Nombre de la empresa

Descripción generada automáticamente con confianza baja

**Transformación XML a HTML5**

**Dennis Gabriela Coy Calderón**

**Universidad de Oviedo**

**Escuela de Ingeniería Informática**

**Software y Estándares para la Web**

**Descripción**

El objetivo de este documento es explicar la tranformación del archivo XML a un archivo HTML5 por medio del lenguaje de programación Python.

**Estructura del archivo HTML5**

Para empezar, se definió la estructura del documento HTML5. Además de los elementos que componen a cualquier archivo HTML5, este se compondrá de un título como elemento h1 y varios subtítulos h2 en el que se listen los nombres de cada una de las personas que componen el árbol genealógico. Cada persona tendrá sus datos con un subtítulo h3 y la descripción de cada uno dentro de elementos de párrafo p y una imagen. De esta manera:

<h1>Título</h1>

<h2>Persona</h2>

<h3>Datos:</h3>

<p><strong>Lugar de Nacimiento:</strong> …. </p>

…

<img src=”nombreimagen.extension” alt=”Fotografía de …” />

**Construcción del código en Python**

Para esta transformación se hizo uso de Python 3.8 mediante el entorno de desarrollo Spyder.

El código xml2html.py utiliza la librería xml.etree.ElementTree para la conversión del archivo XML en su árbol y así poder recorrer sus elementos como nodos. Está compuesto de las siguientes funciones:

* datosXML(archivoXML)

Esta función recibe como parámetro el archivo XML para extraer sus datos y retornarlo como una lista de diccionarios, en la que cada diccionario será una persona y su información. Detalle:

Paso 1. Se hace uso de la función parse(archivoXML) para convertir el archivo XML en un árbol.

Paso 2. Se obtiene la raíz del árbol mediante la función getroot()

Paso 3. Con la función iter(elemento) se busca a partir de la raíz todos los elementos que sean del mismo tipo, en este caso, elemento será igual a una cadena de caracteres con el nombre que recibe personas en el árbol XML, es decir, ‘{http://www.uniovi.es}persona’. Todo esto dentro de un ciclo para que cuando encuentre todos esos elementos se pueda extraer la información de cada uno de ellos.

Paso 4. Extraer la información de cada persona. Para esto se indica persona[0] para que tome el primer hijo de persona que es datos y con el uso de la función find(elemento) encontrar cada uno de los datos que lo componen, donde elemento es una cadena de caracteres con el nombre que recibe en el árbol XML. Además, se hace uso del atributo .text para obtener el valor del elemento.

Paso 5. Finalmente, se guardan estos datos en un diccionario y luego el diccionario en la lista. Se retorna la lista.

* prologoHTML(archivo, titulo)

Esta función recibe como parámetros el archivo de salida y el titulo que aparecerá en la página, es decir, el texto que hará parte de los elementos title y h1, para escribir el prólogo del archivo con la definición !DOCTYPE html, el elemento html con el lenguaje, el elemento head con los metadatos que describen al documento y su formato de codificación (UTF-8).

* contenidoHTML(archivo, datos, c)

Esta función recibe como parámetros el archivo de salida, el diccionario con los datos de una persona y un contador para escribir en el archivo de salida la información de la persona en el formato explicado anteriormente con elementos h2, h3 y p. El contador es utilizado para extraer el identificador de la persona, es decir, que relación tiene con el nodo principal de un arreglo llamado ID, declarado como constante global.

* añadirContenidoHTML(archivo, data)

Esta función recibe como parámetros el archivo de salida y la lista de diccionarios con los datos de las personas para hacer un ciclo y enviar cada diccionario a la función contenidoHTML.

* epilogoHTML(archivo)

Esta función recibe como parámetro el archivo de salida para escribir los cierres de las etiquetas body, main y html del documento.

Finalmente, en la función main() se recibe el archivo XML de entrada, se pide un nombre de Salida y el nombre que se mostrará en la página. Se hace un llamado del prologoHTML, añadirContenidoHTML y epilogoHTML y se muestra un mensaje de Creación exitosa.