This Jupyter notebook has been made by Mladen Tsolov

I have written important information in the beginning of this project that is crucial before we get into a web scraping of a website. Please take your time to read it and do not ignore it, because a web scraping can lead to breaking the law.

General information

In this Jupyter notebook we will be doing web scraping of a Bulgarian website for properties.

After we collect the data we will inspect it in more details.

But before we begin we have to make sure that the website allows web scraping. Some websites do not allow it and we can break the law if we collect data without permission.

It is not very hard to see if a website allows web scraping. All we have to do is to append '/robots.txt' to the main website url:

https://www.imot.bg/robots.txt

The result that we get is:

Sitemap: https://www.imot.bg/sitemap/index.xml

User-agent: * - for all users using the website

Disallow: - it is left empty, which means that nothing is restricted

Information about the website.

The website that we will use in this example is only in Bulgarian language. It has many sections like: Buying, Selling, Renting, Looking for new built properties and many more.

In order to navigate within the website you can right click on it and select: 'Translate to English'

Imot.bg lists properties predominantly in Bulgaria and Greece, whereas http://worldwide-search.imot.bg/ has listings all over the world.

In this example we will focus only on https://www.imot.bg/ (imot means property)

The main page shows us the Bulgarian municipalities borders

Clicking on one of them will trigger a search window like the one given here:

https ://ww w.im ot.bg/pcgi/imot.cgi?act=3&slink= 7hou98 &f1=1

PLEASE NOTE: The reason why I have put spaces in the link above is, because in time it will not work, because the code for the municipality will be changed.

7hou98 - is the code for the municipality (please note that the code changes with time)

=1 - that is the current page (=0 and =1 yeald same result)

Each municipality page has exactly 40 property results and 25 pages of results.

Having said that there is a bit more preparation that we need to do.

We have to make sure that the webpage link is in the right format like the one below:

https ://ww w.im ot.bg/pcgi/imot.cgi?act=3&slink= 7hou98 &f1=1

We can see that it ends with '=1', that means if I wanted to see the second page I can change the 1 with 2 and it will open the second page.

In order to navigate in the webpage and find the relevant information that we would like to scrape we have 'inspect' it by right clicking anywhere in the webpage and pressing 'view page source' or 'inspect'. In that way we can see the separate objects/tags and tables that combined together form the webpage that we see. Furthermore we can use 'Ctrl + F', which opens a search window with the help of which we can find information faster.

How the information that we will scrape looks like?

The information is "hidden" in tags (from HTML). Tags are information containers used in websites.

In this cell we can see one table tag that has all the information that we need. The majority of that table tag is deleted for clarity.

table tag - depending on the property the style of this tag can vary. It can be ordinary listing, vip listing or top listing. This table tag represents a single property listing. Within it we can gather additional information.

table width="660" cellspacing="0" cellpadding="0" border="0" style="margin-bottom:0px; border-top:#990000 1px solid; background:url(../images/picturess/top_bg.gif); background-position:bottom; background-repeat:repeat-x;"

td tag - we can get information about the location and the number of rooms of the property. It is a bit tricky to get to the information hidden in the 'alt' parameter (Someone made mistake with the number of rooms for this listing, in the tags below it is said that it is one bedroom apartment).

td a class="photoLink" img src="" style="object-fit: cover;" width="200" height="150" border="0" alt="2-bedroom apartment for sale, Gorna Oryahovitsa, Veliko Tarnovo region"

div tag - We can get the price of the property using the class of the div

div class="price" img BGN 58,000

a tag or link tag - Number of rooms, using the class

a href="" class="lnk1" One-bedroom apartment for sale in

a tag or link tag - Location, using the class

a href="" class="lnk2" Gorna Oryahovitsa, Veliko Tarnovo region

td tag - Additional information about the listing

td width="520" colspan="3" height="50" style="padding-left:4px" 69 sq.m, Prolet quarter, ONE BEDROOM APARTMENT EXCLUSIVELY! WE ARE WITH THE KEY! style="vertical-align: inherit;"> Property Tarnovgrad offers one-bedroom panel apartment ..., tel .: 089

All the tags above are for one property listing.

Language and translation

The Imot.bg website is in Bulgarian. There are ways to convert the information into English.

One is to use Module that will translate the information into this notebook - googletrans or googletrans new (I used google trans new).

Other is to scrape the website, save the dataframe as CSV file and then translate the file with Google Translate.

I will do both and compare them to see which one is better.

Actual Web scraping

Plan of action.

We will first do web scraping to see how it works and later on we will translate the results. I will try to explain along the way where needed.

```python
# Libraries for Web Scraping
from bs4 import BeautifulSoup
import requests
import urllib.request

# Libraries for Data Processing
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# Library for translation into English
from google_trans_new import google_translator

# The two lines below allow us to see all the columns without truncation.
# It allows us to see the dataset better.
pd.options.display.max_columns = None
pd.options.display.max_rows = None
```

As mentioned before it is interesting to see that '=0' gives the same result as '=1' (that was for the page number). We do not want repetitions, so we will start with '=1'. We can define a variable called 'url' that will store string with the website's address.

We can define a function called 'connection' that will make the connection between us and the website and will return number, which will tell us if it was sucessful.

```python
def connection(page):
    '''
    This function is used to change the number of the page from the link.
    So after we finish with collecting the data from page 1, we can continue to page 2
    of that webpage and so on.
    '''

    # https://router-network.com/tools/what-is-my-user-agent    -> to find the 'user agent'
    # Or google -> 'what is my user agent'
    # 'user agent' is a mediator between the user and the internet it holds technical information
    # About the device (the computer used) and the software
    # The 'user agent' is unique for every person on the internet
    headers = {'User-Agent': 'Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) C

    # F-string in which we can change the number of the page
    # Note that 'page' in the f-string is the parameter of the function
    url = f'https://www.imot.bg/pcgi/imot.cgi?act=3&slink=7o949b&f1={page}'

    r = requests.get(url, headers)

    # Checks if there is a connection between the user and the webpage
    return r.status_code

print(connection(1))

# https://developer.mozilla.org/en-US/docs/Web/HTTP/Status
# Informational responses (100-199)
# Successful responses (200-299)
# Redirection messages (300-399)
# Client error responses (400-499)
# Server error responses (500-599)

# If we get 200 in return then all is working fine.
```

```
200
```

