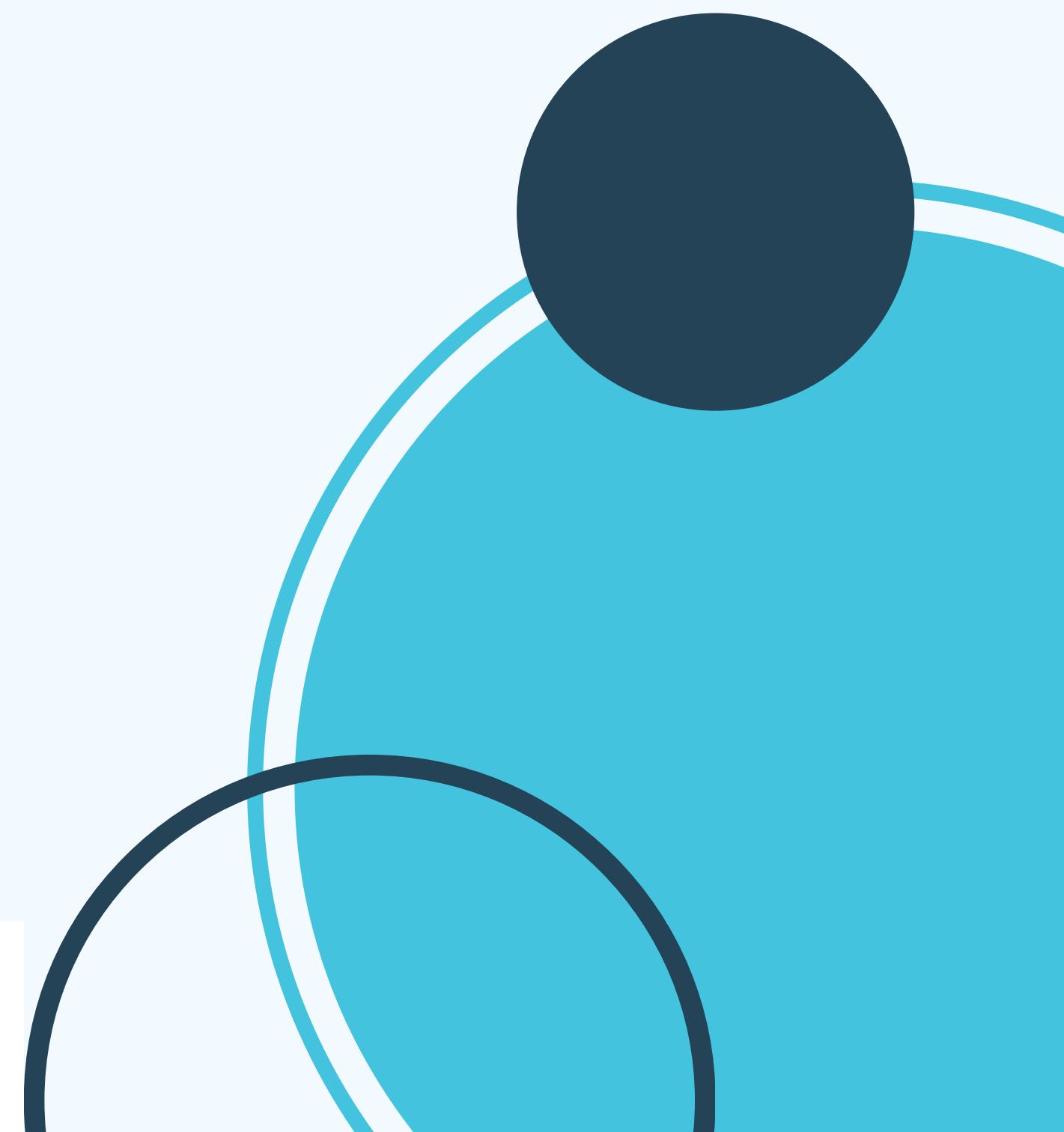


Engenharia de Dados

Projeto Final



APRESENTAÇÃO DA EQUIPE



Pablo
Henrique



Vinícius
Amaral



Vinícius
Silva



Nayara
Oliveira



Rodrigo
Molverstet



SAÚDE: COVID-19

- Problema de grande relevância mundial
- Altos índices de morte
- Impactos sociais, econômicos, políticos, culturais e históricos
- Impacto sobre os sistemas de saúde
- Abalo na saúde mental das pessoas
- Dificuldade de acesso a bens essenciais como alimentação, medicamentos, transporte e etc

ESCOPO DO PROJETO



- Os datasets devem ter formatos diferentes
- Operações com Pandas e Spark (limpezas , transformações e normalizações)
- Os datasets devem ser salvos e operados em Cloud
- Dados originais devem ser armazenados no MySql
- Dados tratados devem ser armazenados no MongoDB
- Análises devem ser realizadas no BigQuery
- Criação de um dashboard no LookerStudio trazendo insights importantes
- Criação de um Workflow simples exibindo as etapas de ETL.



INSIGHTS INICIAIS

- Países com maiores casos seriam Itália, Estados Unidos.
- Quais países tiveram índices elevados e não foram retratados?
- Verificação de vacinas(quantos tipos de fato existem?)
- Primeiras vacinas a serem utilizadas foram a Pfizer e AstraZeneca
- Verificação de períodos de autorização/início da vacinação.

FONTE DOS DADOS

The screenshot shows the WHO COVID-19 dashboard. At the top left is the WHO logo. To its right is a search bar with the placeholder "Search by Country, Territory, or Area". Further right are icons for information, sharing, and a "Covid-19 Response Fund Donate" button. Below the header, the page title is "WHO Coronavirus (COVID-19) Dashboard". Underneath the title are several navigation links: "Overview", "Measures", "Table View", "Data" (which is underlined to indicate it's the active page), and "More Resources".

OMS(Organização mundial de Saúde)

3 bases de dados: COVID -19 Global Data - COVID -19 Global Table Data - Vaccination Data

Link: <https://covid19.who.int/data> | Formato: CSV | última data de atualização 12 - 2022

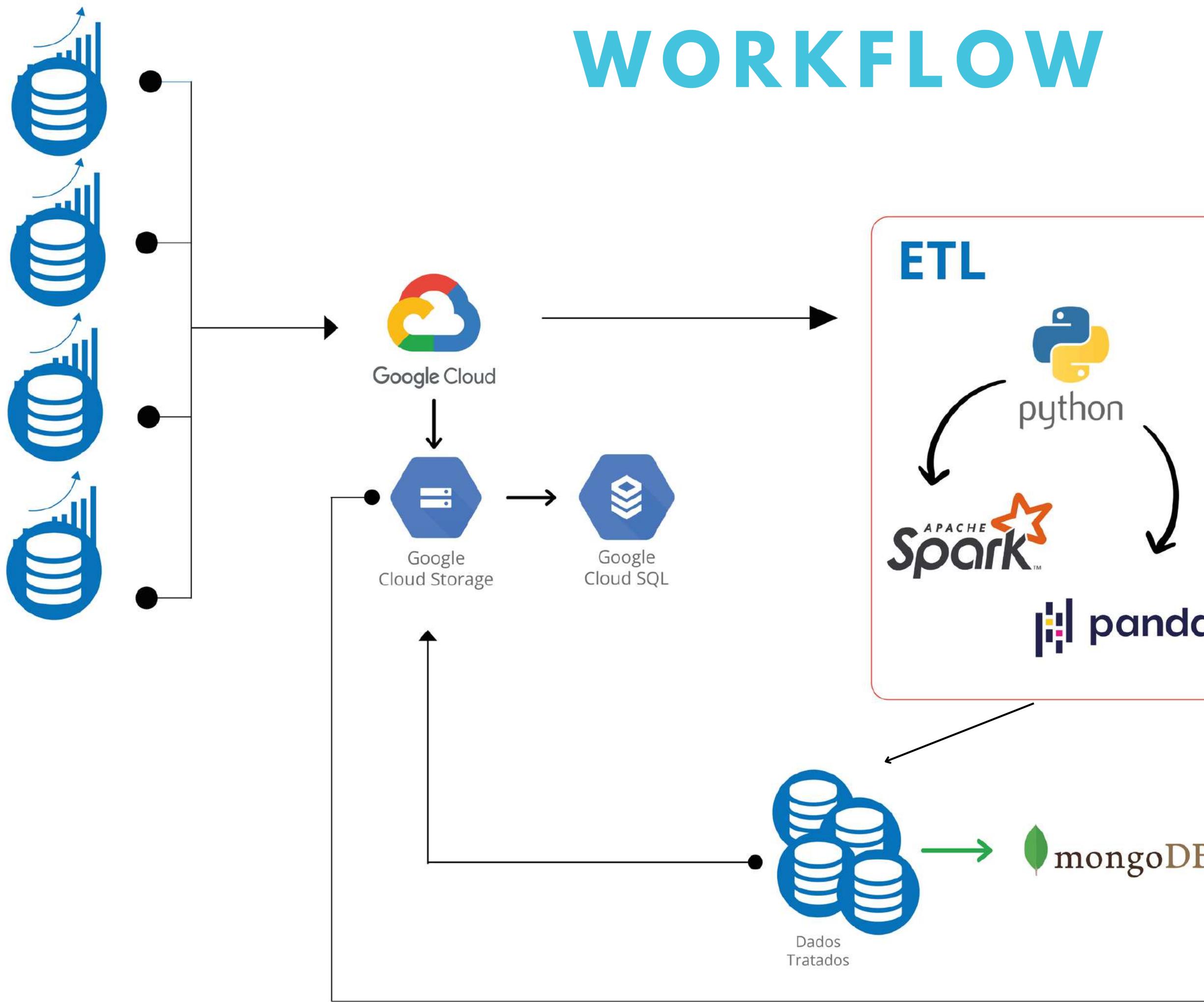
The screenshot shows the ECDC website. At the top left is the ECDC logo, which includes a stylized globe icon and the acronym "ecdc". To the right of the logo is the text "European Centre for Disease Prevention and Control" and "An agency of the European Union". On the right side of the header is a search bar with a dropdown menu set to "All sections" and a text input field with the placeholder "Enter your keyword".

