OTROS LENGUAJES ASOCIADOS A XML.

- **Xpath**.
  - Navegación por documentos XML.
- **XQuery**.
  - Lenguaje de consultas a documentos XML.
- **XSLT**.
  - Lenguaje que permite aplicar una transformación a los datos de un documento XML.

XPath. (XML PATH LANGUAGE)

- **Concepto y características.**
  - Lenguaje que se utiliza para navegar y consultar los elementos y atributos contenidos en la estructura de un documento XML o HTML**.**
  - Es un lenguaje de descripción de rutas.
  - Desarrollado por W3C, es un lenguaje complejo en continuo desarrollo.
  - Se usa en combinación con **XSLT** (eXtensible Stylesheet Language for Transformations).
- **XPath en navegadores**.
  - **Buscar XPath**.
    - Abrir consola (pulsar F12, por ejemplo).
    - Firefox:
      - En la pestaña **inspector** utilizar el buscador HTML con expresiones XPath.
    - Google Chrome:
      - En la pestaña **elementos** abrir el buscador HTML con Control + F y escribir la expresión XPath de búsqueda.
  - **Copiar Xpath.**
    - Abrir consola (pulsar F12, por ejemplo).
    - Firefox:
      - Seleccionar la pestaña **inspector.**
      - Pulsar, con el botón derecho del ratón, sobre la etiqueta HTML de la cual se quiere copiar su XPath.
      - Seleccionar Copiar / XPath.
    - Google Chrome:
      - Seleccionar la pestaña **elementos.**
      - Pulsar, con el botón derecho del ratón, sobre la etiqueta HTML de la cual se quiere copiar su XPath.
      - Seleccionar Copiar / Copiar XPath.
- **Versiones:**
  - 1.0, 2.0, 2.0 segunda edición, 3.0 y 3.1.
- **Uso o procesamiento de un fichero XML**
  - Localizarlo en Internet mediante su URI o URL.
  - Seleccionar su información contenida usando XPath.

ÁRBOL DE UN DOCUMENTO XML.

- Al igual que una página HTML, un documento XML es una estructura jerárquica compuesta por:
    - Nodos ubicados en distintas ramas y niveles.
    - Relaciones entre esos diferentes nodos.
- **Tipos de nodos.**
    - **Nodo raíz**.
        - Contiene al elemento raíz del documento.
        - Todos los documentos XML tienen uno que indica el inicio del documento y fin del documento.
        - Todos los demás nodos dependes de él.
    - **Nodo Elemento**.
        - Representa a una etiqueta XML.
        - Puede tener descendientes.
    - **Nodo Atributo**.
        - Representa a un atributo asociado a una etiqueta XML.
        - No se considera un descendiente de un elemento, sino como una especie de etiqueta adosada a él.
        - No pueden tener descendientes.
    - **Nodo Texto**.
        - Representa al contenido de una etiqueta o elemento XML.
        - Se considera descendiente de un elemento.
        - No pueden tener descendientes.
    - **Otros tipos** de nodos:
        - Comentario.
        - Instrucciones de procesamiento.
        - Espacio de nombres.

EXPRESIONES Y NOTACIONES XPath.

- **Concepto.**
    - Cadena de texto que representa un recorrido por el árbol del documento XML.
- **Características.**
    - Parecidas a rutas de acceso a archivos en Windows o Linux.
    - Evaluar una expresión XPath consiste en buscar nodos que cumplen con lo definido en la expresión.
    - Las expresiones pueden indicarse con sintaxis abreviadas más simples o con sintaxis más largas y complejas.
    - Una expresión está compuesta por paso de búsqueda constituidos por ejes, predicados y selección de nodos.
    - Se pueden usar las propias etiquetas HTML para buscar elementos dentro de una página web a través de la consola.

SINTAXIS ABREVIADA.

EJE.

