UST 과학기술연합대학원대학교
UNIVERSITY OF SCIENCE & TECHNOLOGY

How to use Python for Big data collection..?

4th week (28th Sep)

- Structure analysis of the Global COVID-19 MAP
- Learning python library to crawl the data

Made by Insung Ahn, Ph.D.

KiSTi 한국과학기술정보연구원
Korea Institute of Science and Technology Information

01

Basics of python

Lesson 1: Let's talk to your computer..

- Michael: Hello, how are you?
- Francesca: Fine thank you, and you?
- Michael: Fine thank you.

- You: How are you?
- Computer: Fine thank you and you?
- You: Fine thank you

How to teach your computer?

1. Create a New Notebook file in JupyterLab
   - ✓ File ➔ New ➔ Notebook
2. Create a New Project in PyCharm
   - ✓ File ➔ New Project ➔ main.py
3. We will teach your computer Three Grammars
   - ✓ input()
   - ✓ if ~ else
   - ✓ print()

input()

1. Write the code

   imVariable = input()
2. Save script as ‘hello.py’
3. Run ‘hello.py’

```
python3 hello.py
(no response)
(type) How are you?
(no response)
```

Why is your computer so SHY..?

input()

4. Upgrade the code

```
imVariable = input('Type here: ')
```

5. Run 'hello.py'

```
python3 hello.py
Type here: (wait)
(type) How are you?
(no response)
```

Now, your computer may **LISTEN**, but they **can't TALK yet..**

print()

6. Upgrade the code

```
imVariable = input('Type here: ')
print(imVariable)
```

7. Run 'hello.py'

```
python3 hello.py
Type here: (wait)
(type) How are you?
(response) How are you?
```

Finally, your computer talk something, but… it's not the answer you wanted

if – elif – else

- If you hear the phrase, 'How are you?'
- Tell (or print out) 'Fine thank you and you?'
- Core grammar is if, elif and else

```
imVariable = input('Type here: ')
if imVariable == 'How are you?':
    print('Fine thank you, and you?')
```

One Tab here == Working area for 'if' method

```
python3 hello.py
Type here: (wait)
(type) How are you?
(response) Fine Thank you, and you?
```

if – elif – else

- Let your computer to make the answer regardless of whether the first character is a capital letter or not

```
imVariable = input('Type here: ‘)
answer = ‘Fine thank you, and you?’
if imVariable == 'How are you?':
   print(answer)
elif imVariable == ‘how are you?’:
   print(answer)
```

```
Variable = input('Type here: ‘)
answer = ‘Fine thank you, and you?’
if imVariable == 'How are you?’ or imVariable == ‘how are you?’:
   print(answer)
```

```
python3 hello.py
Type here: (wait)
(type) how are you?
(response) Fine Thank you, and you?
```

if – elif – else

- Let's send an error message when the question is wrong

```
Variable = input('Type here: ')
answer = 'Fine thank you, and you?'
if imVariable == 'How are you?':
   print(answer)
elif imVariable == 'how are you?':
   print(answer)
else:
   print('Wrong question, sorry')
```

```
Variable = input('Type here: ')
answer = 'Fine thank you, and you?'
if imVariable == 'How are you?' or imVariable == 'how are you?':
   print(answer)
else:
   print('Wrong question, sorry')
```

```
python3 hello.py
Type here: (wait)
(type) how do you do?
(response) Wrong question, sorry
```

Lesson 2: Let your computer to read file..

1. Read the file in your computer line by line.
2. If your computer find something in your file, let you know about it.
3. We will teach your computer Three Grammars
   - handle = open(file name, option)
   - for ~ in~
   - find()

open(file name, mode)

1. Create a sample file named 'readMe.txt'
   - This is a file for reading practice in big data class.
   - The sentence you will search is..
   - Hello Insung (🡐 your name here)
2. Find out the whole path of 'readMe.txt' file in your system.
   - Open your Terminal ➔ move to the directory where 'readMe.txt' file located using 'cd' and 'ls (for Windows, dir)' commands ➔ copy the full path to 'readMe.txt' using 'pwd' in Mac OS (for Windows, you can see the full path in prompt session)

open(file name, mode)

3. Open your file using 'open' command
   - handle = open('_full_path/readMe.txt', 'r')
   - r : read the file w : write to the file
4. Close the file object using 'close()' command
   - handle.close()

Tip! :

- ✓ In the case of Windows system, a backslash(\) is included when displaying a file path (i.e. C:\Documents\), but in python, a backslash in a string is recognized as a special character.
- ✓ In python, special characters within a sentence are recognized as a simple string by adding a backslash in front path (i.e. ;'C:\\Documents\\').

for ~ in ~

1. Read the file line by line using 'for' command

```
handle = open ('_full_path/readMe.txt', 'r')
for line in handle:
    print(line)
handle.close()
```

```
python3 readFile.py
(response) This is a file for reading practice in big data class.
(response) The sentence you are looking for is..
(response) Hello Insung
```

find()

- variable.find(target_string)
  - You can find the value of the first position of a specific character placed in parentheses ( ).

```
>>> a = 'Hello Insung'
>>> a.find('Insung')
6
>>> a.find('Thanin')
-1
```

