TODO

Specification for Google Reviews scraper:

1. Read a line from file 'g-properties.txt' (a property record) to check which reviews should be scraped. The structure of the record is as follows, with ';' separator:

id\_property;property\_type;g\_reviews\_link;last\_scrape\_timestamp

Description of the fields:

id\_property – the identifier of each property

property\_type – the type of each property (hotel, restaurant, etc.)

g\_reviews\_link – the URL to retrieve data and reviews about the property

last\_scrape\_timestamp – the timestamp of the last scrape that was done

Follow stages B and C below.

Then:

If property\_type = 'restaurant' then continue to stages D-I below.

If property\_type = 'hotel' then go to stage J-M below.

1. Get the following data elements as the shown in the Powerpoint presentation 'scraper-Google Reviews-11102021.pptx'. Each index number refers to the number in the presentation:

13 – the name of the reviewer and the link (URL) to his/her profile. Store the name of the reviewer in field 'reviewer\_name' (from the example, the text 'Yuval BarHai' will be stored). Store the link (URL) to the reviewer's profile in field 'reviewer\_link'.

14 – this is the reviewer's role. Store the name of the reviewer in field 'reviewer\_role'.

21 – Go to the reviewer's profile (through the URL link that is retrieved as described in index number 13 above).

22 – this is the number of views of the reviewer's reviews. Store this value in field 'reviewer\_views' (from the example, the value is 629).

23 - this is the number of the reviewer's reviews. Store this value in field 'reviewer\_reviews' (from the example, the value is 23).

24 - this is the number of the reviewer's ratings. Store this value in field 'reviewer\_ratings' (from the example, the value is 26).

25 - this is the number of the reviewer's photos. Store this value in field 'reviewer\_photos' (from the example, the value is 3).

26 - this is the number of the reviewer's videos. Store this value in field 'reviewer\_videos' (from the example, the value is 0).

27 - this is the number of the reviewer's answers. Store this value in field 'reviewer\_answers' (from the example, the value is 3).

28 - this is the number of the reviewer's edits. Store this value in field 'reviewer\_edits' (from the example, the value is 0).

29 - this is the number of the reviewer's places added. Store this value in field 'reviewer\_places' (from the example, the value is 0).

30 - this is the number of the reviewer's roads added. Store this value in field 'reviewer\_roads' (from the example, the value is 0).

31 - this is the number of the reviewer's facts checked. Store this value in field 'reviewer\_facts' (from the example, the value is 0).

32 - this is the number of the reviewer's Q&A. Store this value in field 'reviewer\_Q\_and\_A' (from the example, the value is 0).

33 - this is the number of the reviewer's published lists. Store this value in field 'reviewer\_lists' (from the example, the value is 0).

1. Write a data record in the following structure into file 'g\_reviewers.txt' with ';' separator between fields:

id\_reviewer;reviewer\_link;reviewer\_role;reviewer\_views;reviewer\_reviews;reviewer\_ratings;reviewer\_photos;reviewer\_videos;reviewer\_answers;reviewer\_edits;reviewer\_places;reviewer\_roads;reviewer\_facts;reviewer\_Q\_and\_A;reviewer\_lists;scrape\_timestamp

Where: id\_reviewer is a running number index starting from 1, 2, 3, … for each record, scrape\_timestamp is the timestamp when the scraping data retrieval was done and the other field are retrieved as explained above.

1. Get the following data elements as the shown in the Powerpoint presentation 'scraper-Google Reviews-11102021.pptx' (Slide number 1). Each index number refers to the number in the presentation:

1 – the property name. Store it in field 'name'.

2 – the link to the property's website (linked to the button 'website'). Store it in field 'website'.

3 – the rating given to the property. Store it in field 'rating'.

4 – the number of reviews. Store it in field 'reviews\_num' without the ',' thousands sign (from the example, store the value 1435 and not 1,435!).

5 – the price level. Store it in field 'price' as the number of currency icons (from the example, store the value 2 as there are 2 currency icons together).

6 – short description. Store it in field 'subtext' (from the example, the text 'Russian restaurant' will be stored).

7 – full description. Store it in field 'description' (from the example, the text 'Homey dining room for traditional Russian delicacies & decorated with standard lamps & book cases.' will be stored). If there are several passages store it as a long string without passage characters or passes.

8 - download all the content after the words 'Service options:' (including the characters separating between the words). Store it in field 'service' (from the example, the text 'Dine-in · Curbside pickup · No-contact delivery' will be stored).

9 – address of the restaurant. Store it in field 'address' (from the example, the text 'Spiridon'yevskiy Pereulok, 10a, Moscow, Russia, 123104' will be stored).

10 – phone stored in field 'phone' (from the example, the text '+7 495 650-65-00' will be stored).