- Se corresponde con recorridos por el árbol.
- Selecciona un subconjunto de nodos de tipo:

        - *elemento* usando sus nombres.
        - *atributo* usando sus nombres precedidos por una arroba (@).
    - **Caracteres**:
        - **/**
            - Nodo raíz o nodo hijo.
        - **/Nombre**
            - Nodo elemento.
        - **//**
            - Todos los descendientes desde el nodo raíz.
        - **Text()**
            - Nodo texto.
        - **@Nombre**
            - Nodo atributo.
        - **|**
            - Varios recorridos a la vez.
        - **.**
            - Nodo actual.
        - **..**
            - Nodo padre.
        - *****
            - Todos los nodos.
    - **Algunas sintaxis**:
        - /nodo elemento 1/ nodo elemento 2/ …/ nodo elemento N.
        - /nodo elemento 1/ nodo elemento 2/ …/ @atributo.
        - /nodo elemento 1/ nodo elemento 2/ …/ text().
        - //nodo elemento
    - Ejemplos:
        - /biblioteca/libros
            - Selecciona todos los nodos *libros* de la ruta especificada.
        - //editorial
            - Selecciona todos los nodos *editoriales* desde el nodo raíz.
        - /biblioteca/libros/titulo/text()
            - Selecciona todos los títulos de los libros (texto del elemento), de la ruta especificada.
        - /biblioteca/libros/titulo/@codigo
            - Selecciona todos los atributos *código* de la ruta especificada.
        - //títulos | // precios
            - Selecciona todos los nodos *títulos* y *precios* desde el directorio raíz.

PREDICADOS.

- Permiten establecer condiciones para la selección de nodos.
- Los predicados se escriben entre corchetes.

- **Tipos de predicados.**
  - **[@atributo]**
    - Selecciona los elementos que tengan el atributo especificado.
    - Ejemplos:
      - //libros[@idioma]
        - Selecciona los nodos de tipo *libros* desde el elemento raíz, que tengan el atributo idioma.
  - **[número]**
    - Selecciona, de entre varios, el nodo que tenga el número de orden especificado.
    - El elemento debe escribirse entre paréntesis.
    - Ejemplos:
      - (//libros)[1]
        - Selecciona el primer nodo de tipo *libros*.
      - (//libros)[1]/titulo/text()
        - Selecciona el título del primer libro (texto del primer nodo *libros*), desde el directorio raíz.
  - **[last()]**
    - Selecciona, de entre varios, el último en la ruta especificada.
    - Ejemplos:
      - (//libros)[last()]
        - Selecciona el último nodo de tipo *libros*.
      - (//libros)[last()]/titulo/text()
        - Selecciona el título del último libro (texto del último nodo *libros*), desde el directorio raíz.
  - **[last() - número]**
    - Selecciona, de entre varios, el nodo que tenga el número de orden especificado empezando por el último.
    - Ejemplos:
      - (//libros)[last()-1]
        - Selecciona el penúltimo nodo de tipo *libros*.
      - (//libros)[last()-1]/titulo/text()
        - Selecciona el título del penúltimo libro (texto del penúltimo nodo *libros*), desde el directorio raíz.
  - **[condición]**
    - Selecciona los nodos que cumplen una determinada condición.
    - Para establecer una condición se usan operadores de diversos tipos.
    - Las condiciones se establecen entre los valores de elementos, atributos o con otras cadenas de texto o números.
    - Los textos se encierran entre comillas simples o dobles, que son opcionales para los números.
    - Un punto (.) hace referencia al elemento actual a la hora de construir una condición.
    - Operadores:

- **Relacionales**:
  - =, >, <, >=, <=, ¡=
- **Lógicos**:
  - and, or y not().
- **Aritméticos**.
  - +, -, *, div, mod.

