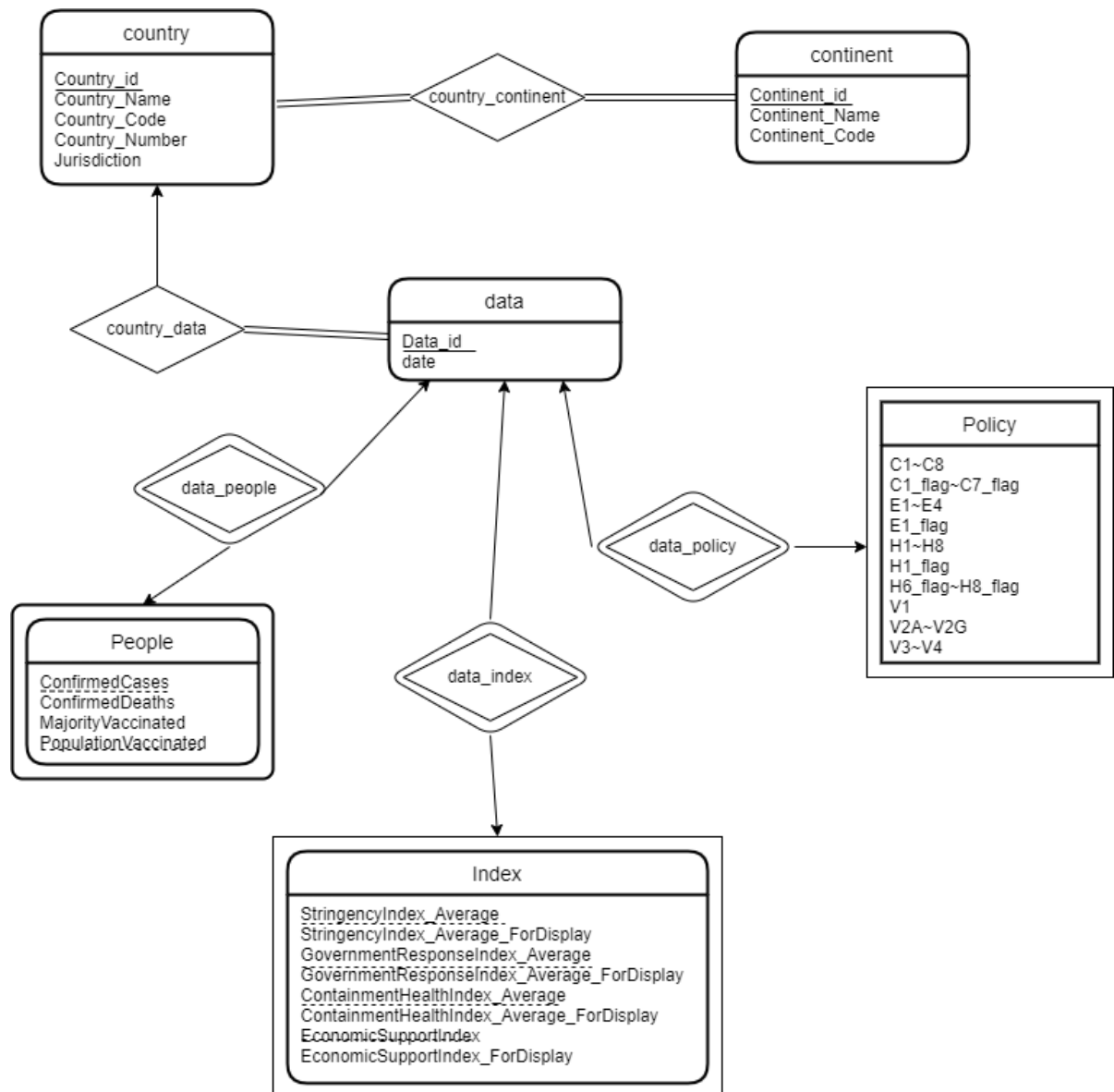


資料庫系統概論 HW2

110550108 施柏江

1. ER diagram



Since "RegionName", "RegionCode", "M1_Wildcard" in OxCGRT_nat_latest.csv have only NULL value, and I think "Two_Letter_Country_Code" in country-and-continent-codes-list-csv.csv are redundancy because we already have "Three_Letter_Country_Code", so I ignore them in the ER diagram.