In the cell above, where I wrote about the tags I mentioned about 'div' tag that has stored the price of the property. We can use it to count the number of properties for a single page -> div class="price" img BGN 58,000

```python
# in this cell we count the number of property listings by counting the number of divs called 'price'

def extract(page):
    '''
    This function is used to change the number of the page from the link.
    So after we finish with collecting the data from page 1, we can continue to page 2
    of that webpage and so on.
    '''

    # https://router-network.com/tools/what-is-my-user-agent    -> to find the 'user agent'
    # or google -> 'what is my user agent'
    # 'user agent' is a mediator between the user and the internet it holds technical information
    # about the device (the computer used) and the software
    # the 'user agent' is unique for every person on the internet
    headers = {'User-Agent': 'Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) C

    # f-string in which we can change the number of the page
    # note that 'page' in the f-string is the parameter of the function
    url = f'https://www.imot.bg/pcgi/imot.cgi?act=3&slink=7o949b&f1={page}'

    r = requests.get(url, headers)

    soup = BeautifulSoup(r.content, 'html.parser')

    return soup


def transform(soup):
    '''
    This function is used to find and collect information.
    '''

    divs = soup.find_all('div', class_ = 'price')

    return len(divs)

search = extract(1)
print(transform(search))

# we get as a result 40
# that is the number of property listings per page
```

```
40
```

If the result is 0, that means that the code for the municipality is changed and that is the main reason. We should get 40 as a result.

We can use the code for the extract function from the cell above. In that way we can focus on the new lines of code.

From the cell above we can see that we get 40 as a result, that is the number of the properties for one page.

In the cell below we will use the same functions 'extract' and 'transform' but this time we will collect the information for one property listing.

```python
# collects general information for 1 property (first property, first page)

def transform_property(soup):
    '''
    This function is used to find and collect information.
    '''

    property_info = soup.find("a", class_ = 'photoLink').img.get('alt')
    print(property_info.strip())

    price_div = soup.find('div', class_ = 'price')
    print(price_div.text.strip())

    rooms_a = soup.find('a', class_ = 'lnk1')
    print(rooms_a.text.strip())

    location_a = soup.find('a', class_ = 'lnk2')
    print(location_a.text.strip())

    info = soup.find('td', width="520", colspan="3", height="50", style="padding-left:4px")
    print(info.text.strip())


    return len(price_div)

search = extract(1)
print(transform_property(search))
```

```
Обява продава къща, гр. Елена, област Велико Търново
50 000 EUR
Продава КЪЩА
гр. Елена, област Велико Търново
146 кв.м, двор 2200 кв.м, с.Дърлевци, Реновирана къща със съхранени елементи на автентичното строителство, хара
ктерно за района от края на ..., тел.: 0887902715
3
```

Yey! We got something.

It would have been nice to go to https://translate.google.com/ . Paste the URL of the website we want translated, select to translate from Bulgarian into English and click the URL on the right hand site. We can see that the website is translated. We can copy that link and paste it as a new value for the 'url' variable in 'extract' function.

https ://www- imot-bg .translate. goog/pcgi/imot.cgi?act=3&slink=7hzunk&f1=1&_x_tr_sl=auto&_x_tr_tl=en&_x_tr_hl=en-GB

just replace '=1' with '={page}'

Unfortunately, it does not work.

Each property listing has a unique style for the table that contains the basic information about it - location, price and general information. Depending on the property there can be three styles: Regular, TOP and VIP listing.

We can use them to collect the information from all listings.

