XML Schema의 모델 그룹과 애트리뷰트 그룹의 활용 방법을 설명할 수 있다.
- XML Schema의 유일 제약 정의 구현을 설명할 수 있다.
- XML Schema 문서를 작성하는 방법을 설명할 수 있다.

- 학습 내용

- 컴플렉스 타입
- 컴플렉스 타입 확장과 제한
- 모델 그룹과 애트리뷰트 그룹
- 유일 제약 정의
- XML Schema 실습

컴플렉스 타입

- 자식 엘리먼트 혹은 애트리뷰트를 갖는 타입
- 글로벌 컴플렉스 타입

```
<xs:complexType name="typeName">  
  ....  
</xs:complexType>  
  
<xs:element name="elementName" type="typeName"/>
```

- 로컬 컴플렉스 타입

```
<xs:element name="elementName">  
  <xs:complexType>  
    ....  
  </xs:complexType>  
</xs:element>
```

순차적으로 자식 엘리먼트를 갖는 컴플렉스 타입

- `<xs:sequence>`를 사용하여 하나 이상의 자식 엘리먼트를 순차적으로 갖는 컴플렉스 타입

```
<xs:complexType name="typeName">  
  <xs:sequence>  
    element declarations  
  </xs:sequence>  
</xs:complexType>
```

순차적으로 자식 엘리먼트를 갖는 컴플렉스 타입

[books1.xsd]

```
...  
<xs:element name="books">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element name="book" type="book" maxOccurs="unbounded"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>  
...
```

```
<xs:complexType name="book">  
  <xs:sequence>  
    <xs:element name="title" type="xs:string"/>  
    <xs:element name="author" type="xs:string"/>  
    <xs:element name="publisher" type="xs:string"/>  
    <xs:element name="price" type="price"/>  
    ...  
  </xs:sequence>  
</xs:complexType>  
...
```

[books1.xml]

```
<?xml version="1.0" encoding="UTF-8"?>  
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  xsi:noNamespaceSchemaLocation="books1.xsd">  
  <book>  
    <title>Beginning XML, 5th Edition</title>  
    <author>Joe Fawcett</author>  
    <publisher>Wrox</publisher>  
    <price>20000</price>  
    <published>2012-07-10</published>  
    <isbn>978-1-118-16213-2</isbn>  
    <pages>864</pages>  
  </book>  
</books>
```

선택적으로 자식 엘리먼트를 갖는 컴플렉스 타입

- <xs:complexType> 혹은 <xs:sequence>에서 <xs:choice>를 사용하여 하나 이상의 자식 엘리먼트를 선택적으로 갖는 컴플렉스 타입

```
<xs:complexType name="typeName">  
  <xs:choice minOccurs="1" maxOccurs="unbounded">  
    element declarations  
  </xs:choice>  
</xs:complexType>
```

```
<xs:complexType name="typeName">  
  <xs:sequence>  
    element declarations  
    <xs:choice minOccurs="1" maxOccurs="unbounded">  
      element declarations  
    </xs:choice>  
    element declarations  
  </xs:sequence>  
</xs:complexType>
```

선택적으로 자식 엘리먼트를 갖는 컴플렉스 타입

[books2.xsd]

```
...
<xs:element name="books">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="book" type="book" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
...

<xs:complexType name="book">
  <xs:sequence>
    <xs:element name="title" type="xs:string"/>
    <xs:choice maxOccurs="unbounded">
      <xs:element name="author" type="xs:string"/>
      <xs:element name="translator" type="xs:string"/>
    </xs:choice>
    <xs:element name="publisher" type="xs:string"/>
    ...
  </xs:sequence>
</xs:complexType>
...
```

[books2.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books2.xsd">
  <book>
    <title>Beginning XML, 5th Edition</title>
    <author>Joe Fawcett</author>
    <author>Liam R.E. Quin</author>
    <author>Danny Ayers</author>
    <translator>홍길동</translator>
    <publisher>Wrox</publisher>
    <price>20000</price>
    <published>2012-07-10</published>
    <isbn>978-1-118-16213-2</isbn>
    <pages>864</pages>
  </book>
</books>
```

자식 엘리먼트와 애트리뷰트를 갖는 컴플렉스 타입

- <xs:sequence> 혹은 <xs:choice> 다음에 애트리뷰트를 갖는 컴플렉스 타입

```
<xs:complexType name="typeName">  
  <xs:sequence>  
    element declarations  
  </xs:sequence>  
  attribute declarations  
</xs:complexType>
```


자식 엘리먼트와 애트리뷰트를 갖는 컴플렉스 타입

[books3.xsd]

```
...
<xs:element name="books">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="book" type="book" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
...