KiSTi 한국과학기술정보연구원
Korea Institute of Science and Technology Information

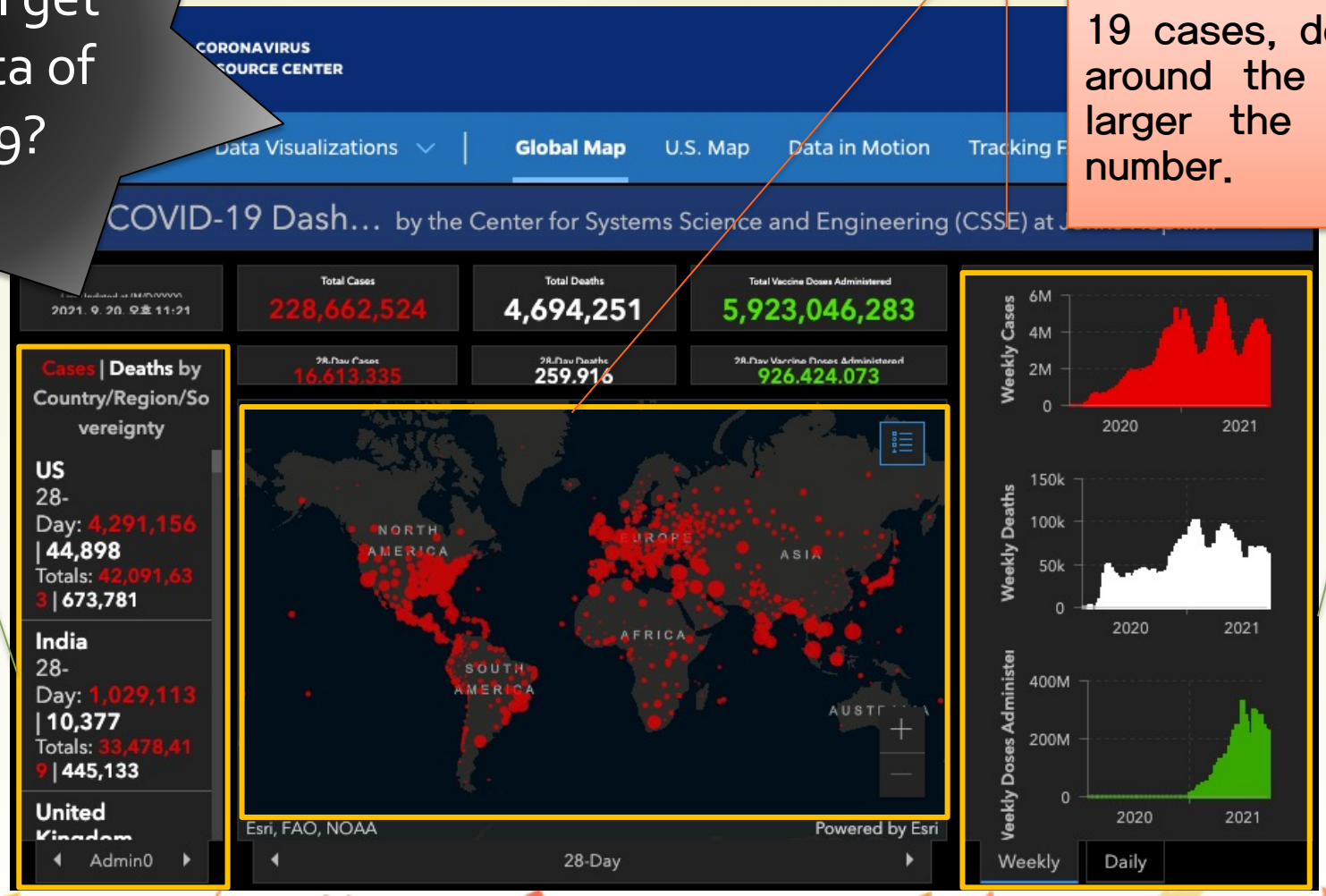
02

Basics of web crawling

COVID-19 Map

Where can I get the raw data of COVID-19?

You can view the status of COVID-19 cases, deaths and vaccinations around the world on a map. The larger the circle, the larger the number.