Starting from the cell below we will focus on the 'transform' function and how to alter the code in order to make it more functional.

```python
def transform(soup):
    '''
    This function is used to find and collect information.
    '''

    # Finds all 'div' tags that store the price for each listing
    # It is used to locate all the listings on the page
    price_div = soup.find_all('div', class_ = 'price')

    # The different styles of the tables that we gather information are defined
    # One table = one property listing
    style_0 = "margin-bottom:0px; border-top:#990000 1px solid; background:url(../images/picturess/top_bg.gif);
    style_1 = "margin-bottom:0px; border-top:#990000 1px solid; background:url(../images/picturess/vip_bg.gif);
    style_2 = "margin-bottom:0px; border-top:#990000 1px solid;"

    # We can find all tables that have the styles from above
    table_0 = soup.find_all('table', style=style_0)
    table_1 = soup.find_all('table', style=style_1)
    table_2 = soup.find_all('table', style=style_2)


    # for loop to collect the listings for tables with style_0
    for i in table_0:
        property_info = soup.find("a", class_ = 'photoLink').img.get('alt').strip()
        price = i.find('div', class_ = 'price').text.strip()
        rooms = i.find('a', class_ = 'lnk1').text.strip()
        location = i.find('a', class_ = 'lnk2').text.strip()
        info = i.find('td', width="520", colspan="3", height="50", style="padding-left:4px").text.strip()

        # dictionary to store the collected information
        property_dictionary_0 = {
                               'property_info' : property_info,
                               'price' : price,
                               'rooms' : rooms,
                               'location' : location,
                               'info' : info
                              }
        property_list.append(property_dictionary_0)


    # for loop to collect the listings for tables with style_1
    for i in table_1:
        property_info = soup.find("a", class_ = 'photoLink').img.get('alt').strip()
        price = i.find('div', class_ = 'price').text.strip()
        rooms = i.find('a', class_ = 'lnk1').text.strip()
        location = i.find('a', class_ = 'lnk2').text.strip()
        info = i.find('td', width="520", colspan="3", height="50", style="padding-left:4px").text.strip()

        # dictionary to store the collected information
        property_dictionary_1 = {
                               'property_info' : property_info,
                               'price' : price,
                               'rooms' : rooms,
                               'location' : location,
                               'info' : info
                              }
        property_list.append(property_dictionary_1)

    # for loop to collect the listings for tables with style_2
    for i in table_2:
        property_info = soup.find("a", class_ = 'photoLink').img.get('alt').strip()
        price = i.find('div', class_ = 'price').text.strip()
        rooms = i.find('a', class_ = 'lnk1').text.strip()
        location = i.find('a', class_ = 'lnk2').text.strip()
        info = i.find('td', width="520", colspan="3", height="50", style="padding-left:4px").text.strip()

        # dictionary to store the collected information
        property_dictionary_2 = {
                               'property_info' : property_info,
                               'price' : price,
                               'rooms' : rooms,
                               'location' : location,
                               'info' : info
                              }
        property_list.append(property_dictionary_2)

    # This return will show how many listings are with style_0, _1 or _2.
    # Please note that the sum must be 40
    return len(table_0), len(table_1), len(table_2), len(price_div)

# we create empty list to which we append info
property_list = []


search = extract(1)
print(transform(search))

print(len(property_list))
print(property_list)
```

```
(21, 19, 0, 40)
40
[{'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '50 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'гр. Елена, област Велико Търново', 'info': '146 кв.м, двор 2200 кв.м, с.Дърлевци, Реновирана къща със съхранени елементи на автентичното строителство, характерно за района от края на ..., тел.: 0887902715'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '65 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'гр. Елена, област Велико Търново', 'info': '144 кв.м, двор 739 кв.м, с.Граматици, Атрактивен имот в балкана. Двуетажна, частично обзаведена тухлена сграда. Имот с целогодишен достъп ..., тел.: 0887902715'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '85 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'гр. Златарица, област Велико Търново', 'info': '185 кв.м, двор 613 кв.м, Primo+ Велико Търново Ви предлага просторна и уютна , тухлена къща след тотален ремонт, с  и добро ..., тел.: 0887177900'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '115 000 лв.', 'rooms': 'Продава КЪЩА', 'location': 'гр. Павликени, област Велико Търново', 'info': '123 кв.м, двор 750 кв.м, Имоти Търновград Ви предлага двуетажна ОБЗАВЕДЕНА къща в гр. Павликени. Имотът се намира на тиха, ас ..., тел.: 0885168782'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '5 999 EUR', 'rooms': 'Продава КЪЩА', 'location': 'гр. Полски Тръмбеш, област Велико Търново', 'info': '140 кв.м, двор 3000 кв.м, село Орловец, Тухлена къща с голям двор 3000 кв.м и стопански постройки в село Орловец на 30 км от Велико Търново  ..., тел.: 0899855101'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '85 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'с. Балван, област Велико Търново', 'info': '160 кв.м, двор 1250 кв.м, ВИРТУАЛЕН ТУР! ЕКСКЛУЗИВНО! Реновирана двуетажна къща с допълнителни постройки в село Балван Атракти ..., тел.: 0899921394'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '155 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'с. Бръчковци, област Велико Търново', 'info': '206 кв.м, двор 826 кв.м, На Вашето внимание представяме един изключителен имот в красивият Еленски Балкан, а именно, новопост ..., тел.: 0884905169'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '252 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'с. Бръчковци, област Велико Търново', 'info': '220 кв.м, двор 900 кв.м, Агенция   Имоти Консулт  предлага имота  разположен в живописния Еленски балкан, на брега на язовир  ..., тел.: 0988871433'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '49 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'с. Джулюница, област Велико Търново', 'info': '87 кв.м, двор 680 кв.м, Агенция за недвижими имоти  SUNNY HOMES  предлага за продажба реновирана къща с двор в с. Джулюница. ..., тел.: 0882373785'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '300 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'с. Лагерите, област Велико Търново', 'info': '1332 кв.м, двор 12400 кв.м, Тухла 1947 г., ЕКСКЛУЗИВНО: ВНУШИТЕЛНА РЕЗИДЕНЦИЯ С ИСТОРИЯ И СТРАТЕГИЧЕСКА ЛОКАЦИЯ -  МЛАДЕЖКИ ЛАГЕР  ХАИНБОАЗ  ..., тел.: 0889590019'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '165 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'с. Леденик, област Велико Търново', 'info': '260 кв.м, двор 488 кв.м, ID:12699 ЕКСКЛУЗИВНО !!! Прекрасен имот с панорамна гледка в предпочитано село!!! ''Ту България''- н ..., тел.: 0876 39 54 29'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '27 000 лв.', 'rooms': 'Продава КЪЩА', 'location': 'с. Лесичери, област Велико Търново', 'info': '120 кв.м, двор 550 кв.м, Агенция за недвижими имоти  Акценти Пропъртис  ООД Ви предлагаме хубав и добре поддържан имот, разпо ..., тел.: 0887 573 633'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '33 000 лв.', 'rooms': 'Продава КЪЩА', 'location': 'с. Палици, област Велико Търново', 'info': '120 кв.м, двор 1500 кв.м, ID:20517 ''Ту България'' недвижими имоти предлага тухлена двуетажна къща с магазин в с.Палици. Селот ..., тел.: 0878 48 47 40'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '160 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'с. Палици, област Велико Търново', 'info': '300 кв.