<xs:complexType name="book">
  <xs:sequence>
    <xs:element name="title" type="xs:string"/>
    ...
  </xs:sequence>
  <xs:attribute name="id" type="xs:ID" use="required"/>
  <xs:attribute name="category" type="xs:string" use="required"/>
</xs:complexType>
...
```

[books3.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books3.xsd">
  <book id="b01" category="information technology & software development">
    <title>Beginning XML, 5th Edition</title>
    <author>Joe Fawcett</author>
    <author>Liam R.E. Quin</author>
    <author>Danny Ayers</author>
    ...
  </book>
  <book id="b02" category="information technology & software development">
    <title>Beginning XSLT and XPath: Transforming XML Documents and Data</title>
    <author>Ian Williams</author>
    <translator>이순신</translator>
    ...
  </book>
</books>
```

애트리뷰트만 갖는 컴플렉스 타입

- 컴플렉스 타입을 이용해 애트리뷰트만 있고 콘텐츠를 갖지 않는 빈 엘리먼트를 기술

```
<xs:complexType name="typeName">  
  attribute declarations  
</xs:complexType>
```

애트리뷰트만 갖는 컴플렉스 타입

[books4.xsd]

```
...
<xs:element name="books">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="book" type="book" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
...
<xs:complexType name="book">
  <xs:sequence>
    <xs:element name="title" type="xs:string"/>
    ...
    <xs:element name="image" type="image" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:ID" use="required"/>
  <xs:attribute name="category" type="xs:string" use="required"/>
</xs:complexType>
```

```
<xs:complexType name="image">
  <xs:attribute name="src" type="xs:anyURI" use="required"/>
</xs:complexType>
```

[books4.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books4.xsd">
  <book id="b01" category="information technology & software development">
    <title>Beginning XML, 5th Edition</title>
    <author>Joe Fawcett</author>
    <author>Liam R.E. Quin</author>
    <author>Danny Ayers</author>
    ...
    <image src="978-1-118-16213-2.jpg"/>
  </book>
  <book id="b02" category="information technology & software development">
    <title>Beginning XSLT and XPath: Transforming XML Documents and Data</title>
    <author>Ian Williams</author>
    <translator>이순신</translator>
    ...
  </book>
</books>
```

애트리뷰트와 콘텐츠를 갖는 컴플렉스 타입

- <xs:simpleContent>와 base 애트리뷰트의 심플 타입 확장을 이용해 애트리뷰트와 콘텐츠를 갖는 엘리먼트를 정의

```
<xs:complexType name="typeName">
  <xs:simpleContent>
    <xs:extension base="simpleType">
      attribute declarations
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

애틀리뷰트와 컨텐트를 갖는 컴플렉스 타입

[books5.xsd]

```
...
<xs:complexType name="book">
  <xs:sequence>
    <xs:element name="title" type="xs:string"/>
    ...
    <xs:element name="price" type="price"/>
    ...
  </xs:sequence>
  <xs:attribute name="id" type="xs:ID" use="required"/>
  <xs:attribute name="category" type="xs:string" use="required"/>
</xs:complexType>

<xs:complexType name="price">
  <xs:simpleContent>
    <xs:extension base="xs:integer">
      <xs:attribute name="currency" type="xs:string"
        use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
...
```

[books5.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="books5.xsd">
  <book id="b01" category="information technology & software development">
    <title>Beginning XML, 5th Edition</title>
    ...
    <price currency="원">35000</price>
    ...
  </book>
  <book id="b02" category="information technology & software development">
    <title>Beginning XSLT and XPath: Transforming XML Documents and Data</title>
    ...
    <price currency="원">27000</price>
    ...
  </book>
</books>
```

컴플렉스 타입의 확장과 제한

- 컴플렉스 타입의 확장:
컴플렉스 타입에 엘리먼트 혹은 애트리뷰트를 추가하는 것
- 컴플렉스 타입의 제한:
컴플렉스 타입의 엘리먼트를 일부 제거하거나 컨텐츠값 혹은 애트리뷰트값의 범위를 제한하는 것

<xs:simpleContent>를 갖는 컴플렉스 타입의 확장

- 컴플렉스 타입에 대해 확장을 정의한 <xs:simpleContent>를 갖는 컴플렉스 타입을 정의하는 것

```
<xs:complexType name="typeName">  
  <xs:simpleContent>  
    <xs:extension base="complexType">  
      attribute declarations  
    </xs:extension>  
  </xs:simpleContent>  
</xs:complexType>
```

- base 애트리뷰트의 값으로 <xs:simpleContent>를 갖는 컴플렉스 타입만 사용할 수 있음

<xs:simpleContent>를 갖는 컴플렉스 타입의 확장

[books6.xsd]

