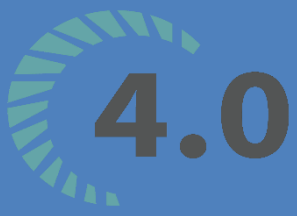
ĐẠI HỌC KHOA HỌC TỰ NHIÊN THÀNH PHỐ HỒ CHÍ MINH, ĐẠI HỌC QUỐC GIA TP HCM

MÔN TRỰC QUAN HÓA DỮ LIỆU

LAB1: MỐI QUAN HỆ CỦA DỮ LIỆU



GV hướng dẫn: Bùi Tiến Lên, Lê Ngọc Thành

ĐỒ ÁN/BÀI TẬP MÔN HỌC - TRỰC QUAN HÓA DỮ LIỆU

HỌC KỲ II – NĂM HỌC 2021-2022

KHOA CÔNG NGHỆ THÔNG TIN
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
227 Nguyễn Văn Cừ, Phường 4, Quận 5, TP.HCM
Điện Thoại: (08) 38.354.266 - Fax:(08) 38.350.096

BẢNG THÔNG TIN CHI TIẾT NHÓM

Số lượng:	**3**		
MSSV	**Họ tên**	**Email**	**Điện thoại**
19120559	Hà Duy Lãm	19120559@student.hcmus.edu.vn	0948311305
19120545	Lê Ngọc Khoa	19120545@student.hcmus.edu.vn	0337175835
19120677	Nguyễn Diệp Minh Tiến	19120677@student.hcmus.edu.vn	0939437078

Bảng phân công & đánh giá hoàn thành công việc			
Công việc thực hiện	**Người thực hiện**	**Mức độ hoàn thành**	**Đánh giá của nhóm**
Làm báo cáo, làm heat map	Hà Duy Lãm	100%	10/10
Làm Pie Chart và Dot and Line Chart	Lê Ngọc Khoa	100%	10/10
Làm Bar chart và Stack Vertical Bar Char	Nguyễn Diệp Minh Tiến	100%	10/10

MỤC LỤC

I. Tiền xử lý:

1. Thêm các thư viện cần thiết và lấy dữ liệu:

- Thêm các thư viện cần thiết:

THÊM THƯ VIỆN

```
import requests
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from bs4 import BeautifulSoup
import lxml.html as lh
import seaborn as sns
```

- Lấy dữ liệu thống kê trên trang web www.worldometers.info bằng code Python để lấy dữ liệu và lưu trữ file dưới định dạng csv.

Lấy dữ liệu hàng ngày

```
URL = 'https://www.worldometers.info/coronavirus/'
page = requests.get(URL)
soup = BeautifulSoup(page.content, 'html.parser')
table = soup.find(id='nav-tabContent')
table = table.find(id = 'nav-today')
table = table.find(id = '')
table = table.find(id = 'main_table_countries_today')
table_rows = table.find_all('tr')
l = []
for tr in table_rows:
    td = tr.find_all('td')
    row = [tr.text for tr in td]
    if (len(row) == 0 ):
        continue
    row = row[:10]
    l.append(row)

dataset = pd.DataFrame(l, columns=["Ranking","Country","Total Cases","New Cases","Total Deaths","New Deaths","Total Recoverd","Ne
dataset
```

- Dữ liệu trước khi xử lí được lưu trữ trong file csv với định dạng tên CoronaData_{Y/m/d}.csv

Lưu dữ liệu hàng ngày

```
import datetime
datestr = datetime.date.today().strftime("%Y%m%d")
dataset.to_csv('CoronaData_{}.csv'.format(datestr))
dataset
```

2. Hàm xử lý dữ liệu.

- Tiền xử lí dữ liệu:

