**Trends scraper**

1. **Setup**

You need to have python installed on your machine. Download it here - <https://www.python.org/downloads/>

You then need to install some dependencies that this script uses, you do it by accessing windows terminal. You then call the following commands in terminal to install them:

python3 -m pip install pytrends

python3 -m pip install requests

python3 -m pip install pandas

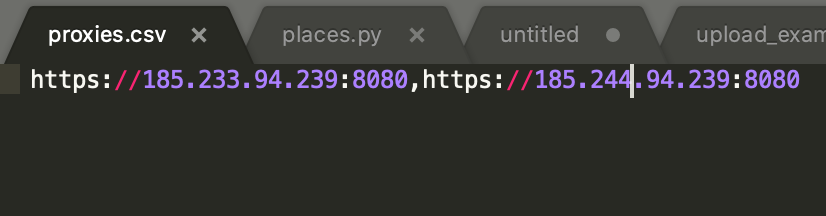
1. **Running the script:**

Run terminal and call

python3 trends.py

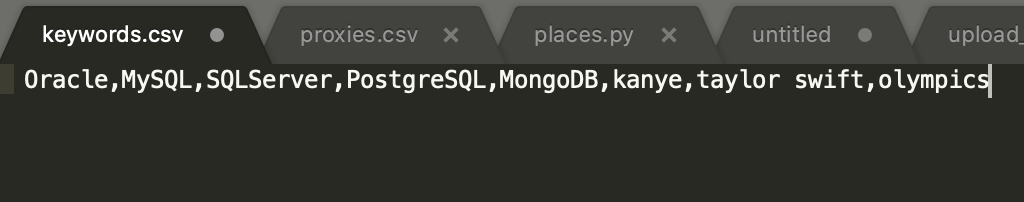
1. **Specify proxies**

If you want to use proxies (and you have to if you want to fetch a lot of keywords) you need to use proxies. The way the library works is that it fetches records 5 at a time – because Trends allows only this many. So for 500 records we make 100 requests to google and it would likely flag us as a bot. Specify proxy addresses in proxies.csv like so, in a comma separated file. The script will then rotate between them when making requests.