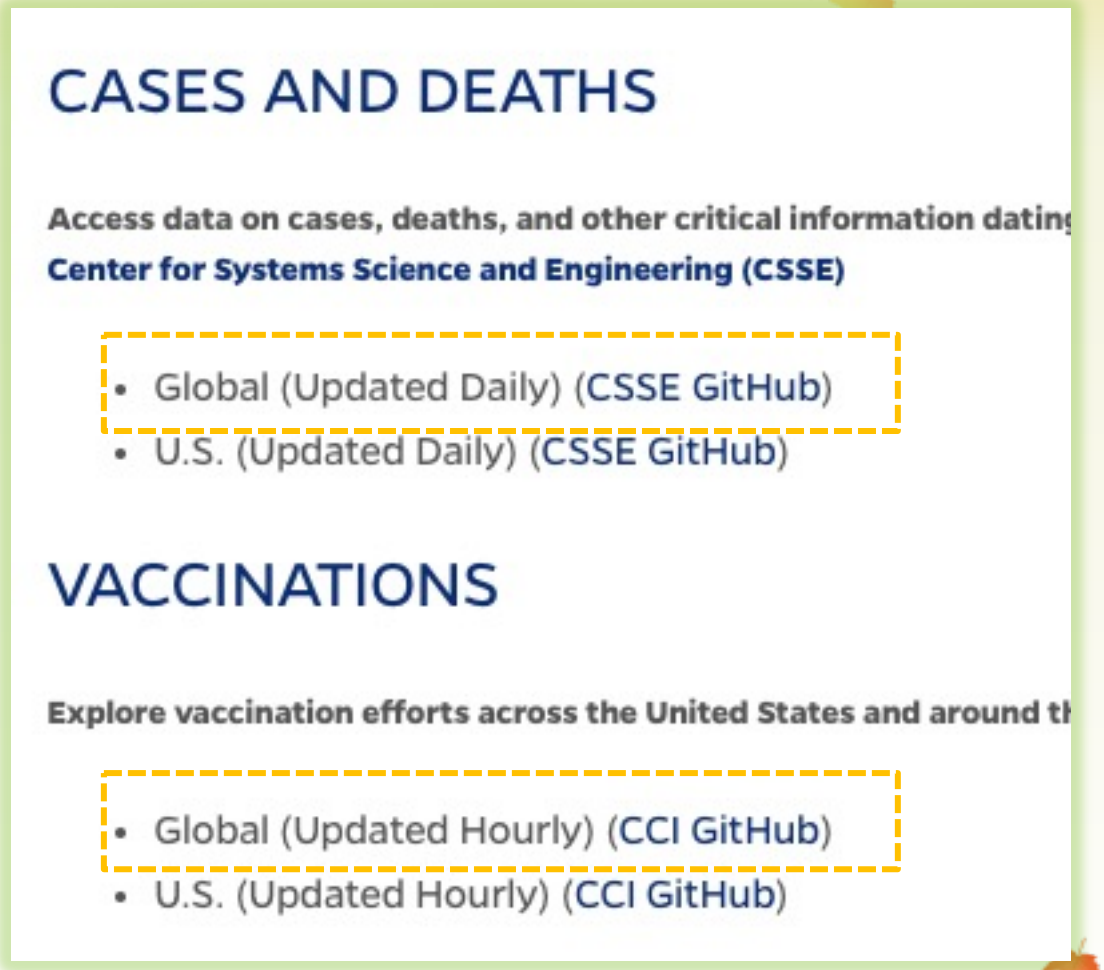


Number of COVID-19 cases and deaths by country or region

You can view the status of COVID-19 confirmed cases, deaths, and vaccinations by week in a graph.

