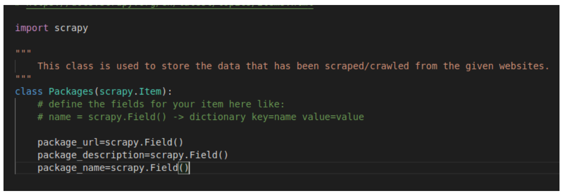
Documentation

Aabishkar Karki
karkiaabishkar@gmail.com

April 5 2021

To create the crawler project I have used scrapy framework from python which helps in crawling data from websites easily. Explanations of the functions and classes can be found in the comments of the code. In this documentation I have just highlighted the important parts of the project.

When the project is run it goes to "packages_spider.py" inside the spiders folder. This file contains the definition of two crawlers (spiders) that are going to be used to crawl data from those two websites. To store the crawled/scraped data a container class [Figure 1] has been created in "items.py" file that is inside Crawler/Crawler.



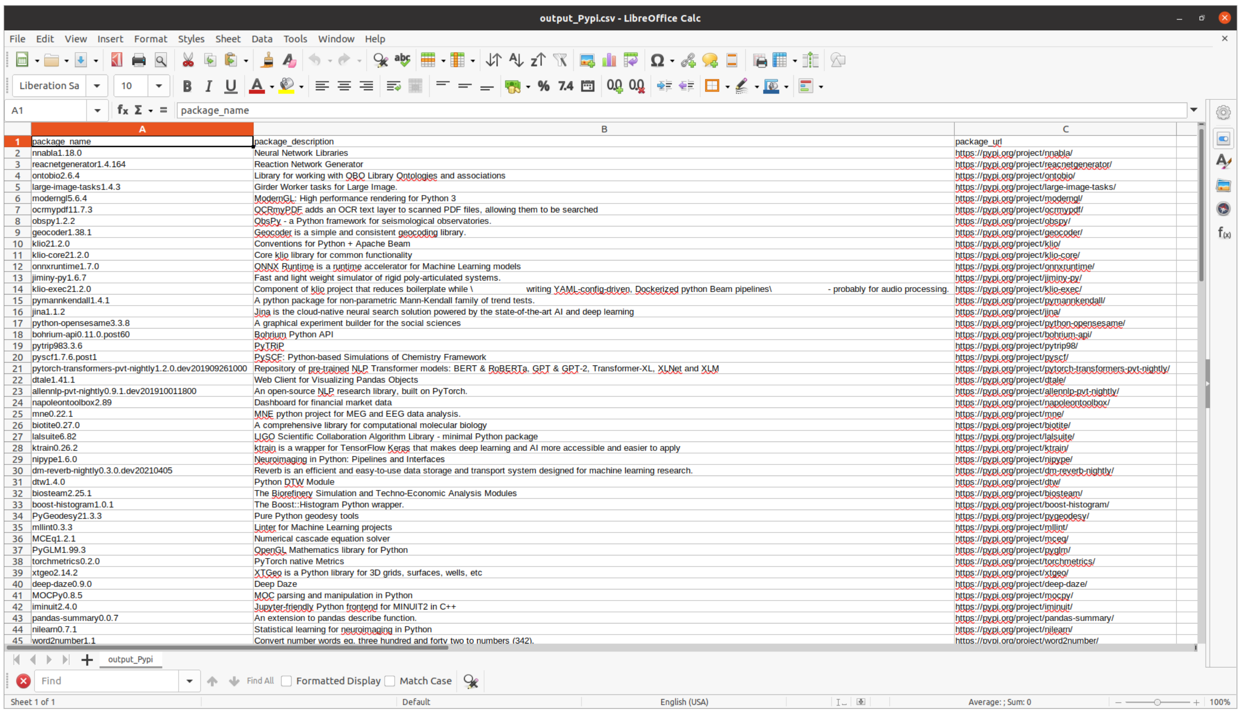
```python
import scrapy

"""
    This class is used to store the data that has been scraped/crawled from the given websites.
"""
class Packages(scrapy.Item):
    # define the fields for your item here like:
    # name = scrapy.Field() -> dictionary key=name value=value

    package_url=scrapy.Field()
    package_description=scrapy.Field()
    package_name=scrapy.Field()
```