м, двор 1600 кв.м, SUNNY HOMES недвижими имоти  предлага  къща с двор, РЗП 300 кв.метра, дворно място 1600 кв.м.,  камъ ..., тел.: 0882373785'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '54 000 лв.', 'rooms': 'Продава КЪЩА', 'location': 'с. Самоводене, област Велико Търново', 'info': '80 кв.м, двор 500 кв.м, 2 ет., Агенция за недвижими имоти 'Оникс' Ви предлага масивна двуетажна къща на калкан във високата част в  ..., тел.: 0885 971 518'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '80 000 лв.', 'rooms': 'Продава КЪЩА', 'location': 'с. Хотница, област Велико Търново', 'info': '100 кв.м, двор 1490 кв.м, ЕКСКЛУЗИВНО! Имоти Търновград Ви предлага реновирана къща в село Русаля, намираща се в близост до ре ..., тел.: 0895476397'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '250 000 EUR', 'rooms': 'Продава ХОТЕЛ', 'location': 'гр. Елена, област Велико Търново', 'info': '6878 кв.м, Тухла 2005 г., ARCO REAL ESTATE, ПРЕДЛАГА КОМПЛЕКС С КЪЩИ ЗА ОТДИХ. Близко до комплекса е град Елена, който е извес ..., тел.: 0894550682'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '400 000 EUR', 'rooms': 'Продава ХОТЕЛ', 'location': 'гр. Елена, област Велико Търново', 'info': '1427 кв.м, Тухла 2007 г., ЕРА ИМОТИ РУСЕ Ви предлага уникалната възможност да инвестирате в собствен приказен хотел в живописн ..., тел.: 0886601920'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '300 000 EUR', 'rooms': 'Продава ХОТЕЛ', 'location': 'с. Лагерите, област Велико Търново', 'info': '1332 кв.м, Тухла 1947 г., ЕКСКЛУЗИВНО: Емблематичен обект с история и стратегическа локация до границата на област Велико Търн ..., тел.: 0889590019'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '55 000 EUR', 'rooms': 'Продава ПАРЦЕЛ', 'location': 'гр. Елена, област Велико Търново', 'info': '4974 кв.м, ID:20556 Продава се атрактивен имот с голям понеицал в Еленския Балкан разположен в село на 5км гр.  ..., тел.: 0878 48 47 40'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '16 000 EUR', 'rooms': 'Продава ПАРЦЕЛ', 'location': 'с. Ново село, област Велико Търново', 'info': '2400 кв.м, Регулация, Предлагаме ЕКСКЛУЗИВНО, дворно място в регулацията на с. Ново село, с площ на парцела 2400кв.м.. Имо ..., тел.: 0878818845'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '65 000 лв.', 'rooms': 'Продава 2-СТАЕН', 'location': 'гр. Горна Оряховица, област Велико Търново', 'info': '57 кв.м, 7-ми ет. от 7, Агенция за недвижими имоти ОНИКС Ви предлага двустаен апартамент в гр. Горна Оряховица, в най-предпо ..., тел.: 0885 971 518'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '67 000 лв.', 'rooms': 'Продава 2-СТАЕН', 'location': 'гр. Горна Оряховица, област Велико Търново', 'info': '60 кв.м, 2-ри ет. от 8, Предлагаме двустаен, панелен апартамент на междинен етаж, с изложение юг. Имота се състои от: коридо ..., тел.: 0878818845'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '70 000 лв.', 'rooms': 'Продава 2-СТАЕН', 'location': 'гр. Горна Оряховица, област Велико Търново', 'info': '68 кв.м, 8-ми ет. от 8, Газ, ДВУСТАЕН АПАРТАМЕНТ ЕКСКЛУЗИВНО ! Агенция за недвижими имоти ОНИКС Ви предлага двустаен панелен апар ..., тел.: 0885 971 518'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '26 300 EUR', 'rooms': 'Продава 3-СТАЕН', 'location': 'гр. Горна Оряховица, област Велико Търново', 'info': '110 кв.м, 1-ви ет. от 3, ЕКСКЛУЗИВНО! Болярски Имоти предлага за продажба изцяло Южен Тристаен апартамент в квартал Гарата гр ..., тел.: 0877559233'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '62 000 лв.', 'rooms': 'Продава 3-СТАЕН', 'location': 'гр. Горна Оряховица, област Велико Търново', 'info': '80 кв.м, 4-ти ет. от 4, Имоти Търновград Ви предлага тристаен апартамент на шпакловка и замазка в тих район на гр. Горна Оря ..., тел.: 0895476397'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '52 000 EUR', 'rooms': 'Продава 4-СТАЕН', 'location': 'гр. Горна Оряховица, област Велико Търново', 'info': '115 кв.м, 5-ти ет. от 5, Ексклузивна оферта! Агенция за имоти продава четиристаен тухлен апартамент в кв.Пролет, близо до пар ..., тел.: 0887999086'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '89 000 EUR', 'rooms': 'Продава 4-СТАЕН', 'location': 'гр. Горна Оряховица, област Велико Търново', 'info': '170 кв.м, 2-ри ет. от 3, ТЕЦ, Екипът на агенция 'ОНИКС' Ви предлага луксозно, обзаведен етаж от къща, находящ се в широкия център  ..., тел.: 0885971518'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '52 500 EUR', 'rooms': 'Продава МНОГОСТАЕН', 'location': 'гр. Горна Оряховица, област Велико Търново', 'info': '93 кв.м, 1-ви ет. от 4, Имоти Търновград има удоволствието да Ви предложи многостаен тухлен апартамент с гараж в град Горна  ..., тел.: 0884796221'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '57 000 EUR', 'rooms': 'Продава МНОГОСТАЕН', 'location': 'гр. Горна Оряховица, област Велико Търново', 'info': '155 кв.м, 4-ти ет. от 4, Тухла 2011 г., Ексклузивна оферта! Без комисионна за купувача! Продава уникален панорамен ново строителство апартам ..., тел.: 0887999086'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '60 000 лв.', 'rooms': 'Продава ЕТАЖ ОТ КЪЩА', 'location': 'гр. Горна Оряховица, област Велико Търново', 'info': '100 кв.м, Калтинец, 2-ри ет. от 2, Агенция за недвижими имоти ОНИКС има удоволствието да Ви предложи етаж от къща в град Горна Оряховиц ..., тел.: 0885 971 518'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '92 300 EUR', 'rooms': 'Продава ЕТАЖ ОТ КЪЩА', 'location': 'гр. Горна Оряховица, област Велико Търново', 'info': '130 кв.м, 1-ви ет. от 2, Агенция за имоти продава ексклузивно луксозно жилище с панорамна гледка в тих и спокоен квартал на Г ..., тел.: 0887999086'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '88 500 лв.', 'rooms': 'Продава ЕТАЖ ОТ КЪЩА', 'location': 'гр. Дебелец, област Велико Търново', 'info': '89 кв.м, 1-ви ет. от 2, ЕКСКЛУЗИВНО!Екипът на фирма за недвижими имоти 'АкцентИ Пропъртис' ООД Ви предлага самостоятелен ета ..., тел.: 0887/573 633'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '65 000 EUR', 'rooms': 'Продава ЕТАЖ ОТ КЪЩА', 'location': 'гр. Елена, област Велико Търново', 'info': '117 кв.м, гр.Елена, 2-ри ет. от 3, Тухла 1978 г., Просторен етаж от масивна двуфамилна къща. Намира се на първи жилищен етаж, над гараж / магазин/скл ..., тел.: 0887902715'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '43 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'гр. Бяла черква, област Велико Търново', 'info': '130 кв.м, двор 4000 кв.м, 1 ет., Агенция за недвижими имоти БОЛКАН ЕСТЕЙТ предлага ЕКСКЛУЗИВНО на Вашето внимание просторна къща за п ..., тел.: 0886129319'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '55 700 EUR', 'rooms': 'Продава КЪЩА', 'location': 'гр. Горна Оряховица, област Велико Търново', 'info': '150 кв.м, двор 351 кв.м, Болярски Имоти Ви предлага за продажба САМОСТОЯТЕЛНА  двуетажна къща със собствен двор на 5 минути п ..., тел.: 0877559233'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '22 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'гр. Елена, област Велико Търново', 'info': '214 кв.м, двор 1800 кв.м, с.Беброво, Здрава тухлена двуетажна къща . Сградата е на два етажа, като на първият се намира три стаи, баня и  ..., тел.: 0887902715'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '22 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'гр. Елена, област Велико Търново', 'info': '150 кв.м, двор 620 кв.м, с.Беброво, Здрава тухлена къща с каменни основи. Сградата е на две нива, като на първото се намира  магазин, пр ..., тел.: 0887902715'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '25 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'гр. Елена, област Велико Търново', 'info': '120 кв.м, двор 2530 кв.м, с.Средни Колиби, Къща в близост до яз. Йовковци. Собствен водоизточник - кладенец, централно водоснабдяване на ВиК, ел ..., тел.: 0887902715'}, {'property_info': 'Обява продава къща, гр. Елена, област Велико Търново', 'price': '33 000 EUR', 'rooms': 'Продава КЪЩА', 'location': 'гр. Елена, област Велико Търново', 'info': '90 кв.м, двор 270 кв.м, гр.Елена, Старинна жилищна сграда с дворно място и построената в него двуетажна допълваща сграда. Продава се ж ..., тел.: 0887902715'}]
```