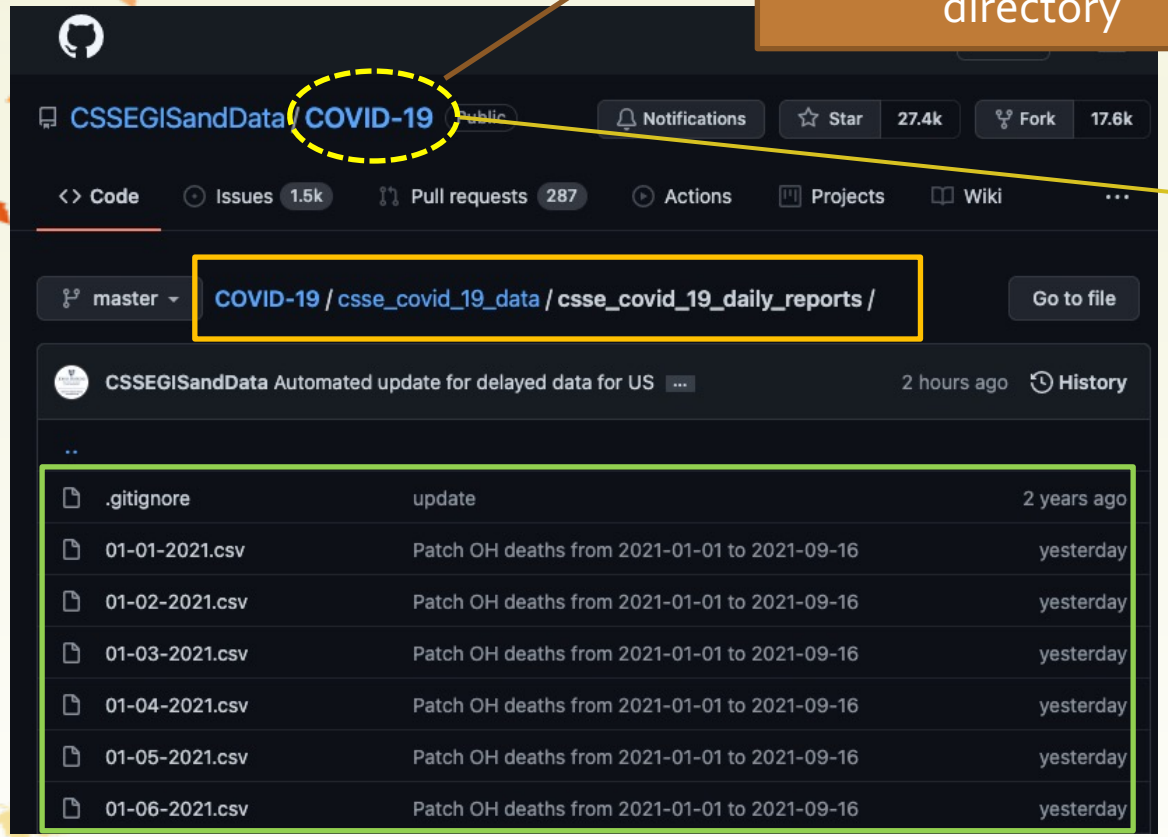
COVID-19 Map

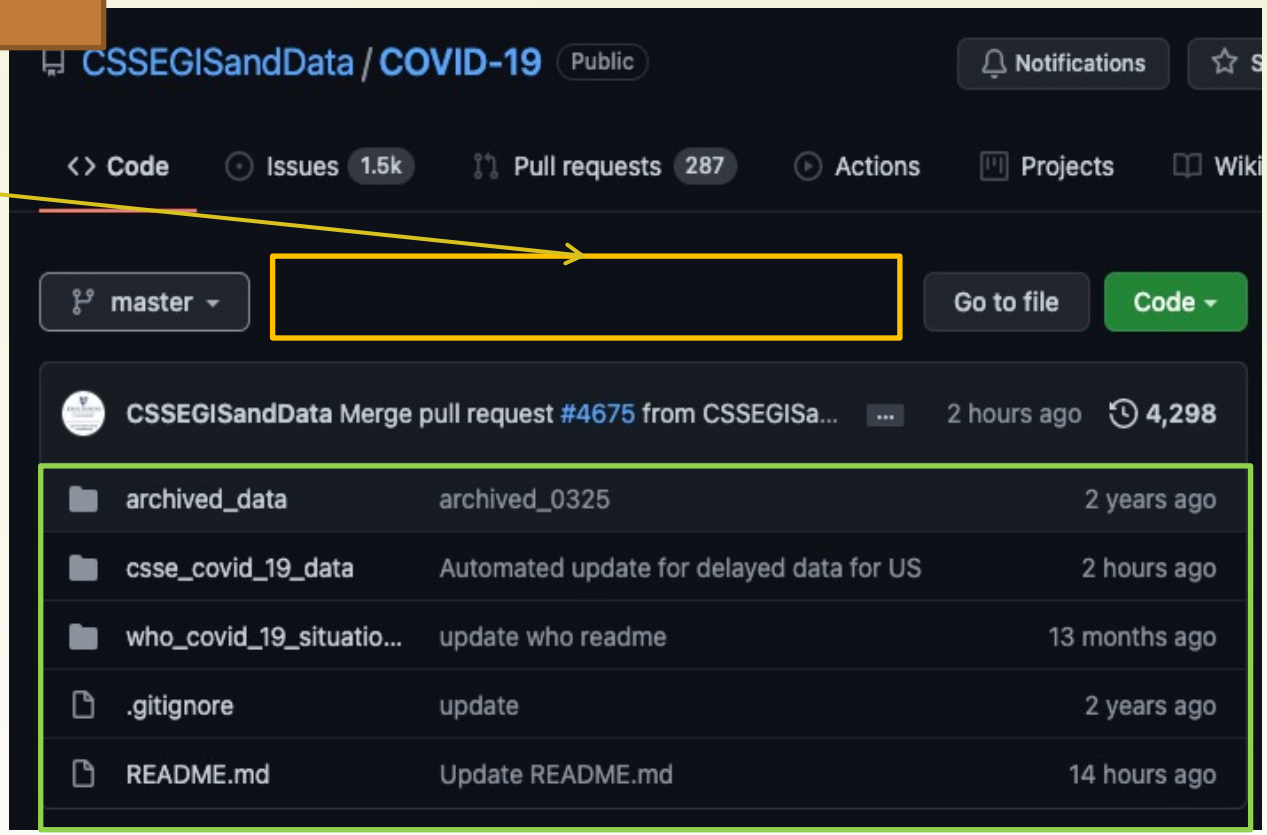

EVENTS & NEWS
Webcasts & Videos
30-Minute COVID-19 Briefing
E-Learning
All News
ABOUT
About Us
World Map FAQ
U.S. Map FAQ
How to Use Our Data


CASES AND DEATHS
Access data on cases, deaths, and other critical information dating
Center for Systems Science and Engineering (CSSE)
Global (Updated Daily) (CSSE GitHub)
U.S. (Updated Daily) (CSSE GitHub)
VACCINATIONS
Explore vaccination efforts across the United States and around th
Global (Updated Hourly) (CCI GitHub)
U.S. (Updated Hourly) (CCI GitHub)

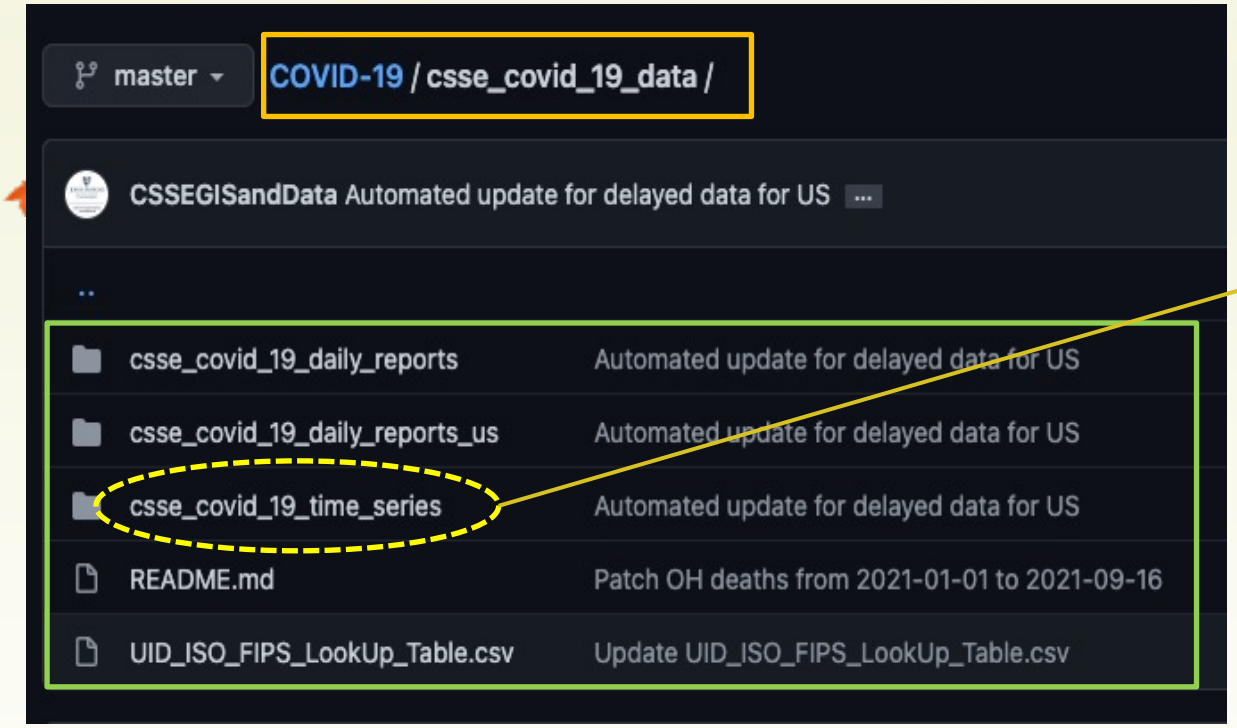
COVID-19 Map

Move to the root directory





COVID-19 Map


master
COVID-19 / csse_covid_19_data /
CSSEGISandData Automated update for delayed data for US
..
csse_covid_19_daily_reports Automated update for delayed data for US
csse_covid_19_daily_reports_us Automated update for delayed data for US
csse_covid_19_time_series Automated update for delayed data for US
README.md Patch OH deaths from 2021-01-01 to 2021-09-16
UID_ISO_FIPS_LookUp_Table.csv Update UID_ISO_FIPS_LookUp_Table.csv


master
COVID-19 / csse_covid_19_data / csse_covid_19_time_series /
CSSEGISandData Automated update for delayed data for India, Pakistan
..
.gitignore update
Errata.csv patch errata
README.md Update README.md
time_series_covid19_confirmed_US.csv Automated update
time_series_covid19_confirmed_global.csv Automated update for delayed data for India, Pakistan
time_series_covid19_deaths_US.csv Automated update
time_series_covid19_deaths_global.csv Automated update for delayed data for India, Pakistan
time_series_covid19_recovered_global.csv Automated update

