Project

Due: Friday, December 10, 2021 at 8:00 p.m. **Points:** 100

In class we went through a program, `getidlist.py`, to produce a list of publication IDs from a keyword search on PubMed. The final project is to produce a list of the publication citations for that keyword.

Homework 4 had you produce a list of publication IDs from a keyword search on PubMed. The final project is to produce a list of the publication citations for that keyword.

Begin with that program. First, modify it so that after it requests the keywords, it asks the user how many references to print. Currently, ir will print 3; look for the variable `numret`. You will have to read in an integer and assign that value to `numret`. Be sure the integer is a positive number!

The program will print out a query string that is a URL ofa list of PubMed publication IDs. Use that URL to get the metadata. The web page you get back is an XML document giving details of the publications.

Your job is to print a bibliography from this record. Your entry for each journal should look like this:

A. Bester, R. Zelazny, and H. Ellison, "On the Role of Viruses in Future Epidemics," Journal of Irreproducible Results 3(4) pp. 29–35 (Mar. 2103). PUBMED: 23456789; DOI 12.1119/2847595.

Then print the abstract, if it is present in the record.

If there is no DOI, use the PII. If neither is there, omit that part of the entry.

You will need to look at the XML records to get the fields. These are delimited by tags with attributes, each of which may have a value. For example, the element

```
<ELocationID EIdType="doi" ValidYN="Y">10.1016/j.vaccine.2015.04.071</ELocationID>
```

has a tag of `ELocationID`, attributes of `EIdType` (with value `doi`) and `ValidYN` (with a value of `Y`), and the field contains `10.1016/j.vaccine.2015.04.071`, which (as the EIdType value indicates) is a DOI.

The easiest way to see what the records look like is to run `getidlist.py` and ask for a single entry. You can then see its structure (you might find the prettyprinter `xmlpp.py`. The fields of interest will have these tags:

- Article — contains the Journal, ArticleTitle (article title), Pagination (page numbers), ElocationID, which gives both the DOI and PII (if those exist), the Abstract, and the AuthorList.
- Journal — this consists of several elements, including JournalIssue, which contains the Volume, Issue, and PubDate (publication date), and Title (article title).
- AuthorList — this lists the authors, each author being in a field called Author. Subfields of interest are LastName and Initial (the initial of the first name)

Those will be enough to build the reference, as described above.

You can find methods for processing XML in the Python Library Reference at `https://docs.python.org/3.7/library/xml.etree.elementtree.html`