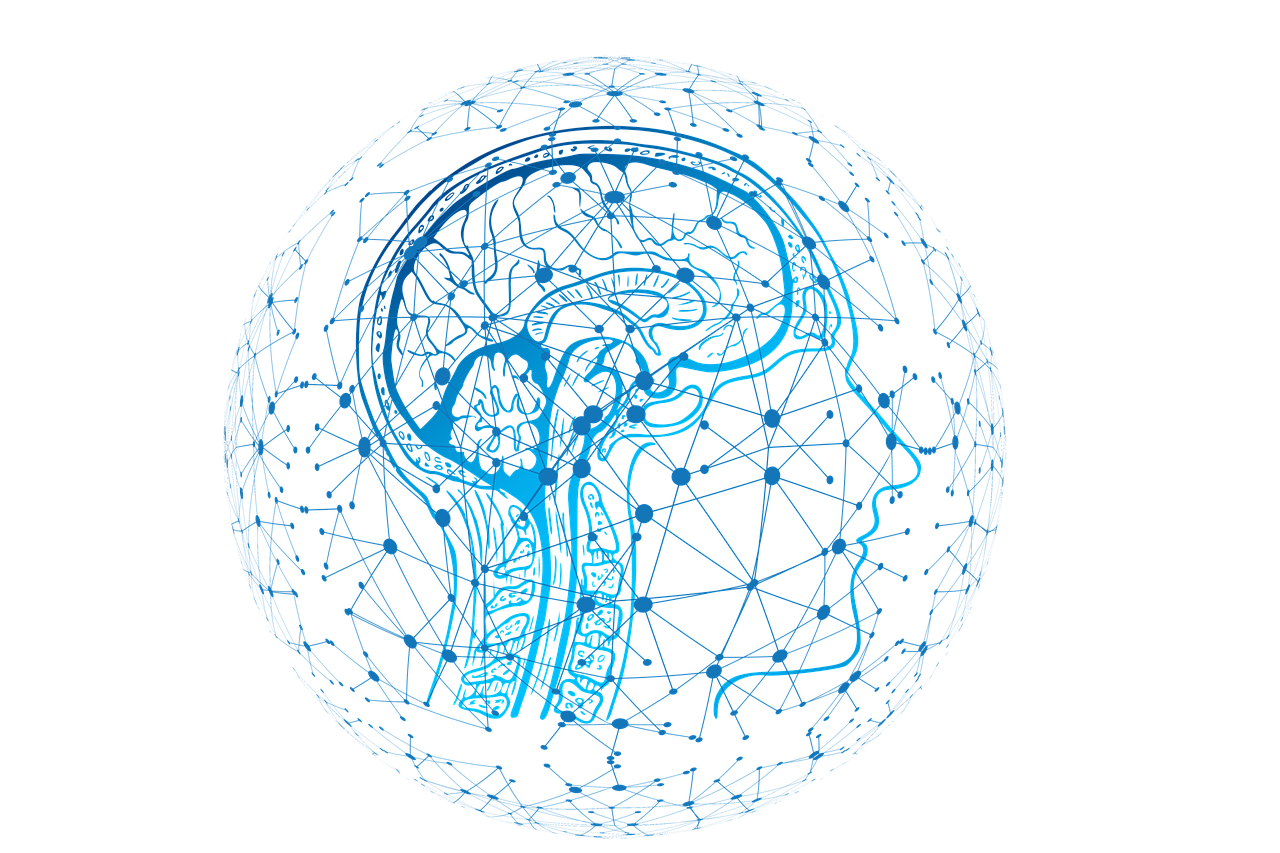
***Artificial Intelligence for Knowledge Discovery***



Une image contenant personne, cravate, homme, portant

Description générée automatiquement

Kevin **JUBERT**

*Master II – BI & Analytics*

# Python-Logo-PNG-ImageJupyter — WikipédiaCreation of the dataset : « AIKD\_project.csv »

The articles’s datas collected for my project are coming from news website named Silicon, having for topic actualites about the data universe. You can find bellow the url link of this website :

<https://www.silicon.fr/actualites/data-storage>

To collect this data and create a CSV file of 50 rows I‘ve coded a Python program who scrap data from the Sillicon website to generate a data file in accordance with the specs of the project. The dataset generated is complete and named « AIKD\_project.csv » in the folllowing path of the folder :

**Projet - AIKD Kevin JUBERT M2\code\Python**

In this path their are two python codes the first one in Jupyter format untitled ‘Article Exctractor.ipynb’ and the second one python script format of this note book untitled ‘Article Exctractor.py’. I’ve coded on the Jupyter notebook environment then I parsed it into a script (because .py code are easier to execute with a git bash script).