Tiền xử lí dữ liệu

```
dataset.info()
def dataframeCleaner(dataset):

    for columnname in dataset:
        temp = []
        for column in dataset[columnname]:
            column = str(column)
            column = column.replace(',','')
            column = column.replace('+','')
            try:
                column = int(column)
            except:
                pass

            temp.append(column)
        dataset[columnname] = temp

    dataset = dataset.replace('N/A', '', regex=True)
    dataset = dataset.replace(r'^\s*$', 0, regex=True)
    dataset.replace(['\n'], '', regex=True, inplace=True)
    dataset.replace([','], '', regex=True, inplace=True)
    return dataset
```

- Các dữ liệu sau khi được xử lý.
  - **a.** Loại bỏ các ký tự rác như: '\n\', '+', ','…
  - **b.** Loại bỏ các giá trị không xác định như NaN thay bằng 0
  - **c.** Chỉnh lại các kiểu dữ liệu cho dữ liệu
- Dữ liệu sau khi xử lí được lưu trữ trong file csv với định dạng tên CoronaData_{Y/m/d}_Daxuli.csv
- Dữ liệu sau xử lí:

Lưu dữ liệu sau xử lí

```
dataset = dataframeCleaner(dataset)
datestr = datetime.date.today().strftime("%Y%m%d")
dataset.to_csv('CoronaData_{}_Daxuli.csv'.format(datestr))
dataset
```

	Ranking	Country	Total Cases	New Cases	Total Deaths	New Deaths	Total Recoverd	New Recoverd	Active Cases	Serious Cases
0	0	North America	98656294	0	1462943	0	94348717	0	2844634	7207
1	0	Asia	148451844	79181	1425428	171	126415821	65544	20610595	12645
2	0	South America	56879752	0	1294985	0	53031574	0	2553193	10756
3	0	Europe	192059241	0	1821484	0	178300163	172138	11937594	8864
4	0	Oceania	7385763	48560	10872	67	6858416	9096	516475	170
...	...	...	...	...	...	...	...	...	...	...
239	0	Total:	192059241	0	1821484	0	178300163	0	11937594	8864
240	0	Total:	7385763	48560	10872	67	6858416	9096	516475	170
241	0	Total:	11921074	0	253860	0	11120021	0	547193	964
242	0	Total:	721	0	15	0	706	0	0	0
243	0	Total:	515354689	127741	6269587	238	470075418	246742	39009684	40606

3. Đọc dữ liệu trong file csv lên để bắt đầu trực quan.

Lấy dữ liệu ngày 4/5/2022 để trực quan

```
: dataset = pd.read_csv('CoronaData_20220505_Daxuli.csv')
  dataset
```

II. Pie Chart:

A. Pie Chart là gì?

- Pie chart là biểu đồ dạng hình tròn thể hiện mối quan hệ theo phần trăm giữa các phần so với tổng thể.
- Biểu đồ có dạng tròn gồm những phần được chia nhỏ có màu sắc (hoặc kí hiệu) khác nhau, ứng với những đối tượng được phân tích.
- Bên góc biểu đồ thường có chú thích làm rõ hơn về đối tượng. Đơn vị thường gặp trong dạng biểu đồ này là phần trăm

B. Lựa chọn dữ liệu:

- Đối với loại biểu đồ này, và dựa vào bảng dữ liệu. Nhóm quyết định chọn 3 trường dữ liệu là Total Recoverd, Total Deaths, Active Cases từ dòng có “Country, Other” = “World” (là tổng hợp từ tất cả các nước trên thế giới) để trực quan hóa cũng như quan sát mối liên hệ giữa các trường với nhau