For example on the first line we may have: (20, 20, 0, 40) - that means that there are 20 properties with 'style 0', 20 properties with 'style 1' and 0 properties with 'style 2'. The sum of which should be always equal to 40.

The code from above is not hard to understand, but there is a lot of repetition.

I want to explain how to create a DataFrame object and how to translate it into English just to see how this works and then I will amend the code from above and make it more compact.

In the cell below the for loop collects data from multiple pages and saves them on 'property_list' variable.

```python
# Each page has 40 entries, therefore 2 times 40 is equal to 80

# I overwrite the 'property_list' variable to be empty list,
# otherwise the 'property_list' will grow each time the functions above are run
# and will give 'wrong' answers when tested

property_list = []

for i in range(1,3):
    print(f'Page number: {i}')
    search = extract(i)
    transform(search)

print(len(property_list))
```

```
Page number: 1
Page number: 2
80
```

We have scraped 80 listings from the website, that is now much, but proves that the concept works.

Let's not forget that the results are in Bulgarian.

The first way to translate into English is to create a Pandas DataFrame object and save the result into a file. Then we can upload the file here: https://translate.google.com/?sl=bg&tl=en&op=translate and translate with Google Translate so we can see the final result.

Initially, when I began working on this project Google Translate was translating *.txt files.

Now it can translate only - .docx, .pdf, .pptx, or .xlsx.

It can not translate *.CSV files as well

First Save it to a *.CSV file to have the right format and to preserve the encoding.

Open the *.CSV file* '*File*' -> '*Save As*' -> '*Excel Workbook*' *(*.xlsx).

In that way it will allow us to upload the file to Google Translate.

This will prevent it from telling us that the file is corrupted.

```python
# We create DataFrame object
df = pd.DataFrame(property_list)

# Print it to see what we got
print(df.head())

# Apparently, in the description of each listing we can see many commas and semicolons
# that are the usually separators.
# It will cause problems, so we can change the separator to be '~'
df.to_csv('test.csv', encoding = 'utf-8-sig', sep = '~')
```

```
                                       property_info        price  \
0  Обява продава къща, гр. Елена, област Велико Т...   50 000 EUR
1  Обява продава къща, гр. Елена, област Велико Т...   65 000 EUR
2  Обява продава къща, гр. Елена, област Велико Т...   85 000 EUR
3  Обява продава къща, гр. Елена, област Велико Т...  115 000 лв.
4  Обява продава къща, гр. Елена, област Велико Т...    5 999 EUR

          rooms                                   location  \
0  Продава КЪЩА           гр. Елена, област Велико Търново
1  Продава КЪЩА           гр. Елена, област Велико Търново
2  Продава КЪЩА       гр. Златарица, област Велико Търново
3  Продава КЪЩА       гр. Павликени, област Велико Търново
4  Продава КЪЩА  гр. Полски Тръмбеш, област Велико Търново

                                              info
0  146 кв.м, двор 2200 кв.м, с.Дърлевци, Реновира...
1  144 кв.м, двор 739 кв.м, с.Граматици, Атрактив...
2  185 кв.м, двор 613 кв.м, Primo+ Велико Търново...
3  123 кв.м, двор 750 кв.м, Имоти Търновград Ви п...
4  140 кв.м, двор 3000 кв.м, село Орловец, Тухлен...
```

I had to manually delete the first row of the file - 'test_after_translation' (Excel Worksheet), then save it to 'test_after_translation' (*.CSV). Now when I read the* .CSV file in Python I wouldn't get an error about some rows having more elements than others, then as you can see below I added again the column names so no harm done.

```python
# Once we have uploaded and translated the file we can save it again as Excel Workbook 'test_translated.txt'
# We have to make sure that all the rows are translated and not just partly.
# If the file has been translated partly then we can try again until it is fully translated.

translated_df = pd.read_csv('./test_after_translation.csv', sep = '~', header = None)
translated_df = translated_df.drop(axis = 1, columns = 0)
translated_df.columns = ['property_info', 'price', 'rooms', 'location','info']

translated_df.head()
```

	property_info	price	rooms	location	info
0	Ad for sale house, town of Elena, Veliko Tarn...	50 000 EUR	House for sale	c. Helen, Veliko Tarnovo region	146 sq.m., yard 2200 sq.m., The village of Da...
1	Ad sells a house, town of Elena, Veliko Tarno...	65 000 EUR	House for sale	c. Helen, Veliko Tarnovo region	144 sq.m., yard 739 sq.m., Gramatici village,...
2	Ad for sale house, town of Elena,Veliko Tarno...	85 000 EUR	House for sale	c. Zlataritsa, Veliko Tarnovo region	185 sq.m., yard 613 sq.m., Primo + Veliko Tar...
3	Ad for sale house, town of Elena, Veliko Tarn...	115 000 BGN	HOUSE FOR SALE	c. Pavlikeni, Veliko Tarnovo region	123 sq.m., yard 750 sq.m., Property Tarnovgra...
4	Ad sells a house, town of Elena,Veliko Tarnov...	5 999 EUR	House for sale	c. Polski Trambesh, Veliko Tarnovo region	140 sq.m., yard 3000 sq.m., the village of Or...

The above DataFrame is the result I got.

Now let's try the other way to translate the Bulgarian DataFrame we managed to scrape.

For this purpose we will use 'google_trans_new' module.

Important: There is a bug we need to fix before translating anything.

https://github.com/lushan88a/google_trans_new/issues/36

Change line 151 and line 233 in google_trans_new/google_trans_new.py:

response = (decoded_line + ']') TO response = (decoded_line)

```python
# This cell is intended for introducing the 'google_trans_new' module


def basic_translation(text_for_translation, target_language):
    '''
    Basic function introducing 'google_trans_new'.
    '''
    # We instantiate 'google_translator' object called 'translator'
    translator = google_translator(url_suffix="bg", timeout=5)

    # In the brackets we put the text we need translated first, then we define 'lang_tgt' parameter
    # that is used to tell to what language we need to translate the original text
    translate = translator.translate(text_for_translation,
                                     lang_tgt = target_language,
                                     pronounce = True)

    return print(translate)



basic_translation('สวัสดีจีน', 'zh')    # 'zh' is for Chinese
basic_translation('Привет мир', 'en')
basic_translation('Здравей свят', 'en')
```

```
['你好中文 ', 'S̄wạs̄dī cīn', 'Nǐ hǎo zhōngwén']
['Hello World ', 'Privet mir', None]
['Hello world ', 'Zdraveĭ svyat', None]
```

Now we can try to translate the DataFrame from above directly without having to save files and manually translate them with Google.