1. Write a data record in the following structure into file 'g\_details\_restaurants.txt' with ';' separator between fields:

id\_property;name;address;phone;website;price;subtext;description;service;rating;reviews\_num;scrape\_timestamp

Where: id\_property is the value from the record read from file 'g-properties.txt', scrape\_timestamp is the timestamp of the scrape data retrieval and the other field are retrieved as explained above.

1. Get the following data elements as the shown in the Powerpoint presentation 'scraper-Google Reviews-11102021.pptx' (Slide number 2). Each index number refers to the number in the presentation:

11 – aspect of reviews. Store each aspect in field 'aspect' (from the example, the values are 'chef', 'atmosphere', etc.). Pay attention to download ALL the aspects and not only the first that are shown on the screen – the is a button "+NUMBER" (in the example "+6") to open the other aspects that are not shown in the screen.

12 – the number of times the aspect is mentioned. Store each number in field 'mention\_num' (from the example, the value for 'chef' is 35, etc.). Pay attention to download the numbers for all the aspects and not only the first that are shown on the screen.

1. Write a data record in the following structure into file 'g\_aspects\_restaurants.txt' with ';' separator between fields:

Id\_aspect;id\_property;aspect;mention\_num;scrape\_timestamp

Where:

id\_aspect is a running number index starting from 1, 2, 3, … for each record.

id\_property is the value from the record read from file 'g-properties.txt'.

The other field are retrieved as explained above.

scrape\_timestamp is the timestamp when the scraping was done.

1. Get the following data elements as the shown in the Powerpoint presentation 'scraper-Google Reviews-11102021.pptx'. Each index number refers to the number in the presentation:

15 – the ranking of the review in terms of the number of stars. Store the name of the reviewer in field 'review\_ranking' (from the example, store the value 4).

16 – the time of the review. Store the text in field 'review\_time' (from the example, store the text 'a week ago').

17 – open the 'More' link and retrieve the complete original review text. Store the review text in field 'review\_text' as a single long string (no paragraph jumps) in the original language. -> SEE ALSO Slide 4 in the presentation to see an example of the full text to retrieve!

18 – If a translated text exists after the string '(Translated by Google)' then store the translation text in field 'review\_translation' as a single long string (no paragraph jumps). -> SEE ALSO Slide 5 in the presentation to see an example of the full text to retrieve!

19 – the number of the photographs uploaded and attached to the review. Store the text in field 'review\_photos' after opening all the photos if needed (from the example, store the value 5).

20 – the number of likes given to the review (if no number >= 1 appears then the value is 0). Store the text in field 'review\_likes' (from the example, store the value 0).

34 – if the string '**Service**' exists in the field 'review\_text' then take the text from this field and store it in field 'review\_service' (from the example, the text 'Dine in' will be stored). Then, delete the string '**Service**' and the text that follows it from field 'review\_text'.

35 – if the string '**Meal type**' exists in the field 'review\_text' then take the text from this field and store it in field 'review\_meal' (from the example, the text 'Dinner' will be stored). Then, delete the string '**Meal type**' and the text that follows it from field 'review\_text'.

36 – if the string '**Price per person**' exists in the field 'review\_text' then take the text from this field and store it in field 'review\_price\_per\_person' (from the example, the text '$10-20' will be stored). Then, delete the string '**Price per person**' and the text that follows it from field 'review\_text'.

1. Write a data record in the following structure into file 'g\_reviews\_restaurants.txt' with ';' separator between fields:

id\_review;id\_reviewer;id\_property;review\_ranking;review\_time;review\_text;review\_translation;review\_service;review\_meal;review\_price\_per\_person;review\_photos;review\_likes;scrape\_timestamp

Where:

id\_review is a running number index starting from 1, 2, 3, … for each record

id\_reviewer is the index of the reviewer that wrote it

id\_property is the index of the property that was reviewed.

scrape\_timestamp is the timestamp when the scraping data retrieval was done and the other field are retrieved as explained above.

The rest of the fields are described above.

1. Get the following data elements as the shown in the Powerpoint presentation 'scraper-Google Reviews-11102021.pptx' (Slide number 1). Each index number refers to the number in the presentation:

37 – the property name. Store it in field 'name'.

38 - short description. Store it in field 'subtext' (from the example, the text 'Hotel Avenida Palace' will be stored).

39 – the rating given to the property. Store it in field 'rating'.

40 – the number of reviews. Store it in field 'reviews\_num' without the ',' thousands sign (from the example, store the value 5000 and not 5,000!).

41 – the property type. Store it in field 'type' (from the example, the text '5-star hotel' will be stored).

