FIT5196
XML Parsing

Introduction

Data from different resources are stored in different formats, such as CSV, XML, JSON, Excel, and PDF files. In this assessment, you are required to write Python script (version 3.7) to convert data stored in an XML file to a JSON file.

Convert an XML file to a JSON file

This task focuses on converting the data stored in an XML file **xml.xml** into a JSON file. The following figure shows what the JSON file should look like.


```
{
  "thesaurus": [
    {
      "Description": "The standard airgun calibre for international target shooting
        .",
      "RelatedTerms": [
        {
          "Relationship": "Narrower Term",
          "Title": "Shooting sport equipment"
        }
      ],
      "Title": ".177 (4.5mm) Airgun"
    },
    {
      "Description": "A rimfire calibre, much used in target shooting and often
        synonymous with the term smallbore.",
      "RelatedTerms": [
        {
          "Relationship": "Narrower Term",
          "Title": "Shooting sport equipment"
        }
      ],
      "Title": ".22"
    },
    {
      "Description": "The standard .22 rimfire cartridge for target rifle and pistol
        use.",
      "RelatedTerms": [
        {
          "Relationship": "Narrower Term",
          "Title": "Shooting sport equipment"
        }
      ],
      "Title": ".22 Long Rifle"
    },
```

There is a sample output file “output_sample.dat” provided in the same folder. You could view the required structure and format from this sample by using any of the text editor or JSON viewer online.

In order to complete this task:

- You must correctly extract the thesaurus in the XML file and store it in a JSON file;
- You are free to use any existing Python packages that are written to parse XML files (e.g., Beautiful-soup, lxml and ElementTree) .
- Python packages, like json, can be used to save the extracted thesaurus;
- Your script must be written in a Jupyter notebook named as "**xml2json.ipynb**";
- The JSON data should be saved in a file named as "**json.dat**";
- The input file must only be "**xml.xml**".

Marking Criteria

- *Mechanical pass:* Your outputs will be compared against the expected output. Therefore, your mark would be based on the similarity between what we expect (as discussed in the instructions) and what we receive from you. It is extremely important to carefully follow the instructions to produce the expected output. Otherwise, you may easily lose points for simple mistakes (eg typos in the format of the files, not loading essential libraries, different file names/path, etc)
- *Expert pass:* Your code will be checked by an expert to validate the logic and flow, proper use of libraries and functions, clarity of codes, comments, structure and presentation.
- The expert will **NOT** fix your code’s problem(s) even if it is a simple typo in a file name or an imported library.

How to Submit

Once you have completed your work, take the following steps to submit it:

1. Rename “rename_me” directory to your student number. This directory should contain your Notebook(s) and output(s).
2. Restart your kernel and rerun your code from beginning to the end to make sure everything works as expected.
3. Save your notebook as .ipynb file once again while you are keeping all the outputs (eg print outputs) in the notebook.

4. Make sure the output files are properly named and your code does not produce any unnecessary file.
5. Zip the containing folder with the same name (ie <student_numnber>.zip) and upload it.

Good luck!