Centro Europeu de Prevenção e Controle de Doenças

1 base de dados: EUROPA-COVID-TEST

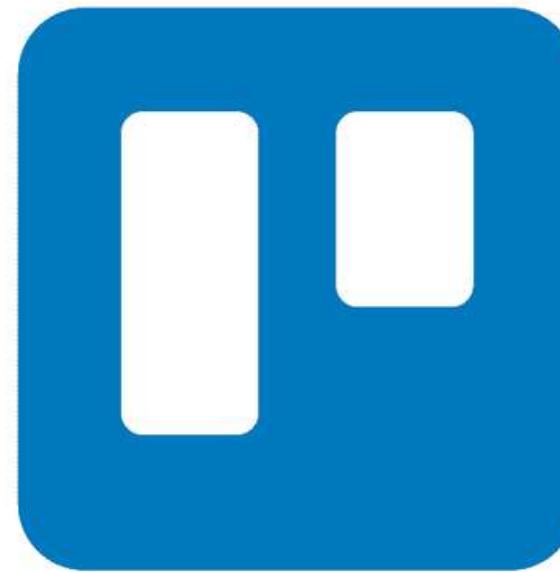
Link: www.ecdc.europa.eu/en/publications-data/download-todays-data-geographic-distribution-covid-19-cases-worldwide | Formato: JSON | última data de atualização 01-2022

WORKFLOW



ORGANIZAÇÃO DAS ATIVIDADES

TRELLO

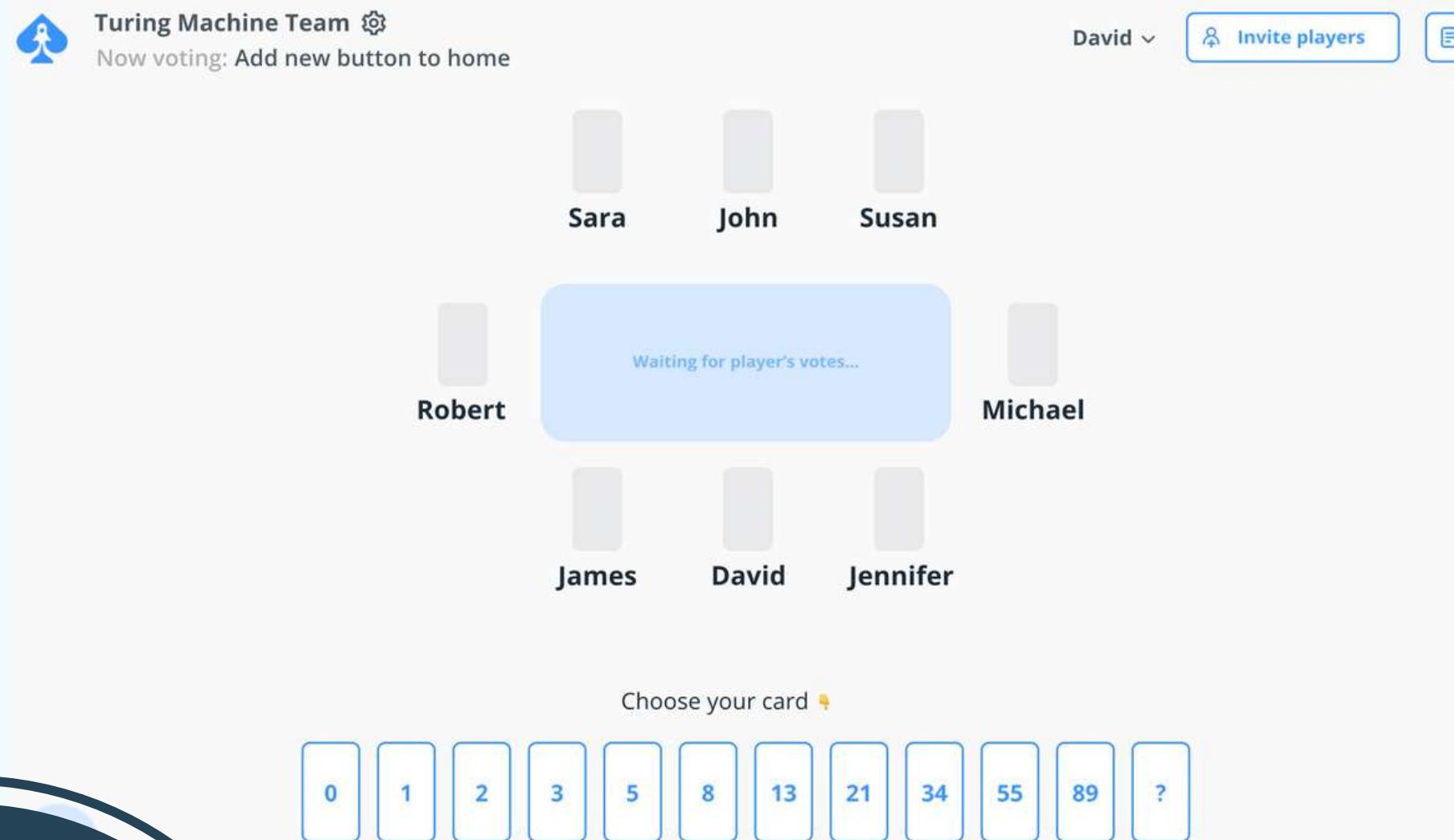


Trello

- Visualização de projetos divididos em tarefas e ações
- Metodologia Kanban
- Possibilidade de agregar recursos visuais e links aos cartões
- Compartilhamento do painel de tarefas com outros membros da equipe
- Facilidade de colaboração dentro do sistema
- Integrações do Trello com outros softwares, como planilhas do Google

ORGANIZAÇÃO DAS ATIVIDADES

PLANNING POKER



- Estimativas mais precisas e melhor mensuradas;
- Evita uma grande variação nas estimativas;
- Melhor interação entre a equipe de desenvolvimento;
- Maior entendimento do que deve ser realizado em cada requisito;
- Ajuda na organização do backlog da sprint, pois, por meio da classificação da estimativa por pontos, é possível adequar em cada sprint suas respectivas User Stories

PROCESSO ETL

EXTRAÇÃO, TRANSFORMAÇÃO E CARREGAMENTO DOS DADOS

SOBRE OS DATASETS

Dataset 1 - EUROPA-COVID-TEST

1 df_pd_1

	country	country_code	year_week	level	region	region_name	population	new_cases	tests_done	testing_rate	positivity_rate	testing_data_source
0	Austria	AT	2020-W01	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	object
1	Austria	AT	2020-W02	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	object
2	Austria	AT	2020-W03	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	object
3	Austria	AT	2020-W04	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	object
4	Austria	AT	2020-W05	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	object
...
4645	Sweden	SE	2022-W46	national	SE	Sweden	10379295	3678.0	20052.0	193.192	1.83	float64
4646	Sweden	SE	2022-W47	national	SE	Sweden	10379295	4259.0	21445.0	206.613	1.97	float64
4647	Sweden	SE	2022-W48	national	SE	Sweden	10379295	6724.0	25361.0	244.342	2.62	float64
4648	Sweden	SE	2022-W49	national	SE	Sweden	10379295	9769.0	25361.0	244.342	3.81	float64
4649	Sweden	SE	2022-W50	national	SE	Sweden	10379295	NaN	NaN	NaN	NaN	object

4650 rows × 12 columns

```
→ country          object  
→ country_code    object  
→ year_week       object  
→ level           object  
→ region          object  
→ region_name     object  
→ population      int64  
→ new_cases        float64  
→ tests_done       float64  
→ testing_rate    float64  
→ positivity_rate float64  
→ testing_data_source object  
dtype: object
```

SOBRE OS DATASETS

Dataset 2 - COVID -19 Global Data

	Unnamed: 0	Date_reported	Country_code	Country	WHO_region	New_cases	Cumulative_cases	New_deaths	Cumulative_deaths
0	0	2020-01-03	AF	Afghanistan	EMRO	0	0	0	0
1	1	2020-01-04	AF	Afghanistan	EMRO	0	0	0	0
2	2	2020-01-05	AF	Afghanistan	EMRO	0	0	0	0
3	3	2020-01-06	AF	Afghanistan	EMRO	0	0	0	0
4	4	2020-01-07	AF	Afghanistan	EMRO	0	0	0	0
...
257377	257377	2022-12-19	ZW	Zimbabwe	AFRO	219	259981	10	10
257378	257378	2022-12-20	ZW	Zimbabwe	AFRO	0	259981	0	0
257379	257379	2022-12-21	ZW	Zimbabwe	AFRO	0	259981	0	0
257380	257380	2022-12-22	ZW	Zimbabwe	AFRO	0	259981	0	0
257381	257381	2022-12-23	ZW	Zimbabwe	AFRO	0	259981	0	0

→ Unnamed: 0 int64
Date_reported object
Country_code object
Country object
WHO_region object
New_cases int64
Cumulative_cases int64
New_deaths int64
Cumulative_deaths int64
dtype: object

SOBRE OS DATASETS

Dataset 3 COVID -19 Global Table Data

	Unnamed: 0	Name	WHO Region	Cases - cumulative total	Cases - total per 100000 population	Cases - newly reported in last 7 days	Cases - reported in last 7 days per 100000 population	Cases - newly reported in last 24 hours	Deaths - cumulative total
0	Global	NaN	651918402	8363.782494	3538858	45.401753	778897	6656601	85.400815
1	United States of America	Americas	99027628	29917.473000	501758	151.587000	501758	1080010	326.284000
2	India	South-East Asia	44676678	3237.430000	1069	0.077000	163	530690	38.456000
3	France	Europe	37988187	58407.997000	327753	503.930000	47594	157364	241.952000
4	Germany	Europe	37177845	44702.796000	215424	259.027000	29261	160611	193.119000
...
233	Holy See	Europe	26	3213.844000	0	0.000000	0	0	0.000000
234	Tokelau	Western Pacific	5	370.370000	0	0.000000	0	0	0.000000
235	Pitcairn Islands	Western Pacific	4	8000.000000	0	0.000000	0	0	0.000000
236	Democratic People's Republic of Korea	South-East Asia	0	0.000000	0	0.000000	0	0	0.000000

```
Unnamed: 0
Name
WHO Region
Cases - cumulative total
Cases - cumulative total per 100000 population
Cases - newly reported in last 7 days
Cases - newly reported in last 7 days per 100000 population
Cases - newly reported in last 24 hours
Deaths - cumulative total
Deaths - cumulative total per 100000 population
Deaths - newly reported in last 7 days
Deaths - newly reported in last 7 days per 100000 population
Deaths - newly reported in last 24 hours
dtype: object
```

SOBRE OS DATASETS

Dataset 4 - Vaccination Data

		Unnamed: 0	ISO3	VACCINE_NAME	PRODUCT_NAME	COMPANY_NAME	AUTHORIZATION_DATE	START_DATE	END_DATE
0	0	SHN	AstraZeneca	AstraZeneca - AZD1222	AZD1222	AstraZeneca	NaN	NaN	NaN
1	1	GRL	Moderna	Moderna - mRNA-1273	mRNA-1273	Moderna	NaN	NaN	NaN
2	2	FRO	Moderna	Moderna - mRNA-1273	mRNA-1273	Moderna	NaN	NaN	NaN
3	3	FRO	Pfizer BioNTech	Comirnaty	Pfizer BioNTech	Comirnaty	NaN	NaN	NaN
4	4	BIH	AstraZeneca	AstraZeneca - AZD1222	AZD1222	AstraZeneca	NaN	NaN	NaN
...
1066	1066	ITA	Pfizer BioNTech	Comirnaty Bivalent Original/Omicron BA.4/BA.5	Comirnaty Bivalent Original/Omicron BA.4/BA.5	NaN	NaN	2022-09-28	NaN
1067	1067	LUX	Pfizer BioNTech	Comirnaty Bivalent Original/Omicron BA.4/BA.5	Comirnaty Bivalent Original/Omicron BA.4/BA.5	NaN	NaN	2022-10-12	NaN
1068	1068	PRT	Pfizer BioNTech	Comirnaty Bivalent Original/Omicron BA.4/BA.5	Comirnaty Bivalent Original/Omicron BA.4/BA.5	NaN	NaN	2022-09-28	NaN
			Moderna - Spikevax	Moderna - Spikevax					

```
Unnamed: 0          int64
ISO3              object
VACCINE_NAME      object
PRODUCT_NAME       object
COMPANY_NAME       object
AUTHORIZATION_DATE object
START_DATE         object
END_DATE           float64
COMMENT            float64
DATA_SOURCE        object
dtype: object
```

