

DTD

DECLARACIÓN DE TIPO DE DOCUMENTO:

- Sintaxis DTD interna:
`<!DOCTYPE elemento-raíz [declaraciones]>`
- Sintaxis DTD externa privada:
`<!DOCTYPE elemento-raíz SYSTEM "URI">`

DECLARACIÓN DE ELEMENTOS:

- Sintaxis:
`<!ELEMENT nombre-del-elemento tipo-de-contenido>`
- En el tipo-de-contenido se especifica el contenido permitido en el elemento, pudiendo ser:
 - Texto, (`#PCDATA`).
 - Otros elementos (hijos).
 - Estar vacío, `EMPTY`.
 - Mixto (texto y otros elementos), `ANY`.
- Operadores de cardinalidad en DTD:
 - ? (interrogación): 0-1
 - * (asterisco): 0-n
 - + (signo más): 1-n

DECLARACIÓN DE ATRIBUTOS:

- Sintaxis:
`<!ATTLIST nombre-del-elemento nombre-del-atributo tipo-de-atributo valor-del-atributo>`

TIPOS DE DECLARACIÓN DE ATRIBUTOS:

- `valor` entre comillas dobles (") o simples (').
- `#REQUIRED`
- `#IMPLIED`
- `#FIXED valor` entre comillas dobles (") o simples (').

TIPOS DE ATRIBUTOS:

- `CDATA`, Enumerado, `ID`, `IDREF`, `IDREFS`, `NMTOKEN`, `NMTOKENS`, `NOTATION`, `ENTITY`, `ENTITIES`, Especiales

Referencias a entidades

<code><</code> (<i>menor que</i>)	<code>lt</code> (<i>less than</i>)	<code>&lt;</code>
<code>></code> (<i>mayor que</i>)	<code>gt</code> (<i>greater than</i>)	<code>&gt;</code>
<code>"</code> (<i>comilla doble</i>)	<code>quot</code> (<i>quotation mark</i>)	<code>&quot;</code>
<code>'</code> (<i>comilla simple</i>)	<code>apos</code> (<i>apostrophe</i>)	<code>&apos;</code>
<code>&</code> (<i>ampersand</i>)	<code>amp</code> (<i>ampersand</i>)	<code>&amp;</code>

XSD
<p><u>DECLARACIÓN:</u></p> <pre><?xml version="1.0" encoding="UTF-8"?> <marcadores xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="marcadores.xsd"></pre>
<p><u>ELEMENTOS SIMPLES:</u></p> <pre><xs:element name="nombre_del_elemento" type="tipo_de_dato"/></pre> <p><u>TIPOS DE DECLARACIÓN DE ELEMENTOS SIMPLES:</u></p> <pre><xs:element name="mes" type="xs:string" fixed="agosto"/> <xs:element name="mes" type="xs:string" default="agosto"/> <xs:element name="nombre del elemento" type="tipo global de datos" ref="declaración del elemento global" id="identificador" form="qualified/unqualified" minOccurs="número mínimo de veces, por defecto 1" maxOccurs="máximo número de veces, por defecto 1" default="valor por defecto" fixed="valor fijo" /></pre>
<p><u>ATRIBUTOS:</u></p> <pre><xs:attribute name="nombre_del_atributo" type="tipo_de_dato"/></pre> <p><u>TIPOS DE DECLARACIÓN DE ATRIBUTOS</u></p> <pre><xs:attribute name="grupo" type="xs:string" fixed="B"/> <xs:attribute name="grupo" type="xs:string" default="B"/> <xs:attribute name="grupo" type="xs:string" use="required"/> <xs:attribute default=string fixed=string form=qualified unqualified id=ID name=NCName ref=QName type=QName use=optional prohibited required any attributes ></pre>
<p>xs:extension sirve para extender un elemento simpleType o complexType.</p>
<p><u>Indicadores de orden (xs:sequence, xs:all, xs:choice)</u></p> <p><u>Indicadores de ocurrencia (maxOccurs, minOccurs)</u></p> <p><u>Indicadores de grupo (xs:group, xs:attributeGroup)</u></p>

TIPOS DE DATOS PREDEFINIDOS:

- **Tipos cadena**
 - **string**: secuencia de longitud finita de caracteres *
 - **anyURI**: una uri estándar de Internet
 - **NOTATION**: declara enlaces a contenido externo no-XML
 - **Qname**: una cadena legal QName (nombre con cualificador)
- **Tipos binario codificado**
 - **boolean**: toma los valores "true" o "false" *
 - **hexBinary**: dato binario codificado como una serie de pares de dígitos hexadecimales
 - **base64Binary**: datos binarios codificados en base 64
- **Tipos numéricos**
 - **decimal**: número decimal de precisión (dígitos significativos) arbitraria *
 - **float**: número de punto flotante de 32 bits de precisión simple *
 - **double**: número de punto flotante de 64 bits de doble precisión *
- **Tipos de fecha/hora**
 - **duration**: duración de tiempo
 - **dateTime**: instante de tiempo específico, usando calendario gregoriano, en formato "YYYYMM-DDThh:mm:ss"
 - **date**: fecha específica del calendario gregoriano, en formato "YYYY-MM-DD" *
 - **time**: una instancia de tiempo que ocurre cada día, en formato "hh:mm:ss"
 - **gYearMonth**: un año y mes del calendario gregoriano
 - **gYear**: año del calendario gregoriano
 - **gMonthDay**: día y mes del calendario gregoriano
 - **gMonth**: un mes del calendario gregoriano
 - **gDay**: una fecha del calendario gregoriano (día)

<i>Símbolos</i>	<i>Significado</i>
.	Cualquier carácter.
\d	Cualquier dígito del 0 al 9.
\D	Cualquier carácter que no sea un dígito del 0 al 9.
x*	x puede aparecer cero o más veces.
x+	x debe aparecer al menos una vez.
x?	x puede aparecer una vez o no aparecer.
[abc] o [a b c]	Cualquier carácter indicado entre los corchetes: a, b o c.
[a-z]	Cualquier carácter de la a a la z.
x{n}	x debe aparecer n veces.
x{n,m}	x debe aparecer entre n y m veces.
x{n,}	x debe aparecer al menos n veces.