C. Code Python:

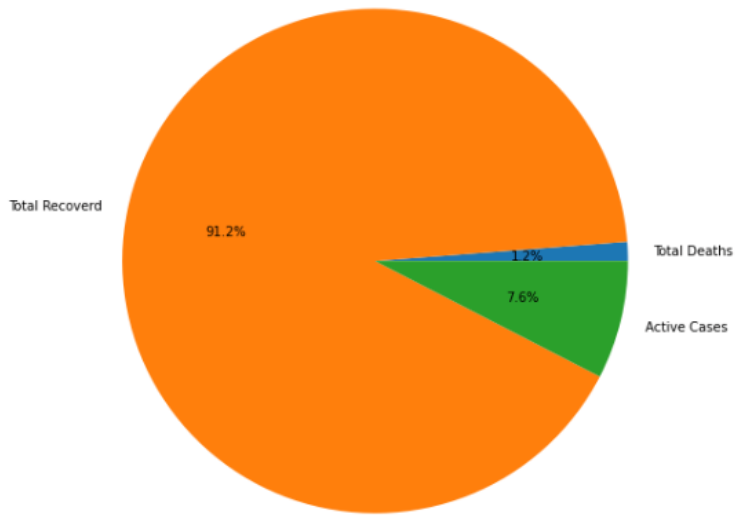
- Để vẽ biểu đồ Pie Chart, sử dụng thư viện pandas.
- Lọc ra các cột dữ liệu, dòng dữ liệu cần thiết đối với biểu đồ, loại bỏ những giá trị thừa, không cần thiết, làm biểu đồ không được chính xác.

D. Biểu đồ:

Pie chart

```
labels = ["Total Deaths","Total Recoverd","Active Cases"]
world_total = dataset[dataset["Country"] == "World"][labels]
world_data = world_total[labels].values[0].tolist()
print(world_total)
world_series = pd.Series(world_data,index = labels, name = "")
plot = world_series.plot.pie(figsize = (9,9),autopct='%1.1f%%',subplots = True)
plt.title('Biểu đồ thể hiện tỉ lệ số ca khỏi bệnh, số ca tử do mắc bệnh, số ca cò
```

```
   Total Deaths  Total Recoverd  Active Cases
7       6269602       470078613      39037955
```



E. Nhận xét mối quan hệ:

- Từ biểu đồ trên ta nhận thấy tỉ lệ ca nhiễm hiện tại đã giảm đáng kể chỉ còn 7,6%. Tỉ lệ khỏi bệnh chiếm mức đa số lên tới 91,2 %.
- Bệnh này có tỉ lệ tử vong thấp 1,2% trên toàn cầu.

III. Bar chart:

A. Bar Chart là gì?

- Bar Chart là biểu đồ thể hiện dữ liệu phân loại với các thanh hình chữ nhật có chiều cao hoặc chiều dài tỷ lệ với các giá trị mà chúng đại diện. Các thanh có thể được vẽ theo chiều dọc hoặc chiều ngang. Biểu đồ thanh dọc đôi khi được gọi là biểu đồ cột.
- Biểu đồ cột là một trong những cách thông dụng nhất để trực quan hoá dữ liệu. Chúng gồm các thanh đứng hoặc thanh ngang, và các trục để hiển thị và so sánh nhiều dữ liệu khác nhau. Biểu đồ cột đặc biệt hiệu quả khi thể hiện các dữ liệu về số, giúp người xem có thể thấy được xu hướng thay đổi trong dữ liệu thông qua những cột dữ liệu.

B. Lựa chọn dữ liệu:

- Nhóm lựa chọn tất cả các trường dữ liệu của bảng dữ liệu.
- Xét 10 nước có tổng số ca mắc covid cao nhất.