```
...
<xs:complexType name="book">
  <xs:sequence>
    <xs:element name="title" type="xs:string"/>
    ...
    <xs:element name="price" type="price"/>
    <xs:element name="sellingprice" type="sellingprice"/>
    ...
  </xs:sequence>
</xs:complexType>

...
<xs:complexType name="price">
  <xs:simpleContent>
    <xs:extension base="xs:integer">
      <xs:attribute name="currency" type="xs:string" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="sellingprice">
  <xs:simpleContent>
    <xs:extension base="price">
      <xs:attribute name="save" type="xs:decimal" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
...
```

[books6.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books6.xsd">
  <book id="b01" category="information technology & software development">
    <title>Beginning XML, 5th Edition</title>
    ...
    <price currency="원">35000</price>
    <sellingprice currency="원" save="0.3">24500</sellingprice>
  </book>
  <book id="b02" category="information technology & software development">
    <title>Beginning XSLT and XPath: Transforming XML Documents and Data</title>
    ...
    <price currency="원">27000</price>
    <sellingprice currency="원" save="0.2">21600</sellingprice>
  </book>
</books>
```


<xs:simpleContent>를 갖는 컴플렉스 타입의 제한

- 컴플렉스 타입에 대해 제한을 정의한 <xs:simpleContent>를 갖는 컴플렉스 타입을 정의하는 것

```
<xs:complexType name="typeName">  
  <xs:simpleContent>  
    <xs:restriction base="complexType">  
      facet elements  
    </xs:restriction>  
  </xs:simpleContent>  
</xs:complexType>
```

- base 애트리뷰트의 값으로 <xs:simpleContent>를 갖는 컴플렉스 타입만 사용할 수 있음

<xs:simpleContent>를 갖는 컴플렉스 타입의 제한

| facet | description |
|----------------|-------------------------|
| minExclusive | 최소값(지정값 불포함) |
| maxExclusive | 최대값(지정값 불포함) |
| minInclusive | 최소값(지정값 포함) |
| maxInclusive | 최대값(지정값 포함) |
| totalDigits | 유효 숫자의 갯수 |
| fractionDigits | 소수부 숫자의 갯수 |
| length | 문자열 길이, 리스트 항목 갯수 |
| minLength | 문자열 최소 길이, 리스트 항목 최소 갯수 |
| maxLength | 문자열 최대 길이, 리스트 항목 최대 갯수 |
| enumeration | 이뉴머레이션 타입의 값 |
| pattern | 정규표현식 패턴 |

<xs:simpleContent>를 갖는 컴플렉스 타입의 제한

[books7.xsd]

```
...
<xs:complexType name="book">
  <xs:sequence>
    ...
    <xs:element name="price" type="price"/>
    <xs:element name="sellingprice" type="sellingprice2"/>
    ...
  </xs:sequence>
</xs:complexType>
...
<xs:complexType name="sellingprice">
  <xs:simpleContent>
    <xs:extension base="price">
      <xs:attribute name="save" type="xs:decimal" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="sellingprice2">
  <xs:simpleContent>
    <xs:restriction base="sellingprice">
      <xs:minInclusive value="1000"/>
      <xs:maxInclusive value="1000000"/>
    </xs:restriction>
  </xs:simpleContent>
</xs:complexType>
...
```

