**EXTRACTING INFORMATION FROM DATASET SEARCH WEBSITE USING SELENIUM AND BEAUTIFUL SOUP**

**INTRODUCTION:**

The Dataset Search website is the Google's search engine that helps researchers locate free-to-use data online. The company launched the service and said the product was targeted at scientists and data journalists.

Selenium Webdriver is a collection of open source APIs used to test Web applications. The tool is used to automate testing of web applications to verify that it is working as expected or not. It supports mostly browsers such as Firefox , Chrome, Safari and Internet Explorer. It also allows for cross-browser checking.

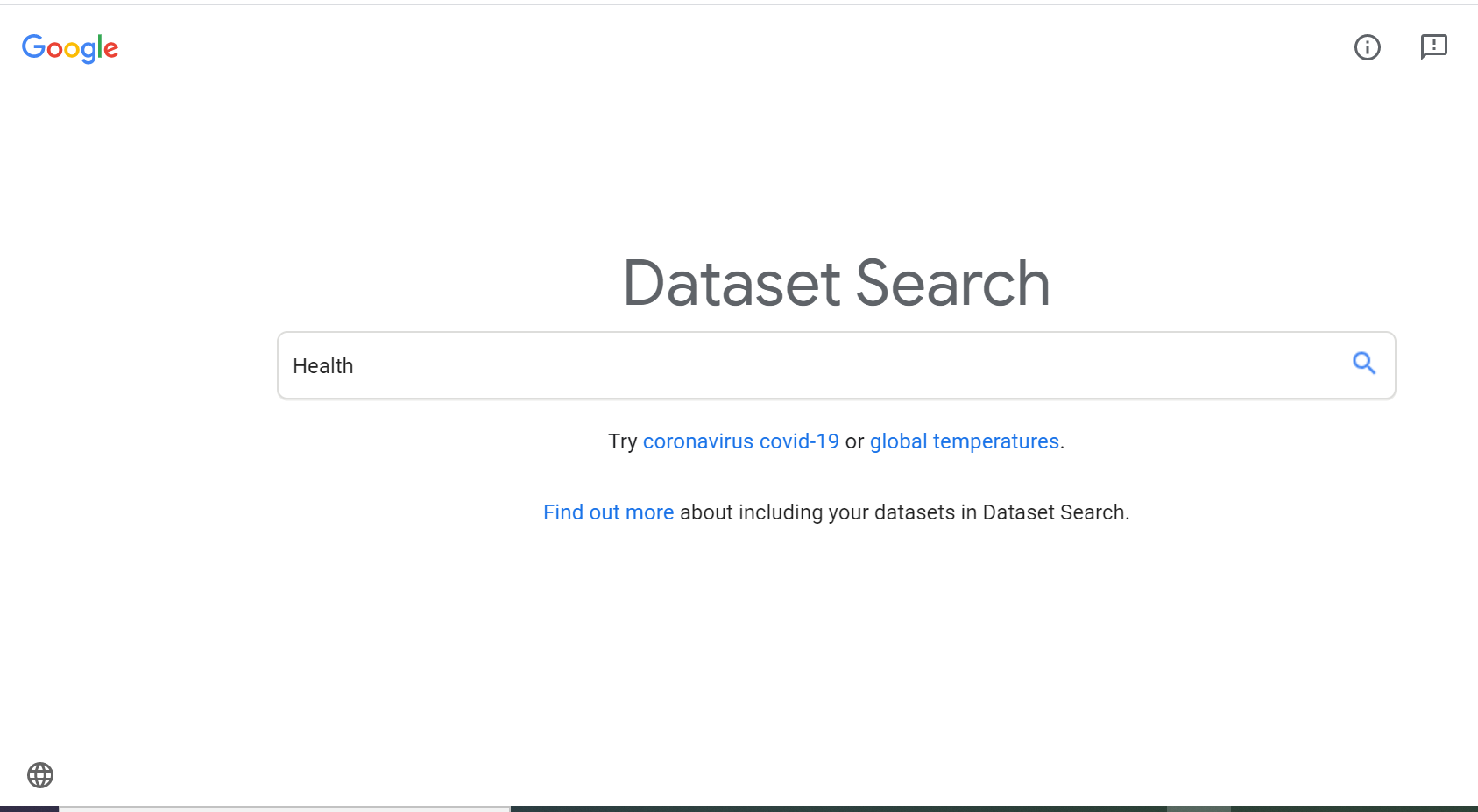
Beautiful Soup is a Python library designed to extract data from both HTML and XML files. Operating with your favorite parser, it offers idiomatic ways to access, scan and change the parser tree.