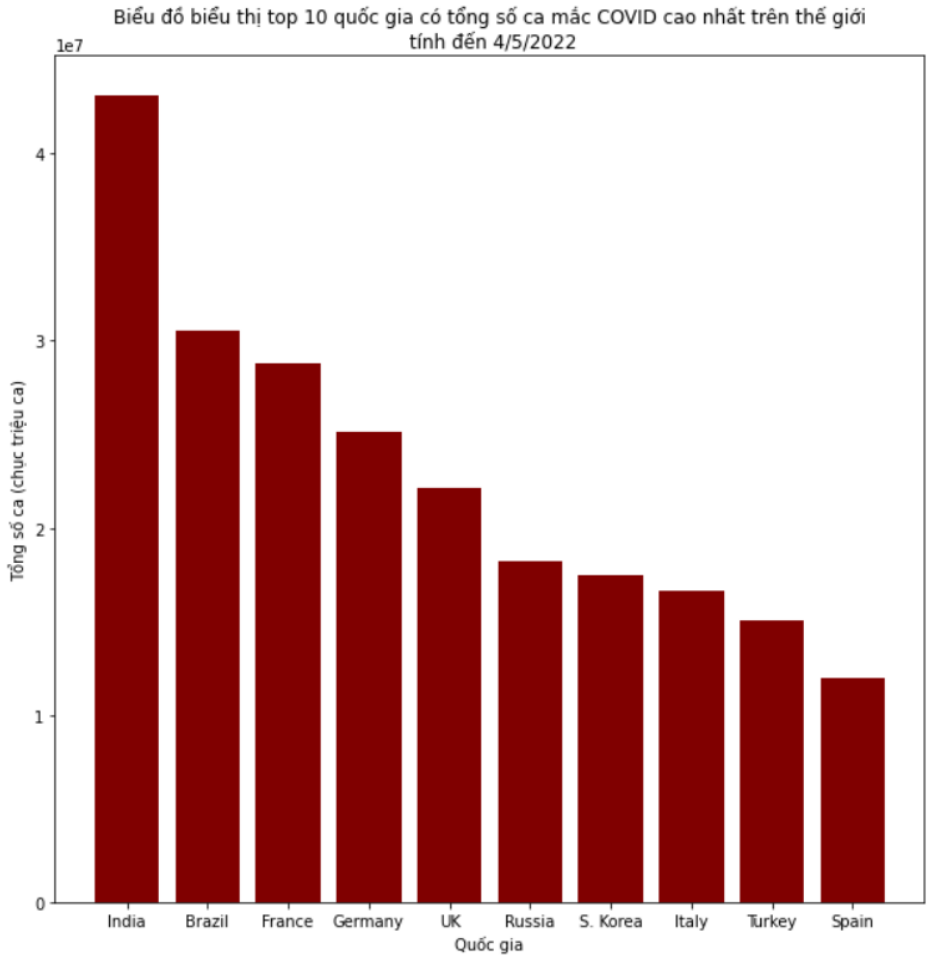
C. Code Python:

Bar chart

```
]: total_case_death_reco = df
features = ["Country","Total Cases","Total Deaths","Total Recoverd","Active Cases","Serious Cases"]
df_total_case_death_reco=total_case_death_reco[features]
df_total_case_death_reco = df_total_case_death_reco.drop(df.head(1).index)
df_total_case_death_reco=df_total_case_death_reco.head(10)

plt.bar('Country' , 'Total Cases',data=df_total_case_death_reco,color ='maroon')
plt.ylabel('Tổng số ca (chục triệu ca)')
plt.xlabel('Quốc gia')
plt.title('Biểu đồ biểu thị top 10 quốc gia có tổng số ca mắc COVID cao nhất trên thế giới\n tính đế
plt.show()
```

D. Biểu đồ:



E. Nhận xét:

- Mỗi cột thể hiện tổng số ca nhiễm của 10 nước có tổng số ca nhiễm lớn nhất trên thế giới
- Từ biểu đồ trên ta thấy:

+ Nước có số tổng số ca nhiễm covid lớn nhất là India(Ấn Độ), ngược lại nước có tổng số ca nhiễm ít nhất trong 10 nước là Spain(Tây Ban Nha)

+ Từ biểu đồ trên cho thấy các ca nhiễm covid chủ yếu ở các nước Châu Âu. Từ các nước có ngành du lịch phát triển và đông dân.

