

# Actividad 2- Investigacion de DOM xml y XPath xml en Python

## XML.dom

### Contenido del Módulo

El módulo xml.dom contiene las siguientes funciones:

`xml.dom.registerDOMImplementation(name, factory)`

Registra la función `factory` con el nombre `name`. La función fábrica (`factory`) debe retornar un objeto que implemente la interfaz `DOMImplementation`. La función fábrica puede retornar el mismo objeto cada vez que se llame, o uno nuevo por cada llamada, según sea apropiado para la implementación específica (e.g. si la implementación soporta algunas personalizaciones).

`xml.dom.getDOMImplementation(name=None, features=())`

Retorna una implementación del DOM apropiada. El `name` es o bien conocido, el nombre del módulo de una implementación DOM, o `None`. Si no es `None` importa el módulo correspondiente y retorna un objeto `DomImplementation` si la importación tiene éxito. Si no se le pasa un nombre, y el entorno de variable `PYTHON_DOM` ha sido puesto, dicha variable es usada para encontrar la información de la implementación.

Si no se le pasa un nombre, examina las implementaciones disponibles para encontrar uno con el conjunto de características requeridas. Si no se encuentra ninguna implementación, levanta una excepción `ImportError`. La lista de características debe ser una secuencia de pares (`feature, version`) que son pasados al método `hasFeature()` en objetos disponibles de `DOMImplementation`.

Algunas constantes convenientes son proporcionadas:

`xml.dom.EMPTY_NAMESPACE`

El valor usado para indicar que ningún espacio de nombres es asociado con un nodo en el DOM. Se encuentra típicamente con el `namespaceURI` de un nodo, o usado como el parámetro `namespaceURI` para un método específico del namespace.

`xml.dom.XML_NAMESPACE`

El espacio de nombres de la URI asociada con el prefijo `xml`, como se define por Namespaces in XML (section 4).

`xml.dom.XMLNS_NAMESPACE`

El espacio de nombres del URI para declaraciones del espacio de nombres, como se define en Document Object Model (DOM) Level 2 Core Specification (section 1.1.8).

xml.dom.XHTML\_NAMESPACE

El URI del espacio de nombres del XHTML como se define en XHTML 1.0: The Extensible HyperText Markup Language (section 3.1.1).

Además, xml.dom contiene una clase base Node y las clases de excepciones del DOM. La clase Node proporcionada por este módulo no implementa ninguno de los métodos o atributos definidos en la especificación DOM; las implementaciones del DOM concretas deben definirlas. La clase Node propuesta por este módulo sí proporciona las constantes usadas por el atributo nodeType en objetos concretos de Node; estas son localizadas dentro de la clase en vez de estar al nivel del módulo para cumplir las especificaciones del DOM.

### Objetos en DOM

Interfaz	Sección	Propósito
DOMImplementation	<a href="#">Objetos DOMImplementation</a>	Interfaz para las implementaciones subyacentes.
Node	<a href="#">Objetos Nodo</a>	Interfaz base para la mayoría de objetos en un documento.
NodeList	<a href="#">Objetos NodeList</a>	Interfaz para una secuencia de nodos.
DocumentType	<a href="#">Objetos DocumentType</a>	Información acerca de la declaraciones necesarias para procesar un documento.
Document	<a href="#">Objetos Documento</a>	Objeto que representa un documento entero.
Element	<a href="#">Objetos Elemento</a>	Nodos elemento en la jerarquía del documento.
Attr	<a href="#">Objetos Atributo</a>	Nodos de los valores de los atributos en los elementos nodo.
Comment	<a href="#">Objetos Comentario</a>	Representación de los comentarios en el documento fuente.
Text	<a href="#">Objetos Texto y CDATASection</a>	Nodos con contenido textual del documento.
ProcessingInstruction	<a href="#">Objetos ProcessingInstruction</a>	Representación de instrucción del procesamiento.

## Xpath

Xpath es un módulo que es parte de la librería `xml.etree.ElementTree` por lo general la misma se importa de la siguiente manera:

```
1 import xml.etree.ElementTree as ET
```

Xpath provee una serie de expresiones para localizar elementos en un árbol, su finalidad es proporcionar un conjunto de sintaxis, por lo que debido a su limitado alcance no se considera un motor en si mismo.

Ejemplos de uso de Xpath:

```
1
2 import xml.etree.ElementTree as ET
3
4 root = ET.fromstring(docxml)
5
6 # Elementos de nivel superior
7 root.findall(".")
8
9 # todos los hijos de neighbor o nietos de country en el nivel superior
10 root.findall("./country/neighbor")
11
12 # Nodos xml con name='Singapore' que sean hijos de 'year'
13 root.findall("./year/..[@name='Singapore']")
14
15 # nodos 'year' que son hijos de etiquetas xml cuyo name='Singapore'
16 root.findall("./*[@name='Singapore']/year")
17
18 # todos los nodos 'neighbor' que son el segundo hijo de su padre
19 root.findall("./neighbor[2]")
```

Como vemos nos permite extraer facilmente partes del xml haciendo referencia a su ubicación nodal representada a forma de path, lo que nos hace una sintaxis familiarmente sencilla a la hora de construir un parser xml.

# Xpath Sintaxis:

SINTAXIS	Descripción
<b>tag</b>	Selecciona todos los elementos hijos contenidos en la etiqueta "tag", Por ejemplo: spam, selecciona todos los elementos hijos de la etiqueta spam y así sucesivamente en un path de nodos spam/egg, /spam/egg/milk
<b>*</b>	Selecciona todos los elementos hijos. Ejemplo: */egg, seleccionara todos los elementos nietos bajo la etiqueta egg
<b>.</b>	Selecciona el nodo actual, este es muy usado en el inicio del path, para indicar que es un path relativo.
<b>//</b>	Selecciona todos los sub elementos de todos los niveles bajo el nodo expresado. Por ejemplo: //egg selecciona todos los elementos bajo egg a través de todo el arbol bajo la etiqueta
<b>..</b>	Selecciona el elemento padre
<b>[@attrib]</b>	Selecciona todos los elementos que contienen el atributo tras el "@"
<b>[@attrib='value']</b>	Seleccione todos los elementos para los cuales el atributo dado tenga un valor dado, el valor no puede contener comillas
<b>[tag]</b>	Selecciona todos los elementos que contienen una etiqueta hijo llamada tag. Solo los hijos inmediatos son admitidos
<b>[tag='text']</b>	Selecciona todos los elementos que tienen una etiqueta hijo llamada tag incluyendo descendientes que sean igual al texto dado
<b>[position]</b>	Selecciona todos los elementos que se encuentran en la posición dada. La posición puede contener un entero siendo 1 la primera posición, la expresión last( ) para la ultima, o la posición relativa con respecto a la ultima posición last( )-1