



XSD



tutorialspoint
SIMPLY EASY LEARNING

www.tutorialspoint.com



<https://www.facebook.com/tutorialspointindia>



<https://twitter.com/tutorialspoint>

About the Tutorial

XML Schema Definition commonly known as XSD is a way to describe precisely the XML language. XSDs check the validity of structure and vocabulary of an XML document against the grammatical rules of the appropriate XML language.

This tutorial will teach you the basics of XSD. It contains chapters that explain all the basic components of XSD with suitable examples.

Audience

This tutorial has been prepared for beginners to help them understand the basic concepts related to XSD. It will give you enough understanding on XSD from where you can take yourself to a higher level of expertise.

Prerequisites

Before proceeding with this tutorial, you should have basic knowledge of XML, HTML, and JavaScript.

Copyright & Disclaimer

© Copyright 2018 by Tutorials Point (I) Pvt. Ltd.

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com

Table of Contents

About the Tutorial.....	i
Audience.....	i
Prerequisites	i
Copyright & Disclaimer	i
Table of Contents	ii
1. XSD – OVERVIEW	1
2. XSD – SYNTAX	4
<Schema> Element.....	5
Referencing Schema.....	5
3. XSD – VALIDATION	7
4. XSD – SIMPLE TYPES	12
XSD – Element.....	12
XSD – Attribute.....	13
XSD – Restriction	15
Types of Restrictions	16
5. XSD – COMPLEX TYPES	18
XSD – Complex Empty Element	19
XSD – Complex Element Only	20
XSD – Complex Text Only Element	22
XSD – Complex Mix Element	23
XSD – Complex Indicators	24
XSD – <any>	28
Use <xs:any>	30
XSD – <anyAttribute>	31
6. XSD – STRING.....	34

7. XSD – DATE TIME	36
8. XSD – NUMERIC	39
9. XSD – MISCELLANEOUS	42

1. XSD – Overview

XML Schema Definition, commonly known as XSD, is a way to describe precisely the XML language. XSD checks the validity of structure and vocabulary of an XML document against the grammatical rules of the appropriate XML language.

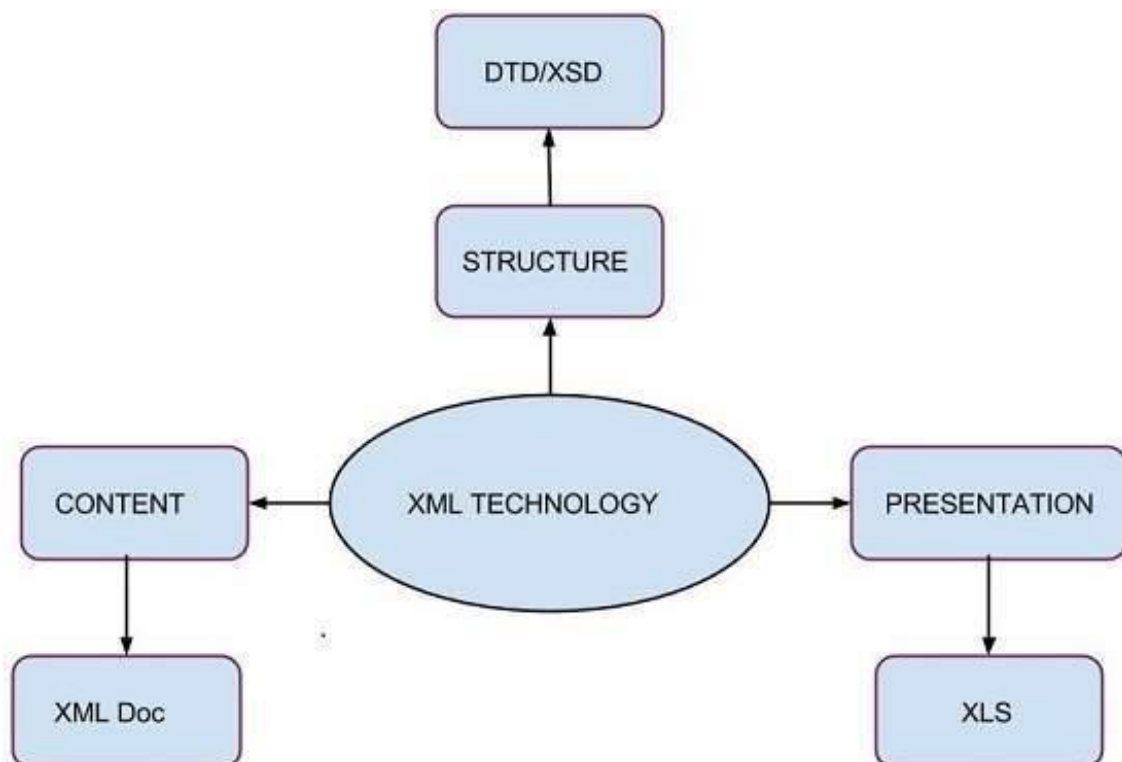
An XML document can be defined as:

- **Well-formed:** If the XML document adheres to all the general XML rules such as tags must be properly nested, opening and closing tags must be balanced, and empty tags must end with '</>', then it is called as *well-formed*.