42 – highlights of the property. Retrieve all the highlights (also those that a continuation arrow should be clicked to show them). Store it in field 'highlights' with a '#' separator sign between them. If a highlight text is cut by a '-' sign and continuous in the next line (like 'great break- fast' in the example) then remove the '-' sign and make it one word. If a highlight is cut over the next line then write it as a single string (like 'Near public transit' in the example).

43 – address of the hotel. Store it in field 'address' (from the example, the text 'R. 1º de Dezembro 123, 1200-359 Lisboa, Portugal' will be stored).

44 – the link to the property's website. Store it in field 'website' (from the example, the text 'hotelavenidapalace.pt' will be stored).

45 – phone stored in field 'phone' (from the example, the text '+351213218100' will be stored).

46 – environment rating. Store it in field 'environment\_rating'.

47 – environment description. Store it as a single string without paragraph jumps in field 'environment\_description'.

48 – click on the 'More' link to receive the complete description of the hotel and on 'View more amenities' to show the whole list of amenities.

49 –full description of the hotel. Store it in field 'description'. If there are several passages store it as a long string without passage characters or passes.

50 – the list of amenities -> write the available amenities (those in darker grey font and with icons different from the 'unavailable' icon next to them) with a '#' separator between amenities. Store the string in field 'amenities'.

51 – take all the categories and write them with a ',' separator between the category text and the rating, and with a '#' separator between one category and the other. Store it in field 'categories' (from the example, the string to be stored will look like 'Families,4.8#Couples,4.6').

1. Write a data record in the following structure into file 'g\_details\_hotels.txt' with ';' separator between fields:

id\_property;name;subtext;rating;reviews\_num;type;highlights;address;website;phone;environment\_rating;environment\_description;description;amenities;categories;scrape\_timestamp

Where: id\_property is the value from the record read from file 'g-properties.txt', scrape\_timestamp is the timestamp of the scrape data retrieval and the other field are retrieved as explained above.

1. Get the following data elements as the shown in the Powerpoint presentation 'scraper-Google Reviews-11102021.pptx'. Each index number refers to the number in the presentation:

52 - open the 'More' link and retrieve the complete original review text.

53 – keep the id\_reviewer of the reviewer that wrote it.

54 - the rating given to the property. Store it in field 'rating'.

55 – the maximal rating given to the property. Store it in field 'max\_rating'.

56 - the time of the review. Store the text before the word 'on' in field 'review\_time' (from the example, store the text 'a month ago').

57 – venue in which the review was published. Store the value after the word 'on' and the icon of the venue in field 'venue' (from the example, store the text 'Google').

58 - open the 'More' link and retrieve the complete original review text. Store the review text in field 'review\_text' as a single long string (no paragraph jumps) in the original language. -> SEE ALSO point 61 below about the translation and SEE ALSO Slide 16 in the presentation to see an example of the full text to retrieve!

59 – the number of photos attached to the review. Open the "+NUMBER" sign to see how many are attached in total (from the example, there are 3 plus 8 photos = 11 photos in total, so 11 will be the value). Store the total number of photos in field 'review\_photos'.

60 – the number of likes given to the review (if no number >= 1 appears then the value is 0). Store the text in field 'review\_likes' (from the example, store the value 0).

61 – If a translated text exists after the string '(Translated by Google)' then store the translation text in field 'review\_translation' as a single long string (no paragraph jumps). -> SEE ALSO Slide 16 in the presentation to see an example of the full text to retrieve!

62 – the type of trip. Store the values in a string with '#' separator between each type in field 'trip' (from the example, the string to store will be 'Family#Couples#Vacation').

63 - if the string '**Location**' exists in the field 'review\_text' then take the number from this field and store it in field 'review\_location' (from the example, the number 5 will be stored). Then, delete the string '**Location**' and the number that follows it from field 'review\_text'.

64 - if the string '**Service**' exists in the field 'review\_text' then take the number from this field and store it in field 'review\_service' (from the example, the number 5 will be stored). Then, delete the string '**Service**' and the number that follows it from field 'review\_text'.

65 - if the string '**Rooms**' exists in the field 'review\_text' then take the number from this field and store it in field 'review\_rooms' (from the example, the number 5 will be stored). Then, delete the string '**Rooms**' and the number that follows it from field 'review\_text'.

1. Write a data record in the following structure into file 'g\_reviews\_hotels.txt' with ';' separator between fields:

id\_review;id\_reviewer;id\_property;rating;max\_rating;review\_time;venue;review\_text;review\_translation;review\_photos;review\_likes;trip;review\_location;review\_service;review\_rooms; scrape\_timestamp

Where:

id\_review is a running number index starting from 1, 2, 3, … for each record

id\_reviewer is the index of the reviewer that wrote it

id\_property is the index of the property that was reviewed.

scrape\_timestamp is the timestamp when the scraping data retrieval was done and the other field are retrieved as explained above.

The rest of the fields are described above.