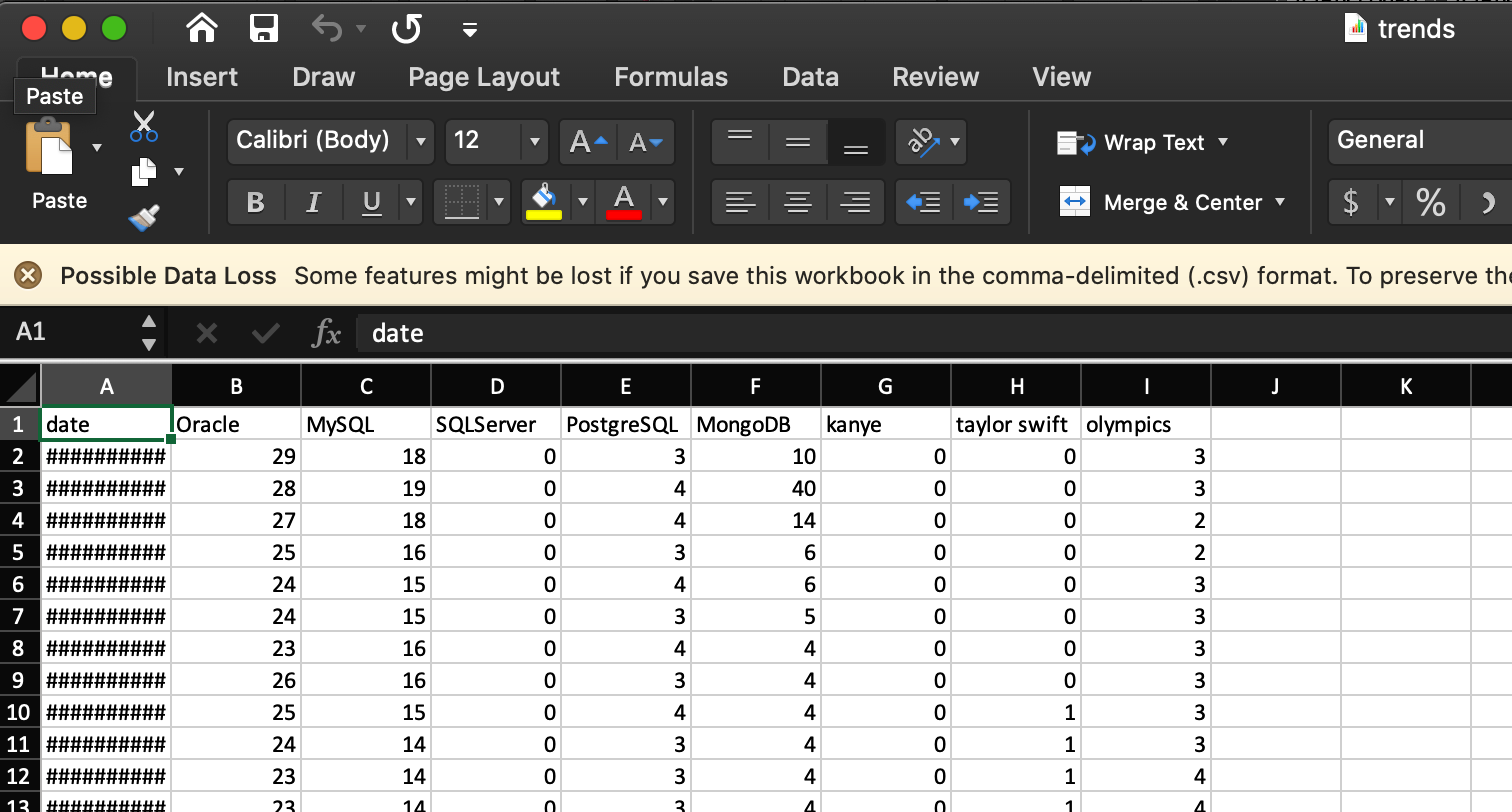
1. **Specify keywords to search**

Like proxies, specify them in a comma separated list in file keywords.csv, example file looks like this



1. **How the script works.**

The script once ran every 1 hour will scrape Trends data for keywords specified in 4. It then puts them in an Excel file trends.csv for manual inspection.



But we also alert the user via Telegram of any detected Trending keywords. Conditions for Trending will be described in 6.

The channel on Telegram the alerts appear in are those:

<https://t.me/joinchat/e4X4rC9C1x42Yjk5>

1. **Config and what it controls**

The main config of the script that dictates how it runs is present in the config.json file, here’s sample content

{

"option":"a",

"avg\_multiplier": 0.5,

"alert\_threshold": 1,

"time\_scale": 10,

"hours\_to\_compare": 5

}

The values dictate how the script works

Option – two possible values ‘a’ or ‘b’. This dictates how we detect whether the keywords is trending.

First we divide the trending values into two sets – time scale ones and last hours we compare against.

For example, if we have following trend values:

26,25,23,24,22,23,23,23,22,22,22,23,23,24,24,24,26,24,26,27,28,29,31

Time scale means how many latest hours we look for to detect trending. Hour\_to\_compare means how many hours before time scale we compare against time scale For example if time\_scale=5 and hours\_to\_compare = 10

26,25,23,24,22,23,23,23,22,22,22,23,23,24,24,24,26,24,26,27,28,29,31

Now that we’ve divided trend values into two sets we do perform the trending detection algorithm depending on the value of option.

**Option A**

We first search for max in hour to compare set, so this 22,22,22,23,23,24,24,24,26,24.

Max is 26. We then go over time scale 26,27,28,29,31 and calculate how many parameters exceed that max. In this case all of them, so 5. We look at alarm threshold, which is set to 1. Since 5 > 1 we alert telegram that keyword is trending.

**Option B**

Here we compare average of those two sets, hours to compare avg=23.4, time scale avg=28.2. We read in multiplier for average and multiple first one against it. Say multiplier is 2. Then we compare values 2\*23.4 and 28.2 .Since time scale has smaller avg then hours to compare times multiplier we do not alert on telegram that keyword is trending.