It might take some time to complete the translation process.

```python
def google_trans(dataframe, source_language, target_language):

    '''
    Simple function for translating a DataFrame.
    '''

    # We instantiate 'google_translator' object called 'translator'
    translator = google_translator(url_suffix="bg", timeout=5)

    # Creating a deep copy of the dataframe
    google_df = dataframe.copy(deep = True)

    # Instantiating 'columns' variable to store the names of the columns
    columns = google_df.columns

    # For-looping through each column and overwriting each text with translated text
    for column in columns:
        google_df[column] = google_df[column].apply(translator.translate,
                                                    lang_src = source_language,
                                                    lang_tgt = target_language)
    return google_df
```

Translating the DataFrame and saving it into file.

```python
google_trans_df = google_trans(df, 'bg', 'en')

google_trans_df.to_csv('google_trans_test.csv')
```

Reading from the saved file.

```python
google_trans_df = pd.read_csv('./google_trans_test.csv')
google_trans_df = google_trans_df.drop(axis = 1, columns = 'Unnamed: 0')
```

Finally, lets compare the results.

First the Original Dataframe followed by the Manually translated DataFrame and third will be the 'Google Trans New' DataFrame.

```python
df.head(3)
```

	property_info	price	rooms	location	info
0	Обява продава къща, гр. Елена, област Велико Т...	50 000 EUR	Продава КЪЩА	гр. Елена, област Велико Търново	146 кв.м, двор 2200 кв.м, с.Дърлевци, Реновира...
1	Обява продава къща, гр. Елена, област Велико Т...	65 000 EUR	Продава КЪЩА	гр. Елена, област Велико Търново	144 кв.м, двор 739 кв.м, с.Граматици, Атрактив...
2	Обява продава къща, гр. Елена, област Велико Т...	85 000 EUR	Продава КЪЩА	гр. Златарица, област Велико Търново	185 кв.м, двор 613 кв.м, Primo+ Велико Търново...

```python
translated_df.head(3)
```

	property_info	price	rooms	location	info
0	Ad for sale house, town of Elena, Veliko Tarn...	50 000 EUR	House for sale	c. Helen, Veliko Tarnovo region	146 sq.m., yard 2200 sq.m., The village of Da...
1	Ad sells a house, town of Elena, Veliko Tarno...	65 000 EUR	House for sale	c. Helen, Veliko Tarnovo region	144 sq.m., yard 739 sq.m., Gramatici village,...
2	Ad for sale house, town of Elena,Veliko Tarno...	85 000 EUR	House for sale	c. Zlataritsa, Veliko Tarnovo region	185 sq.m., yard 613 sq.m., Primo + Veliko Tar...

```python
google_trans_df.head(3)
```

	property_info	price	rooms	location	info
0	Listing House for sale, Elena, Veliko Tarnovo ...	EUR 50 000	House for sale	Elena, Veliko Tarnovo District	146 sq.m, yard 2200 sq.m, S.Mrulevtsi, renovat...
1	Listing House for sale, Elena, Veliko Tarnovo ...	EUR 65 000	House for sale	Elena, Veliko Tarnovo District	144 sq.m, yard 739 sq.m, villageGamatsi, attra...
2	Listing House for sale, Elena, Veliko Tarnovo ...	85 000 EUR	House for sale	Zlataritsa, Veliko Tarnovo region	185 sq.m, yard 613 sq.m, Primo + Veliko Tarnov...

Overall, both translations are good, both have flaws and good sides, but personally I prefer the Manual translation : 'translated_df'

Now we can upgrade the code and scrape listings from multiple municipalities.

Might be useful to see how we can get the last page for this website.

Keep in mind every website is designed differently and the code below will not work with other websites. The way I found the last page number was to find the text representing it in the HTML code.

span class="pageNumbersInfo">Страница 1 от 25</span

Translated: span class="pageNumbersInfo">PAGE 1 OF 25</span

It is interesting to mention that this website has 25 pages of properties for each municipality and each page has 40 property listings.

In [16]:
```python
def transform_page_number(soup):

    '''
    This simple function is used to find the last page number for each municipality.
    Assumes that the page number is 2 digit number
    '''

    # We are looking for 'span' tag from class 'pageNumbersInfo'
    last_page = soup.find('span', class_ = 'pageNumbersInfo').text.strip()[-2:]
    last_page_number.append(last_page)

    return last_page_number

# we create empty list to which we append info
property_list = []
last_page_number = []


search = extract(1)
print(transform_page_number(search))
```

```
['25']
```

In order to get results from multiple municipalities we need to alter the code for the 'extract' function. The code for 'transform' function stays the same, it is not the best code. We can see repetition occuring.

The 'modified_extract' function takes another argument called 'municipality' that is the code of the municipality from the URL for it. We will use it instead of 'extract' from now on.

In [17]:
```python
def modified_extract(page, municipality):
    '''
    This function is used to change the number of the page from the link.
    So after we finish with collecting the data from page 1, we can continue to page 2
    of that webpage and so on.
    '''

    # https://router-network.com/tools/what-is-my-user-agent    -> to find the 'user agent'
    # or google -> 'what is my user agent'
    # 'user agent' is a mediator between the user and the internet it holds technical information
    # about the device (the computer used) and the software
    # the 'user agent' is unique for every person on the internet
    headers = {'User-Agent': 'Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) C

    # f-string in which we can change the number of the page
    # note that 'page' in the f-string is the parameter of the function
    # in addition to that now we can input diferent strings that represent minicipalities
    url = f'https://www.imot.bg/pcgi/imot.cgi?act=3&slink={municipality}&f1={page}'

    r = requests.get(url, headers)

    soup = BeautifulSoup(r.content, 'html.parser')

    return soup
```

