trial

Release v.1.0.0

celine

May 02, 2022

CONTENTS:

Welcome to the documentation of RSS-reader.

RSS (RDF Site Summary or Really Simple Syndication) is a web feed that allows users and applications to access updates to websites in a standardized, computer-readable format. Subscribing to RSS feeds can allow a user to keep track of many different websites in a single news aggregator, which constantly monitor sites for new content, removing the need for the user to manually check them. News aggregators (or "RSS readers") can be built into a browser, installed on a desktop computer, or installed on a mobile device.[Wikipedia]

This RSS-reader comes in a command line interface format, and provided the RSS link of a webpage, will provide the RSS feeds of that webpage in two formats, including a JSON form. The user has the option to decide how many feeds they are willing to get and based on this option, they will be shown that many feeds.

CHAPTER

ONE

HOW TO USE THE RSS-READER

The RSS-reader is represented in CMI form, by running the program, user can enter the RSS of a webpage and the number of feeds they have selected as the option.

Below is demonstrated the help of the program

```
(iter1) E:\EPAM\Exercises\iter1>py rss.1.6.py --help
usage: rss.1.6.py [-h] [--version] [--json] [--verbose] [--limit LIMIT] [source]

Pure Python command-line RSS reader.

positional arguments:
  source         RSS URL

optional arguments:
  -h, --help     show this help message and exit
  --version      Print version info
  --json         Print result as JSON in stdout
  --verbose      Outputs verbose status messages
  --limit LIMIT  Limit news topics if this parameter provided

(iter1) E:\EPAM\Exercises\iter1>
```

As it is mentioned in the help of the program, user have multiple options as follows:

`help` which displays the options, flags and arguments for the program.
`version` displays the version in which the program comes in If this option is selected, only the version will be printed.
`json` if selected, the news feed will be printed in json format
`verbose` if selected, the user will be informed by the logs shown to them that how the program progresses
`limit` the number of news feeds the user is willing to be shown. If not specified, all the feeds available on the RSS will be output.

And as argument, it is required to input an RSS URL.

`source` the RSS URL which the user expects to see feeds from
`date` the date specified by the user to be shown the feed from that date
`--to-html` path to which the HTML file should be stored
`--to-pdf` path to which the PDf file should be stored
`--colorize` prints the output in colorized format

If the distribution is installed, program can also be run using the rss_reader command, without needing the py command.

Below is illustrate an example of running the program with multiple options:

```
(RSS-parser) E:\EPAM\Exercises\final\first_iteration\RSS-parser\source>py rss.1.0.0.py https://news.yahoo.com/rss/ --verbose --limit 2
INFO:Verbose is set to ON!
INFO:Requesting the URL webpage
INFO:Webpage retrieved successfully!
INFO:RSS file Parsed successfully!
INFO:Getting the content of the news feed.
INFO:Printing the RSS feed
--------------------------------------------------------------------------------
Title: What are howitzers? A look at the cannons in latest U.S. military aid to Ukraine
Date: Fri, 22 Apr 2022 13:22:19 +0000
Link: https://news.yahoo.com/what-are-howitzers-the-weapons-included-in-us-military-aid-to-ukraine-132219093.html

[image: What are howitzers? A look at the cannons in latest U.S. military aid to Ukraine][2]
The first shipments of the Biden administration's $800 million military aid package have arrived in Ukraine. Included
among the first round of weapons are 18 155 mm howitzers, in addition to another 72 cannons that were announced this
week. The howitzers heading to Ukraine will have a "significant" impact on Ukrainian firepower, according to a senior
U.S. defense official, as the war with Russia enters its third month.[1]


[1]: https://news.yahoo.com/what-are-howitzers-the-weapons-included-in-us-military-aid-to-ukraine-132219093.html (link)
[2]: https://s.yimg.com/os/creatr-uploaded-images/2022-04/e42f2010-c23d-11ec-bf5b-04cafe848750 (image)


INFO:Getting the content of the news feed.
--------------------------------------------------------------------------------
Title: 'It wasn't a good feeling': An NFL player says he was turned away from a French restaurant in Atlanta due to his attire
Date: Sat, 23 Apr 2022 17:20:41 +0000
Link: https://news.yahoo.com/wasnt-good-feeling-nfl-player-172041099.html

[image: 'It wasn't a good feeling': An NFL player says he was turned away from a French restaurant in Atlanta due to his attire][2]
Atlanta Falcons player Grady Jarrett was turned away from Le Bilboquet while wearing a Gucci tracksuit worth thousand of
dollars and a tennis chain.[1]


[1]: https://news.yahoo.com/wasnt-good-feeling-nfl-player-172041099.html (link)
[2]: https://s.yimg.com/uu/api/res/1.2/CPTmegJnSbU_KGuImu_8kA--~B/aD0zMDE1O3c9NDAyMDthcHBpZD15dGFjaHlvbg--/https://media.zenf
s.com/en/insider_articles_922/89e402822a958519e8ce41ed73914fef (image)
```