CARREGAMENTO

Envio dos arquivos originais para o Cloud Storage
Extração da base de dados do Cloud Storage

```
2
3 df.to_json('gs://projfinal;brutos/Europa-Covid-Test.json',storage_options={'token': '/content/bc26-rdg-ed27-7fdc00e9e239.json'})
4 df2.to_csv('gs://projfinal;brutos/WHO-COVID-19-global-data.csv',storage_options={'token': '/content/bc26-rdg-ed27-7fdc00e9e239.json'})
5 df3.to_csv('gs://projfinal;brutos/WHO-COVID-19-global-table-data.csv',storage_options={'token': '/content/bc26-rdg-ed27-7fdc00e9e239.json'})
6 df4.to_csv('gs://projfinal;brutos/vaccination-metadata.csv',storage_options={'token': '/content/bc26-rdg-ed27-7fdc00e9e239.json'})
```

```
[ ] 1 #Service acc
2
3 serviceAccount = '/content/bc26-rdg-ed27-7fdc00e9e239.json'
4 os.environ['GOOGLE_APPLICATION_CREDENTIALS'] = serviceAccount

▶ 1 #Extração Arquivos GCP/bucket
2
3 client = storage.Client('/content/bc26-rdg-ed27-7fdc00e9e239.json')
4 #CRIAR UMA VARIÁVEL PARA RECEBER O NOME DA BUCKET
5 bucket = client.get_bucket('projfinal')
6
7 #ESCOLHER O ARQUIVO DENTRO DA BUCKET
8 bucket.blob('Europa-Covid-Test')
9
10 #CRIAR UMA VARIÁVEL PARA RECEBER O CAMINHO DO ARQUIVO
11 path1 = 'gs://projfinal;brutos/Europa-Covid-Test.json'
```

<input type="checkbox"/>	Name	Size
<input type="checkbox"/>	Europa-Covid-Test.json	821.8 KB
<input type="checkbox"/>	WHO-COVID-19-global-data.csv	12.8 MB
<input type="checkbox"/>	WHO-COVID-19-global-table-data...	17.1 KB
<input type="checkbox"/>	vaccination-metadata.csv	92.8 KB

CARREGAMENTO

Envio dos arquivos originais para o MySQL(Banco Relacional)

```
1 Envio dos DF para MySQL
2
3 Não esquecer de criar o database no shell: gcloud sql connect servidor-db --user=root --quiet
4 CREATE DATABASE projfinal;
5
6 conexao = '34.139.131.208'
7 nome_do_banco = 'projfinal'
8 usuario = 'root'
9 senha = 'root'
10 engine = create_engine("mysql+pymysql://{}user:{}pw@{}host:{}db{}".format(host=conexao, db=nome_do_banco, user= usuario, pw=senha))
11
12 df_pd_1.to_sql("Europa-Covid-Test",engine,index=True,index_label='id')
13 df_pd_2.to_sql("WHO-COVID-19-global-data",engine,index=True,index_label='id')
14 df_pd_3.to_sql("WHO-COVID-19-global-table-data",engine,index=True,index_label='id')
15 df_pd_4.to_sql("vaccination-metadata",engine,index=True,index_label='id')
```

Name ↑	Collation	Character set	Type	
information_schema	utf8_general_ci	utf8	System	⋮
mysql	utf8_general_ci	utf8	System	⋮
performance_schema	utf8mb4_0900_ai_ci	utf8mb4	System	⋮
projfinal	utf8mb4_0900_ai_ci	utf8mb4	User	⋮
sys	utf8mb4_0900_ai_ci	utf8mb4	System	⋮

CARREGAMENTO

Acesso ao MySQL(Banco Relacional) visualização do Database juntamente com os Datasets

```
mysql> SHOW DATABASES;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| projfinal      |
| sys            |
+-----+
5 rows in set (0.00 sec)

mysql> USE projfinal;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> SHOW TABLES;
+-----+
| Tables_in_projfinal |
+-----+
| Europa-Covid-Test   |
| WHO-COVID-19-global-data |
| WHO-COVID-19-global-table-data |
| vaccination-metadata |
+-----+
4 rows in set (0.00 sec)

mysql> SELECT * FROM `Europa-Covid-Test` LIMIT 5;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id   | country | country_code | year_week | level    | region   | region_name | population | new_cases | tests_done | testing_rate | positivity_rate | testing_data_source |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 0    | Austria | AT          | 2020-W01   | national  | AT       | Austria     | 8932664   | NULL      | NULL      | NULL      | NULL      | NULL      | NULL      |
| 1    | Austria | AT          | 2020-W02   | national  | AT       | Austria     | 8932664   | NULL      | NULL      | NULL      | NULL      | NULL      | NULL      |
| 2    | Austria | AT          | 2020-W03   | national  | AT       | Austria     | 8932664   | NULL      | NULL      | NULL      | NULL      | NULL      | NULL      |
| 3    | Austria | AT          | 2020-W04   | national  | AT       | Austria     | 8932664   | NULL      | NULL      | NULL      | NULL      | NULL      | NULL      |
| 4    | Austria | AT          | 2020-W05   | national  | AT       | Austria     | 8932664   | NULL      | NULL      | NULL      | NULL      | NULL      | NULL      |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

TRANSFORMAÇÃO

Tratamento de Dados/Pandas

Conferência dos elementos das colunas com o objetivo de buscar inconsistências

```
1 pd.unique(df_pd_2['Date_reported'])  
  
1 pd.unique(df_pd_2['Country_code'])  
  
1 pd.unique(df_pd_2['Country'])  
  
1 pd.unique(df_pd_2['WHO_region'])
```

```
[ ] 1 pd.unique(df_pd_2['Cumulative_cases'])  
array([ 0, 5, 8, ..., 259350, 259762,  
  
[ ] 1 pd.unique(df_pd_2['New_deaths'])  
array([ 0, 1, 2, ..., 2658, 2881, -35])  
  
[ ] 1 pd.unique(df_pd_2['Cumulative_deaths'])  
array([ 0, 1, 2, ..., 5427, 5444, 5498])
```

TRANSFORMAÇÃO

Manipulação de Dados/Pandas

Renomeação e Tradução das colunas com base no significado pré definido pela OMS

```
2 df_pd_2.rename(columns={  
3     'Date_reported': 'data_relatada',  
4     'Country_code': 'cod_pais',  
5     'Country': 'pais',  
6     'WHO_region': 'regiao_who',  
7     'New_cases': 'novos_casos',  
8     'Cumulative_cases': 'casos_cumulativos',  
9     'New_deaths': 'novas_mortes',  
10    'Cumulative_deaths': 'mortes_cumulativas',  
11 }, inplace=True)
```

```
2 df_pd_2.replace({'pais':'Afghanistan'}, 'Afeganistão', regex=True, inplace=True)  
3 df_pd_2.replace({'pais':'Albania'}, 'Albânia', regex=True, inplace=True)  
4 df_pd_2.replace({'pais':'Algeria'}, 'Argélia', regex=True, inplace=True)  
5 df_pd_2.replace({'pais':'American Samoa'}, 'Samoa Americana', regex=True, inplace=True)  
6 df_pd_2.replace({'pais':'Andorra'}, 'Andorra', regex=True, inplace=True)  
7 df_pd_2.replace({'pais':'Angola'}, 'Angola', regex=True, inplace=True)  
8 df_pd_2.replace({'pais':'Anguilla'}, 'Anguilha', regex=True, inplace=True)  
9 df_pd_2.replace({'pais':'Antigua and Barbuda'}, 'Antígua e Barbuda', regex=True, inplace=True)  
10 df_pd_2.replace({'pais':'Argentina'}, 'Argentina', regex=True, inplace=True)  
11 df_pd_2.replace({'pais':'Armenia'}, 'Armênia', regex=True, inplace=True)  
12 df_pd_2.replace({'pais':'Aruba'}, 'Aruba', regex=True, inplace=True)  
13 df_pd_2.replace({'pais':'Australia'}, 'Austrália', regex=True, inplace=True)  
14 df_pd_2.replace({'pais':'Austria'}, 'Austria', regex=True, inplace=True)  
15 df_pd_2.replace({'pais':'Azerbaijan'}, 'Azerbaijão', regex=True, inplace=True)  
16 df_pd_2.replace({'pais':'Bahamas'}, 'Bahamas', regex=True, inplace=True)  
17 df_pd_2.replace({'pais':'Bahrain'}, 'Bahrein', regex=True, inplace=True)  
18 df_pd_2.replace({'pais':'Bangladesh'}, 'Bangladesh', regex=True, inplace=True)  
19 df_pd_2.replace({'pais':'Barbados'}, 'Barbados', regex=True, inplace=True)
```

TRANSFORMAÇÃO

Limpeza de Inconsistências/Pandas

Drop e Padronização de colunas

```
[ ] # Identificamos a coluna 'Unnamed: 0' como padrão nos 3 datasets  
  
df_pd_3.rename(columns={'Unnamed: 0':'pais'}, inplace=True)  
  
df_pd_2.drop(['Unnamed: 0']), axis=1, inplace=True  
df_pd_4.drop(['Unnamed: 0']), axis=1, inplace=True
```

TRANSFORMAÇÃO

Limpeza de Inconsistências/Pandas

Filtros para redução de dados Nulos nos Datasets

```
# Filtro parte I
filtro1 = df_pd_2.regiao_who == 'Mediterrâneo Oriental'
df_f1 = df_pd_2.loc[filtro1]
df_f1
# 23892 rows
```

	data_relatada	cod_pais	país	regiao_who	novos_casos	casos_cumulativos	novas_mortes	mortes_cumulativas
0	2020-01-03	AF	Afeganistão	Mediterrâneo Oriental	0	0	0	0
1	2020-01-04	AF	Afeganistão	Mediterrâneo Oriental	0	0	0	0
2	2020-01-05	AF	Afeganistão	Mediterrâneo Oriental	0	0	0	0
3	2020-01-06	AF	Afeganistão	Mediterrâneo Oriental	0	0	0	0
4	2020-01-07	AF	Afeganistão	Mediterrâneo Oriental	0	0	0	0

TRANSFORMAÇÃO

Limpeza de Inconsistências/Pandas

Filtros para redução de dados Nulos nos Datasets e concatenação dos Datasets a partir destes filtros

```
# Filtro parte II  
filtro1_1 = (df_f1.novos_casos != 0) | (df_f1.casos_cumulativos != 0) | (df_f1.novas_mortes != 0) | (df_f1.mortes_cumulativas != 0)  
df_f1_1 = df_f1.loc[filtro1_1]  
df_f1_1  
# novos casos 18536
```

52	2020-02-24	AF	Afeganistão	Mediterrâneo Oriental	5	5	0	0
53	2020-02-25	AF	Afeganistão	Mediterrâneo Oriental	0	5	0	0
54	2020-02-26	AF	Afeganistão	Mediterrâneo Oriental	0	5	0	0
55	2020-02-27	AF	Afeganistão	Mediterrâneo Oriental	0	5	0	0
56	2020-02-28	AF	Afeganistão	Mediterrâneo Oriental	0	5	0	0