In [18]:
```python
# downloads content from different locations/municipalities

def transform(soup):
    '''
    This function is used to find and collect information.
    '''

    # We can find all 'div' tags that store the price of each property listing
    # in that way we can get all 40 listings
    price_div = soup.find_all('div', class_ = 'price')

    # Each listing is associated with a particular style that actually shows if the listing
    # is 'Top', 'Vip' or just ordinary listing
    style_0 = "margin-bottom:0px; border-top:#990000 1px solid; background:url(../images/picturess/top_bg.gif);
    style_1 = "margin-bottom:0px; border-top:#990000 1px solid; background:url(../images/picturess/vip_bg.gif);
    style_2 = "margin-bottom:0px; border-top:#990000 1px solid;"

    # We locate all tags 'table' that contain all the information we can scrape and we associate
    # the 'table' tag with the different styles
    table_0 = soup.find_all('table', style=style_0)
    table_1 = soup.find_all('table', style=style_1)
    table_2 = soup.find_all('table', style=style_2)

    # The following three for loops go through each 'table' tag and scrape information from it
    # The each for loop stores the information into a dictionary. Each dictionary is then appended
    # to a list that contains all the information.
    for i in table_0:
        property_info = soup.find("a", class_ = 'photoLink').img.get('alt').strip()
        price = i.find('div', class_ = 'price').text.strip()
        rooms = i.find('a', class_ = 'lnk1').text.strip()
        location = i.find('a', class_ = 'lnk2').text.strip()
        info = i.find('td', width="520", colspan="3", height="50", style="padding-left:4px").text.strip()

        # dictionary to store the collected information
        property_dictionary_0 = {
                            'property_info' : property_info,
                            'price' : price,
                            'rooms' : rooms,
                            'location' : location,
                            'info' : info
                          }
        property_list.append(property_dictionary_0)



    for i in table_1:
        property_info = soup.find("a", class_ = 'photoLink').img.get('alt').strip()
        price = i.find('div', class_ = 'price').text.strip()
        rooms = i.find('a', class_ = 'lnk1').text.strip()
        location = i.find('a', class_ = 'lnk2').text.strip()
        info = i.find('td', width="520", colspan="3", height="50", style="padding-left:4px").text.strip()

        # dictionary to store the collected information
        property_dictionary_1 = {
                            'property_info' : property_info,
                            'price' : price,
                            'rooms' : rooms,
                            'location' : location,
                            'info' : info
                          }
        property_list.append(property_dictionary_1)



    for i in table_2:
        property_info = soup.find("a", class_ = 'photoLink').img.get('alt').strip()
        price = i.find('div', class_ = 'price').text.strip()
        rooms = i.find('a', class_ = 'lnk1').text.strip()
        location = i.find('a', class_ = 'lnk2').text.strip()
        info = i.find('td', width="520", colspan="3", height="50", style="padding-left:4px").text.strip()

        # dictionary to store the collected information
        property_dictionary_2 = {
                            'property_info' : property_info,
                            'price' : price,
                            'rooms' : rooms,
                            'location' : location,
                            'info' : info
                          }
        property_list.append(property_dictionary_2)


    return len(table_0), len(table_1), len(table_1), len(price_div)

# we create empty list to which we append the information gathered from the three for loops
property_list = []


# These are two list variables that store strings for the municipality and
# the actual name of the municipality
municipality = ['7o9q5q', '7o9q7x']
municipality_translation = ['Veliko Tarnovo', 'Sofia City']


# For loop that does the web scraping
for i in range(len(municipality)):
    x = municipality[i]
    for j in range(1,3):
        print(f'Page number: {j} - {municipality_translation[i]}')
        search = modified_extract(j,x)
        transform(search)

print(len(property_list))
```

```
Page number: 1 - Veliko Tarnovo
Page number: 2 - Veliko Tarnovo
Page number: 1 - Sofia City
Page number: 2 - Sofia City
160
```

What is important is that we have managed to extract all the property listings of two pages for two municipalities. That means that we can scrape all the property listings if we just add the code for each municipality to the list 'municipality'. In addition to that we can see that for the four pages that we scraped the sum of listings is equal to 160, which means that all is working as it should.

What is left to do is to make the code for the 'transform' function more compact and to reduce the amount of code. In that way we will reduce the repetition and improve the quality of the code.

In [19]:
```python
def modified_transform(soup):

    # finds the total number of pages
    # assumes that the page number is 2 digit number
    # span class="pageNumbersInfo">Страница 1 от 25</span>    -   Page 1 of 25

    last_page = soup.find('span', class_ = 'pageNumbersInfo').text.strip()[-2:]
    last_page_number.append(int(last_page))



    # We can find all 'div' tags that store the price of each property listing
    # in that way we can get all 40 listings
    price_div = soup.find_all('div', class_ = 'price')

    # Each listing is associated with a particular style that actually shows if the listing
    # is 'Top', 'Vip' or just ordinary listing
    styles = ["margin-bottom:0px; border-top:#990000 1px solid; background:url(../images/picturess/top_bg.gif);
              "margin-bottom:0px; border-top:#990000 1px solid; background:url(../images/picturess/vip_bg.gif);
              "margin-bottom:0px; border-top:#990000 1px solid;"]


    # The 'table' tags that contain the information do not have class, the only way to distinguish
    # one from another is the 'style' attribute that they have
    # We can create a for loop that will collect automatically all the information instead repeating
    # the process three times for each table style and if something is changed we can easily
    # amend where it is needed.
    for i in range(len(styles)):

        table_x = soup.find_all('table', style=styles[i])

        for j in table_x:
            property_info = soup.find("a", class_ = 'photoLink').img.get('alt').strip()
            price = j.find('div', class_ = 'price').text.strip()
            rooms = j.find('a', class_ = 'lnk1').text.strip()
            location = j.find('a', class_ = 'lnk2').text.strip()
            info = j.find('td', width="520", colspan="3", height="50", style="padding-left:4px").text.strip()

            # dictionary to store the collected information
            property_dictionary = {
                                'property_info' : property_info,
                                'price' : price,
                                'rooms' : rooms,
                                'location' : location,
                                'info' : info
                              }
            property_list.append(property_dictionary)


    return len(price_div), last_page_number[0], len(property_info)


# we create empty list to which we append info
property_list = []

# variable to store the the last page number
last_page_number = []

# FOR INFORMATION
# The URL code for each municipality is different every now and again.
# This is just an example on how to collect them.
# Keep in mind that those codes have expired and will give error,
# because it can not find the page number if the municipality code from the URL is changed.
# municipality = ['7gf7vh', '7gk4sj', '7gjb7j', '7gjay6', '7gk56o', '7gk4gh']
# municipality_translation = ['Veliko Tarnovo', 'Sofia City', 'Sofia Municipality',
#                             'Lovetch', 'Burgas', 'Varna']


# You can see that at the moment 'Veliko Tarnovo' code is '7lzaba' and not '7gf7vh'.
municipality = ['7o9q5q']
municipality_translation = ['Veliko Tarnovo']


# In order to get the last page number we need to run 'modified_extract' function once
# so we can get the value of 'last_page_number' variable and empty the 'property_list' variable
# For information - it turns out that this site has only 25 pages with 40 properties
# per page for each municipality
search = modified_extract(1, '7o9q5q')
modified_transform(search)
# #print(transform(search))
property_list = []
```

The following cell contains two identical in what they do for loops. The difference between them is that the first one finds the last page number, where as the second one has the number hard-coded.

