# **DTD**

DTD stands for **Document Type Definition**. It defines the legal building blocks of an XML document. It is used to define document structure with a list of legal elements and attributes.

Its main purpose is to define the structure of an XML document. It contains a list of legal elements and define the structure with the help of them.

Before proceeding with XML DTD, you must check the validation. An XML document is called "well-formed" if it contains the correct syntax.

A well-formed and valid XML document is one which have been validated against DTD.

Let's take an example of well-formed and valid XML document. It follows all the rules of DTD.

*employee.xml*

1. **<?xml** version="1.0"**?>**
2. <!DOCTYPE employee SYSTEM "employee.dtd"**>**
3. **<employee>**
4. **<firstname>**vimal**</firstname>**
5. **<lastname>**jaiswal**</lastname>**
6. **<email>**vimal@javatpoint.com**</email>**
7. **</employee>**

In the above example, the DOCTYPE declaration refers to an external DTD file. The content of the file is shown in below paragraph.

*employee.dtd*

1. <!ELEMENT employee (firstname,lastname,email)**>**
2. <!ELEMENT firstname (#PCDATA)**>**
3. <!ELEMENT lastname (#PCDATA)**>**
4. <!ELEMENT email (#PCDATA)**>**

## Description of DTD

**<!DOCTYPE employee :** It defines that the root element of the document is employee.

**<!ELEMENT employee:** It defines that the employee element contains 3 elements "firstname, lastname and email".

**<!ELEMENT firstname:** It defines that the firstname element is #PCDATA typed. (parse-able data type).

**<!ELEMENT lastname:** It defines that the lastname element is #PCDATA typed. (parse-able data type).

**<!ELEMENT email:** It defines that the email element is #PCDATA typed. (parse-able data type).

# **XSL**

XSL is a language for expressing style sheets. An XSL style sheet is, like with [CSS](https://www.w3.org/Style/CSS), a file that describes how to display an XML document of a given type. XSL shares the functionality and is compatible with CSS2 (although it uses a different syntax). It also adds:

* A transformation language for XML documents: **XSLT**. Originally intended to perform complex styling operations, like the generation of tables of contents and indexes, it is now used as a general purpose XML processing language. XSLT is thus widely used for purposes other than XSL, like generating HTML web pages from XML data.
* Advanced styling features, expressed by an XML document type which defines a set of elements called **Formatting Objects**, and attributes (in part borrowed from CSS2 properties and adding more complex ones.

Styling requires a source XML documents, containing the information that the style sheet will display and the style sheet itself which describes how to display a document of a given type.

The following shows a sample XML file and how it can be transformed and rendered.

## The XML file

<scene>

<FX>General Road Building noises.</FX>

<speech speaker="Prosser">

Come off it Mr Dent, you can't win

you know. There's no point in lying

down in the path of progress.

</speech>

<speech speaker="Arthur">

I've gone off the idea of progress.

It's overrated

</speech>

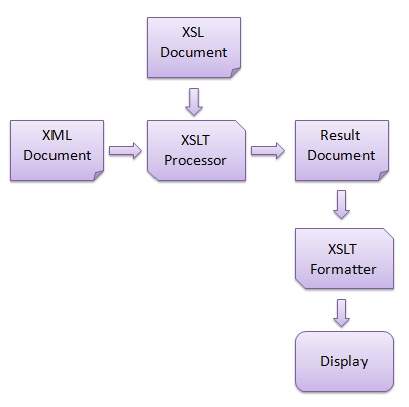
</scene>

This XML file doesn't contain any presentation information, which is contained in the stylesheet. Separating the document's content and the document's styling information allows displaying the same document on different media (like screen, paper, cell phone), and it also enables users to view the document according to their preferences and abilities, just by modifying the style sheet.

**XSLT**

Extensible Stylesheet Language Transformations, provides the ability to transform XML data from one format to another automatically.

An XSLT stylesheet is used to define the transformation rules to be applied on the target XML document. XSLT stylesheet is written in XML format. XSLT Processor takes the XSLT stylesheet and applies the transformation rules on the target XML document and then it generates a formatted document in the form of XML, HTML, or text format. This formatted document is then utilized by XSLT formatter to generate the actual output which is to be displayed to the end-user.



Advantages

Here are the advantages of using XSLT −

* Independent of programming. Transformations are written in a separate xsl file which is again an XML document.
* Output can be altered by simply modifying the transformations in xsl file. No need to change any code. So Web designers can edit the stylesheet and can see the change in the output quickly.