[books7.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books7.xsd">
  <book id="b01" category="information technology & software development">
    <title>Beginning XML, 5th Edition</title>
    ...
    <price currency="원">35000</price>
    <sellingprice currency="원" save="0.3">24500</sellingprice>
    ...
  </book>
  <book id="b02" category="information technology & software development">
    <title>Beginning XSLT and XPath: Transforming XML Documents and Data</title>
    ...
    <price currency="원">27000</price>
    <sellingprice currency="원" save="0.2">21600</sellingprice>
    ...
  </book>
</books>
```

<xs:complexContent>를 갖는 컴플렉스 타입의 확장

- 기존의 컴플렉스 타입에 엘리먼트와 애트리뷰트를 추가

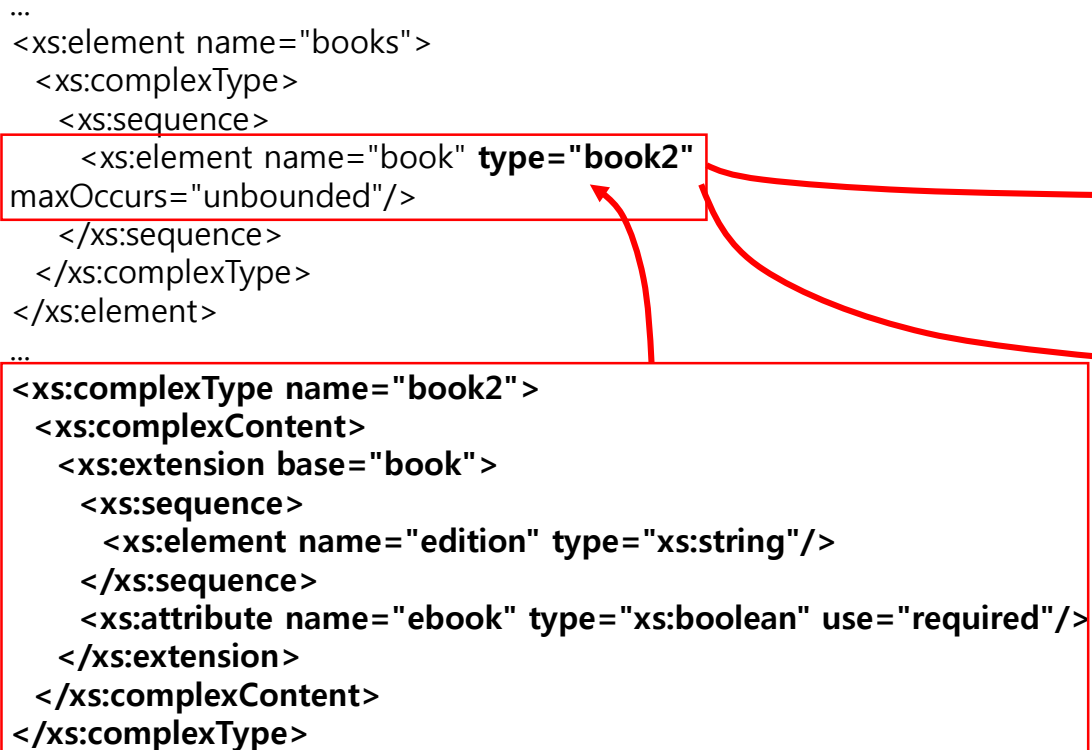
```
<xs:complexType name="typeName">  
  <xs:complexContent>  
    <xs:extension base="complexType">  
      <xs:sequence>  
        element declarations  
      </xs:sequence>  
      attribute declarations  
    </xs:extension>  
  </xs:complexContent>  
</xs:complexType>
```

- base 애트리뷰트의 값으로 자식 엘리먼트와 속성을 갖는 컴플렉스 타입만 사용할 수 있음

<xs:complexContent>를 갖는 컴플렉스 타입의 확장

[books8.xsd]

```
...
<xs:element name="books">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="book" type="book2"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
...
<xs:complexType name="book2">
  <xs:complexContent>
    <xs:extension base="book">
      <xs:sequence>
        <xs:element name="edition" type="xs:string"/>
      </xs:sequence>
      <xs:attribute name="ebook" type="xs:boolean" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
...
```



[books8.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books8.xsd">
  <book id="b01" category="information technology & software development"
ebook="false">
    <title>Beginning XML, 5th Edition</title>
    <author>Joe Fawcett</author>
    ...
    <image src="978-1-118-16213-2.jpg"/>
    <edition>5th</edition>
  </book>
  <book id="b02" category="information technology & software development"
ebook="false">
    <title>Beginning XSLT and XPath: Transforming XML Documents and Data</title>
    <author>Ian Williams</author>
    ...
    <isbn>978-0-470-47725-0</isbn>
    <pages>431</pages>
    <edition>1st</edition>
  </book>
</books>
```

<xs:complexContent>를 갖는 컴플렉스 타입의 제한

- 기존 컴플렉스 타입에 대해 자식 엘리먼트의 콘텐츠값이나 애트리뷰트값의 범위 제한을 추가

```
<xs:complexType name="typeName">
  <xs:complexContent>
    <xs:restriction base="complexType">
      <xs:sequence>
        element redeclarations
      </xs:sequence>
      attribute redeclarations
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>
```

- base 애트리뷰트의 값으로 자식 엘리먼트와 속성을 갖는 컴플렉스 타입만 사용할 수 있음
- 기존의 엘리먼트와 애트리뷰트만을 사용해야 하며, 적은 범위의 제한은 가능하나 더 큰 범위의 확장은 안됨