CHAPTER

TWO

NEWS FEED IN JSON FORMAT

If the user selects the option of being presented the feed in json format, depending on the –limit option, they will be shown that number of feeds in json format.

The JSON fomat implemented for the RSS-reader is as follows: The feeds based on their relevence on the RSS file will be numbered starting from 1 and the output would appear in the followig form:

```
{
        "News Number n":{
        "title":
        "date":
        "link":
        "content":
        "image link":
        }
}
```

with corresponding info about each news in front of the fields.

Below is an actual ouput of the news feed in JSON format is represented:


```
{
    "News Number 1": {
        "title": "What are howitzers? A look at the cannons in latest U.S. military aid to Ukraine"
        "date": "Fri, 22 Apr 2022 13:22:19 +0000"
        "link": "https://news.yahoo.com/what-are-howitzers-the-weapons-included-
                in-us-military-aid-to-ukraine-132219093.html"
        "content": "The first shipments of the Biden administration's $800 million
                military aid package have arrived in Ukraine. Included among the
                first round of weapons are 18 155 mm howitzers, in addition to
                another 72 cannons that were announced this week. The howitzers
                heading to Ukraine will have a "significant" impact on Ukrainian
                firepower, according to a senior U.S. defense official, as the
                war with Russia enters its third month."
        "image link": "https://s.yimg.com/os/creatr-uploaded-
                images/2022-04/e42f2010-c23d-11ec-bf5b-04cafe848750"
    },

    "News Number 2": {
        "title": "'It wasn't a good feeling': An NFL player says he was turned away from a French restaurant in Atlanta due to his attire"
        "date": "Sat, 23 Apr 2022 17:20:41 +0000"
        "link": "https://news.yahoo.com/wasnt-good-feeling-nfl-
                player-172041099.html"
        "content": "Atlanta Falcons player Grady Jarrett was turned away from Le
                Bilboquet while wearing a Gucci tracksuit worth thousand of
                dollars and a tennis chain."
        "image link": "https://s.yimg.com/uu/api/res/1.2/CPTmegJnSbU_KGuImu_8kA--~B/aD0
                zMDE1O3c9NDAyMDthcHBpZD15dGFjaHlvbg--/https://media.zenfs.com/en
                /insider_articles_922/89e402822a958519e8ce41ed73914fef"
    },
}
```

