

# XML Namespace

(13주차)

# 학습개요

- 학습 목표

- XML Namespace을 사용하는 이유에 대해 설명할 수 있다.
- XML Namespace 권고안을 설명할 수 있다.
- XML Namespace 설계 및 사용 방법을 설명할 수 있다.
- 디폴트 XML Namespace에 대해 설명할 수 있다.
- XML Namespace를 갖는 스키마 사용 방법에 대해 설명할 수 있다.
- XML Namespace 적용 방법을 설명할 수 있다.

- 학습 내용

- XML Namespace
- XML Namespace 권고안
- XML Namespace 선언과 사용
- 디폴트 Namespace
- XML Namespace를 갖는 스키마
- XML Namespace 실습

# 엘리먼트 이름 충돌

- XML 문서에서 여러 마크업 언어를 사용할 경우 같은 이름을 가진 엘리먼트간에 충돌 발생으로 인해 엘리먼트 식별이 불가능
- 엘리먼트의 이름을 모두 다르게 주는 것은 현실적으로 불가능
- XML Namespace를 이용해 엘리먼트의 이름 충돌 문제를 해결할 수 있음

# XML Namespace를 이용한 엘리먼트 이름 충돌 문제 해결

- 접두어:엘리먼트명의 사용으로 엘리먼트 이름 충돌 문제 해결
- 동일 마크업 언어의 엘리먼트는 동일 접두어를 사용
- XML 파서는 ':'가 들어간 엘리먼트도 유효한 엘리먼트로 인식
- 접두어를 사용하기 위해서는 XML Namespace 선언이 필요

# XML Namespace 권고안



## Namespaces in XML 1.0 (Third Edition)

W3C Recommendation 8 December 2009

**This version:**

<http://www.w3.org/TR/2009/REC-xml-names-20091208/>

**Latest version:**

<http://www.w3.org/TR/xml-names/>

**Previous versions:**

<http://www.w3.org/TR/2006/REC-xml-names-20060816/> <http://www.w3.org/TR/2009/PER-xml-names-20090806/>

**Editors:**

Tim Bray, Textuality <[tbray@textuality.com](mailto:tbray@textuality.com)>

Dave Hollander, Contivo, Inc. <[dmh@contivo.com](mailto:dmh@contivo.com)>

Andrew Layman, Microsoft <[andrewl@microsoft.com](mailto:andrewl@microsoft.com)>

Richard Tobin, University of Edinburgh and Markup Technology Ltd <[richard@inf.ed.ac.uk](mailto:richard@inf.ed.ac.uk)>

Henry S. Thompson, University of Edinburgh and W3C <[ht@w3.org](mailto:ht@w3.org)> - Third Edition

Please refer to the [errata](#) for this document, which may include normative corrections.

See also [translations](#).

This document is also available in these non-normative formats: [XML](#) and [HTML highlighting differences from the second edition](#).

Copyright © 2009 W3C® (MIT, ERCIM, Keio), All Rights Reserved. W3C [liability](#), [trademark](#) and [document use](#) rules apply.

### Abstract

- W3C의 Namespace in XML 권고안 - <http://www.w3.org/TR/xml-names/>

# XML Namespace

- XML Namespace는 URI 참조[RFC3986]를 이용해 식별
- URI는 Uniform Resource Indicator로 리소스를 식별하는 문자열
- URI의 대표적인 형태가 웹에서 사용하는 URL(Uniform Resource Locator)
- XML Namespace를 URL형태로 만드는 이유는 유일한 이름을 만들 수 있기 때문
- 마크업 언어가 XML Namespace를 가지지 않을 경우 XML Namespace 선언 없이 XML 문서를 작성할 수 있지만 XML Namespace를 가지고 있다면 반드시 XML Namespace 선언 후 사용해야 함

# XML Namespace 선언

- XML Namespace 선언은 XML Namespace와 이를 대신해 사용할 접두어와 연결짓는 것
- XML Namespace 선언은 xmlns:접두어 애트리뷰트로 기술하고 애트리뷰트 값으로 XML Namespace 이름에 해당하는 URL을 사용
- XML Namespace 선언은 일반적으로 루트 엘리먼트에서 기술

# XML Namespace 선언

- XML Namespace 접두어 작성 규칙
  - 첫 번째 문자는 유니코드 문자나 '\_'로 시작해야 하며 숫자나 '.'으로 시작할 수 없음
  - 두 번째 문자부터 숫자 또는 '-', '.' 사용이 가능
  - 이름 안에 공백문자를 포함할 수 없음
  - ':'는 XML Namespace 관련 기호이므로 사용하지 않을 것을 권장



# XML Namespace 선언

- ```
<!-- xml namespace 선언 -->
<order xmlns:c='http://www.example.com/2015/customer-schema'>

</order>
```

```
<!-- xml namespace 선언 -->
<o:order xmlns:o='http://www.example.com/2015/order-schema'
        xmlns:c='http://www.example.com/2015/customer-schema'>

</o:order>
```

# XML Namespace 사용

- 엘리먼트 이름 앞에 XML Namespace 접두어를 붙임으로써 동일 XML Namespace에 속하게 됨
- XML Namespace의 사용 범위는 XML Namespace 선언을 한 엘리먼트와 자손 엘리먼트로 한정됨
- 루트 엘리먼트에 XML Namespace를 선언하면 모든 엘리먼트에 XML Namespace 접두어를 붙일 수 있음