<xs:complexContent>를 갖는 컴플렉스 타입의 제한

[books9.xsd]

```
...
<xs:element name="books">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="book" type="book3"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
...
<xs:complexType name="book3">
  <xs:complexContent>
    <xs:restriction base="book2">
      <xs:sequence>
        ...
        <xs:element name="publisher">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="O'Reilly Media, Inc"/>
              <xs:enumeration value="Wrox"/>
              <xs:enumeration value="Addison-Wesley"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
```

```
<xs:element name="price" type="price"/>
<xs:element name="sellingprice" type="sellingprice2"/>
<xs:element name="published" type="xs:date"/>
<xs:element name="isbn" type="isbn"/>
<xs:element name="pages" type="pages"/>
<xs:element name="image" type="image" minOccurs="0"/>
<xs:element name="edition" type="xs:string"/>
</xs:sequence>
```

```
<xs:attribute name="id" type="xs:ID" use="required"/>
```

```
<xs:attribute name="category" use="required">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="programming"/>
      <xs:enumeration value="database"/>
      <xs:enumeration value="security"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
```

```
<xs:attribute name="ebook" type="xs:boolean" use="required"/>
</xs:restriction>
</xs:complexContent>
</xs:complexType>
```

...

<xs:complexContent>를 갖는 컴플렉스 타입의 제한

[books9.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books9.xsd">
  <book id="b01" category="programming" ebook="false">
    <title>Beginning XML, 5th Edition</title>
    <author>Joe Fawcett</author>
    <author>Liam R.E. Quin</author>
    <author>Danny Ayers</author>
    <translator>홍길동</translator>
    <publisher>Wrox</publisher>
    <price currency="원">35000</price>
    <sellingprice currency="원" save="0.3">24500</sellingprice>
    <published>2012-07-10</published>
    <isbn>978-1-118-16213-2</isbn>
    <pages>864</pages>
    <image src="978-1-118-16213-2.jpg"/>
    <edition>5th</edition>
  </book>
```

```
<book id="b02" category="programming" ebook="false">
  <title>Beginning XSLT and XPath: Transforming XML Documents and Data</title>
  <author>Ian Williams</author>
  <translator>이순신</translator>
  <publisher>Wrox</publisher>
  <price currency="원">27000</price>
  <sellingprice currency="원" save="0.2">21600</sellingprice>
  <published>2009-09-08</published>
  <isbn>978-0-470-47725-0</isbn>
  <pages>431</pages>
  <edition>1st</edition>
</book>
</books>
```


컴플렉스 타입과 다형성

- 다형성(polymorphism): 하나의 타입이 여러 타입에 속하는 성질
- 엘리먼트 선언 시 type 값으로 특정 종류의 컴플렉스 타입을 지정하면 이를 확장하거나 제한한 컴플렉스 타입도 type 값으로 사용할 수 있음

```
<element xsi:type="complexType"/>
```

- xsi: 는 스키마 인스턴스 네임스페이스의 접두사임
- 스키마 인스턴스 애트리뷰티인 type의 값으로 컴플렉스 타입을 지정
-

컴플렉스 타입과 다형성

[books10.xsd]

```
...
<xs:element name="books">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="book" type="book"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
...
```

[books10.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books10.xsd">
  <book id="b01" category="programming">
    <title>Beginning XML, 5th Edition</title>
    <author>Joe Fawcett</author>
    <author>Liam R.E. Quin</author>
    <author>Danny Ayers</author>
    <translator>홍길동</translator>
    <publisher>Wrox</publisher>
    <price currency="원">35000</price>
    <sellingprice currency="원" save="0.3">24500</sellingprice>
    <published>2012-07-10</published>
    <isbn>978-1-118-16213-2</isbn>
```

```

    <pages>864</pages>
    <image src="978-1-118-16213-2.jpg"/>
  </book>
  <book id="b02" category="programming" ebook="false" xsi:type="book2">
    <title>Beginning XSLT and XPath: Transforming XML Documents and Data</title>
    <author>Ian Williams</author>
    <translator>이순신</translator>
    <publisher>Wrox</publisher>
    <price currency="원">27000</price>
    <sellingprice currency="원" save="0.2">21600</sellingprice>
    <published>2009-09-08</published>
    <isbn>978-0-470-47725-0</isbn>
    <pages>431</pages>
    <edition>1st</edition>
  </book>
  <book id="b03" category="database" ebook="false" xsi:type="book3">
    <title>MongoDB: The Definitive Guide, 2nd Edition</title>
    <author>Kristina Chodorow</author>
    <publisher>O'Reilly Media, Inc.</publisher>
    <price currency="원">45000</price>
    <sellingprice currency="원" save="0.2">36000</sellingprice>
    <published>2013-05-16</published>
    <isbn>978-1-449-34468-9</isbn>
    <pages>432</pages>
    <edition>2nd</edition>
  </book>
</books>
```