CHAPTER

THREE

CACHING THE NEWS

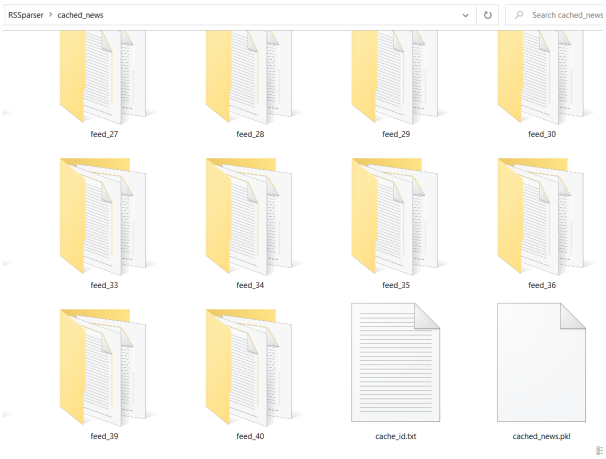
When the user inputs a RSS URL, and no date is entered, the rss-reader fetches the feed items from the specified source and prints it in normal or json format, based on the options selected. While doing this, it also caches the read news.

The utility caches the feeds data as follows: When a feed is read, a dictionary of the feed's information is created, storing its title, date, content, news link and image's link, the RSS source and a path to the feed's cache directory. The utility creates a cache directory in the cached_news folder for each feed. In the feed's directory, the article of the feed from its news page is downloaded in a text file, the links in that article are extracted and stored in a text file and the images in the article are downloaded in another directory named "images" in the feed's directory. This is done for when the utility wants to convert the feeds into HTML or PDF.

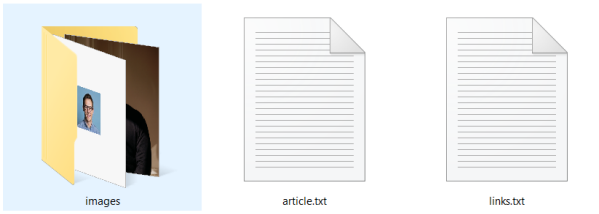
Then for each feed, a tuple is constructed, first element being the news date and the second one the previously mentioned dictionary and all tuples each corresponding to one feed are stored in a list that is saved in a file in the cached_news directory. The cached news are fetched by the news date, hence this implementation is designed that is demonstrated in the image below.

```
[("feed number 1 date", {"title": "news feed title",
                         "date": "feed's publish date",
                         "link": "link to the feed's news page",
                         "content": "feed's content",
                         "image_link": "image_link",    # if exists
                         "cache_directory": "path to the cache directory",
                         "rss_source": "RSS URL entered by user"}), ("feed number 2", {...})]
```

The cached_news directory would look like this:



And inside each feed's directory, would look like this:


images
article.txt
links.txt

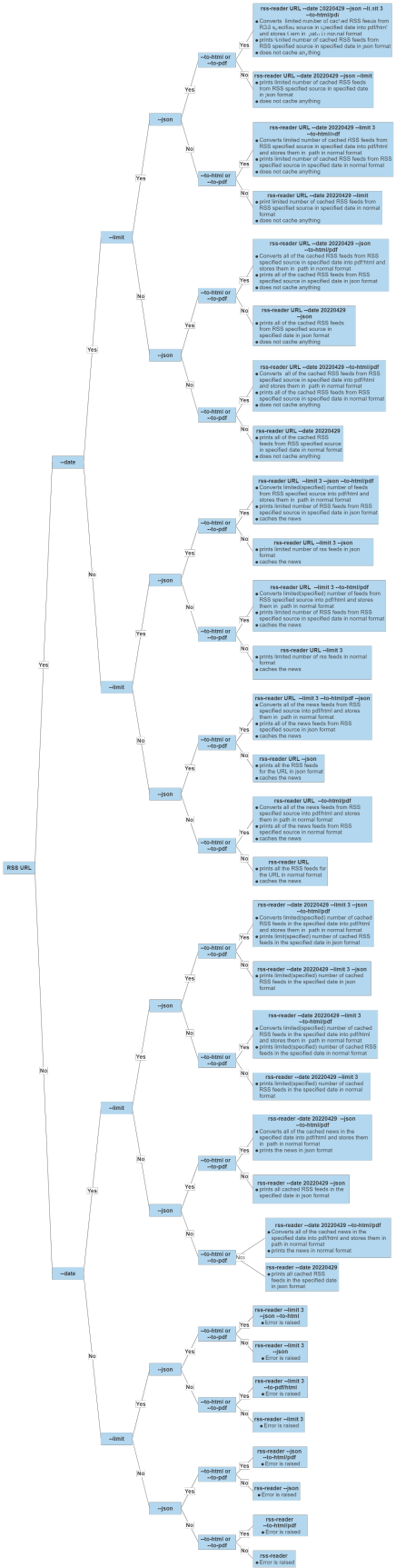
CHAPTER

FOUR

HOW RSS-READER UTILITY WORKS

- If the user selects –version option as an argument, the version of the utility will be printed and the program will be ended.
- If the user selects –verbose option, verbose will be printed in the stdout
- If the user selects –colorize option, the output on stoud would be printed in colorful format.
- If the user doesn't enter neither RSS URL nor –date, an error will be raised.

Otherwise, the behavior of the utility would be as it is illustrated in the diagram below:


RSS URL
--date
--limit
--json
--to-html or --to-pdf
rss-reader URL --date 20220429 --json --limit 3 --to-html/pdf
rss-reader URL --date 20220429 --json --limit
rss-reader URL --date 20220429 --limit 3 --to-html/pdf
rss-reader URL --date 20220429 --limit
rss-reader URL --date 20220429 --json --to-html/pdf
rss-reader URL --date 20220429 --json
rss-reader URL --date 20220429 --to-html/pdf
rss-reader URL --date 20220429
rss-reader URL --limit 3 --json --to-html/pdf
rss-reader URL --limit 3 --json
rss-reader URL --limit 3 --to-html/pdf
rss-reader URL --limit 3
rss-reader URL --limit 3 --to-html/pdf --json
rss-reader URL --json
rss-reader URL --to-html/pdf
rss-reader URL
rss-reader --date 20220429 --limit 3 --json --to-html/pdf
rss-reader --date 20220429 --limit 3 --json
rss-reader --date 20220429 --limit 3 --to-html/pdf
rss-reader --date 20220429 --limit 3
rss-reader --date 20220429 --json --to-html/pdf
rss-reader --date 20220429 --json
rss-reader --date 20220429 --to-html/pdf
rss-reader --date 20220429
rss-reader --limit 3 --json --to-html
rss-reader --limit 3 --json
rss-reader --limit 3 --to-pdf/html
rss-reader --limit 3
rss-reader --json --to-html/pdf
rss-reader --json
rss-reader --to-html/pdf
rss-reader
Error is raised

CHAPTER

FIVE

USED LIBRARIES IN THE PROJECT

For the implementation of this project, a number of libraries have been made use of. The most important of them are as follows: | These libraries are as follows:

`argparse` for parsing the arguments of the CMI
`xml.etree.ElementTree` for parsing the XML file of RSS into XML objects
`requests` for fetching web pages
`json` for converting the dictionary into json format This library has used in the second approach for this goal, and is not in the initial implementation (in the commented section)
`logging` for logging info/warning/error messages when the verbose option is set to on
`re` for regular expression operations
`datetime` and `dateutil` for the date format conversions
`textwrap` for wrapping the text in 120 characters format
`pickle` for data serialization during caching process
`reportlab` for producing PDF documents
`BeautifulSoup` for parsing HTML content into elements

CHAPTER

SIX

NOTES FOR THE REVIEWERS

Dear reviewers, during the implementation of this project, I faced a few vague points and complications that I will present in this section, some of them for the purpose of clarification.

6.1 pycodestyle errors

In the output of the pycodestyle, there were few **too many blank lines** error. They were regarding the 2 blank lines I surrounded my module functions with, according the pep-8 guideline.

6.2 Future works

I will be improving this project every time I have time, as I have learned a lot from this project and I am still learning. My next step is going to implement test, as this step was not mandatory. Then I will be working on the implementation of the 6th iteration.