Patrones

Restricción	Descripción
enumeration	Define una lista de valores aceptables
fractionDigits	Especifica el número máximo de decimales permitidos. Debe ser igual o mayor que cero
length	Especifica el número exacto de caracteres o elementos de lista de permitidos. Debe ser igual o mayor que cero
maxExclusive	Especifica los límites superiores para los valores numéricos (el valor debe ser inferior a este valor)
maxInclusive	Especifica los límites superiores para los valores numéricos (el valor debe ser menor o igual a este valor)
maxLength	Especifica el número máximo de caracteres permitidos o elementos de la lista. Debe ser igual o mayor que cero
minExclusive	Especifica los límites inferiores para los valores numéricos (el valor debe ser mayor que este valor)
minInclusive	Especifica los límites más bajos para los valores numéricos (el valor debe ser mayor que o igual a este valor)
minLength	Especifica el número mínimo de caracteres permitidos o elementos de lista. Debe ser igual o mayor que cero
pattern	Define la secuencia exacta de caracteres que son aceptables
totalDigits	Especifica el número exacto de dígitos permitidos. Debe ser mayor que cero
whiteSpace	Especifica el modo en el espacio blanco (saltos de línea, tabulaciones, espacios y retornos de carro) se maneja



Jednostavna XML Schema

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://interoperabilnost.hr" xmlns="http://interoperabilnost.hr"
elementFormDefault="qualified">
  <xs:element name="note">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="to" type="xs:string"/>
        <xs:element name="from" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Simple Types

- xs:string
- xs:decimal
- xs:integer
- xs:boolean
- xs:date
- xs:time

```
<xs:element name="start_date" type="xs:date"/>
```

XML Facets

- xs:minInclusive
- xs:maxInclusive
- xs:minExclusive
- xs:maxExclusive
- xs:enumeration
- xs:pattern
- xs:whiteSpace
- xs:length

- xs:minLength
- xs:maxLength
- xs:totalDigits
- xs:fractionDigits

Min-max

```
<xs:element name="age">
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="120"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

Enumeration

```
<xs:element name="car" type="carType"/>

<xs:simpleType name="carType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Audi"/>
    <xs:enumeration value="Golf"/>
    <xs:enumeration value="BMW"/>
  </xs:restriction>
</xs:simpleType>
```

Pattern

- regexp

```
<xs:restriction base="xs:string">
  <xs:pattern value="([a-z])+"/>
</xs:restriction>
```

WhiteSpace

- preserve: keep whitespaces
- replace: replace whitespace characters with spaces
- collapse: collapse all whitespace characters to a single space

```
<xs:restriction base="xs:string">
  <xs:whiteSpace value="preserve"/>
</xs:restriction>
```

Complex Elements

- empty elements
- elements that contain only other elements
- elements that contain only text
- elements that contain both other elements and text

Complex Example

```
<xs:element name="employee" type="personinfo"/>
<xs:element name="student" type="personinfo"/>

<xs:complexType name="personinfo">
  <xs:sequence>
    <xs:element name="firstname" type="xs:string"/>
    <xs:element name="lastname" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

<xs:element name="professor" type="fullpersoninfo">

<xs:complexType name="fullpersoninfo">
  <xs:complexContent>
    <xs:extension base="personinfo">
      <xs:sequence>
        <xs:element name="address" type="xs:string"/>
        <xs:element name="city" type="xs:string"/>
        <xs:element name="country" type="xs:string"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Empty Element

```
<xs:complexType name="prodtype">
  <xs:attribute name="prodid" type="xs:positiveInteger"/>
</xs:complexType>
```

Text only + attribute

```
<xs:complexType name="shoetype">
  <xs:simpleContent>
    <xs:extension base="xs:integer">
      <xs:attribute name="country" type="xs:string" />
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Element Indicators

- All - any order, only once
- Choice - order, at most once
- Sequence - order, only once
- minOccurs
- maxOccurs

Data Types

String

- xs:string

Date

Name	Description
xs:date	YYYY-MM-DD
xs:dateTime	YYYY-MM-DDThh:mm:ss
xs:duration	PnYnMnDTnHnMnS
xs:gDay	DD
xs:gMonth	MM
xs:gMonthDay	MM-DD
xs:gYear	YYYY
xs:gYearMonth	YYYY-MM
xs:time	hh:mm:ss

Numeric

- xs:byte
- xs:decimal
- xs:int
- xs:integer
- xs:long
- xs:negativeInteger
- xs:nonNegativeInteger
- xs:nonPositiveInteger
- xs:positiveInteger
- xs:short
- xs:unsignedLong
- xs:unsignedInt
- xs:unsignedShort
- xs:unsignedByte
- xs:float

- xs:double

Boolean

- xs:boolean