모델 그룹과 애트리뷰트 그룹

- 모델 그룹:

중복되는 엘리먼트 선언을 모아 모델 그룹으로 만들어 참조함으로써 스키마 문서의 복잡성을 줄여주고 가독성을 높임

- 애트리뷰트 그룹:

중복되는 애트리뷰트 선언을 모아 애트리뷰트 그룹으로 만들어 참조함으로써 스키마 문서의 복잡성을 줄여주고 가독성을 높임

모델 그룹

- 순서와 반복 횟수가 지정된 엘리먼트들의 그룹

```
<xs:group name="groupName">  
  ( all | sequence | choice )  
</xs:group>  
  
<xs:group ref="groupName"/>
```

- <xs:all>: 엘리먼트 선언의 순서에 상관없이 한번씩 사용 (부분적으로 적용 불가능하며 반드시 <xs:complexType>, <xs:restriction>, <xs:extension>에 포함되어야함)
- <xs:sequence>: 엘리먼트 선언의 순서에 맞게 한번씩 사용
- <xs:choice>: 선언된 엘리먼트 중 한 개만 선택해 사용

모델 그룹

[books11.xsd]

```
<xs:complexType name="book">
  <xs:sequence>
    <xs:element name="title" type="xs:string"/>
    <xs:choice maxOccurs="unbounded">
      <xs:element name="author" type="profile"/>
      <xs:element name="translator" type="profile"/>
    </xs:choice>
    ...
  </xs:sequence>
  ...
</xs:complexType>
...
<xs:complexType name="profile">
  <xs:sequence>
    <xs:element name="name" type="xs:string"/>
    <xs:group ref="contact"/>
  </xs:sequence>
</xs:complexType>
<xs:group name="contact">
  <xs:sequence>
    <xs:element name="email" type="xs:string" minOccurs="0"/>
    <xs:element name="facebook" type="xs:anyURI" minOccurs="0"/>
    <xs:element name="tweeter" type="xs:anyURI" minOccurs="0"/>
  </xs:sequence>
</xs:group>
...
```

[books11.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books11.xsd">
  <book id="b01" category="programming">
    <title>Beginning XML, 5th Edition</title>
    <author>
      <name>Joe Fawcett</name>
    </author>
    ...
    <translator>
      <name>홍길동</name>
      <email>gildonghong@example.com</email>
      <facebook>https://www.facebook.com/gildonghong</facebook>
      <twitter>https://twitter.com/gildonghong</twitter>
    </translator>
    <publisher>Wrox</publisher>
    <price currency="원">35000</price>
    <sellingprice currency="원" save="0.3">24500</sellingprice>
    <published>2012-07-10</published>
    <isbn>978-1-118-16213-2</isbn>
    <pages>864</pages>
    <image src="978-1-118-16213-2.jpg"/>
  </book>
  ...
</books>
```

애트리뷰트 그룹

- 애트리뷰트 선언들의 모임

```
<xs:attributeGroup name="groupName">  
  attrinute declarations  
</xs:attrinuteGroup>  
  
<xs:attributeGroup ref="groupName"/>
```

애트리뷰트 그룹

[books12.xsd]

```
<xs:complexType name="book">
  ...
  <xs:element name="publisher" type="publisher"/>
  ...
</xs:complexType>
...
<xs:complexType name="publisher">
  <xs:sequence>
    <xs:element name="name">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="O'Reilly Media, Inc."/>
          <xs:enumeration value="Wrox"/>
          ...
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:group ref="contact"/>
  </xs:sequence>
  <xs:attributeGroup ref="phone"/>
</xs:complexType>
<xs:attributeGroup name="phone">
  <xs:attribute name="tel" type="xs:string" use="required"/>
  <xs:attribute name="fax" type="xs:string" use="optional"/>
</xs:attributeGroup>
...
```

[books12.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books12.xsd">
  <book id="b01" category="programming">
    <title>Beginning XML, 5th Edition</title>
    <author>
      <name>Joe Fawcett</name>
    </author>
    ...
    <translator>
      <name>홍길동</name>
      <email>gildonghong@example.com</email>
      <facebook>https://www.facebook.com/gildonghong</facebook>
      <twitter>https://twitter.com/gildonghong</twitter>
    </translator>
    <publisher tel="(877) 762-2974" fax="(800) 597-3299">
      <name>Wrox</name>
      <facebook>https://www.facebook.com/WroxPress</facebook>
      <twitter>https://twitter.com/wrox</twitter>
    </publisher>
    <price currency="원">35000</price>
    <sellingprice currency="원" save="0.3">24500</sellingprice>
    ...
  </book>
  ...
</books>
```