COVID-19 Map

- The US and other countries are at the country level.
- Global cases:
  - time_series_covid19_confirmed_global.csv
  - time_series_covid19_deaths_global.csv
  - time_series_covid19_recovered_global.csv

COVID-19 Vaccination data

CASES AND DEATHS

Access data on cases, deaths, and other critical information datin
Center for Systems Science and Engineering (CSSE)

- Global (Updated Daily) (CSSE GitHub)
- U.S. (Updated Daily) (CSSE GitHub)

VACCINATIONS

Explore vaccination efforts across the United States and around t

- Global (Updated Hourly) (CCI GitHub)
- U.S. (Updated Hourly) (CCI GitHub)

master COVID-19 / data_tables / vaccine_data / global_data / readme.md Go to file

marycvaughan Added sources for Bahrain; Estonia; Greenland; Kazakhsta... Latest commit 978f094 on 11 Aug History

2 contributors

100 lines (94 sloc) 8.7 KB Raw Blame

International vaccine data

Files in this folder

- time_series_covid19_vaccine_global.csv: Contains time series data. Each row is uniquely defined by `country` and `date`. Long format.
- time_series_covid19_vaccine_doses_admin_global.csv: Contains time series da Each row is uniquely defined by `country` and `date` Wide format.
- vaccine_data_global.csv: Contains the most recent data collected for is uniquely defined by `country`

Unfortunately, we can not access directly…

master COVID-19 / data_tables / vaccine_data / global_data /

jhu-crc-data-bot Automated hourly vaccination data products update

data_dictionary.csv	Update data_dictionary.csv
readme.md	Added sources for Bahrain; Estonia; Greenland; Kazakh
time_series_covid19_vaccine_dose...	Automated JHU vaccination time series update
time_series_covid19_vaccine_globa...	09-20-21 vaccine time series daily update
vaccine_data_global.csv	Automated hourly vaccination data products update