Figure 1: Class to store crawled data

Each of the crawler has been given a name so once the Crawler is activated using its name it opens the links that are present in its 'start_urls' property. Both of the crawler have been given only one start url that opens the first page of each websites list of packages. After that the parse function is automatically called by scrappy and using "XPATH" the relevant details are extracted/crawled from the webpage like package name, url and descriptions and stored in the Class that was created earlier. Since both the websites need to extract same data the same class has been used in both of the crawler to store data. The parse function also checks whether we have extracted 100 package datas from the website. Once 100 packages are extracted the function ends. If the initial page contains < 100 packages then the url of the next page is extracted using XPATH and then the 'parse' function is called on that url and again the same process is repeated.

The output of both the crawler can be seen in Figure 2 and Figure 3. For the project description I have extracted the project description that was written in the main web page that contained the list of packages. I tried extracting the project description by opening each of the package's url but the description was too large hence could not be easily stored in tables like this. Output of Unittests can be seen in Figure 4. In this unittests I have only tested the parse function of the crawler as it is the most important part of the program.
New websites can be easily scraped using this project if similar data needs to be extracted. We just need to add another crawler /spider class that does the scraping. However for website where we need to extract a variety of other data a new Class has to be created to store the data but the other processes are still the same and we just need to write a different "parse" function.
Resources that have been used in the project:

- Scrapy
- Xpath
- Unittests

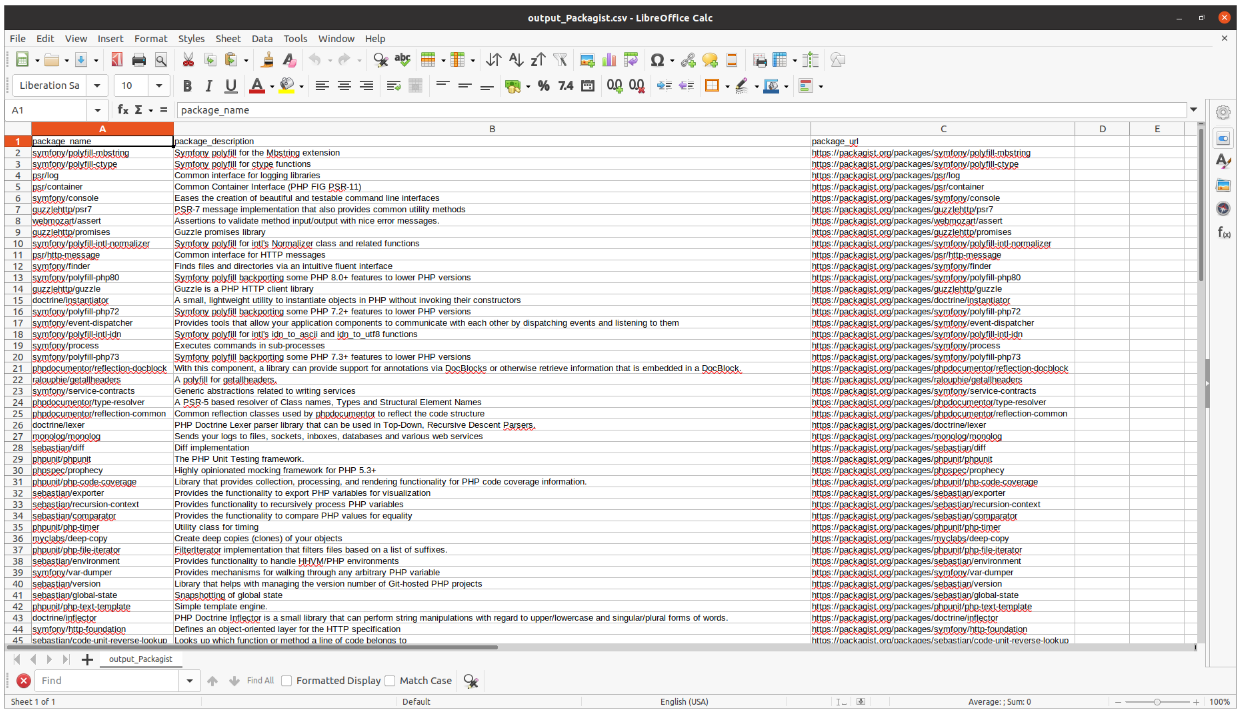


output_Pypi.csv - LibreOffice Calc

A1 package_name

	A	B	C
1	package_name	package_description	package_url
2	nnabla1.18.0	Neural Network Libraries	https://pypi.org/project/nnabla/
3	reacnetgenerator1.4.164	Reaction Network Generator	https://pypi.org/project/reacnetgenerator/
4	ontobio2.6.4	Library for working with OBO Library Ontologies and associations	https://pypi.org/project/ontobio/
5	large-image-tasks1.4.3	Girder Worker tasks for Large Image.	https://pypi.org/project/large-image-tasks/
6	moderngl5.6.4	ModernGL: High performance rendering for Python 3	https://pypi.org/project/moderngl/
7	ocrmypdf11.7.3	OCRmyPDF adds an OCR text layer to scanned PDF files, allowing them to be searched	https://pypi.org/project/ocrmypdf/
8	obspy1.2.2	ObsPy - a Python framework for seismological observatories.	https://pypi.org/project/obspy/
9	geocoder1.38.1	Geocoder is a simple and consistent geocoding library.	https://pypi.org/project/geocoder/
10	klio21.2.0	Conventions for Python + Apache Beam	https://pypi.org/project/klio/
11	klio-core21.2.0	Core klio library for common functionality	https://pypi.org/project/klio-core/
12	onnxruntime1.7.0	ONNX Runtime is a runtime accelerator for Machine Learning models	https://pypi.org/project/onnxruntime/
13	jiminy-py1.6.7	Fast and light weight simulator of rigid poly-articulated systems.	https://pypi.org/project/jiminy-py/
14	klio-exec21.2.0	Component of klio project that reduces boilerplate while \ writing YAML-config-driven, Dockerized python Beam pipelines\ - probably for audio processing.	https://pypi.org/project/klio-exec/
15	pymannkendall1.4.1	A python package for non-parametric Mann-Kendall family of trend tests.	https://pypi.org/project/pymannkendall/
16	jina1.1.2	Jina is the cloud-native neural search solution powered by the state-of-the-art AI and deep learning	https://pypi.org/project/jina/
17	python-opensesame3.3.8	A graphical experiment builder for the social sciences	https://pypi.org/project/python-opensesame/
18	bohrium-api0.11.0.post60	Bohrium Python API	https://pypi.org/project/bohrium-api/
19	pytrip983.3.6	PyTRiP	https://pypi.org/project/pytrip98/
20	pyscf1.7.6.post1	PySCF: Python-based Simulations of Chemistry Framework	https://pypi.org/project/pyscf/
21	pytorch-transformers-pvt-nightly1.2.0.dev201909261000	Repository of pre-trained NLP Transformer models: BERT & RoBERTa, GPT & GPT-2, Transformer-XL, XLNet and XLM	https://pypi.org/project/pytorch-transformers-pvt-nightly/