2. AWS RDS

RDS > 数据库 > database-2

database-2

修改 動作 ▼

摘要

資料庫識別符
database-2

CPU
5.04%

狀態
可用

類別
db.t3.micro

角色

目前活動
0.00 sessions

引擎
PostgreSQL

區域與可用區域 (AZ)
us-east-1d

連線與安全性

監控

日誌與事件

組態

維護與備份

標籤

連線與安全性

端點與連接埠

端點
database-2.cwhx9uqpnqq.us-east-1.rds.amazonaws.com

聯網

可用區域
us-east-1d

VPC
vpc-0bd3f6211f5a1876d

子網路群組
default-vpc-0bd3f6211f5a1876d

子網路
subnet-056e12d44e6393cd2
subnet-04ea4df37186fb0b6
subnet-0e39e03ede09efb65
subnet-0745e05f949dfcd43
subnet-03e56127503d92ef9
subnet-0bcb114c92807afc3

網路類型
IPv4

安全性

VPC 安全群組
dbhw1 (sg-04e6f5620fbe2cdcd)
作用中

公開存取情形
是

憑證授權單位
rds-ca-2019

憑證授權單位日期
August 23, 2024, 01:08 (UTC+08:00)

PGAdmin

File Object Tools Help

Dashboard Properties SQL Statistics Dependencies Dependents Processes

Servers (2)
PostgreSQL 15
dbhw1

General

ID
2

Name
dbhw1

Server type
PostgreSQL

Version
PostgreSQL 13.7 on x86_64-pc-linux-gnu, compiled by gcc (GCC) 7.3.1 20180712 (Red Hat 7.3.1-12), 64-bit

