# Web Scrapping documentation

## Definition:

Web scraping, web harvesting, or web data extraction is data scraping used for extracting data from websites. In Our case, we have extracted data from the following site;

<https://www.rateinflation.com/inflation-rate/germany-historical-inflation-rate>.

## Requirement:

* Install Python SDK
* Clone the project from the following repo:

<https://git-codecommit.us-west-2.amazonaws.com/v1/repos/Web-Scraping-Inflation>

* Downloading the different package necessary to run the project.

## Understanding the code Structure.

The main package (file) where our main code is found is inside the main folder where we can see;

### Asset’s folder:

Here we have principally two files.

* + The Style.css file where we have all our CSS code.
  + The favicon.ico file. This is simply the favicon picture of mobile hup which is added our URL when the code is lunch.

### Inflation Data (CSV) folder:

In this folder we have the stored information extracted from a website on the internet in the csv file format. Every time the code is executed, we have a new file which is created with the date and hours as the file name. the actual file name is 2022-11-20-11-31-15.csv. This indicates the last time the code was executed for the last time.

### \_init\_.py:

Here is the main project file having 2 functions.

* The principal if function use to execute script. When the program is lunch it gets the actual time and stores it in the date\_time variable. Next it call the InflationScraper() Class from the inflation\_scrapper.py which collects the data from our site and call next the save\_inflation\_rate\_as\_csv() function which takes in parameter the date, the file\_name and the table\_heading to save the collected data in an CSV file format.

### Inflation\_scraper.py:

Here we have the principal class InflationScraper having 4 functions use to properly extract our data.

* \_\_init\_\_(self): is the initial functions which runs when the code is executed. It takes the URL of the site we want to extract data from in parameters and called while loop saying: While the site is available, parse all the data then stop.
* get\_table\_heading: In this method we define the exact type of data we want to extract. Depending on the type of data found on the website, we also define the type of data it will be extracting. The BeautifulSoup package is use here to parse the obtained data into readable html code. Then we specify we want to obtain the information present in the table (soup.find('table')). In the table now, we are finding all the table heads (thead) from the table row (tr), and getting all the table headers from 0 to 14 rows (table.find('thead').find('tr').find\_all('th')[0:14]).
* extract\_inflation\_data: Then main objective of this function is to collects now the information in the table body.