- Ejemplos:
  - //libros[@idioma =”*klingonés*”]
    - Selecciona los nodos de tipo *libros* desde el elemento raíz, que tengan el atributo idioma con el valor *klingonés*.
  - //libros/[autor =”Lope de Vega”]
    - Selecciona los nodos de tipo *libros* desde el elemento raíz, que tengan el subelemento autor con el valor *Lope de Vega.*
  - //libros/autor[. =”Lope de Vega”]
    - El punto representa al elemento actual que es *autor*.
    - Equivale al ejemplo anterior.
  - //libros/[autor =”J. K. Rowling” and año_publicación=”1997”]
    - 2 condiciones encadenadas con la *y* lógica.
    - Selecciona los nodos de tipo *libros* desde el elemento raíz, cuyo autor sea J. K. Rowling y estén escritos 1997.
  - //libros/[autor =”J. K. Rowling”] [año_publicación=”1997”]
    - Expresión similar a la anterior.
    - Dos condiciones distintas para el mismo nodo equivalen a 2 condiciones encadenadas con la *y* lógica.

SELECCIÓN DE NODOS.

- La selección de nodos se especifica a continuación del eje y del predicado.
- Esta opción permite selecciona elementos, el texto que contienen o ambos.
- También se pueden seleccionar atributos.
- **Opciones**:
  - **/node()**
    - Selecciona todos los hijos (elementos, texto, atributos, etc.), de un nodo.
    - Ejemplos:
      - //libros/node()
        - Selecciona todos los elementos hijos de *libros* (*autor, editorial, titulo, etc*.), con sus textos y los atributos que lleve cada uno.

        - Si autor por ejemplo incluye los elementos *nombre* y *seudónimo*, éstos no son seleccionados.
    - //libros/editorial/node()
        - Si *editorial* no tiene descendientes ni atributos, sólo mostraría el nombre (texto) de las editoriales.
- **//node()**
    - Selecciona todos los descendientes (elementos, texto, atributos, etc.), de un nodo.
    - Ejemplo:
        - //libros//node()
            - Selecciona todos los elementos hijos de *libros* (a*utor, editorial, titulo, etc*.), y lo hijos de éstos, y así sucesivamente, incluyendo todos sus textos y atributos.
- **/text()**
    - Selecciona sólo el texto contenido en un nodo.
    - Podría no mostrar nada si el nodo no contiene texto.
    - Ejemplos:
        - //titulo/text()
            - Selecciona todos los textos de los títulos.
- **//text()**
    - Selecciona sólo el texto contenido en un nodo y los sus elementos descendientes, si los hay.
    - Ejemplo:
        - //libros//text()
            - Selecciona todos los textos de elementos hijos de *libros* (a*utor, editorial, titulo, etc*.), y el de los hijos de éstos, como, por ejemplo, *nombre* y *seudónimo* si fueran subelementos de *autor*.
- **/***
    - Selecciona todos los elementos hijos de un nodo.
    - Ejemplos:
        - //biblioteca/*
            - Selecciona todos los elementos hijos de biblioteca, es decir, *libros*.
            - Los descendientes de libros (a*utor, editorial, titulo, etc*.), no son seleccionados.
- **//***
    - Selecciona todos los elementos descendientes de un nodo.
    - Ejemplo:
        - //biblioteca//*
            - Selecciona todos los elementos hijos de biblioteca (*libros),* así como sus descendientes (a*utor, editorial, titulo, etc*.), lo hijos de éstos últimos, y así sucesivamente, incluyendo todos sus textos y atributos.

FUNCIONES.

- XPath incluye una librería funciones para realizar operaciones con cadenas de caracteres, operaciones numéricas, comparaciones de fechas, etc.
- Web de ejemplo con listado de funciones:
  - https://www.data2type.de/es/xml-xslt-xslfo/xslt/referencia-xslt-y-xpath
- **Algunas funciones.**
  - Funciones para cadenas de caracteres.
    - concat(), substring(), contains(), substring-before(), substring-after(), translate(), normalize-space(), string-length()
  - Funciones para datos numéricos.
    - sum(), count(), round(), floor(), ceiling(), max(), min(), avg(), abs().
  - Funciones para obtener las propiedades de un nodo.
    - name().
  - Funciones de conversión.
    - string(), number(), boolean().