22	dtale1.41.1	Web Client for Visualizing Pandas Objects	https://pypi.org/project/dtale/
23	allennlp-pvt-nightly0.9.1.dev201910011800	An open-source NLP research library, built on PyTorch.	https://pypi.org/project/allennlp-pvt-nightly/
24	napoleontoolbox2.89	Dashboard for financial market data	https://pypi.org/project/napoleontoolbox/
25	mne0.22.1	MNE python project for MEG and EEG data analysis.	https://pypi.org/project/mne/
26	biotite0.27.0	A comprehensive library for computational molecular biology	https://pypi.org/project/biotite/
27	lalsuite6.82	LIGO Scientific Collaboration Algorithm Library - minimal Python package	https://pypi.org/project/lalsuite/
28	ktrain0.26.2	ktrain is a wrapper for TensorFlow Keras that makes deep learning and AI more accessible and easier to apply	https://pypi.org/project/ktrain/
29	nipype1.6.0	Neuroimaging in Python: Pipelines and Interfaces	https://pypi.org/project/nipype/
30	dm-reverb-nightly0.3.0.dev20210405	Reverb is an efficient and easy-to-use data storage and transport system designed for machine learning research.	https://pypi.org/project/dm-reverb-nightly/
31	dtw1.4.0	Python DTW Module	https://pypi.org/project/dtw/
32	biosteam2.25.1	The Biorefinery Simulation and Techno-Economic Analysis Modules	https://pypi.org/project/biosteam/
33	boost-histogram1.0.1	The Boost::Histogram Python wrapper.	https://pypi.org/project/boost-histogram/
34	PyGeodesy21.3.3	Pure Python geodesy tools	https://pypi.org/project/pygeodesy/
35	mllint0.3.3	Linter for Machine Learning projects	https://pypi.org/project/mllint/
36	MCEq1.2.1	Numerical cascade equation solver	https://pypi.org/project/mceq/
37	PyGLM1.99.3	OpenGL Mathematics library for Python	https://pypi.org/project/pyglm/
38	torchmetrics0.2.0	PyTorch native Metrics	https://pypi.org/project/torchmetrics/
39	xtgeo2.14.2	XTGeo is a Python library for 3D grids, surfaces, wells, etc	https://pypi.org/project/xtgeo/
40	deep-daze0.9.0	Deep Daze	https://pypi.org/project/deep-daze/
41	MOCPy0.8.5	MOC parsing and manipulation in Python	https://pypi.org/project/mocpy/
42	iminuit2.4.0	Jupyter-friendly Python frontend for MINUIT2 in C++	https://pypi.org/project/iminuit/
43	pandas-summary0.0.7	An extension to pandas describe function.	https://pypi.org/project/pandas-summary/
44	nilearn0.7.1	Statistical learning for neuroimaging in Python	https://pypi.org/project/nilearn/
45	word2number1.1	Convert number words eg. three hundred and forty two to numbers (342).	https://pypi.org/project/word2number/