Comments

Connection

Connected?
是

Host name/address
database-2.cwhx9uqpnqq.us-east-1.rds.amazonaws.com

Port
5432

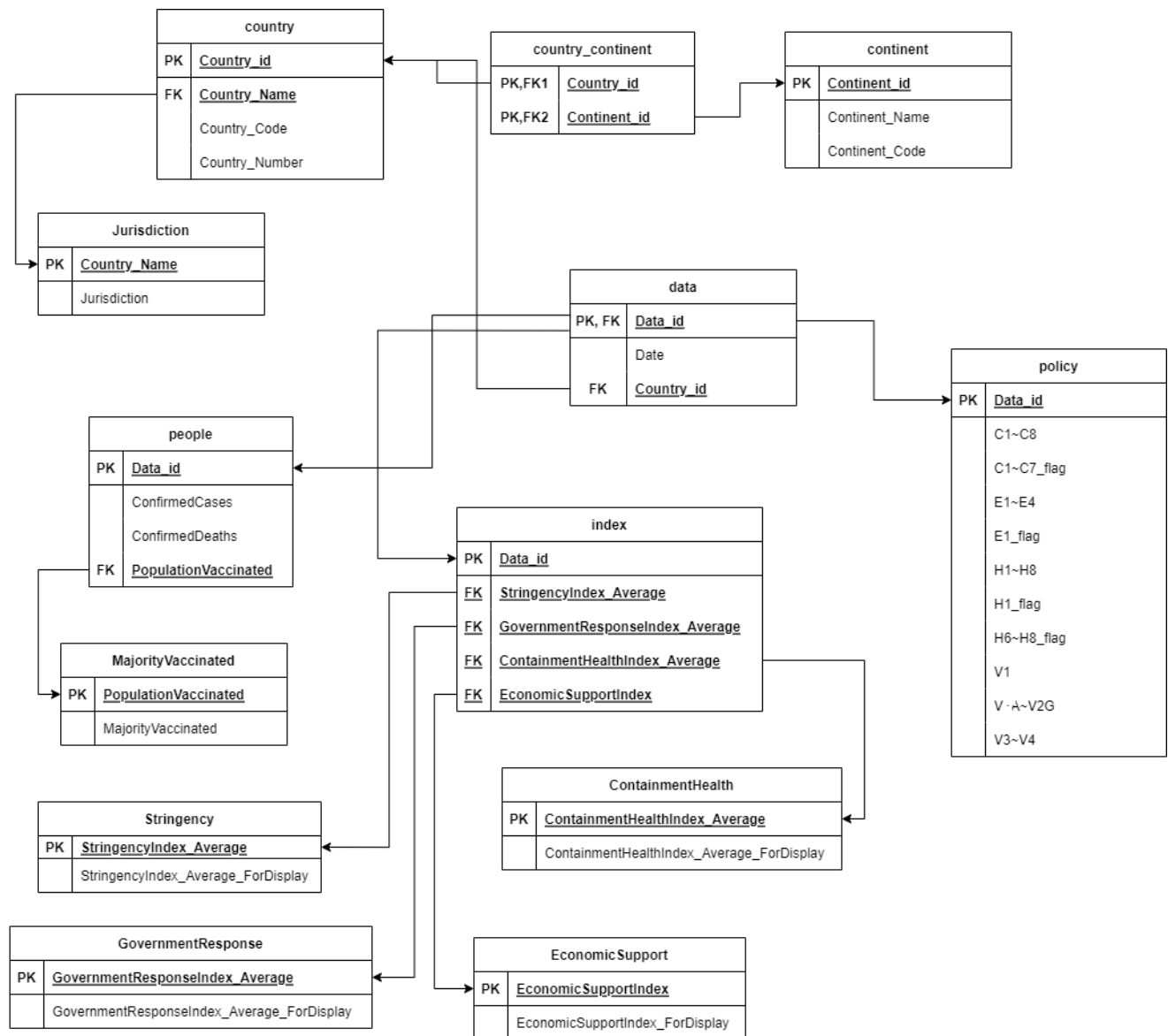
Maintenance database
oxcgrt

Username
brianshih59

Kerberos authentication?
否

GSS authenticated?
否

3. Relational schema



Since “Kosovo” is not included in country-and-continent-codes-list-csv.csv, so I add it manually.

Create table “country”

The screenshot shows the pgAdmin interface with the 'Schemas' tree on the left. The 'Query' tab is active, displaying the following SQL code:

```
1 create table country(  
2     Country_id int,  
3     Country_Name varchar(80),  
4     Country_Code char(3),  
5     Country_Number int,  
6     primary key(Country_id),  
7     foreign key(Country_Name) references jurisdiction(Country_Name)  
8 )
```

The 'Messages' tab shows the execution result:

```
CREATE TABLE  
Query returned successfully in 288 msec.
```

The status bar at the bottom indicates: 'Total rows: 0 of 0 Query complete 00:00:00.280' and a green checkmark with the message 'Query returned successfully in 288 msec.'

Create table “continent”

The screenshot shows the pgAdmin interface with the 'Schemas' tree on the left. The 'Query' tab is active, displaying the following SQL code:

```
1 create table continent(  
2     Continent_id int,  
3     Continent_Name varchar(50),  
4     Continent_Code char(2),  
5     primary key(Continent_id)  
6 )
```

The 'Messages' tab shows the execution result:

```
CREATE TABLE  
Query returned successfully in 276 msec.
```

The status bar at the bottom indicates: 'Total rows: 0 of 0 Query complete 00:00:00.270' and a green checkmark with the message 'Query returned successfully in 270 msec.'

Create table “country_continent”

The screenshot shows the pgAdmin interface with the 'Schemas' tree on the left. The 'Query' tab is active, displaying the following SQL code:

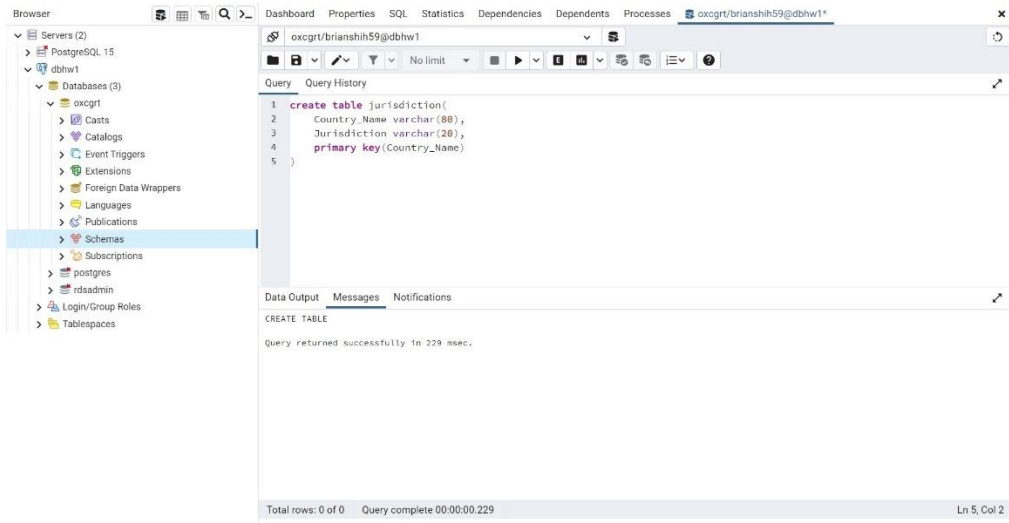
```
1 create table country_continent(  
2     Country_id int,  
3     Continent_id int,  
4     primary key(Country_id, Continent_id),  
5     foreign key(Country_id) references country(Country_id),  
6     foreign key(Continent_id) references continent(Continent_id)  
7 )
```

The 'Messages' tab shows the execution result:

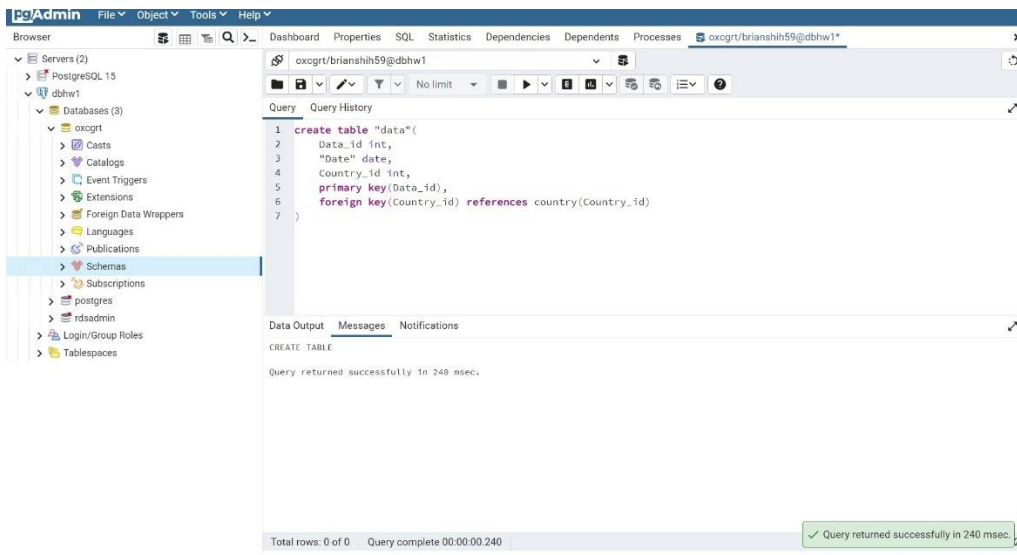
```
CREATE TABLE  
Query returned successfully in 268 msec.
```

The status bar at the bottom indicates: 'Total rows: 0 of 0 Query complete 00:00:00.268' and a green checkmark with the message 'Query returned successfully in 268 msec.'