OR

- **Valid:** An XML document said to be valid when it is not only *well-formed*, but it also conforms to available XSD that specifies which tags it uses, what attributes those tags can contain, and which tags can occur inside other tags, among other properties.

The following diagram shows how XSD is used to structure XML documents:



Here is a simple XSD code. Take a look at it.

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

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.tutorialspoint.com"
xmlns="http://www.tutorialspoint.com"
elementFormDefault="qualified">

  <xs:element name='class'>
    <xs:complexType>
      <xs:sequence>
        <xs:element name='student' type='StudentType'
          minOccurs='0' maxOccurs='unbounded' />
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:complexType name="StudentType">
    <xs:sequence>
      <xs:element name="firstname" type="xs:string"/>
      <xs:element name="lastname" type="xs:string"/>
      <xs:element name="nickname" type="xs:string"/>
      <xs:element name="marks" type="xs:positiveInteger"/>
    </xs:sequence>
    <xs:attribute name='rollno' type='xs:positiveInteger' />
  </xs:complexType>
</xs:schema>
```

Features

Here is a list of some of the popular features of XSD:

- XSDs can be extensible for future additions.
- XSD is richer and more powerful than DTD.
- XSD is written in XML.
- XSD supports data types.
- XSD supports namespaces.
- XSD is W3C recommendation.

2. XSD – Syntax

An XML XSD is kept in a separate document and then the document can be linked to an XML document to use it.

Syntax

The basic syntax of an XSD is as follows:

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

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.tutorialspoint.com"
xmlns="http://www.tutorialspoint.com"
elementFormDefault="qualified">

  <xs:element name='class'>

    <xs:complexType>

      <xs:sequence>

        <xs:element name='student' type='StudentType'
          minOccurs='0' maxOccurs='unbounded' />

      </xs:sequence>

    </xs:complexType>

  </xs:element>

  <xs:complexType name="StudentType">

    <xs:sequence>

      <xs:element name="firstname" type="xs:string"/>

      <xs:element name="lastname" type="xs:string"/>

      <xs:element name="nickname" type="xs:string"/>

      <xs:element name="marks" type="xs:positiveInteger"/>

    </xs:sequence>

    <xs:attribute name='rollno' type='xs:positiveInteger' />

  </xs:complexType>
```

```
</xs:schema>
```

<Schema> Element

Schema is the root element of XSD and it is always required.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

The above fragment specifies that elements and datatypes used in the schema are defined in "http://www.w3.org/2001/XMLSchema" namespace and these elements/data types should be prefixed with **xs**. It is always required.

```
targetNamespace="http://www.tutorialspoint.com"
```

The above fragment specifies that elements used in this schema are defined in "http://www.tutorialspoint.com" namespace. It is optional.

```
xmlns="http://www.tutorialspoint.com"
```

The above fragment specifies that default namespace is "http://www.tutorialspoint.com".

```
xmlns="http://www.tutorialspoint.com"
```

The above fragment indicates that any elements declared in this schema must be namespace qualified before using them in any XML Document. It is optional.

Referencing Schema

Take a look at the following Referencing Schema:

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

<class xmlns="http://www.tutorialspoint.com"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.tutorialspoint.com student.xsd">

    <student rollno="393">

        <firstname>Dinkar</firstname>

        <lastname>Kad</lastname>

        <nickname>Dinkar</nickname>
```

```

        <marks>85</marks>
    </student>
    <student rollno="493">
        <firstname>Vaneet</firstname>
        <lastname>Gupta</lastname>
        <nickname>Vinni</nickname>
        <marks>95</marks>
    </student>
    <student rollno="593">
        <firstname>Jasvir</firstname>
        <lastname>Singh</lastname>
        <nickname>Jazz</nickname>
        <marks>90</marks>
    </student>
</class>
xmlns="http://www.tutorialspoint.com"

```

The above fragment specifies default namespace declaration. This namespace is used by the schema validator check that all the elements are part of this namespace. It is optional.

```

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.tutorialspoint.com student.xsd">

```

After defining the XMLSchema-instance xsi, use **schemaLocation** attribute. This attribute has two values, namespace and location of XML Schema, to be used separated by a space. It is optional.

End of ebook preview

If you liked what you saw...

Buy it from our store @ **<https://store.tutorialspoint.com>**