유일 제약 정의

- 유일 제약 정의:
중복된 값을 가지지 않도록 제한하는 것
- xs:ID 애트리뷰트, <xs:key>와 <xs:unique> 엘리먼트를 이용해
유일 제약을 구현

xs:ID 애트리뷰트를 이용한 유일 제약 정의

- xs:ID 애트리뷰트 타입은 빌트인 심플 타입으로 XML 문서 전체에서 유일한 값을 가지도록 애트리뷰트를 지정할 수 있음
- xs:ID 애트리뷰트 타입의 값은 숫자로 시작하면 안되고, 공백을 포함할 수 없음
- xs:ID 애트리뷰트 타입으로 선언된 애트리뷰트의 값은 xs:IDREF 애트리뷰트 타입으로 선언된 애트리뷰트에서 참조
- xs:IDREF 애트리뷰트 타입으로 선언된 애트리뷰트는 XML 문서 전체에서 모든 xs:ID 애트리뷰트 타입으로 선언된 애트리뷰트 값을 사용할 수 있음

xs:ID 애트리뷰트를 이용한 유일 제약 정의

[books13.xsd]

```
...
<xs:element name="books">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="categories" type="categories"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
<xs:complexType name="categories">
  <xs:sequence>
    <xs:element name="category" type="category"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="category">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="id" type="xs:ID" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

```
<xs:complexType name="book">
```

```
  ...
  <xs:attribute name="id" type="xs:ID" use="required"/>
```

```
  <xs:attribute name="category" type="xs:IDREF" use="required"/>
</xs:complexType>
```

```
...
<xs:complexType name="book3">
```

```
  ...
  <xs:attribute name="id" type="xs:ID" use="required"/>
```

```
  <xs:attribute name="category" type="xs:IDREF" use="required"/>
  ...
</xs:complexType>
```

xs:ID 애트리뷰트를 이용한 유일 제약 정의

[books13.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books13.xsd">
  <categories>
    <category id="c01">programming</category>
    <category id="c02">database</category>
    <category id="c03">security</category>
  </categories>
  <book id="b01" category="c01">
    <title>Beginning XML, 5th Edition</title>
    <author>
      <name>Joe Fawcett</name>
    </author>
    ...
  </book>
  <book id="b02" category="c01" ebook="false" xsi:type="book2">
    <title>Beginning XSLT and XPath: Transforming XML Documents and
Data</title>
    <author>
      <name>Ian Williams</name>
    </author>
    ...
  </book>
```

```
<book id="b03" category="b02" ebook="false" xsi:type="book3">
  <title>MongoDB: The Definitive Guide, 2nd Edition</title>
  <author>
    <name>Kristina Chodorow</name>
  </author>
  ...
</book>
</books>
```

<xs:key> 엘리먼트를 이용한 유일 제약 정의

- <xs:key> 엘리먼트의 name 애트리뷰트는 필수 애트리뷰트
- name 애트리뷰트의 값은 <xs:keyref> 엘리먼트에 의해 참조됨
- <xs:selector> 엘리먼트의 xpath 애트리뷰트 값은 대상이 되는 엘리먼트의 경로를 XPath 표현식으로 나타냄
- <xs:field> 엘리먼트의 xpath 애트리뷰트 값은 유일 제약이 적용될 애트리뷰트(XPath 표현식에 의해 '@' 기호를 붙임)를 지정함

```
<xs:key name="keyName">  
  <xs:selector xpath="element xpath expression"/>  
  <xs:field xpath="attribute xpath expression"/>  
</xs:key>
```

<xs:key> 엘리먼트를 이용한 유일 제약 정의

- <xs:keyref> 엘리먼트의 name과 refer 애트리뷰트는 필수 애트리뷰트
- ref 애트리뷰트의 값은 <xs:key> 엘리먼트의 name 애트리뷰트 값을 참조
- <xs:selector> 엘리먼트의 xpath 애트리뷰트 값은 대상이 되는 엘리먼트의 경로를 XPath 표현식으로 나타냄
- <xs:field> 엘리먼트의 xpath 애트리뷰트 값은 유일 제약이 적용될 애트리뷰트(XPath 표현식에 의해 '@' 기호를 붙임)를 지정함

```
<xs:keyref name="keyrefName" refer="keyName">  
  <xs:selector xpath="element xpath expression"/>  
  <xs:field xpath="attribute xpath expression"/>  
</xs:keyref>
```