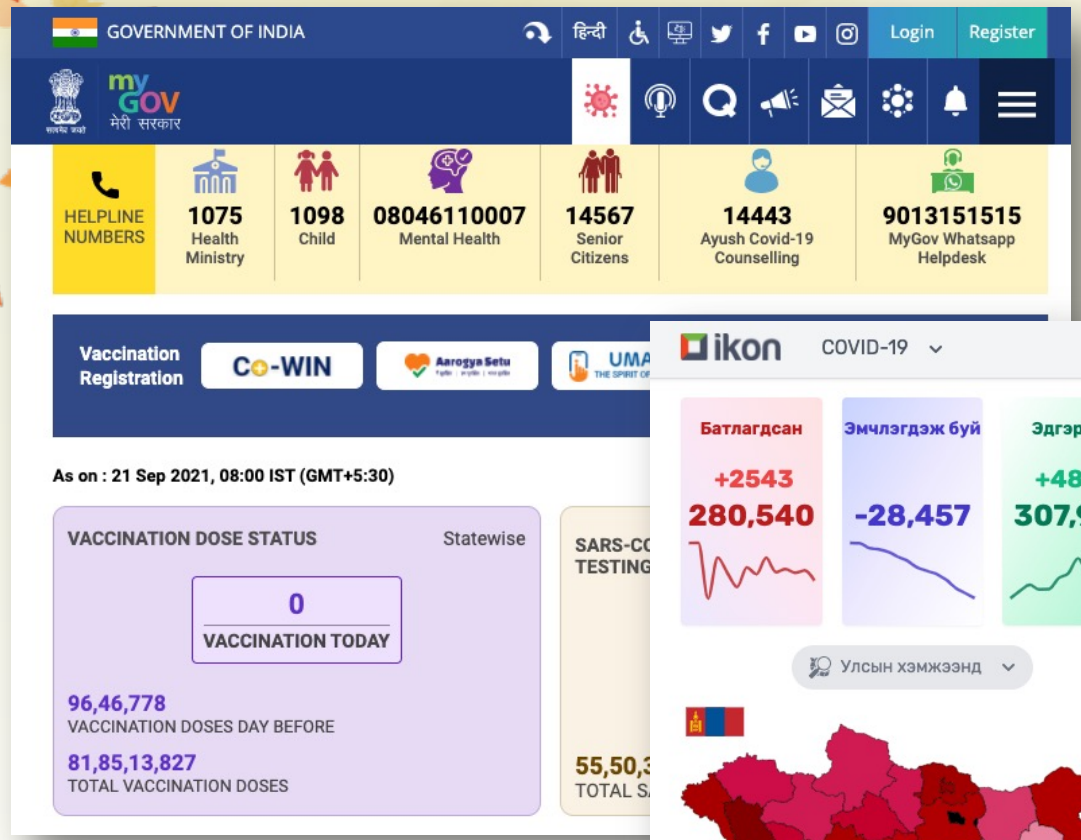
COVID-19 Vaccination data

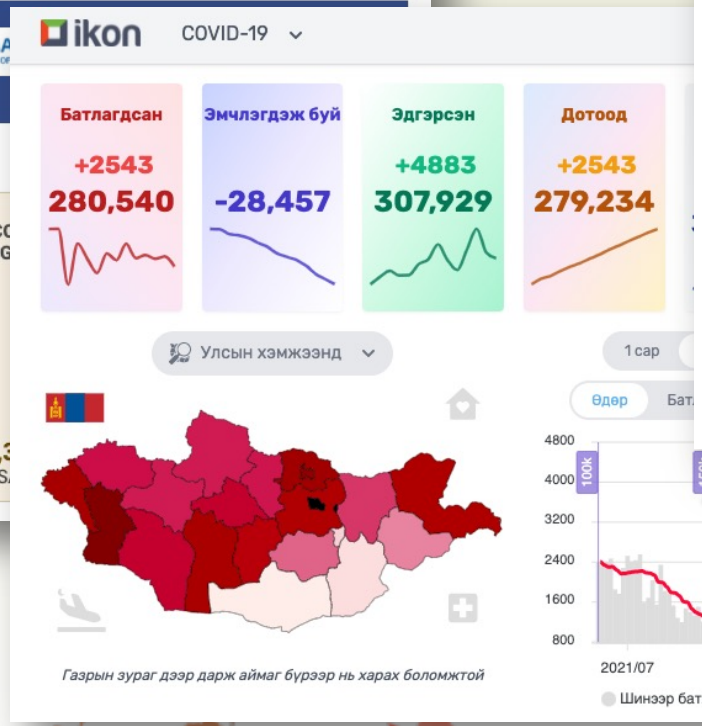
- Aggregated data sources:
  - US Centers for Disease Control and Prevention (CDC): https://covid.cdc.gov/covid-data-tracker/#vaccinations
  - Our World in Data (OWiD): https://ourworldindata.org/covid-vaccinations
  - World Health Organization (WHO): https://covid19.who.int/who-data/vaccination-data.csv

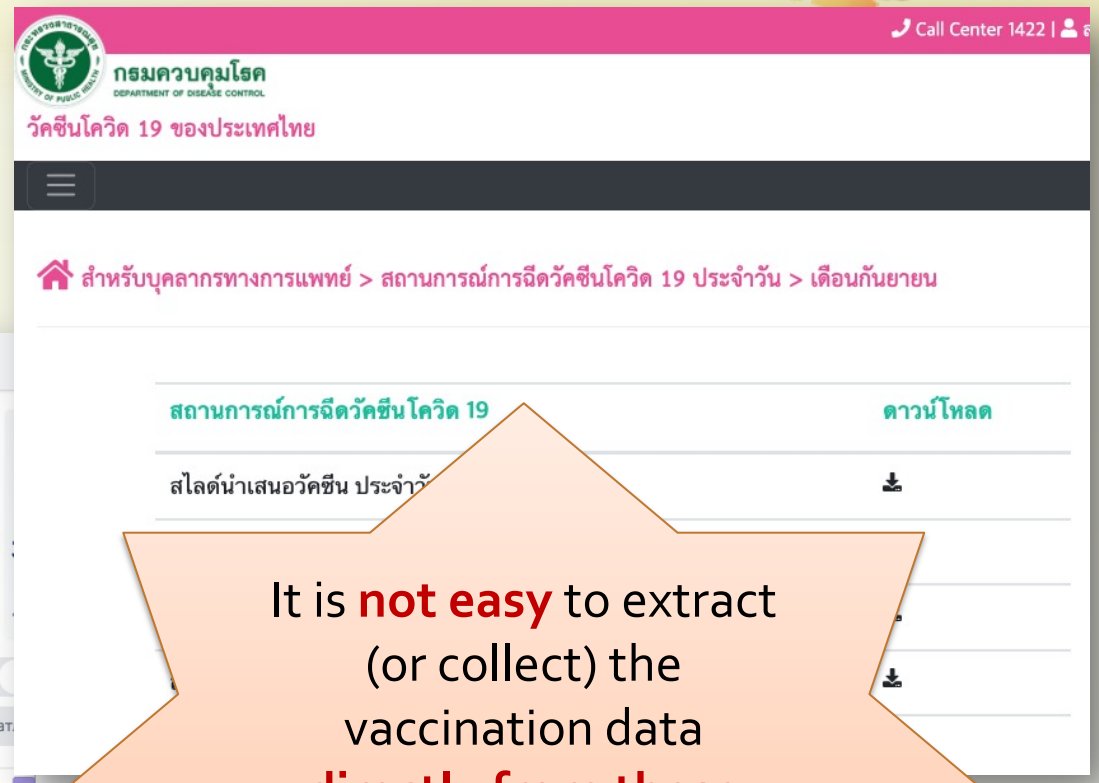
This is the list of source links for vaccination data

- Argentina: Ministry of Health: http://datos.salud.gob.ar/dataset/vacunas-contra-covid-19-dosis-aplicadas-en-la-republica-argentina/archivo/b4684dd9-3cb7-45f7-9c0e-086550013e22
- Australia: COVID Live: https://covidlive.com.au/vaccinations
- Austria: Department of Health: https://info.gesundheitsministerium.gv.at/?re=opendata
- Bahrain: Ministry of Health: https://healthalert.gov.bh/en/
- Bangladesh: Directorate General of Health Services: http://103.247.238.92/webportal/pages/covid19-vaccination-update.php
- Belgium: Institute of Health (Sciensano): https://covid-vaccinatie.be/en
- Bolivia: Ministry of Health and Sports: https://www.minsalud.gob.bo/
- Brazil: Ministry of Health: https://qsprod.saude.gov.br/extensions/DEMAS_C19Vacina/DEMAS_C19Vacina.html
- Bulgaria: Unified Information Portal: https://coronavirus.bg/bg/statistika
- Canada: COVID-19 Tracker: https://covid19tracker.ca/vaccinationtracker.html