Docker is a platform designed to facilitate the development, deployment, and running of applications using containers. Containers allow a developer to assemble and deploy an application as one assemble, with all the parts it needs, such as libraries and other dependencies.

**1.DATASEARCH TITLE**

The dataset search website can be accessed through the URL: <https://datasetsearch.research.google.com/>, in order to access the search bar in this website, the below link can be used to <https://datasetsearch.research.google.com/search?query=>, here we can type any related dataset related keywords to check for datasets.

For instance, I have just entered keyword as health, thus URL appears to be [https://datasetsearch.research.google.com/search?query=health](https://datasetsearch.research.google.com/search?query=), this website provides all the dataset related to health.



**2.SELECTING DATASET TITLE**

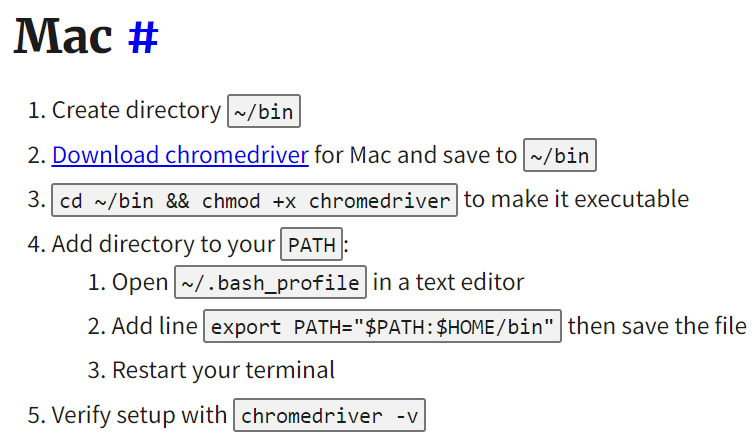
Using beautifulSoup library we can extract the title of each dataset, thus we can retrieve about 100+ datasets title. The HTML parser is used to locate the **H1(header tag)** for all datasets by using a function named as **soup.findAll**, it can fetch all the H1 tag title and then the output of all dataset title can be displayed.