IV. Dot and Line Chart:

A. Dot and Line Chart là gì?

- Là biểu đồ kết hợp giữa biểu đồ đường và biểu đồ chấm

B. Lựa chọn dữ liệu:

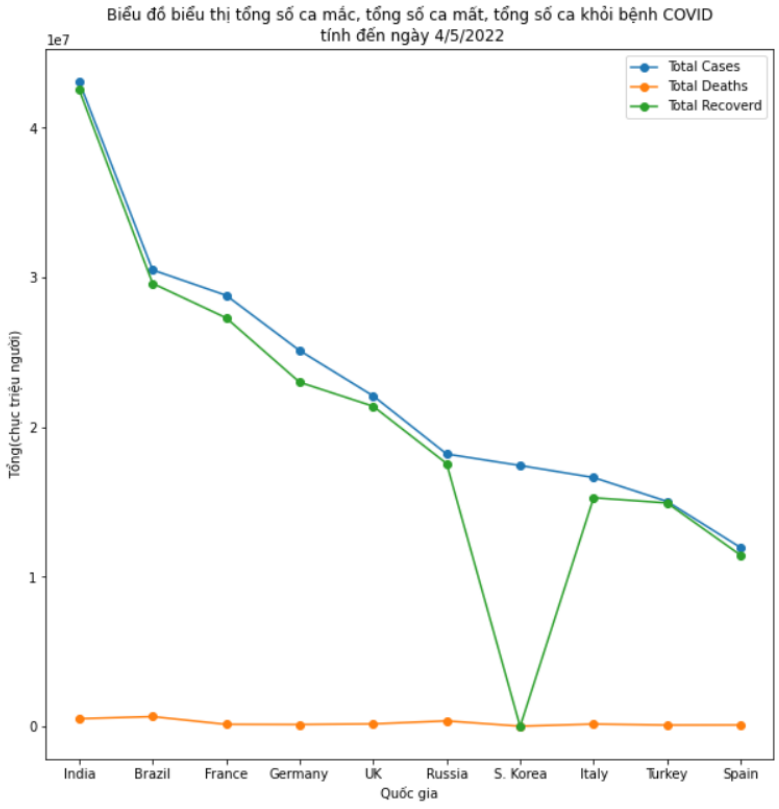
- Chọn 10 quốc gia có Total Cases thuộc top[2,11]

C. Code Python:

```
In [402]: plt.rcParams["figure.figsize"] = [10,10]
          plt.xlabel('Quốc gia')
          plt.ylabel('Tổng(chục triệu người)')
          plt.title('Biểu đồ biểu thị tổng số ca mắc, tổng số ca mất, tổng số ca khỏi bệnh COVID\n tính đến ngày 4/5/2022')
          plt.plot( 'Country', 'Total Cases', data=df_total_case_death_reco, linestyle='-', marker='o')
          plt.plot( 'Country', 'Total Deaths', data=df_total_case_death_reco, linestyle='-', marker='o')
          plt.plot( 'Country', 'Total Recoverd', data=df_total_case_death_reco, linestyle='-', marker='o')
          plt.legend()
```

D. Biểu đồ:

```
Out[402]: <matplotlib.legend.Legend at 0x15b67207f10>
```



E. Nhận xét:

- Từ biểu đồ ta thấy India là nơi có số cả nhiễm cao nhất và Spain là nơi có số ca nhiễm thấp nhất.

- Hầu hết các quốc gia đều có tổng số người mắc và tổng số người hồi phục gần như bằng nhau, Cho thấy dịch bệnh đang được khắc phục rất tốt. Trong đó Turkey là quốc gia có tỉ lệ hồi phục cao nhất.
- Ngoại trừ S.Korea không có thông tin ghi nhận về số lượng ca hồi phục.