Une image contenant texte

Description générée automatiquement

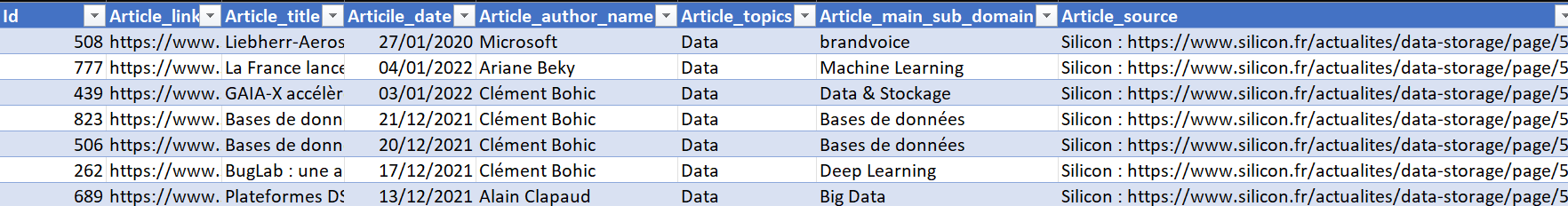
The code uses the Python library « BeautifulSoup » to scrap data from the [first Silicon website](https://www.silicon.fr/actualites/data-storage) where I collect the Title, the domain and the link of the article.

Une image contenant texte

Description générée automatiquement

Une image contenant texte

Description générée automatiquementThen I’ve web scrapped and collected consecutively the links of articles the authors fullname and the dates of publication.

I have generated a dataframe with the pandas library that I’ve filtered to keep only the full lines and limited it to only 50 rows however it could generate more than 275 articles if you by simply modify the numberOfPage variable value to 10.



# Java : Load CSV file into an array

The first step of the project was to load the CSV file that I paste inside the code/Java folder, then I have created a Java class untitle read\_CSV and the « Read » function that allow to my Java code to read the csv File and gather/return data into an array of Strings values.

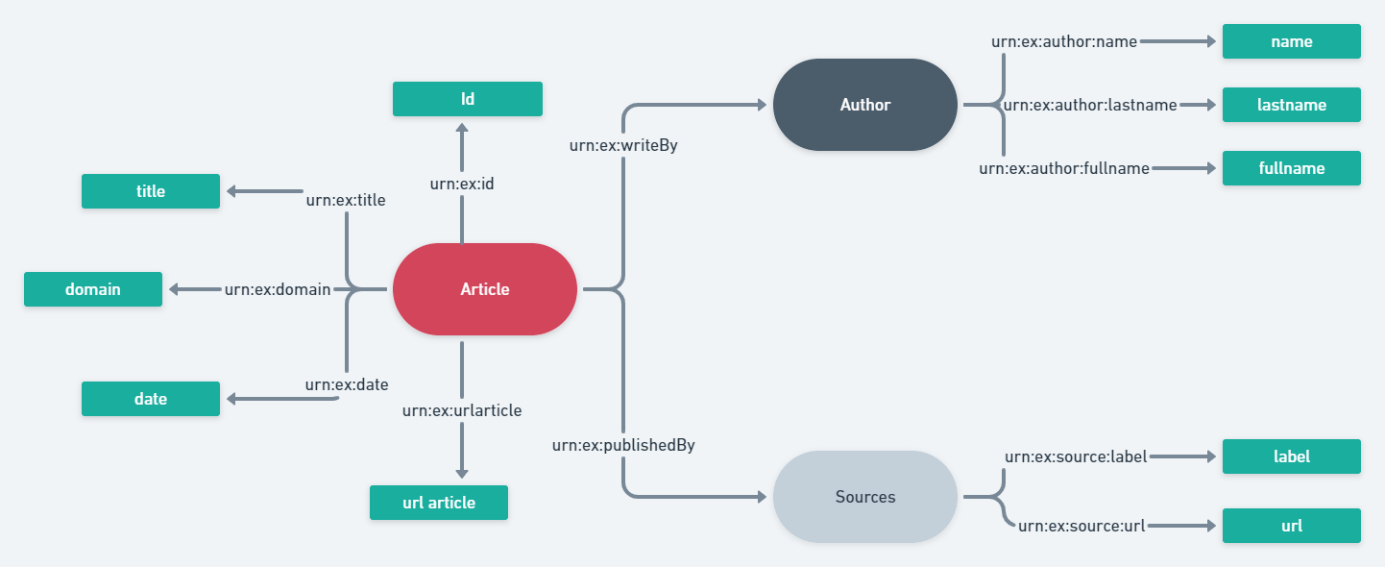
This method tale only two parameters one to get the name of the .csv file and the other one is a Boolean true/false to print the data of each row in the console.