# XML Namespace 사용

- ```
<!-- xml namespace 선언 -->
<order xmlns:c='http://www.example.com/2015/customer-schema'>
  <id>20150102361003</id>
  <price>99000</price>
```

```
<!-- xml namespace 사용 -->
<c:id>1513667005</c:id>
<c:name>홍길동</c:name>
<c:address>서울 송파구 올림픽로35길 104</c:address>
</order>
```

```
<!-- xml namespace 선언 -->
<o:order xmlns:o='http://www.example.com/2015/order-schema'
        xmlns:c='http://www.example.com/2015/customer-schema'>
  <!-- xml namespace 사용 -->
  <o:id>20150102361003</id>
  <o:price>99000</price>
```

```
<!-- xml namespace 사용 -->
<c:id>1513667005</c:id>
<c:name>홍길동</c:name>
<c:address>서울 송파구 올림픽로35길 104</c:address>
</o:order>
```

# 디폴트 Namespace

- 디폴트 Namespace는 접두어가 없는 모든 엘리먼트가 기본적으로 속하게 되는 XML Namespace
- xmlns 다음에 접두어를 생략해 선언하면 디폴트 Namespace를 선언하게 됨
- 디폴트 Namespace는 엘리먼트에 대해서만 적용되며 애트리뷰트에 대해서는 적용되지 않음
- 애트리뷰트에 적용할 XML Namespace를 추가로 선언하여 애트리뷰트에 접두사를 붙여 기술

# 디폴트 Namespace

```
<!-- default namespace 선언 -->  
<product xmlns='http://www.example.com/2015/product-schema'>  
  <id>3823811</id>  
  <name>JBL Pulse Blooth Speaker</name>  
  <price>249000</price>  
</product>
```

```
<!-- default namespace 선언 -->  
<product xmlns='http://www.example.com/2015/product-schema'  
  xmlns:p='http://www.example.com/2015/product-schema'>  
  <id>3823811</id>  
  <name>JBL Pulse Blooth Speaker</name>  
  <price p:currency="원">249000</price>  
</product>
```

# XML Namespace를 갖는 스키마 문서

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.example.com/2015/books-schema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  xmlns="http://www.example.com/2015/books-schema">
...
</xs:schema>
```

- targetNamespace
  - 스키마 문서에서 정의하는 새로운 마크업 언어의 XML Namespace를 지정
- elementFormDefault
  - 스키마 문서에 선언된 엘리먼트들이 targetNamespace에 속할 것인지 지정. (qualified => targetNamespace에 속하며 접두사를 반드시 붙여야 함, unqualified => targetNamespace에 속하지 않으며 접두사없이 사용)
- attributeFormDefault
  - 스키마 문서에 선언된 애트리뷰트들이 targetNamespace에 속할 것인지 지정 (qualified => targetNamespace에 속하며 접두사를 반드시 붙여야 함, unqualified => targetNamespace에 속하지 않으며 접두사없이 사용)
- xmlns
  - xmlns="xml\_namespace\_name" => 디폴트 Namespace 선언으로 엘리먼트를 접두어 없이 사용

# XML Namespace를 갖는 스키마 문서의 인스턴스 지정

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.example.com/2015/books-schema books.xsd"
      xmlns="http://www.example.com/2015/books-schema">

  ...
</books>
```

- `xsi:schemaLocation`

- XML Namespace가 있는 스키마 문서의 경우 `xsi:noNamespaceSchemaLocation` 애트리뷰트 대신 스키마 문서를 지정하는 애트리뷰트
- `xsi:schemaLocation="xml_namespace_name xml_schema_uri"`
- `xsi:schemaLocation="xml_namespace1_name xml_schema1_uri  
xml_namespace2_name xml_schema2_uri"`

# XML Namespace를 갖는 스키마 문서의 인스턴스 지정

```
<?xml version="1.0" encoding="UTF-8"?>
<books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.example.com/2015/books-schema books.xsd"
      xmlns="http://www.example.com/2015/books-schema">
  ...
</books>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<bm:books xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:schemaLocation="http://www.example.com/2015/books-schema books.xsd"
          xmlns:bm="http://www.example.com/2015/books-schema">
  ...
</bm:books>
```

- xmlns

- xmlns="xml\_namespace\_name" => 디폴트 Namespace 선언으로 엘리먼트를 접두어 없이 사용
- xmlns:prefix="xml\_namespace\_name" => 모든 엘리먼트에 접두어를 붙여야 함



# 학습정리

- XML Namespace를 적용함으로써 엘리먼트의 이름 충돌 문제를 "XML Namespace의 접두어:엘리먼트명"의 사용으로 해결할 수 있다.
- XML 문서가 XML Namespace를 가지면 반드시 XML Namespace 선언 후 사용해야 한다.
- 엘리먼트 이름 앞에 XML Namespace 접두어를 붙이면 동일 XML Namespace에 속하게 된다.
- XML Namespace의 사용 범위는 XML Namespace 선언을 한 엘리먼트와 자손 엘리먼트로 한정된다.
- 루트 엘리먼트에 XML Namespace를 선언하면 모든 엘리먼트에 XML Namespace 접두어를 붙일 수 있다.
- xmlns 애트리뷰트에 접두어를 생략해 선언하면 디폴트 Namespace를 선언하게 된다.
- 디폴트 Namespace는 엘리먼트에 대해서만 적용되며 애트리뷰트에 대해서는 적용되지 않는다.
- 스키마 문서에서 targetNamespace 애트리뷰트에는 스키마 문서에서 정의하는 새로운 마크업 언어의 XML Namespace를 지정한다.
- 스키마 문서에서 elementFromDefault 애트리뷰트에는 일반적으로 qualified를 attributeFromDefault는 unqualified를 지정한다.