```
df_pd_2_final = pd.concat([df_f1_1, df_f2_1, df_f3_1, df_f4_1, df_f5_1, df_f6_1, df_f7_1], ignore_index = True)
```

TRANSFORMAÇÃO

Limpeza de Inconsistências/Pandas

Identificação de redundâncias em colunas

	país	vacina	produto	empresa
22	Turquia	Turkovac	NaN	NaN

```
# Verificando a coluna produto
pd.unique(dfback['produto'])
#Temos 37 resultados /valores nulos 1

array(['AZD1222', 'mRNA-1273', 'Comirnaty', 'Coronavac', 'EpiVacCorona',
       'n_informado', 'Zifivax', 'LV-SMENP-DC', 'VLA2001', 'Soberana-02',
       'Covidful', 'Gam-Covid-Vac', 'QazVac', 'BBIBP-CorV', 'ZyCov-D',
       'Corbevax', 'CIGB-66', 'Vaxzevria', 'Convidecia', 'Covi-Vac',
       'Inactivated SARS-CoV-2 vaccine', 'Spikevax', 'Covaxin',
       'Ad26.COV 2-S', 'NUVAXOVID', 'Covishield', 'Soberana Plus',
       'Sputnik-Light', 'Hayat-Vax', 'COVIran Barakat', 'Covavax',
       'Covovax', 'Spikevax Bivalent Original/Omicron BA.1',
       'Comirnaty Bivalent Original/Omicron BA.1',
       'Comirnaty Bivalent Original/Omicron BA.4/BA.5',
       'Moderna - Spikevax Bivalent Original/Omicron - Generic',
       'Pfizer BioNTech - Comirnaty Bivalent Original/Omicron - Generic'],
      dtype=object)
```

TRANSFORMAÇÃO

Limpeza de Inconsistências/Pandas

Filtro realizado para padronização de colunas

```
# Realizando o filtro na vacina 'Turkovac'
filtro1 = dfback.vacina == 'Turkovac'
filtro1_1 = dfback.loc[filtro1]
filtro1_1
```

	país	vacina	produto	empresa	data_de_autorizacao	data_de_inicio	fonte_de_dados
22	Turquia	Turkovac		NaN	NaN	NaN	OWID


```
[16] # Realizando as alterações necessárias
    filtro1_1.iat[0,2] = 'Turkovac'
    filtro1_1.iat[0,3] = 'Health Institutes of Turkey'
    filtro1_1
```

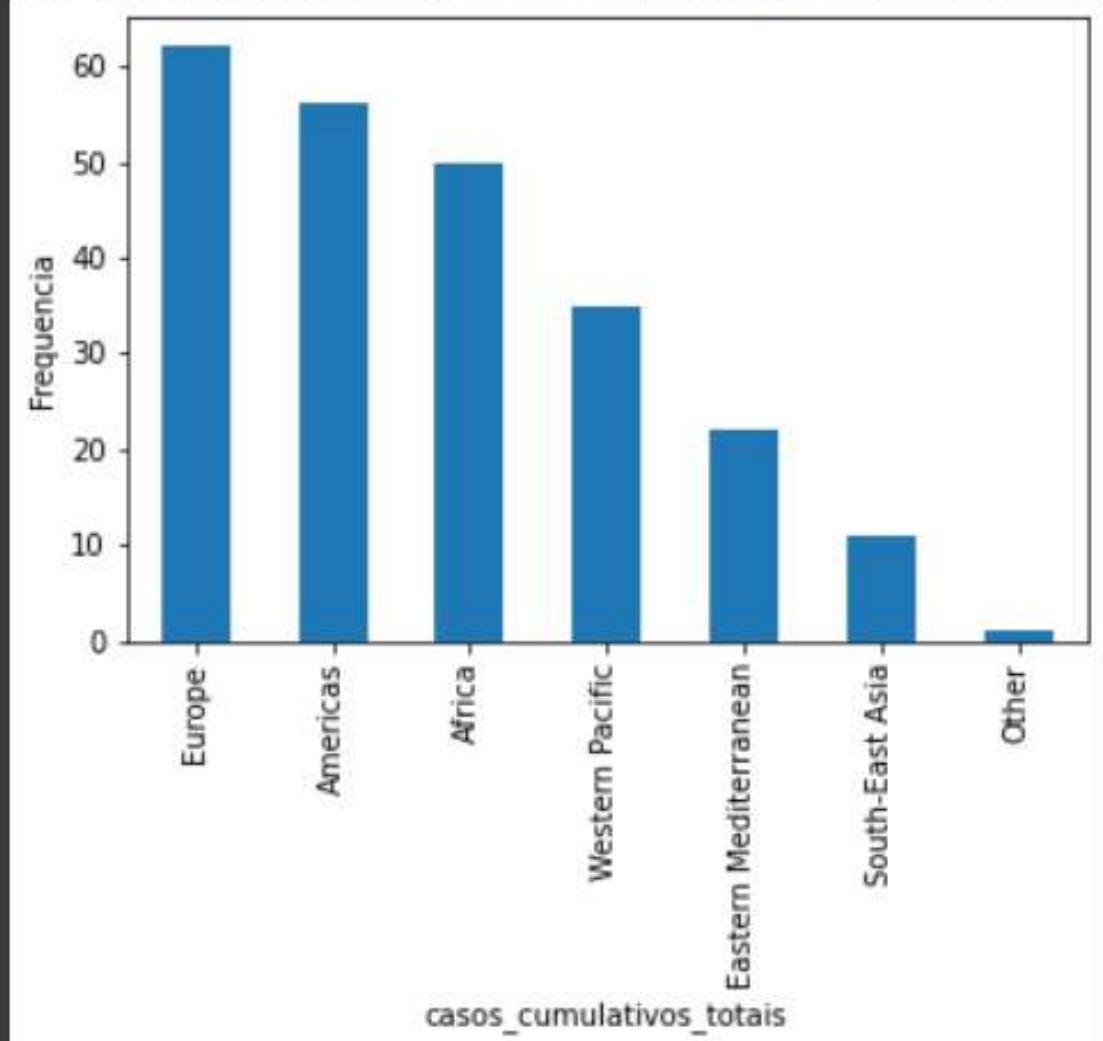
	país	vacina	produto	empresa	data_de_autorizacao	data_de_inicio	fonte_de_dados	
22	Turquia	Turkovac	Turkovac	Health Institutes of Turkey		NaN	NaN	OWID

TRANSFORMAÇÃO

Groupbys com auxílio de Plot para gerar Insights

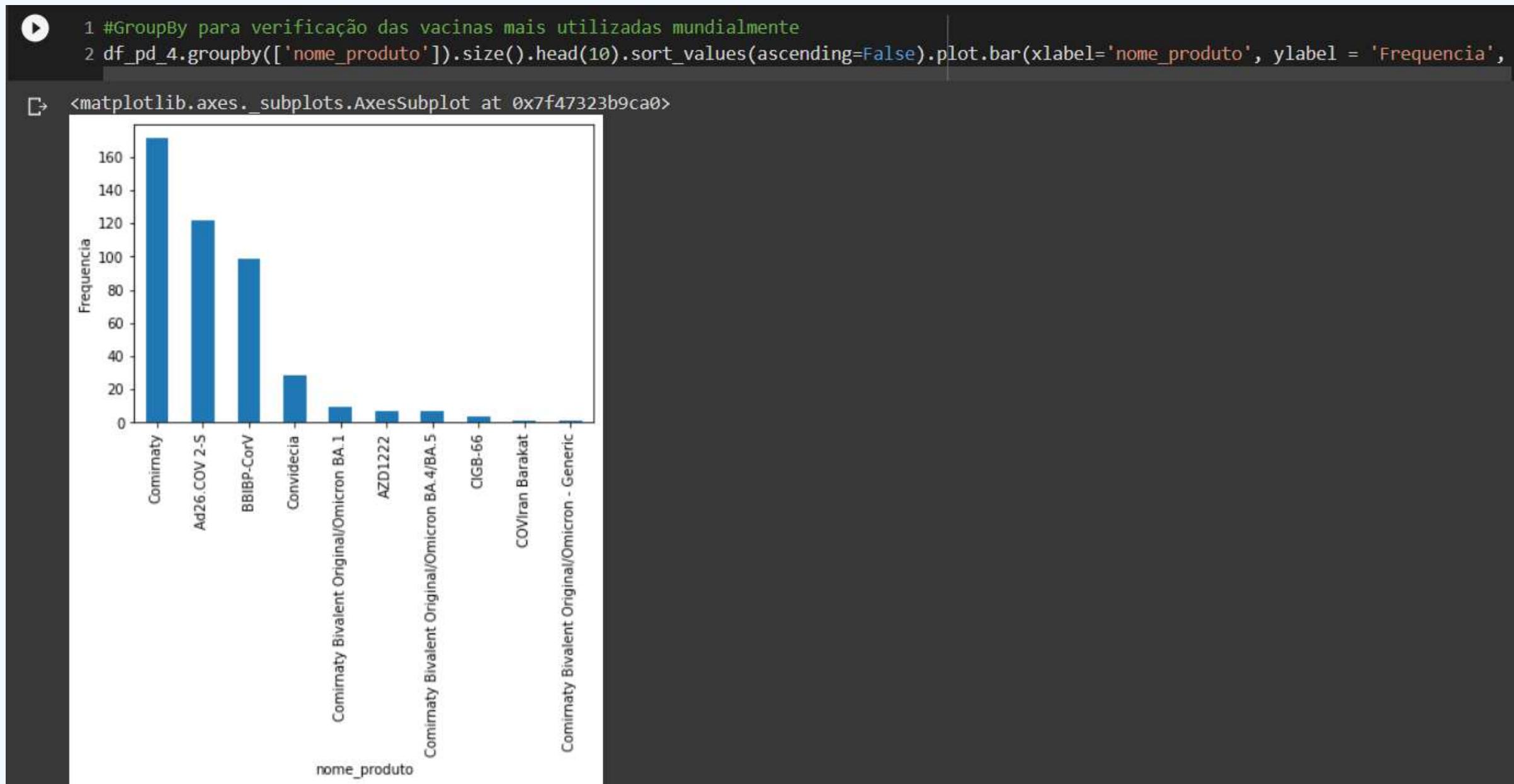
```
1 #GroupBy com auxílio de Plot para verificação dos casos totais por regiao  
2 df_pd_3.groupby(['regiao']).size().head(10).sort_values(ascending=False).plot.bar(xlabel='casos_cumulativos_totais', ylabel
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f47325807c0>
```



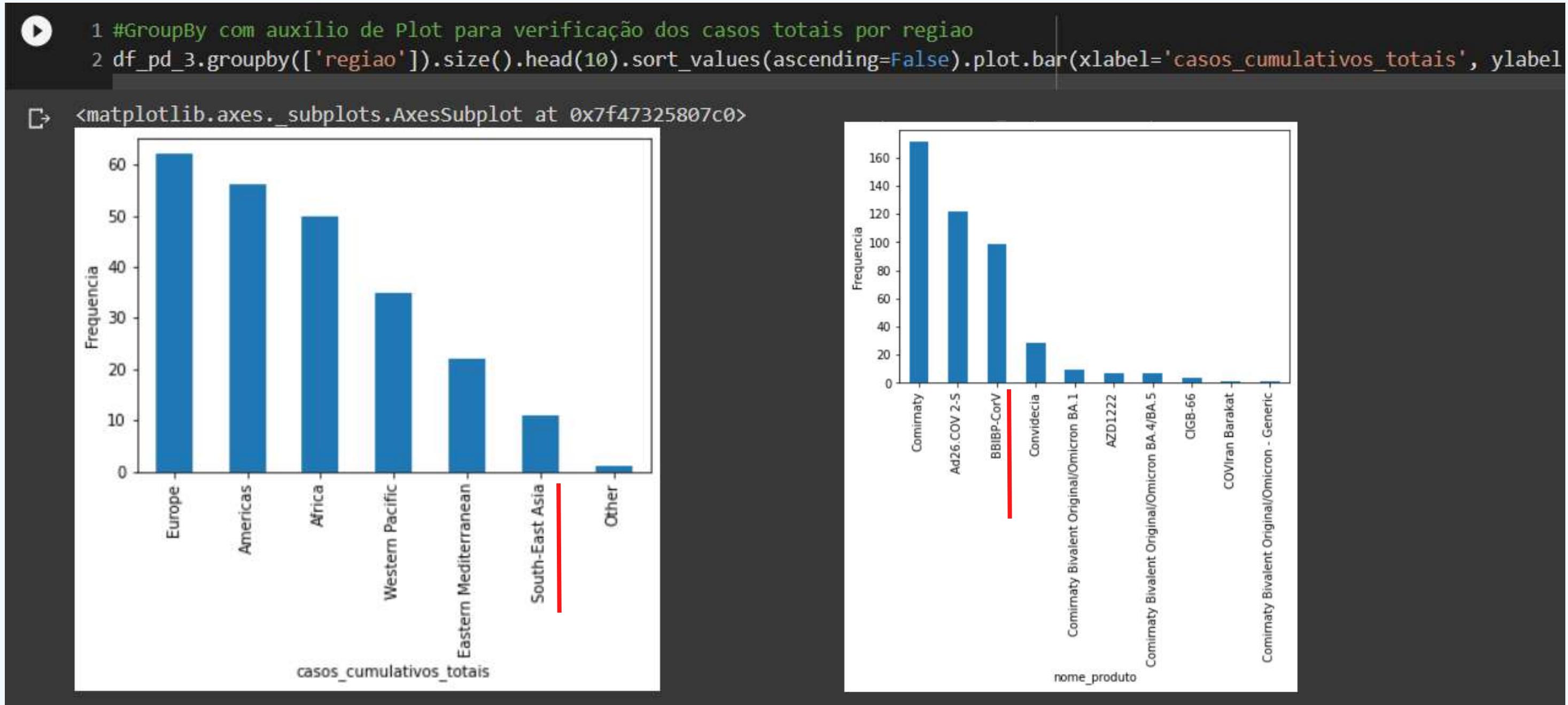
TRANSFORMAÇÃO

Groupbys com auxílio de Plot para gerar Insights



TRANSFORMAÇÃO

Groupbys com auxílio de Plot para gerar Insights



TRANSFORMAÇÃO

início da sessão/Spark

Configuração da SparkSession/Criação do Dataframe Spark