Figure 2: Output of First Crawler

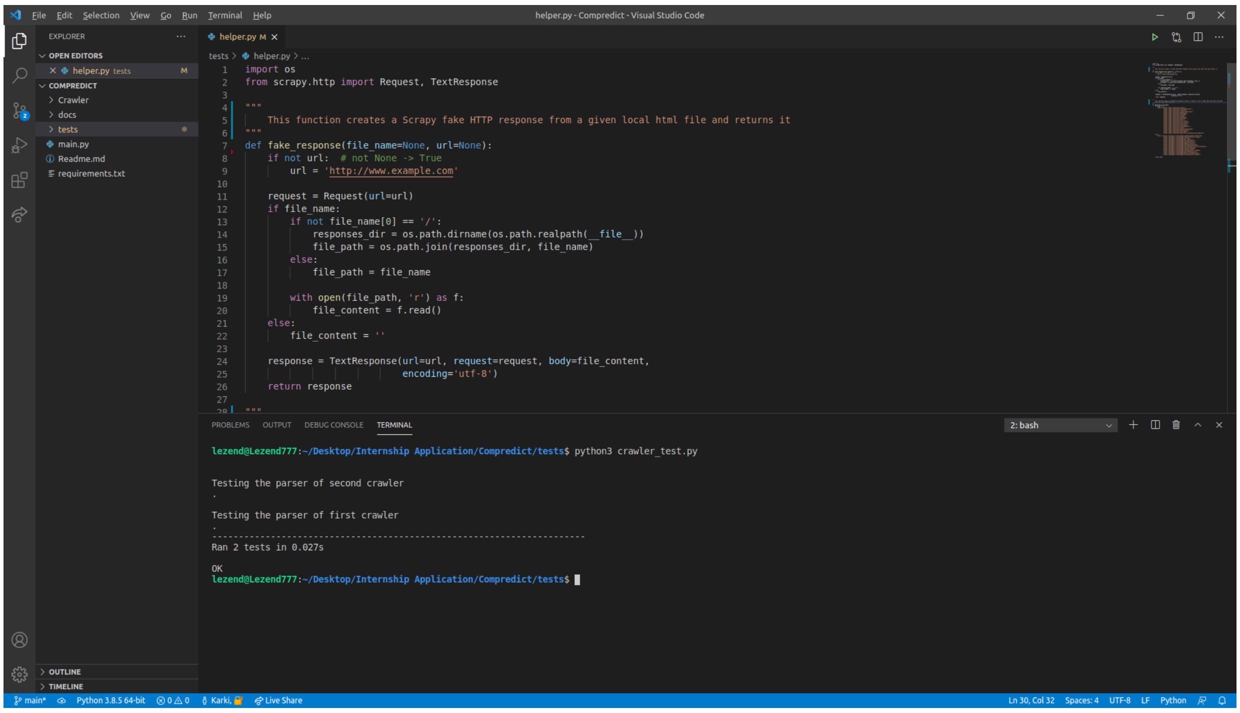


output_Packagist.csv - LibreOffice Calc

A1 package_name

	A	B	C	D	E
1	package_name	package_description	package_url		
2	symfony/polyfill-mbstring	Symfony polyfill for the Mbstring extension	https://packagist.org/packages/symfony/polyfill-mbstring		
3	symfony/polyfill-ctype	Symfony polyfill for ctype functions	https://packagist.org/packages/symfony/polyfill-ctype		
4	psr/log	Common interface for logging libraries	https://packagist.org/packages/psr/log		
5	psr/container	Common Container Interface (PHP FIG PSR-11)	https://packagist.org/packages/psr/container		
6	symfony/console	Eases the creation of beautiful and testable command line interfaces	https://packagist.org/packages/symfony/console		
7	guzzlehttp/psr7	PSR-7 message implementation that also provides common utility methods	https://packagist.org/packages/guzzlehttp/psr7		
8	webmozart/assert	Assertions to validate method input/output with nice error messages.	https://packagist.org/packages/webmozart/assert		
9	guzzlehttp/promises	Guzzle promises library	https://packagist.org/packages/guzzlehttp/promises		
10	symfony/polyfill-intl-normalizer	Symfony polyfill for intl's Normalizer class and related functions	https://packagist.org/packages/symfony/polyfill-intl-normalizer		
11	psr/http-message	Common interface for HTTP messages	https://packagist.org/packages/psr/http-message		
12	symfony/finder	Finds files and directories via an intuitive fluent interface	https://packagist.org/packages/symfony/finder		
13	symfony/polyfill-php80	Symfony polyfill backporting some PHP 8.0+ features to lower PHP versions	https://packagist.org/packages/symfony/polyfill-php80		
14	guzzlehttp/guzzle	Guzzle is a PHP HTTP client library	https://packagist.org/packages/guzzlehttp/guzzle		
15	doctrine/instantiator	A small, lightweight utility to instantiate objects in PHP without invoking their constructors	https://packagist.org/packages/doctrine/instantiator		
16	symfony/polyfill-php72	Symfony polyfill backporting some PHP 7.2+ features to lower PHP versions	https://packagist.org/packages/symfony/polyfill-php72		
17	symfony/event-dispatcher	Provides tools that allow your application components to communicate with each other by dispatching events and listening to them	https://packagist.org/packages/symfony/event-dispatcher		
18	symfony/polyfill-intl-idn	Symfony polyfill for intl's idn_to_ascii and idn_to_utf8 functions	https://packagist.org/packages/symfony/polyfill-intl-idn		
19	symfony/process	Executes commands in sub-processes	https://packagist.org/packages/symfony/process		
20	symfony/polyfill-php73	Symfony polyfill backporting some PHP 7.3+ features to lower PHP versions	https://packagist.org/packages/symfony/polyfill-php73		
21	phpdocumentor/reflection-docblock	With this component, a library can provide support for annotations via DocBlocks or otherwise retrieve information that is embedded in a DocBlock.	https://packagist.org/packages/phpdocumentor/reflection-docblock		
22	ralouphie/getallheaders	A polyfill for getallheaders.	https://packagist.org/packages/ralouphie/getallheaders		
23	symfony/service-contracts	Generic abstractions related to writing services	https://packagist.org/packages/symfony/service-contracts		
24	phpdocumentor/type-resolver	A PSR-5 based resolver of Class names, Types and Structural Element Names	https://packagist.org/packages/phpdocumentor/type-resolver		
25	phpdocumentor/reflection-common	Common reflection classes used by phpdocumentor to reflect the code structure	https://packagist.org/packages/phpdocumentor/reflection-common		
26	doctrine/lexer	PHP Doctrine Lexer parser library that can be used in Top-Down, Recursive Descent Parsers.	https://packagist.org/packages/doctrine/lexer		