<xs:key> 엘리먼트를 이용한 유일 제약 정의

[books14.xsd]

```
...
<xs:element name="books" type="books">
  <xs:key name="categoryKey">
    <xs:selector xpath="categories/category"/>
    <xs:field xpath="@id"/>
  </xs:key>
  <xs:key name="bookKey">
    <xs:selector xpath="book"/>
    <xs:field xpath="@id"/>
  </xs:key>
  <xs:keyref refer="categoryKey" name="categoryKeyRef">
    <xs:selector xpath="book"/>
    <xs:field xpath="@category"/>
  </xs:keyref>
</xs:element>
```

```
...
<xs:complexType name="books">
  <xs:sequence>
    <xs:element name="categories" type="categories"/>
    <xs:element name="book" type="book" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
...
```

```
<xs:complexType name="category">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="id" type="xs:string" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
...
<xs:complexType name="book">
  ...
  <xs:attribute name="id" type="xs:string" use="required"/>
  <xs:attribute name="category" type="xs:string" use="required"/>
</xs:complexType>
...
<xs:complexType name="book3">
  ...
  <xs:attribute name="id" type="xs:string" use="required"/>
  <xs:attribute name="category" type="xs:string" use="required"/>
  ...
</xs:complexType>
...
```

<xs:key> 엘리먼트를 이용한 유일 제약 정의

[books14.xml]

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="books14.xsd">
  <categories>
    <category id="c01">programming</category>
    <category id="c02">database</category>
    <category id="c03">security</category>
  </categories>
  <book id="b01" category="c01">
    <title>Beginning XML, 5th Edition</title>
    <author>
      <name>Joe Fawcett</name>
    </author>
    ...
  </book>
  <book id="b02" category="c01" ebook="false" xsi:type="book2">
    <title>Beginning XSLT and XPath: Transforming XML Documents and
Data</title>
    <author>
      <name>Ian Williams</name>
    </author>
    ...
  </book>
```

```
<book id="b03" category="c02" ebook="false" xsi:type="book3">
  <title>MongoDB: The Definitive Guide, 2nd Edition</title>
  <author>
    <name>Kristina Chodorow</name>
  </author>
  ...
</book>
</books>
```

<xs:unique> 엘리먼트를 이용한 유일 제약 정의

- 값을 참조하지 않고 그 값에 대한 유일성만 보장할 필요가 있을 경우
<xs:unique> 엘리먼트를 적용
- <xs:unique> 엘리먼트의 name 애트리뷰트는 필수 속성
- <xs:selector> 엘리먼트의 xpath 애트리뷰트 값은 대상이 되는 엘리먼트의 경로를 XPath 표현식으로 나타냄
- <xs:field> 엘리먼트의 xpath 애트리뷰트 값은 유일 제약이 적용될 애트리뷰트(XPath 표현식에 의해 '@' 기호를 붙임)를 지정함

```
<xs:unique name="uniqueName">  
  <xs:selector xpath="element xpath expression"/>  
  <xs:field xpath="attribute xpath expression"/>  
</xs:unique>
```


학습정리

- 자식 엘리먼트 혹은 애트리뷰트를 갖는 엘리먼트를 사용하려면 `<xs:complexType>`으로 컴플렉스 타입을 정의해 사용한다.
- `<xs:sequence>`를 사용하여 하나 이상의 자식 엘리먼트를 순차적으로 갖는 컴플렉스 타입을 정의해 사용한다.
- `<xs:complexType>` 혹은 `<xs:sequence>`에서 `<xs:choice>`를 사용하여 하나 이상의 자식 엘리먼트를 선택적으로 갖는 컴플렉스 타입을 정의해 사용한다.
- `<xs:simpleContent>`와 `base` 애트리뷰트의 심플 타입 확장을 이용해 애트리뷰트와 콘텐츠를 갖는 엘리먼트를 정의해 사용한다.
- 컴플렉스 타입에 대한 확장 혹은 제한을 정의한 `<xs:simpleContent>`와 `<xs:complexContent>`로 컴플렉스 타입을 정의해 사용한다.
- 중복되는 엘리먼트와 애트리뷰트 선언을 모아 모델 그룹 `<xs:group>`과 애트리뷰트 그룹 `<xs:attributeGroup>`을 만들고 참조함으로써 스키마 문서의 복잡성을 줄여주고 가독성을 높일 수 있다.
- 유일 제약 정의란 중복된 값을 가지지 않도록 제한하는 것으로 `xs:ID` 애트리뷰트, `<xs:key>`와 `<xs:unique>` 엘리먼트를 이용해 유일 제약을 구현한다.