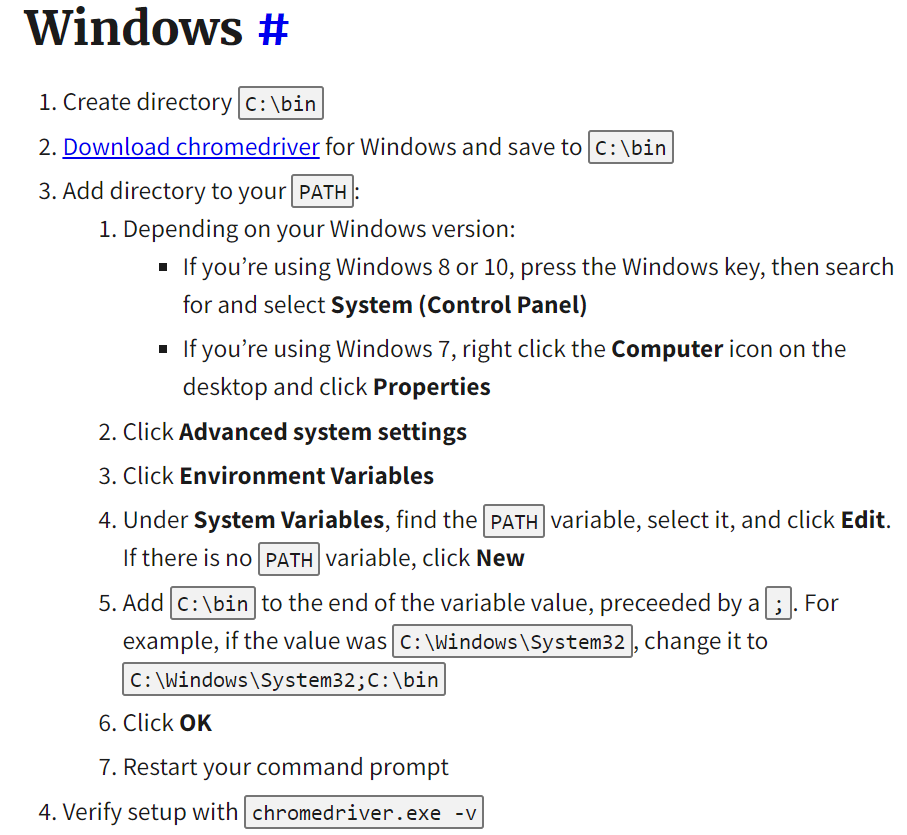


**3.METADATA OF THE DATASET**

**3.1 SELENIUM CHROMEDRIVER**

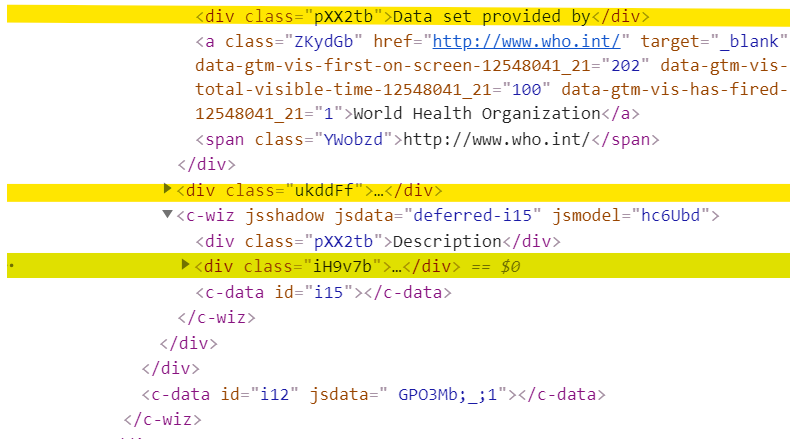
The user can select a title from the list . After the selecting the title, using selenium we must access the chromedriver.exe application to fetch the page. Firstly, we must download the chromedriver.exe application from the website and save the location in environment variables. Import functions are used to access the Webdriver application, thus it can fetch the PATH of the chromedriver.exe application in environment variables(system settings). The title chose by the user will be opened virtually using headless chrome(Open chrome application virtually). Below images are used to download the chromedriver application and store them in the path for both MAC and Windows or docker file can be used to run the chromedriver virtually in all platforms which will be explained in the end.





**3.2 BEAUTIFUL SOUP**

The information in the current webpage can be fetched using beautiful Soup library. The HTML tags can be found through inspecting the webpage(ctrl+shift+I). The metadata of each dataset is included inside the <div> tags in various classes. The DATAINFO function consist of Dataset provided by and area covered information in <div> class = ‘ukddFf’ which can be selected by using SOUP.SELECT function in beautifulSoup. After selecting we must print the result as text, so we use text() function. Similarly, we can extract the CITED BY information, but to extract the link we must use Soup.findAll function to retrieve all HREF links in that div class. Then, using the div class ‘cycj3’ and ‘iH9v7b’ we can extract download format and description of the metadata.



**3.3 DATASET DOWNLOADING LINKS**

The above method can be used to check the metadata of all datasets or user selected datasets. To download the dataset, we must access the div class='WpHeLc'. Each dataset have multiple links to download the dataset, thus we can use Soup.findAll function for all HREF links which provides access to their corresponding websites to download the dataset.



**4. DOCKERFILE**

The developed program should have the ability to run on all platforms(MAC,Linux and Windows). Thus, the solution is provided by Selenium DockerFile. The first step is to download the Docker application in the system using this link <https://www.docker.com/>. After downloading the application, we must access the **selenium/standalone-chrome** docker image in the link <https://hub.docker.com/>. Then, click on the github link to know more about the docker file.

1.To pull the chrome dockerfile use this command in the terminal : **docker pull selenium/standalone-chrome:latest, it will start downloading the images for few seconds.**

**2. Now we must represent a port for our remote webdriver, so execute the following command in terminal : $ docker run -d -p 4444:4444 -v /dev/shm:/dev/shm selenium/standalone-chrome:4.0.0-alpha-6-20200721,which creates a new container.**

**3.Inorder to run our program in the container, use the library in the program:Import selenium from selenium.webdriver.common.desired\_capabilities import DesiredCapabilities and call the driver using following command driver = webdriver.Remote("http://127.0.0.1:4444/wd/hub", DesiredCapabilities.CHROME).**

**5. CONCLUSION**

The Dataset Search website has many datasets which can be accessed for the user in simple ways. Selenium and BeautifulSoup libraries are the best package for accessing information in the webpage. Inspect function is used to retrieve all the tags like div, H1 and P. The text information can be extracted using text() function and which can be displayed as output. The downloading links can also be provided for the dataset. Docker file is used to run the program in all platforms which acts as a virtual computer.