```
2 spark = (
3     SparkSession.builder
4         .master('local')
5         .appName('dataset')
6         .config('spark.ui.port', '4050')
7         .getOrCreate()
8 )
```

```
2 df = (
3     spark.createDataFrame(pd.read_json('/content/Europa-Covid-Test.json'))
4
5 )
```

TRANSFORMAÇÃO

Pré-Análise/Spark

Exibição do Dataset

```
1 | df.show(truncate = True)
```

country	country_code	year_week	level	region	region_name	population	new_cases	tests_done	testing_rate	positivity_rate	testing_data_source
Austria	AT	2020-W01	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W02	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W03	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W04	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W05	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W06	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W07	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W08	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W09	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W10	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W11	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W12	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W13	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W14	national	AT	Austria	8932664	NaN	NaN	NaN	NaN	null
Austria	AT	2020-W15	national	AT	Austria	8932664	1838.0	12339.0	138.1335	14.8959	Manual webscraping
Austria	AT	2020-W16	national	AT	Austria	8932664	686.0	58488.0	654.7655	1.1729	Manual webscraping
Austria	AT	2020-W17	national	AT	Austria	8932664	450.0	33443.0	374.39	1.3456000000000001	Manual webscraping
Austria	AT	2020-W18	national	AT	Austria	8932664	309.0	26598.0	297.7611	1.1617	Country website
Austria	AT	2020-W19	national	AT	Austria	8932664	260.0	42153.0	471.8973	0.6168	Country website
Austria	AT	2020-W20	national	AT	Austria	8932664	324.0	46001.0	514.9752	0.7043	Country website

only showing top 20 rows

TRANSFORMAÇÃO

Manipulação de Dados/Spark

Criação de duas novas colunas: 'Year'/'Week', contendo ano e semana respectivamente

Remoção da coluna 'year_week'

```
[ ] 1 split_cols = pyspark.sql.functions.split(df['year_week'], '-')

[ ] 1 df1 = df.withColumn('year', split_cols.getItem(0)) \
  2 | | | .withColumn('week', split_cols.getItem(1))
```

TRANSFORMAÇÃO

Manipulação de Dados/Spark

Tipagem das colunas/Utilização do StructType

```
[ ] 1 esquema = (
2     structType([
3         StructField('country', StringType()),
4         StructField('country_code', StringType()),
5         StructField('level', StringType()),
6         StructField('region', StringType()),
7         StructField('region_name', StringType()),
8         StructField('population', LongType()),
9         StructField('new_cases', FloatType()),
10        StructField('tests_done', FloatType()),
11        StructField('testing_rate', FloatType()),
12        StructField('positivity_rate', FloatType()),
13        StructField('testing_data_source', StringType()),
14        StructField('year', StringType()),
15        StructField('week', StringType()),
16    ])
17 )
```

TRANSFORMAÇÃO

Manipulação de Dados/Spark

Renomeação e tradução das colunas com base no significado pré definido pelo Centro Europeu de Prevenção e Controle de Doenças



```
1 df4 = (df3.withColumnRenamed("country","pais")
2     .withColumnRenamed("country_code","cod_pais")
3     .withColumnRenamed("level","nivel")
4     .withColumnRenamed("region","sigla")
5     .withColumnRenamed("region_name","nome_regiao")
6     .withColumnRenamed("population","populacao")
7     .withColumnRenamed("new_cases","novos_casos")
8     .withColumnRenamed("tests_done","testes_feitos")
9     .withColumnRenamed("testing_rate","taxa_teste")
10    .withColumnRenamed("positivity_rate","taxa_positiva")
11    .withColumnRenamed("testing_data_source","fonte_dados")
12    .withColumnRenamed("year","ano")
13    .withColumnRenamed("week","semana")
14 )
```

TRANSFORMAÇÃO

Manipulação de Dados/Spark

Renomeação e tradução das colunas com base no significado pré definido pelo Centro Europeu de Prevenção e Controle de Doenças

```
1 df5 = df4.withColumn("pais", regexp_replace("pais","Sweden","Suécia")) \
2     .withColumn("pais", regexp_replace("pais","Germany","Alemanha")) \
3     .withColumn("pais", regexp_replace("pais","France","França")) \
4     .withColumn("pais", regexp_replace("pais","Greece","Grécia")) \
5     .withColumn("pais", regexp_replace("pais","Slovakia","Eslováquia")) \
6     .withColumn("pais", regexp_replace("pais","Belgium","Bélgica")) \
7     .withColumn("pais", regexp_replace("pais","Finland","Finlândia")) \
8     .withColumn("pais", regexp_replace("pais","Malta","Malta")) \
9     .withColumn("pais", regexp_replace("pais","Croatia","Croácia")) \
10    .withColumn("pais", regexp_replace("pais","Italy","Itália")) \
11    .withColumn("pais", regexp_replace("pais","Lithuania","Lituânia")) \
12    .withColumn("pais", regexp_replace("pais","Norway","Noruega")) \
13    .withColumn("pais", regexp_replace("pais","Spain","Espanha")) \
14    .withColumn("pais", regexp_replace("pais","Czechia","Tchéquia")) \
15    .withColumn("pais", regexp_replace("pais","Denmark","Dinamarca")) \
```

TRANSFORMAÇÃO

Manipulação de Dados/Spark

Contagem de linhas gerais/Contagem de linhas duplicatas

```
[ ] 1 df5.count()
```

```
4649
```

```
▶ 1 df6 = df5.dropDuplicates()  
2 df6.count()
```

```
4649
```

TRANSFORMAÇÃO

Manipulação de Dados/Spark

Drop de colunas, verificação de valores nulos nas demais colunas



```
1 df7 = df6.drop('nome_regiao','sigla','nivel')
2 df7.show()
```

```
[ ] 1 df2 = df1.drop('year_week')
```

```
[ ] 1 df11 = df10.drop('fonte_dados')
```

TRANSFORMAÇÃO

Manipulação de Dados/Spark

Drop de linhas com valores com inconsistentes(NaN)

```
2 df8 = df7.where(F.col("fonte_dados").isNull())
3 df8.show()
```

pais	cod_pais	populacao	novos_casos	teste_ok	taxa_teste	taxa_positiva	fonte_dados	ano	s
belgica	BE	11554767	NaN	NaN	NaN	NaN	null	2020	
islandia	IS	368792	NaN	NaN	NaN	NaN	null	2020	
portugal	PT	10298252	NaN	NaN	NaN	NaN	null	2020	
belgica	BE	11554767	NaN	NaN	NaN	NaN	null	2021	
hungria	HU	9730772	NaN	NaN	NaN	NaN	null	2020	
letonia	LV	1893223	NaN	NaN	NaN	NaN	null	2021	
letonia	LV	1893223	NaN	NaN	NaN	NaN	null	2021	
austria	AT	8932664	NaN	NaN	NaN	NaN	null	2022	
italia	IT	59236213	NaN	NaN	NaN	NaN	null	2022	
belgical	RF	11554767	NaN	NaN	NaN	NaN	null	2021	

TRANSFORMAÇÃO

Manipulação de Dados/Spark

Drop de linhas com valores com inconsistentes(NaN)

```
2 df8 = df7.where(~ F.col("fonte_dados").isNull())
3 df8.show()
```

pais	cod_pais	populacao	novos_casos	testes_feitos	taxa_teste	taxa_positiva	fonte_dados
Bulgária	BG	6916548	7746.0	83755.0	1210.9364	9.2484	TES
Croácia	HR	4036355	2445.0	57322.0	1420.1427	4.2654	TES
Croácia	HR	4036355	9811.0	108810.0	2695.749	9.0166	TES
Croácia	HR	4036355	11152.0	107017.0	2651.3276	10.4208	TES
Chipre	CY	896007	1834.0	290194.0	32387.47	0.632	TES
Finlândia	FI	5533793	4390.0	163609.0	2956.5435	2.6832	TES
Finlândia	FI	5533793	10623.0	35361.0	639.0011	30.0416	TES
França	FR	67656682	3392.0	151072.0	223.2921	2.2453	TES
França	FR	67656682	85437.0	2364348.0	3494.626	3.6136	TES
Alemanha	DE	83155031	144675.0	1436474.0	1727.4648	10.0715	TES

TRANSFORMAÇÃO

Manipulação de Dados/Spark

Substituição de valores nulos por 0 com o objetivo de facilitar operações de insights

Verificação se a substituição foi efetivada