27	monolog/monolog	Sends your logs to files, sockets, inboxes, databases and various web services	https://packagist.org/packages/monolog/monolog		
28	sebastian/diff	Diff implementation	https://packagist.org/packages/sebastian/diff		
29	phpunit/phpunit	The PHP Unit Testing framework.	https://packagist.org/packages/phpunit/phpunit		
30	phpspec/prophecy	Highly opinionated mocking framework for PHP 5.3+	https://packagist.org/packages/phpspec/prophecy		
31	phpunit/php-code-coverage	Library that provides collection, processing, and rendering functionality for PHP code coverage information.	https://packagist.org/packages/phpunit/php-code-coverage		
32	sebastian/exporter	Provides the functionality to export PHP variables for visualization	https://packagist.org/packages/sebastian/exporter		
33	sebastian/recursion-context	Provides functionality to recursively process PHP variables	https://packagist.org/packages/sebastian/recursion-context		
34	sebastian/comparator	Provides the functionality to compare PHP values for equality	https://packagist.org/packages/sebastian/comparator		
35	phpunit/php-timer	Utility class for timing	https://packagist.org/packages/phpunit/php-timer		
36	myclabs/deep-copy	Create deep copies (clones) of your objects	https://packagist.org/packages/myclabs/deep-copy		
37	phpunit/php-file-iterator	FilterIterator implementation that filters files based on a list of suffixes.	https://packagist.org/packages/phpunit/php-file-iterator		
38	sebastian/environment	Provides functionality to handle HHVM/PHP environments	https://packagist.org/packages/sebastian/environment		
39	symfony/var-dumper	Provides mechanisms for walking through any arbitrary PHP variable	https://packagist.org/packages/symfony/var-dumper		
40	sebastian/version	Library that helps with managing the version number of Git-hosted PHP projects	https://packagist.org/packages/sebastian/version		
41	sebastian/global-state	Snapshotting of global state	https://packagist.org/packages/sebastian/global-state		
42	phpunit/php-text-template	Simple template engine.	https://packagist.org/packages/phpunit/php-text-template		
43	doctrine/inflector	PHP Doctrine Inflector is a small library that can perform string manipulations with regard to upper/lowercase and singular/plural forms of words.	https://packagist.org/packages/doctrine/inflector		
44	symfony/http-foundation	Defines an object-oriented layer for the HTTP specification	https://packagist.org/packages/symfony/http-foundation		
45	sebastian/code-unit-reverse-lookup	Looks up which function or method a line of code belongs to	https://packagist.org/packages/sebastian/code-unit-reverse-lookup		

Figure 3: Output of Second Crawler



```python
import os
from scrapy.http import Request, TextResponse

"""
    This function creates a Scrapy fake HTTP response from a given local html file and returns it
"""
def fake_response(file_name=None, url=None):
    if not url:  # not None -> True
        url = 'http://www.example.com'

    request = Request(url=url)
    if file_name:
        if not file_name[0] == '/':
            responses_dir = os.path.dirname(os.path.realpath(__file__))
            file_path = os.path.join(responses_dir, file_name)
        else:
            file_path = file_name

        with open(file_path, 'r') as f:
            file_content = f.read()
    else:
        file_content = ''

    response = TextResponse(url=url, request=request, body=file_content,
                            encoding='utf-8')
    return response

"""
```

```
lezend@Lezend777:~/Desktop/Internship Application/Compredict/tests$ python3 crawler_test.py


Testing the parser of second crawler
.

Testing the parser of first crawler
.
----------------------------------------------------------------------
Ran 2 tests in 0.027s

OK
lezend@Lezend777:~/Desktop/Internship Application/Compredict/tests$
```

Figure 4: Output of Unittests