COVID-19 Vaccination data



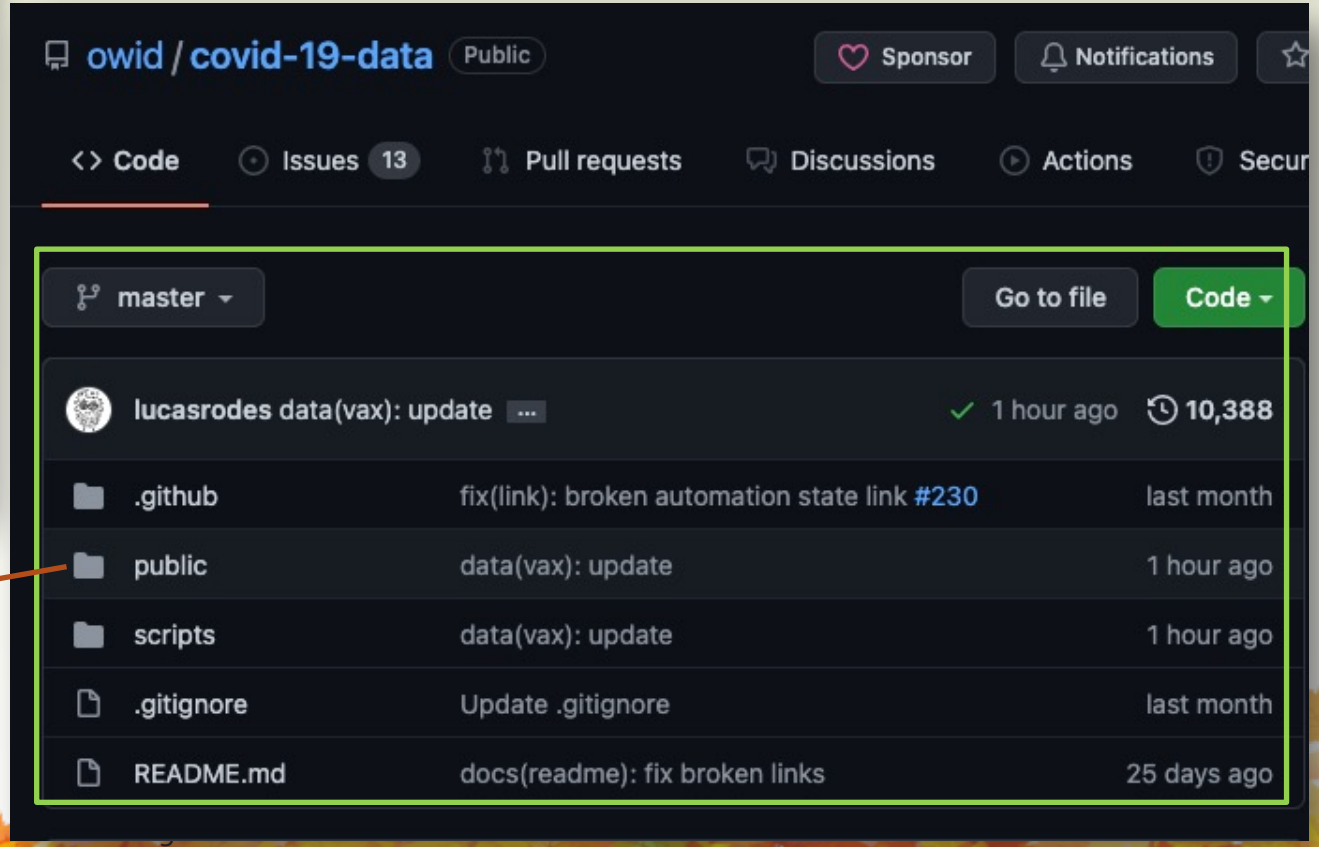




It is **not easy** to extract (or collect) the vaccination data **directly from these source sites..**

OWID COVID-19 Vaccine Data

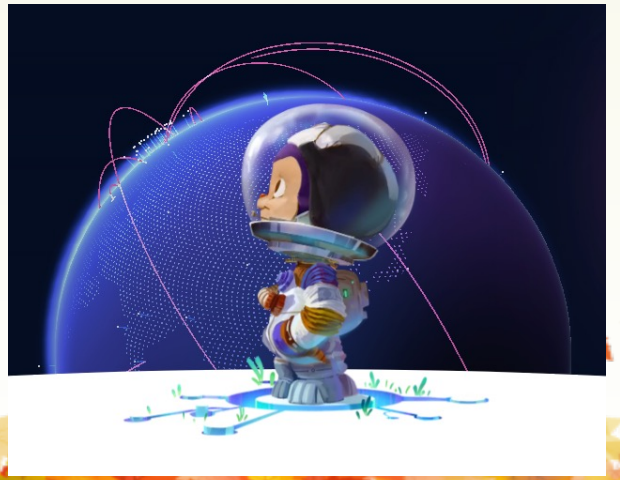
- **Data sources:** at the end of this page you find a detailed list of all our country-specific sources.
- **Open access:** as with all of our data, we are making this dataset openly available, so that everyone can check and use the data that we bring together. You find the vaccination data in our daily-updated repository on GitHub.



For more detailed things for vaccination data, we will discuss later...

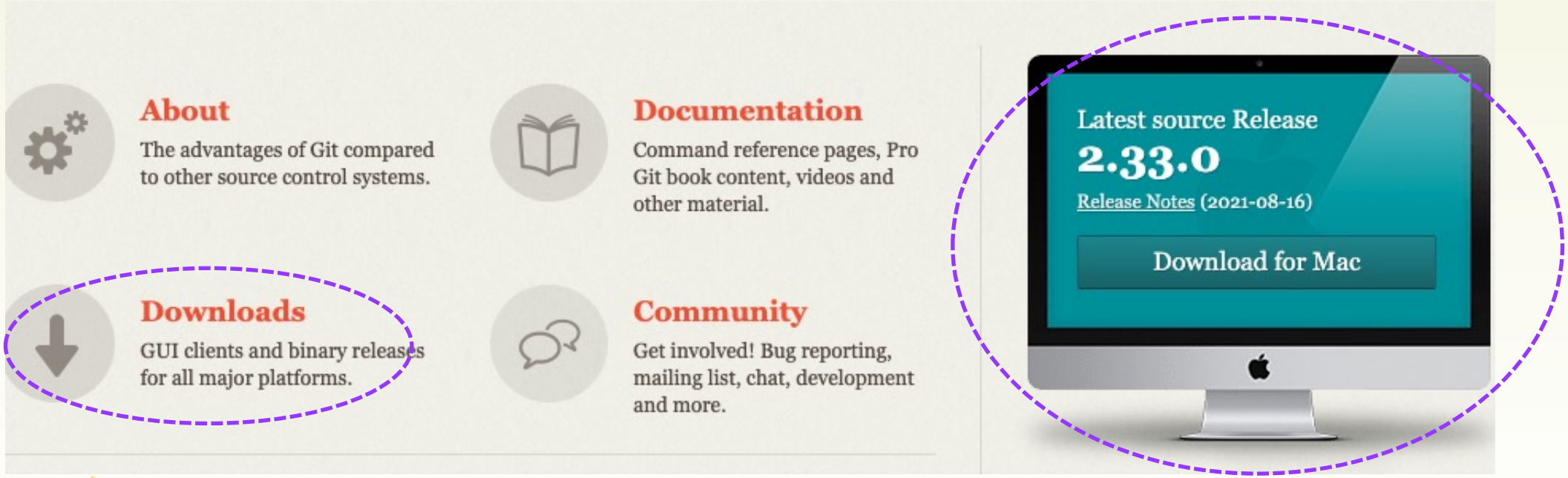
What is Git..?

- Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.
- Projects on GitHub can be accessed and manipulated using the standard Git command-line interface, and all standard Git commands work with this interface.



Installation of Git

- Visit the Git Homepage ➔ Download the installation file (Windows/Mac OS) ➔ Install according to the instructions



git clone

- git clone:
  - If you use git clone, it copies the repository to the current directory as it is on github.
- Assignment:
  - Create a command to download COVID-19 data via gitHub.

KiSTi 한국과학기술정보연구원
Korea Institute of Science and Technology Information

03

pandas DataFrame

What we will do here is..

1. We will make a simple csv file.
2. Open the csv file as pandas library's DataFrame format.
3. Connect to the MySQL in python
4. Create a table named 'myFirstTable' in MySQL to store the data
5. Generate SQL script to input your data into MySQL
6. Send SQL script to MySQL Server

DataFrame in Pandas library

- Pandas stands for Python Data Analysis Library.
- This provides R's Dataframe in a form that can be used in python.
- Pandas Dataframe is very convenient because data can be processed in the form of a table.

Create DataFrame

A dictionary type is an unordered set mapped to immutable keys and mutable values.

Using List

```
import pandas as pd

frame = pd.DataFrame([[1,2,3],[4,5,6],[7,8,9]])
frame
```

	0	1	2
0	1	2	3
1	4	5	6
2	7	8	9

Using Dictionary

```
import pandas as pd

data = {
    'age' : [29, 33, 39],
    'height' : [169, 170, 182],
    'weight' : [70, 74, 80]
}
indexName = ['Insung', 'Thanin', 'Pyunghwa']

frame = pd.DataFrame(data, index=indexName)
frame
```

	age	height	weight
Insung	29	169	70
Thanin	33	170	74
Pyunghwa	39	182	80

Search from DataFrame

- Column search

```
print(frame['age'])
```

```
Insung      29
Thanin      33
Pyunghwa    39
Name: age, dtype: int64
```

```
print(frame.age)
```

```
Insung      29
Thanin      33
Pyunghwa    39
Name: age, dtype: int64
```

	age	height	weight
Insung	29	169	70
Thanin	33	170	74
Pyunghwa	39	182	80

- Row search

```
print(frame.loc['Thanin'])
```

```
age          33
height      170
weight       74
Name: Thanin, dtype: int64
```

```
print(frame.iloc[1])
```

```
age          33
height      170
weight       74
Name: Thanin, dtype: int64
```

Add a new column & row

- Add a new column named 'new_col'
- Add a new row named 'new_row'

```
frame_col_added = pd.DataFrame(frame, columns = ['age','height','weight','new_col'])
frame_col_added
```

	age	height	weight	new_col
Insung	29	169	70	NaN
Thanin	33	170	74	NaN
Pyunghwa	39	182	80	NaN

	age	height	weight
Insung	29	169	70
Thanin	33	170	74
Pyunghwa	39	182	80

```
frame_row_added = frame_col_added.copy()
frame_row_added.loc['Tanus'] = [40, 200, 97, 'new1']
frame_row_added
```

	age	height	weight	new_col
Insung	29	169	70	NaN
Thanin	33	170	74	NaN
Pyunghwa	39	182	80	NaN
Tanus	40	200	97	new1

Save DataFrame as csv format

```
frame_row_added.to
```

- to_clipboard
- to_csv
- to_dict
- to_excel
- to_feather
- to_gbq
- to_hdf
- to_html
- to_json
- to_latex

```
,age,height,weight,new_col
Insung,29,169,70,
Thanin,33,170,74,
Pyunghwa,39,182,80,
Tanus,40,200,97,new1
```

df_to_csv

	age	height	weight	new_col
Insung	29	169	70	
Thanin	33	170	74	
Pyunghwa	39	182	80	
Tanus	40	200	97	new1

Read csv as DataFrame

```
df_from_csv = pd.rea('df_to_csv.csv')
```

```
read_clipboard
read_csv
read_excel
read_feather
read_fwf
read_gbq
read_hdf
read_html
read_json
read_orc
```

```
df_from_csv
```

	Unnamed: 0	age	height	weight	new_col
0	Insung	29	169	70	NaN
1	Thanin	33	170	74	NaN
2	Pyunghwa	39	182	80	NaN
3	Tanus	40	200	97	new1

Set index name

```
df_from_csv.set_index('Unnamed: 0', inplace = True)
```

```
df_from_csv
```

Unnamed: 0	age	height	weight	new_col
Insung	29	169	70	NaN
Thanin	33	170	74	NaN
Pyunghwa	39	182	80	NaN
Tanus	40	200	97	new1

KiSTi 한국과학기술정보연구원
Korea Institute of Science and Technology Information

Assignments

1. Make your first python script with what you practiced today.
2. Collect data from the Johns Hopkins COVID-19 DashBoard using the Git clone command.
3. Homework assigned every week should be included in the midterm exam presentation.

THANK YOU