```
[ ] 1 df_teste = df8.select([count(when  
2 | | | | | | | (isnan(c) | col(c).isNull(), c)).alias(c)  
3 for c in df8.columns]  
4 ).show()
```

+	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	+
	pais	cod_pais	populacao	novos_casos	teste_ok	taxa_teste	taxa_positiva	fonte_dados	ano	semana	
+	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	+
	0	0	0	37	0	0	60	0	0	0	

TRANSFORMAÇÃO

Manipulação de Dados/Spark

Substituição de valores nulos por 0 com o objetivo de facilitar operações de insights

Verificação se a substituição foi efetivada

```
1 df9 = df8.fillna({'taxa_positiva': 0})  
2 df10 = df9.fillna({'novos_casos': 0})
```

pais	cod_pais	populacao	novos_casos	teste_ok	taxa_teste	taxa_positiva	fonte_dados	ano	semana
0	0	0	0	0	0	0	0	0	0

TRANSFORMAÇÃO

Groupbys e filtros para gerar Insights

Quantidade de novos casos por país / Quantidade de testes por país

```
1 df11.groupBy('pais').sum('novos_casos').show(truncate = False)
```

pais	sum(novos_casos)
Luxemburgo	352941.0
Suécia	2626753.0
Polônia	6370020.0
França	3.8802839E7
Alemanha	3.6784535E7
Países Baixos	8557599.0
Áustria	5076291.0
Bélgica	3554348.0
Romênia	3302295.0
Malta	115963.0

```
1 df11.groupBy('pais').sum('testes_feitos').show(truncate = False)
```

pais	sum(testes_feitos)
Luxemburgo	4506448.0
Suécia	1.7697679E7
Polônia	3.8411268E7
França	2.88662064E8
Alemanha	1.48850151E8
Países Baixos	5.1661236E7
Áustria	2.28605777E8
Bélgica	2.2194749E7
Romênia	2.3039716E7
Malta	2120700.0

Dataframes Pós Tratamento

Renomeação, Tradução, Correção de valores nulos, Drop de colunas

	Unnamed: 0	Date_reported	Country_code	Country	WHO_region	New_cases	Cumulative_cases
0	0	2020-01-03	AF	Afghanistan	EMRO	0	0
1	1	2020-01-04	AF	Afghanistan	EMRO	0	0
2	2	2020-01-05	AF	Afghanistan	EMRO	0	0
3	3	2020-01-06	AF	Afghanistan	EMRO	0	0
4	4	2020-01-07	AF	Afghanistan	EMRO	0	0
...
257377	257377	2022-12-19	ZW	Zimbabwe	AFRO	219	259981
257378	257378	2022-12-20	ZW	Zimbabwe	AFRO	0	259981
257379	257379	2022-12-21	ZW	Zimbabwe	AFRO	0	259981
257380	257380	2022-12-22	ZW	Zimbabwe	AFRO	0	259981
257381	257381	2022-12-23	ZW	Zimbabwe	AFRO	0	259981
257382 rows × 9 columns							

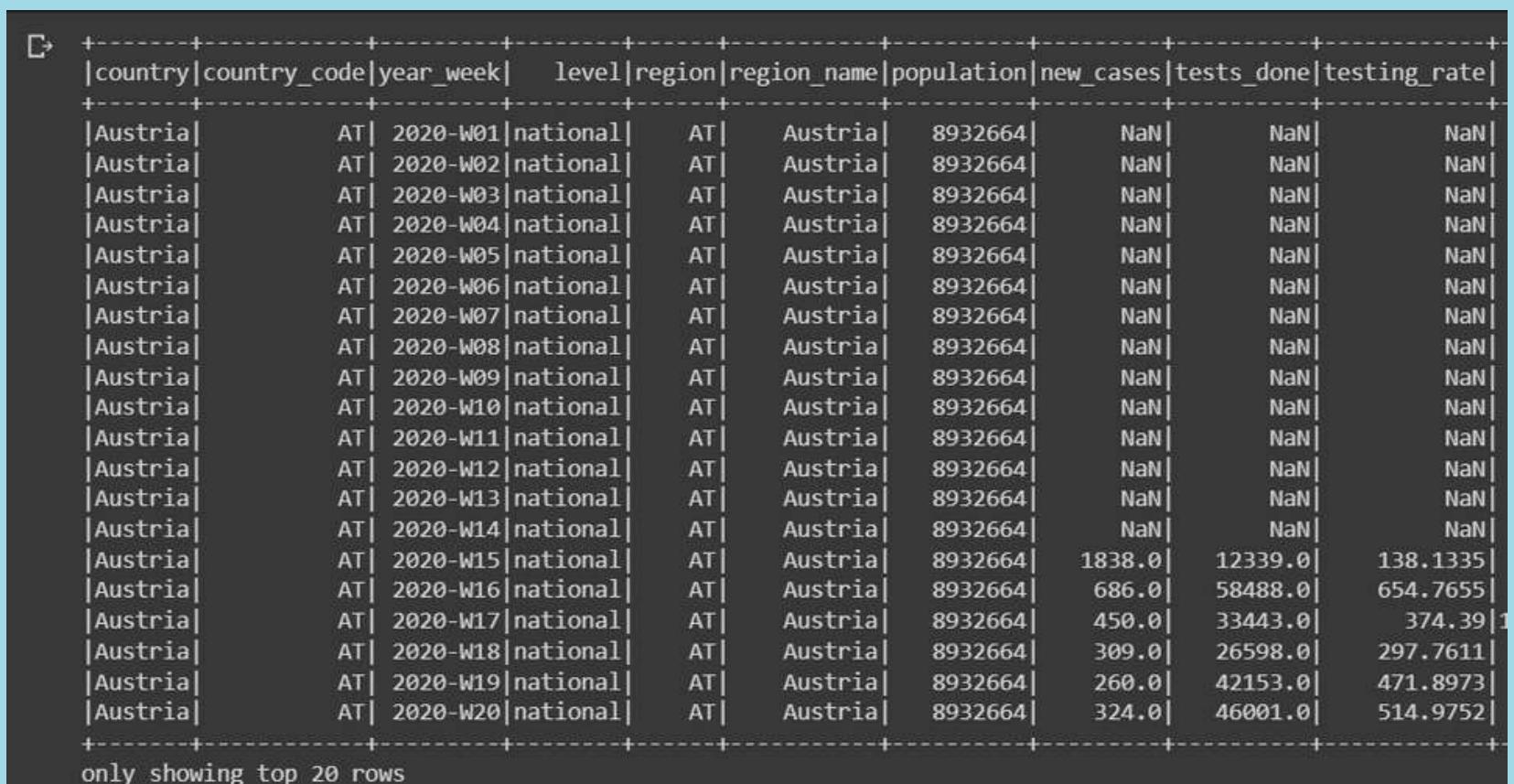
Antes

	data_relatada	cod_pais	país	regiao_who	novos_casos	casos_cumulativos
0	2020-02-24	AF	Afeganistão	Mediterrâneo Oriental	5	5
1	2020-02-25	AF	Afeganistão	Mediterrâneo Oriental	0	5
2	2020-02-26	AF	Afeganistão	Mediterrâneo Oriental	0	5
3	2020-02-27	AF	Afeganistão	Mediterrâneo Oriental	0	5
4	2020-02-28	AF	Afeganistão	Mediterrâneo Oriental	0	5
...
230179	2022-12-19	Other	Outro	Outro	0	764
230180	2022-12-20	Other	Outro	Outro	0	764
230181	2022-12-21	Other	Outro	Outro	0	764
230182	2022-12-22	Other	Outro	Outro	0	764
230183	2022-12-23	Other	Outro	Outro	0	764
230184 rows × 8 columns						

Depois

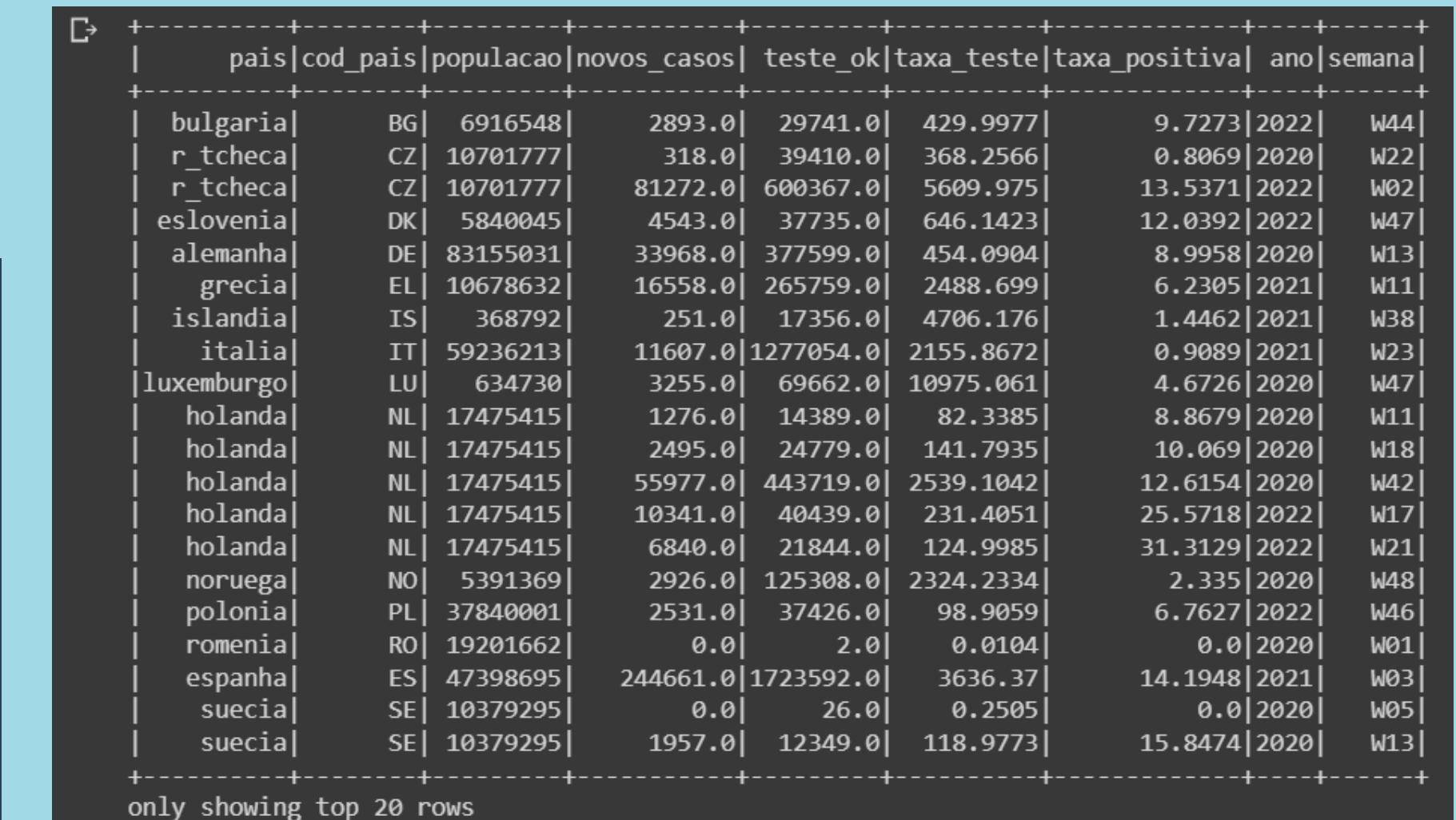
Dataframes Pós Tratamento

Renomeação, Tradução, Correção de valores nulos, Drop de colunas



country	country_code	year_week	level	region	region_name	population	new_cases	tests_done	testing_rate
Austria	AT	2020-W01	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W02	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W03	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W04	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W05	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W06	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W07	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W08	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W09	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W10	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W11	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W12	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W13	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W14	national	AT	Austria	8932664	NaN	NaN	NaN
Austria	AT	2020-W15	national	AT	Austria	8932664	1838.0	12339.0	138.1335
Austria	AT	2020-W16	national	AT	Austria	8932664	686.0	58488.0	654.7655
Austria	AT	2020-W17	national	AT	Austria	8932664	450.0	33443.0	374.391
Austria	AT	2020-W18	national	AT	Austria	8932664	309.0	26598.0	297.7611
Austria	AT	2020-W19	national	AT	Austria	8932664	260.0	42153.0	471.8973
Austria	AT	2020-W20	national	AT	Austria	8932664	324.0	46001.0	514.9752

Antes



pais	cod_pais	populacao	novos_casos	teste_ok	taxa_teste	taxa_positiva	ano	semana
bulgaria	BG	6916548	2893.0	29741.0	429.9977	9.7273	2022	W44
r_tcheca	CZ	10701777	318.0	39410.0	368.2566	0.8069	2020	W22
r_tcheca	CZ	10701777	81272.0	600367.0	5609.975	13.5371	2022	W02
eslovenia	DK	5840045	4543.0	37735.0	646.1423	12.0392	2022	W47
alemanha	DE	83155031	33968.0	377599.0	454.0904	8.9958	2020	W13
grecia	EL	10678632	16558.0	265759.0	2488.699	6.2305	2021	W11
islandia	IS	368792	251.0	17356.0	4706.176	1.4462	2021	W38
italia	IT	59236213	11607.0	1277054.0	2155.8672	0.9089	2021	W23
luxemburgo	LU	634730	3255.0	69662.0	10975.061	4.6726	2020	W47
holanda	NL	17475415	1276.0	14389.0	82.3385	8.8679	2020	W11
holanda	NL	17475415	2495.0	24779.0	141.7935	10.069	2020	W18
holanda	NL	17475415	55977.0	443719.0	2539.1042	12.6154	2020	W42
holanda	NL	17475415	10341.0	40439.0	231.4051	25.5718	2022	W17
holanda	NL	17475415	6840.0	21844.0	124.9985	31.3129	2022	W21
noruega	NO	5391369	2926.0	125308.0	2324.2334	2.335	2020	W48
polonia	PL	37840001	2531.0	37426.0	98.9059	6.7627	2022	W46
romenia	RO	19201662	0.0	2.0	0.0104	0.0	2020	W01
espanha	ES	47398695	244661.0	1723592.0	3636.37	14.1948	2021	W03
suecia	SE	10379295	0.0	26.0	0.2505	0.0	2020	W05
suecia	SE	10379295	1957.0	12349.0	118.9773	15.8474	2020	W13

Depois

CARREGAMENTO

Preparação dos arquivos Envio para a Bucket na GCP