Une image contenant texte

Description générée automatiquement

# Build of a Jena Model

My data correctly imported in a string data array the next step has been to create a RDF/RDFS model to etablish an efficient Jena design pattern adapted to the subject.



In this schema the green rectangles are representing the properties of sources in the red/grey/dark grey ellipses. I made the choice to create and code these three Resources (Article, Source and Author) and their properties inside a Java class named JenaModel in the ‘Build’ method who aim to generate a RDF file of the model and the CSV data.

I have started by create the sources’s properties, with the next code (JenaModel.Build) :

Une image contenant texte

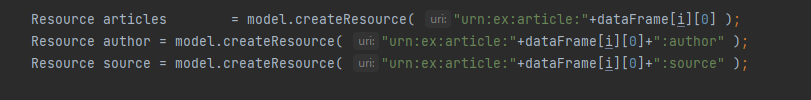
Description générée automatiquement

**Article**

**Author**

**Source**

Then I create all the resources of my dataset and I add them their properties, like in the schema you’have seen upper by using the Jena method « createResource » and « addProperty » to a default model with the data coming from the array of string data return by the « csvFile.Read » method.

Une image contenant texte

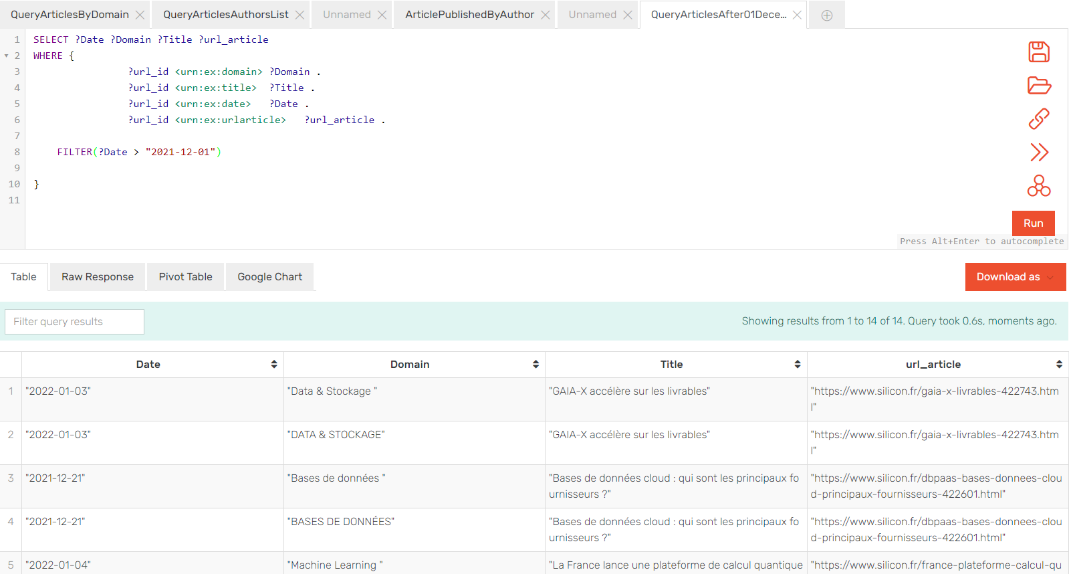
Description générée automatiquementUne image contenant texte, plaque, capture d’écran

Description générée automatiquement

From this « Jenamodel.Build » method I am now able to generate an .rdf file usable to create and execute queries.

# SPARQL queries

Une image contenant texte

Description générée automatiquementNow that the .rdf file is correctly generated as you can see bellow, I’have import this file to test my sparql queries on graphDB API.

Une image contenant texte

Description générée automatiquementThen the last steps of my project was to create the queries and menu to query the .rdf file from the console

Une image contenant texte

Description générée automatiquement