V. Heat map

A. Heat map là gì?

- Bản đồ nhiệt là một kỹ thuật trực quan hóa dữ liệu cho thấy cường độ của một hiện tượng là màu sắc ở hai chiều. Sự thay đổi màu sắc có thể là do màu sắc hoặc cường độ, mang lại tín hiệu thị giác rõ ràng cho người đọc về cách hiện tượng được nhóm lại hoặc thay đổi theo không gian.

B. Lựa chọn dữ liệu:

- Lựa chọn 7 trường dữ liệu bao gồm [Total Cases, New Cases, Total Deaths, New Deaths, Total Recoverd, Active Cases, Serious Cases]

C. Code Python:

In [405]:
```
def heatMap(df):
    corr = df.corr()
    fig, ax = plt.subplots(figsize=(10, 10))
    sns.heatmap(corr,  annot=True, fmt=".2f")
    plt.xticks(range(len(corr.columns)), corr.columns)
    plt.yticks(range(len(corr.columns)), corr.columns)
    plt.show()
data = df
features = ["Total Cases","New Cases","Total Deaths","New Deaths","Total Recoverd","Active Cases","Serious Cases"]
corr_data = data[features]
heatMap(corr_data)
```

D. Biểu đồ:

E. Nhận xét mối quan hệ:

- Với biểu đồ Heatmap, những ô có màu càng đậm, thì sự phụ thuộc giữa các trường càng ít và mối liên quan với nhau càng ít. Còn đối với những ô có màu càng nhạt, thì sự liên quan và tác động lẫn nhau càng lớn
- Nhìn vào biểu đồ ta có thể thấy các trường hợp có tổng số ca nhiễm và tổng số ca hồi phục có tỉ lệ 0.99 chứng tỏ số lượng ca nhiễm hồi phục gần như bằng nhau cho thấy tình hình dịch bệnh đã triển biến rất tốt và hầu như tất cả những người mắc đều đã khỏi bệnh
- Biểu đồ cũng cho thấy mối liên quan giữa tổng số ca nhiễm, số ca mắc mới với tổng số người chết khi cho tỉ lệ lên đến 0.91 và 0.90

VI. Stack Vertical Bar Char:

A. Stack Vertical Bar Char là gì?

- Trong Stack Vertical Bar Char, các danh mục được biểu thị dưới dạng thanh, như trong Bar char, nhưng các thanh bao gồm các chuỗi được "xếp chồng" lên nhau, với mỗi chuỗi đại diện cho giá trị của nó. Do đó, chiều cao của toàn bộ ngăn xếp đại diện cho tổng tất cả các chuỗi trong danh mục đó. Thanh xếp chồng ngang gần giống như thanh xếp chồng dọc, nhưng chuỗi này xuất hiện cạnh nhau, thay vì xếp chồng lên nhau.

B. Chọn dữ liệu:

- Chọn 3 trường dữ liệu bao gồm [Total Recover, Total Death, Active Cases] từ 6 Châu lụcCode C. Python:

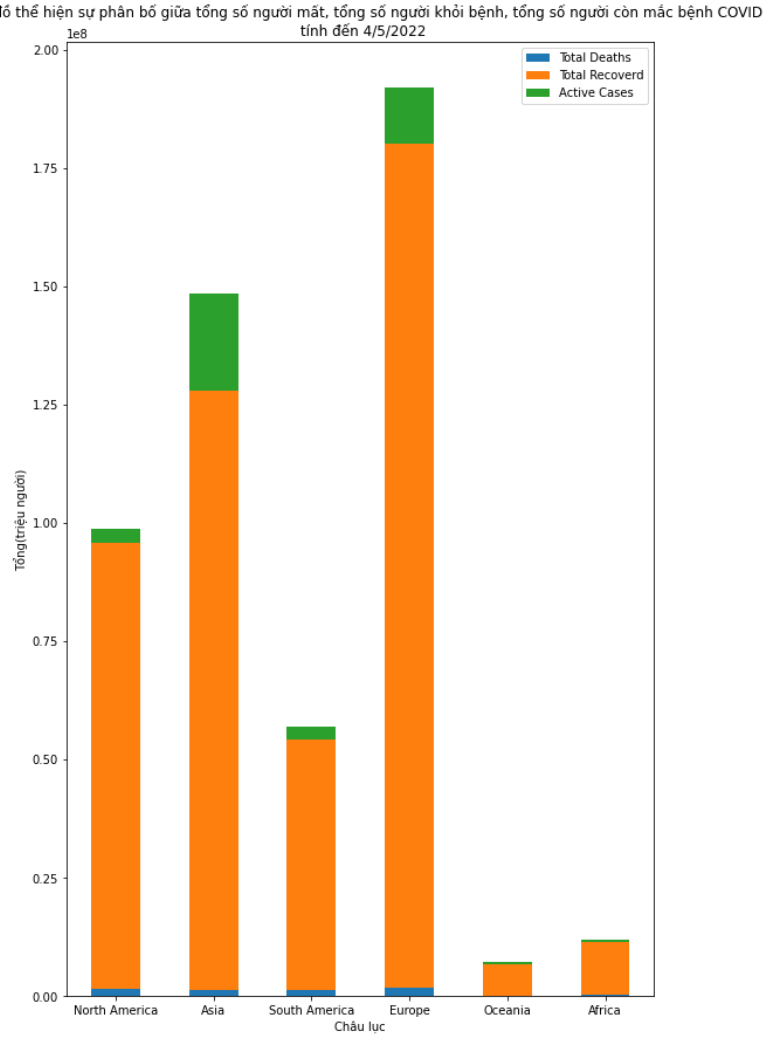
C. Code Python

In [404]:
```
df1 = dataset[0:6]
df1
indexs = ["Country","Total Deaths", "Total Recoverd", "Active Cases"]
large_filter = df1[indexs].set_index("Country")
print(large_filter)
plot = large_filter.plot.bar(stacked=True,figsize = (9,15),rot=0)
plt.xlabel('Châu lục')
plt.ylabel('Tổng(triệu người)')
plt.title('Biểu đồ thể hiện sự phân bố giữa tổng số người mất, tổng số người khỏi bệnh, tổng số người còn mắc bệnh COVID\n tính c
```

```
               Total Deaths  Total Recoverd  Active Cases
Country
North America       1462943        94348717       2844634
Asia                1425428       126415821      20610595
South America       1294985        53031574       2553193
Europe              1821484       178300163      11937594
Oceania               10872         6858416        516475
Africa               253860        11120021        547193
```

D. Biểu đồ:

```
Out[404]: Text(0.5, 1.0, 'Biểu đồ thể hiện sự phân bố giữa tổng số người mất, tổng số người khỏi bệnh, tổng số người còn mắc bệnh COVID\n tính đến 4/5/2022')
```



E. Nhận xét:

- Từ biểu đồ đa thấy Europe đang là Châu lục có tổng số lượng người mắc nhiều nhất kế tiếp là Asia và North America
- Đồng thời Europe cũng là Châu lục có số ca tử vong cao nhất tuy nhiên kế tiếp là North America rồi mới đến Asia cho thấy Asia xử lý dịch bệnh tốt hơn North America
- Tuy nhiên Asia lại là Châu lục có số ca nhiễm hiện hành cao nhất, cho thấy tình hình dịch bệnh vẫn chưa được kiểm soát tốt

VII. NGUỒN THAM KHẢO:

- Trực quan hoá dữ liệu bằng biểu đồ | bởi Bạch Hạnh | Brands Vietnam

- Chart Visualization — pandas 1.4.2 documentation (pydata.org)
- pandas.DataFrame.plot.barh — pandas 1.4.2 documentation (pydata.org)
- (246) Hướng Dẫn Trực Quan Hoá Dữ Liệu với Matplotlib và Python - YouTube