```
[ ] 1 #Convertendo data frame em arquivo csv
[ ] 2 df11.write.format("parquet").mode("overwrite").save("/content/eu_covid_tratado.csv")
[ ] 3

[ ] 1 df_pd_1 = pd.read_parquet("/content/eu_covid_tratado.csv/part-00000-10c9a54c-b0a2-4622-85c2-353c24d4b8d5-c000.snappy.parquet")

[ ] 1 #Enviando data frame tradado em formato csv
[ ] 2 df_pd_1.to_csv('gs://projfinal/tratados/df_pd_1_europa-covid-test.csv',storage_options={'token': '/content/bc26-rdg-ed27-7fdc00e9e239.json'})
[ ] 3
```

<input type="checkbox"/>	Name	Size
<input type="checkbox"/>	df_pd_1_europa-covid-test.csv	293.9 KB
<input type="checkbox"/>	df_pd_2_global_data.csv	13.1 MB
<input type="checkbox"/>	df_pd_3_global_table_data.csv	16.4 KB
<input type="checkbox"/>	df_pd_4_vaccination-metadata.cs...	62.7 KB

CARREGAMENTO

Envio para o MongoDB (Não Relacional)

```
[ ] 1 # Conectando com o MongoDB
2
3 uri = "mongodb+srv://rodrigo-soulcode.lr83n68.mongodb.net/?authSource=%24external&authMechanism=MONGODB-X509&retryWrites=true&w=majority"
4 client = MongoClient(uri,tls=True,tlsCertificateKeyFile='/content/X509-cert-1489879420125386803.pem')

[ ] 1 # Criando coleções para enviar para o mongoDB
2
3 db = client['projfinal']
4 colecaotratado1 = db['tratado1']

[ ] 1 #Enviando os Datasets tratados para o mongoDB
2
3 df_pd_1.reset_index(drop=True)
4 df01 = df_pd_1.to_dict("records")
5 colecaotratado1.insert_many(df01)

<pymongo.results.InsertManyResult at 0x7f7324b890a0>
```

RODRIGO'S ORG - 2022-12-01 > PROJECT 0 > DATABASES

 **rodrigo-SOULCODE** VERSION: 5.0.14 REGION: GCP Sao Paulo (southamerica-east1)

Overview Real Time Metrics Collections Search Profiler Performance Advisor Online Archive Cmd Line Tools

DATABASES: 1 COLLECTIONS: 4

+ Create Database VISUALIZE YOUR DATA REFRESH

Search Namespaces

projfinal

tratado1

tratado2

tratado3

tratado4

projfinal.tratado1

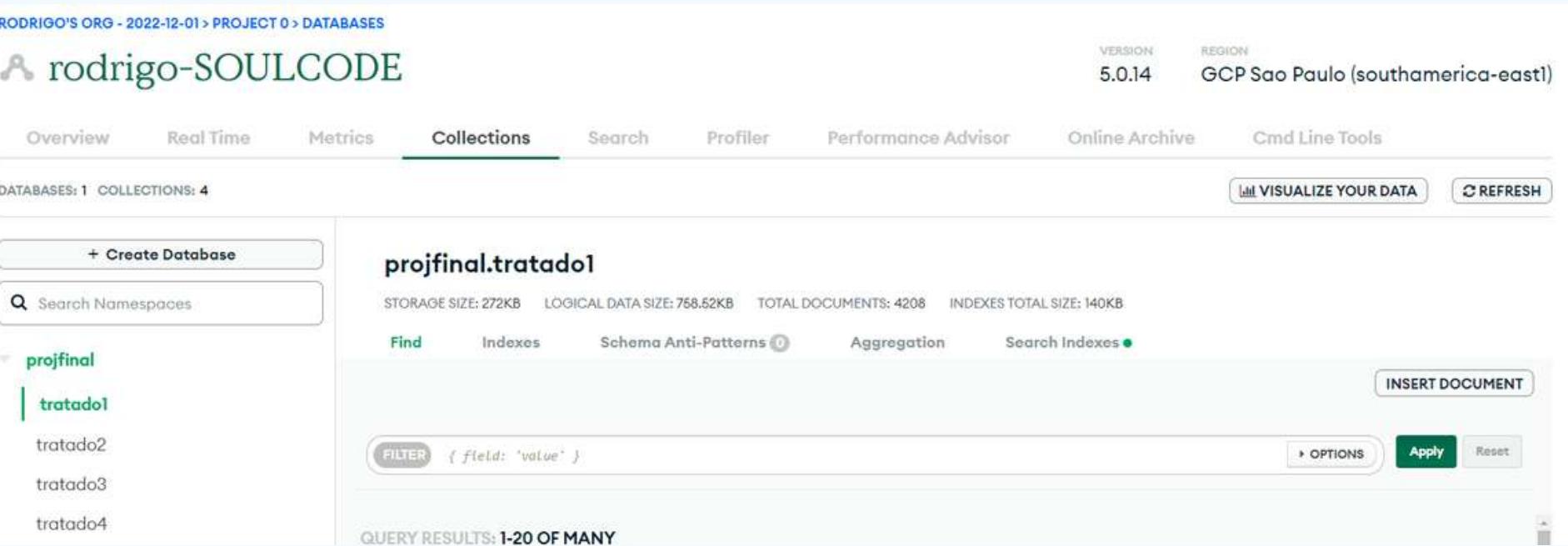
STORAGE SIZE: 272KB LOGICAL DATA SIZE: 758.52KB TOTAL DOCUMENTS: 4208 INDEXES TOTAL SIZE: 140KB

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

INSERT DOCUMENT

OPTIONS Apply Reset

QUERY RESULTS: 1-20 OF MANY



ANÁLISE DE DADOS

ANÁLISE DE DADOS

Querys executadas no BigQuery

Schema dos Datasets

Europa_Covid				
ESQUEMA	DETALHES	VISUALIZAR	LINHAGEM	
<input type="checkbox"/>	Nome do campo	Tipo	Modo	Compilação
<input type="checkbox"/>	int64_field_0	INTEGER	NULLABLE	
<input type="checkbox"/>	pais	STRING	NULLABLE	
<input type="checkbox"/>	cod_pais	STRING	NULLABLE	
<input type="checkbox"/>	populacao	INTEGER	NULLABLE	
<input type="checkbox"/>	novos_casos	FLOAT	NULLABLE	
<input type="checkbox"/>	teste_ok	FLOAT	NULLABLE	
<input type="checkbox"/>	taxa_teste	FLOAT	NULLABLE	
<input type="checkbox"/>	taxa_positiva	FLOAT	NULLABLE	
<input type="checkbox"/>	ano	DATE	NULLABLE	
<input type="checkbox"/>	semana	STRING	NULLABLE	

Global_Data				
ESQUEMA	DETALHES	VISUALIZAR	LINHAGEM	
Filtro Insira o nome ou o valor da propriedade				
<input type="checkbox"/>	Nome do campo	Tipo	Modo	Compilação
<input type="checkbox"/>	int64_field_0	INTEGER	NULLABLE	
<input type="checkbox"/>	data_relatada	DATE	NULLABLE	
<input type="checkbox"/>	cod_pais	STRING	NULLABLE	
<input type="checkbox"/>	pais	STRING	NULLABLE	
<input type="checkbox"/>	regiao_who	STRING	NULLABLE	
<input type="checkbox"/>	novos_casos	INTEGER	NULLABLE	
<input type="checkbox"/>	casos_cumulativos	INTEGER	NULLABLE	
<input type="checkbox"/>	novas_mortes	INTEGER	NULLABLE	
<input type="checkbox"/>	mortes_cumulativas	INTEGER	NULLABLE	

ANÁLISE DE DADOS

Querys executadas no BigQuery

Schema dos Datasets

Global_Table_Data		CONSULTA	COMPARTILHAR	COPIAR
ESQUEMA	DETALHES	VISUALIZAR	LINHAGEM	PRÉ-VISUALIZAÇÃO
<input type="checkbox"/>	int64_field_0		INTEGER	NULLABLE
<input type="checkbox"/>	pais		STRING	NULLABLE
<input type="checkbox"/>	regiao		STRING	NULLABLE
<input type="checkbox"/>	casos_cumulativos_totais		FLOAT	NULLABLE
<input type="checkbox"/>	casos_cumulativos_por_100mil_hab		INTEGER	NULLABLE
<input type="checkbox"/>	casos_nos_ultimos_7dias		FLOAT	NULLABLE
<input type="checkbox"/>	casos_nos_ultimos_7dias_por_100mil_hab		INTEGER	NULLABLE
<input type="checkbox"/>	casos_nas_ultimas_24hrs		INTEGER	NULLABLE
<input type="checkbox"/>	mortes_cumulativas_totais		FLOAT	NULLABLE
<input type="checkbox"/>	mortes_cumulativas_totais_por_100mil_hab		INTEGER	NULLABLE
<input type="checkbox"/>	mortes_nos_ultimos_7dias		FLOAT	NULLABLE
<input type="checkbox"/>	mortes_nos_ultimos_7dias_por_100mil_hab		INTEGER	NULLABLE
<input type="checkbox"/>	mortes_nas_ultimas_24hrs		STRING	NULLABLE

Vaccination		CONSULTA	COMPARTILHAR
ESQUEMA	DETALHES	VISUALIZAR	LINHAGEM
Filtro Insira o nome ou o valor da propriedade			
<input type="checkbox"/>	Nome do campo	Tipo	Modo
<input type="checkbox"/>	int64_field_0	INTEGER	NULLABLE
<input type="checkbox"/>	pais	STRING	NULLABLE
<input type="checkbox"/>	produto	STRING	NULLABLE
<input type="checkbox"/>	empresa	STRING	NULLABLE
<input type="checkbox"/>	data_de_autorizacao	DATE	NULLABLE
<input type="checkbox"/>	data_de_inicio	DATE	NULLABLE

ANÁLISE DE DADOS

Querys executadas no BigQuery

Global_Data_1 - Filtro por intervalo de data em ordem decrescente

The screenshot shows the Google BigQuery web interface. At the top, there are three tabs: '*Unsaved query' (selected), 'DF2_GLOBAL_DATA' (with a warning icon), and '*Unsaved query 2'. Below the tabs are buttons for RUN, SAVE, SHARE, SCHEDULE, MORE, and a green checkmark for Query comp. The main area contains the following SQL query:

```
1 SELECT
2   data_relatada,
3   pais,
4   casos_cumulativos,
5   mortes_cumulativas,
6   FROM `aulas-bc-26-nayara-n3-372217.Projetofinal.DF2_GLOBAL_DATA`
7   WHERE data_relatada >= '2021-01-01' AND data_relatada < '2021-12-31'
8   ORDER BY casos_cumulativos desc |
```

Below the query, a message says 'Press Alt+F1 for Accessibility'. Under the heading 'Query results', there are buttons for SAVE RESULTS and EXPLORE DATA. A table titled 'JOB INFORMATION' shows the following data:

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS	EXECUTION GRAPH	PREVIEW
Row	data_relatada	pais	casos_cumulativos	mortes_cumulativas		
1	2021-12-30	Estados Unidos da America	53059977	817232		
2	2021-12-29	Estados Unidos da America	52670463	814901		
3	2021-12-28	Estados Unidos da America	52205968	813115		
4	2021-12-27	Estados Unidos da America	51994684	812849		

At the bottom, it says 'Results per page: 200' and '1 - 200 of 81467'.

ANÁLISE DE DADOS

Querys executadas no BigQuery

Vaccination Data - Diferença de dias entre data de autorização e início da vacinação em ordem decrescente

```
1 #DIFERENÇA DE DIAS/SEMANA APÓS RECEBER AUTORIZAÇÃO
2 SELECT
3 pais,
4 produto,
5 DATE_DIFF(data_de_inicio, data_de_autorizacao, DAY) AS APOS_DIAS,
6 DATE_DIFF(data_de_inicio, data_de_autorizacao, WEEK) AS APOS_SEMANAS
7 FROM `bc26-rdg-ed27.projfinal.Vaccination`
8 WHERE data_de_inicio is not null and data_de_autorizacao is not null
9 ORDER BY pais desc
```

Resultados da consulta SALVAR RESULTADO

INFORMAÇÕES DO JOB RESULTADOS JSON DETALHES DA EXECUÇÃO GRÁFICO DE EXECUÇÃO

Linha	país	produto	APOS_DIAS	APOS_SEMANAS
1	Índia	Covaxin	13	1
2	Índia	Covishield	13	1
3	Índia	Gam-Covid-Vac	32	4
4	Zâmbia	Covishield	45	6
5	Wallis e Futuna	Spikevax	72	10
6	Vietnã	Spikevax	3	0

ANÁLISE DE DADOS

Querys executadas no BigQuery

Global Data - Top 10 Casos e mortes cumulativas

EXECUTAR SALVAR COMPARTILHAR PROGRAMAÇÃO MAIS Esta consulta pro...

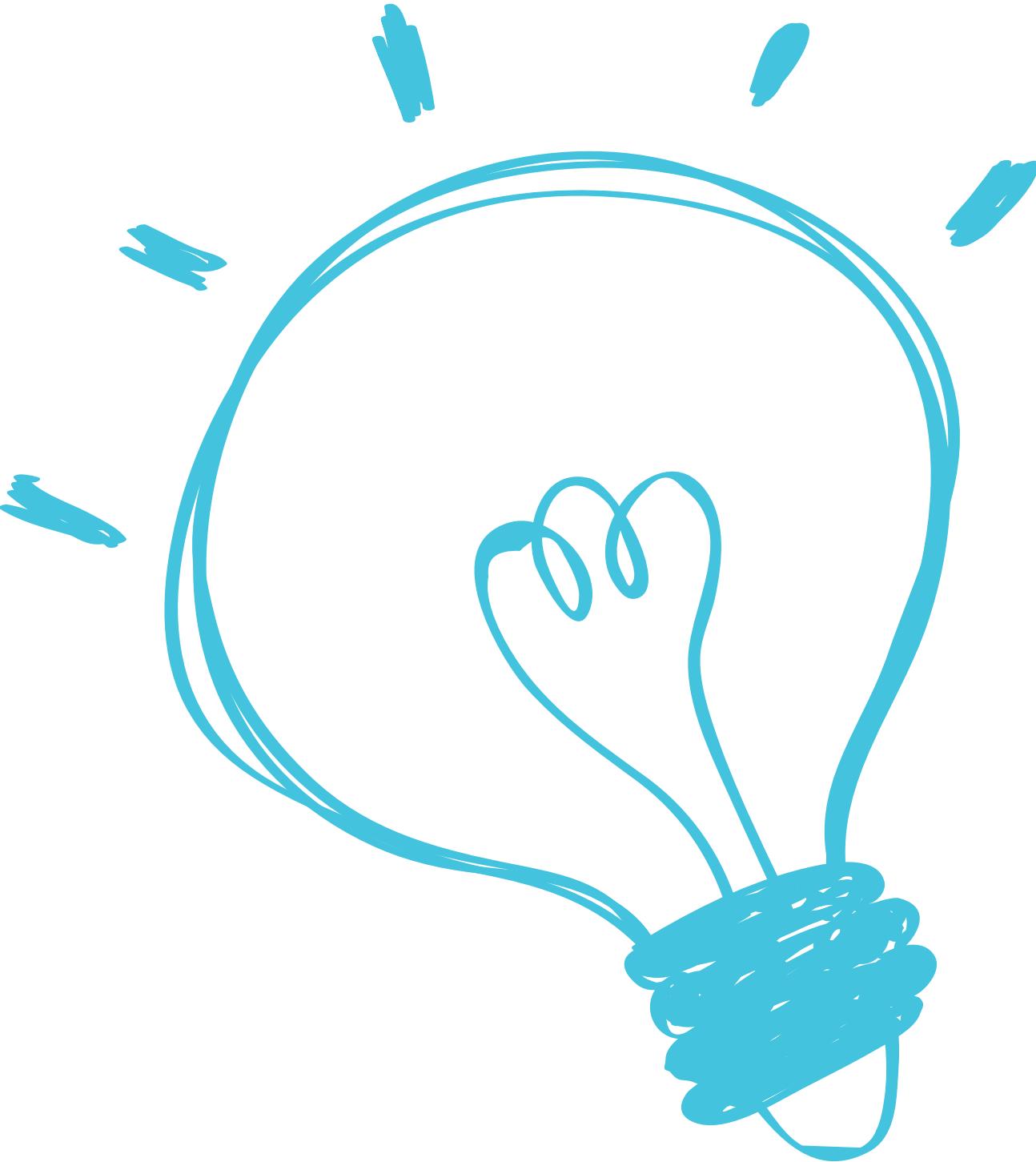
```
1 # TOP 10 SOMA CASOS CUMULATIVOS
2 SELECT
3 pais,
4 SUM(casos_cumulativos_totais) as total_casos,
5 SUM(mortes_cumulativas_totais) as total_mortes
6 FROM
7 `bc26-rdg-ed27.projfinal.Global_Table_Data`
8 GROUP BY pais
```

Pressione Alt+F1 para consultar as opções de acessibilidade.

Resultados da consulta SALVAR RESULTADOS EXPLORAR DADOS

Linha	INFORMAÇÕES DO JOB	RESULTADOS	JSON	DETALHES DA EXECUÇÃO	GRÁFICO DE EXECUÇÃO
1	Índia	3237.43	38.456		
2	África do Sul	6822.951	172.909		
3	Zâmbia	1815.42	21.861		
4	Zimbábue	1749.191	37.927		
5	Wallis e Futuna	30366.352	62.244		
6	Vietnã	11838.643	44.362		
7	Venezuela (Bolivarian Republic ...	1932.582	20.502		
8	Venezuela	2011.107	4.550		

Resultados por página: 50 1 - 50 de 238 < < > >>



Insights Iniciais

- Países com maiores casos seriam Itália, Estados Unidos.
- Quais países tiveram índices elevados e não foram retratados?
- Verificação de vacinas(quantos tipos de fato existem?)
- Primeiras vacinas a serem utilizadas foram a Pfizer e AstraZeneca
- Verificação de períodos de autorização/início da vacinação.



Looker



Conclusões

- Os 4 países com maiores índices de casos cumulativos totais foram: Ilhas Faroé, Chipre, San Marino e Áustria sendo que somente Itália e EUA foram mais debatidos
- Existem 37 vacinas das quais só conhecemos 5
- As primeiras vacinas a serem utilizadas foram a BBIBP-CorV, Coronavac e Convidecia
- Os países que começaram primeiro período de vacinação foram a China, Bahrein e República Democrática Popular do Laos, sendo que só a China foi divulgada.
- Dados faltantes

Obrigado!



pablohenrique93



vinicius-sodreA



vinícius-santos-vsas



naolip



rodrigodataeng