In [20]:
```python
property_list = []


for i in range(len(municipality)):
    for j in range(0, int(last_page_number[0])):
        print(f'Page number: {j} - {municipality_translation[i]}')
        search = modified_extract(j, municipality[i])
        modified_transform(search)


# This is the same for-loop as the one from above, it is just using out knowledge that the last page number is
# for i in range(len(municipality)):
#     for j in range(1, 26, 1):
#         print(f'Page number: {j} - {municipality_translation[i]}')
#         search = modified_extract(j, municipality[i])
#         transform(search)

print(len(property_list))
print(last_page_number)
```

```
Page number: 0 - Veliko Tarnovo
Page number: 1 - Veliko Tarnovo
Page number: 2 - Veliko Tarnovo
Page number: 3 - Veliko Tarnovo
Page number: 4 - Veliko Tarnovo
Page number: 5 - Veliko Tarnovo
Page number: 6 - Veliko Tarnovo
Page number: 7 - Veliko Tarnovo
Page number: 8 - Veliko Tarnovo
Page number: 9 - Veliko Tarnovo
Page number: 10 - Veliko Tarnovo
Page number: 11 - Veliko Tarnovo
Page number: 12 - Veliko Tarnovo
Page number: 13 - Veliko Tarnovo
Page number: 14 - Veliko Tarnovo
Page number: 15 - Veliko Tarnovo
Page number: 16 - Veliko Tarnovo
Page number: 17 - Veliko Tarnovo
Page number: 18 - Veliko Tarnovo
Page number: 19 - Veliko Tarnovo
Page number: 20 - Veliko Tarnovo
Page number: 21 - Veliko Tarnovo
Page number: 22 - Veliko Tarnovo
Page number: 23 - Veliko Tarnovo
Page number: 24 - Veliko Tarnovo
1000
[25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25]
```

We got 25 pages, which is the total amount and 25 times 40 is equal to 1000 and that is the exact number of property listings that we got.

We can just add more municipalities and to the 'municipality' list and it will collect all the data for us.

Then we can save the dataframe as shown before as CSV file and read the CSV into Pandas to start cleaning and analyzing the data.

Lastly, I want to show that it can work for different municipalities, but I will hard code it to take the first 2 pages of each so it will show that it can do the job as it should.

In [21]:
```python
municipality = ['7o9qpr', '7o9q5q', '7o9qs2']
municipality_translation = ['Sofia City', 'Veliko Tarnovo', 'Varna']

last_page_number = []
property_list = []

for i in range(len(municipality)):
#     for j in range(0, int(last_page_number[0])):
    for j in range(0, 2):
        print(f'Page number: {j} - {municipality_translation[i]}')
        search = modified_extract(j, municipality[i])
        modified_transform(search)


print(len(property_list))
print(last_page_number)
```

```
Page number: 0 - Sofia City
Page number: 1 - Sofia City
Page number: 0 - Veliko Tarnovo
Page number: 1 - Veliko Tarnovo
Page number: 0 - Varna
Page number: 1 - Varna
240
[25, 25, 25, 25, 25, 25]
```

We can still check the results and make sure that everything is working fine. 6 * 40 = 240. That is obvious, but let's check what are the municipality names.

In [22]:
```python
# This function is pretty much the same as 'basic_translation' the only difference is that
# this one doesn't have the pronunciation written down, which we do not need.

def basic_translation_list(text_for_translation, target_language):
    '''
    Basic function introducing 'google_trans_new'.
    '''
    # We instantiate 'google_translator' object called 'translator'
    translator = google_translator(url_suffix="bg", timeout=5)

    # In the brackets we put the text we need translated first, then we define 'lang_tgt' parameter
    # that is used to tell to what language we need to translate the original text
    translate = translator.translate(text_for_translation,
                                     lang_tgt = target_language)

    return print(translate)



def dictionary_into_dataframe(dictionary):
    '''
    This function takes as an argument a dictionary and creates a DataFrame using Pandas.
    The function returns as a result the municipalities.
    '''

    # We instantiate a DataFrame object
    dataframe = pd.DataFrame(dictionary)

    # We take only the 'location' column for translation
    dataframe_unique_locations = dataframe['location'].unique()

    # The actual translation
    translation = basic_translation_list(dataframe_unique_locations, 'en')

    return translation
```

In [23]:
```python
dictionary_into_dataframe(property_list)
```

```
['city Sofia, Banishora' 'city Sofia, Vitosha' 'city Sofia, Vrabnitsa 1'
 'city of Sofia, Geo Milev' 'city Sofia, Dianabad' 'city Sofia, Dragalevtsi'
 'city of Sofia, Darvenica' 'city Sofia, Lozenets' 'city Sofia, Lyulin 5'
 'city of Sofia, Lyulin 7' 'city Sofia, Malinova Valley'
 'city of Sofia, Manastirski livadi' 'city Sofia, Mladost 1'
 'city of Sofia, Mladost 3' 'city Sofia, Mladost 4'
 'city of Sofia, modern suburb' 'city Sofia, Hope 1'
 'city of Sofia, Obelya 2' 'city Sofia, Ovcha Kupel 1' 'city Sofia, Pavlovo'
 'city of Sofia, Poduyane' 'city Sofia, the Holy Trinity' 'city Sofia, Slatina'
 'city. Elena, Veliko Tarnovo region '' gr. Zlataritsa, Veliko Tarnovo region
 'city. Pavlikeni, Veliko Tarnovo region
 'city. Polski Trambesh, Veliko Tarnovo region
 's. Balvan, Veliko Tarnovo region '' s. Bridkovtsi, Veliko Tarnovo region
 's. Julyunitsa, Veliko Tarnovo region
 's. The camps, Veliko Tarnovo region Ledenik, Veliko Tarnovo region
 's. Lesichari, Veliko Tarnovo Region '' Pales, Veliko Tarnovo Region
 's. Samepage, Veliko Tarnovo Region
 's. Hotnitsa, Veliko Tarnovo region " Novo selo, Veliko Tarnovo region
 'city. Gorna Oryahovitsa, Veliko Tarnovo region
 'city. Debelets, Veliko Tarnovo region
 'city. White church, Veliko Tarnovo region " Byala, Varna Region
 'kk Kamchia, Varna Region '' Valchi Dol, Varna Region '
 'city. Provadia, Varna Region '' s. Avren, Varna Region '
 's. Gemini, Varna Region '' s. Bolagers, Varna Region '
 's. General Kantardzhievo, Varna Region " Oreshak, Varna Region '
 's. Radevo, Varna Region '' Shkorpilovtsi, Varna Region '
 's. Goritsa, Varna Region "s. Chertricks, Varna Region '
 'city. Beloslav, Varna Region ']
```

If we take a second and look at the result we have from the cell below we can see that the municipalities listed below are as follows: 'city of Sofia', 'Veliko Tarnovo region' and 'Varna Region'. That means that the Web scraper is doing exactly what we want.

The main task that we have specified in the beginning of this Jupyter Notebook is now accomplished. That doesn't mean that we still do not have work to do for the data that we have extracted, but this is for another Jupyter Notebook.

Thank you very much for your attention and time.

Kind regards,

Mladen