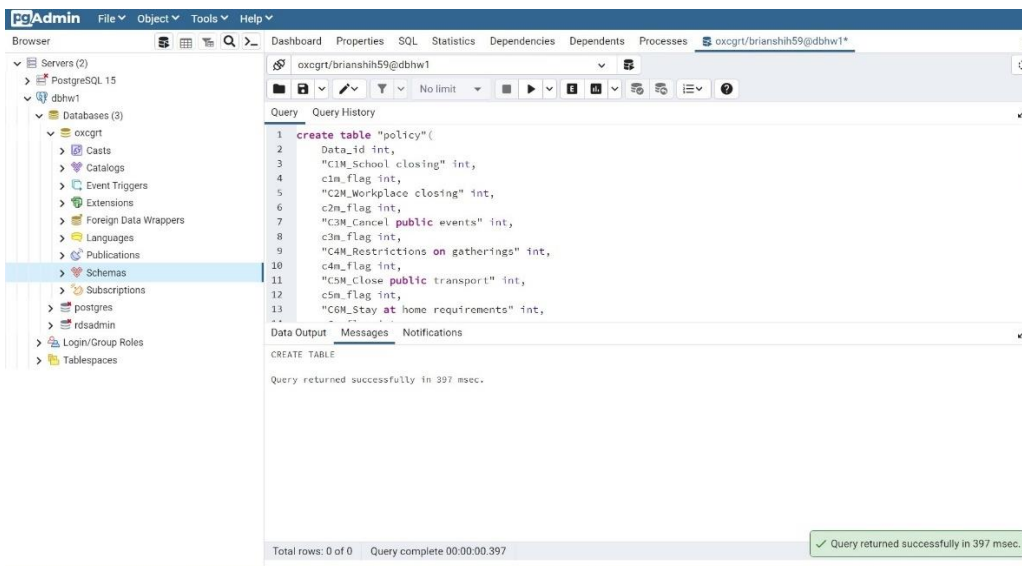
Create table “jurisdiction”



Create table "data"



Create table “policy”



```

14     c6m_flag int,
15     "C7M_Restrictions on internal movement" int,
16     c7m_flag int,
17     "C8EV_International travel controls" int,
18     "E1_Income support" int,
19     e1_flag int,
20     "E2_Debt/contract relief" int,
21     "E3_Fiscal measures" real,
22     "E4_International support" double precision,
23     "H1_Public information campaigns" int,
24     h1_flag int,
25     "H2_Testing policy" int,
26     "H3_Contact tracing" int,
27     "H4_Emergency investment in healthcare" real,
28     "H5_Investment in vaccines" real,
29     "H6M_Facial Coverings" int,
30     h6m_flag int,
31     "H7_Vaccination policy" int,
32     h7_flag int,
33     "H8M_Protection of elderly people" int,
34     h8m_flag int,
35     "V1_Vaccine Prioritisation (summary)" int,
36     "V2A_Vaccine Availability (summary)" int,
37     "V2B_Vaccine age eligibility/availability age floor (general)" varchar(20),
38     "V2C_Vaccine age eligibility/availability age floor (at risk)" varchar(20),
39     "V2D_Medically/ clinically vulnerable (Non-elderly)" int,
40     v2e_education int,
41     "V2F_Frontline workers (non healthcare)" int,
42     "V2G_Frontline workers (healthcare)" int,
43
44     "V3_Vaccine Financial Support (summary)" int,
45     "V4_Mandatory Vaccination (summary)" int,
46     primary key(Data_id)
47 )

```

Create table "people"

The screenshot shows the PGAdmin 4 interface with the following details:

- Browser Panel:** Shows the database structure. The 'Schemas' folder is selected under the 'dbhw1' database.
- Query Editor:** Contains the SQL query to create the 'people' table. The query is:


```

1 create table people(
2     Data_id int,
3     ConfirmedCases int,
4     ConfirmedDeaths int,
5     populationVaccinated real,
6     primary key(Data_id),
7     foreign key(populationVaccinated) references MajorityVaccinated(populationVaccinated)
8 )
      
```
- Messages Panel:** Displays the message "Query returned successfully in 231 msec." under the 'Messages' tab.
- Status Bar:** At the bottom, it shows "Total rows: 0 of 0" and "Query complete 00:00:00.231". A green checkmark icon indicates successful execution.

Create table “majorityVaccinated”

The screenshot shows the pgAdmin interface with the 'majorityVaccinated' table being created. The left sidebar shows the database structure: Servers (2) > PostgreSQL 15 > dbhw1 > Databases (3) > oxcgrt > Schemas. The main pane shows the SQL query:

```
1 create table MajorityVaccinated(  
2     populationVaccinated real,  
3     MajorityVaccinated varchar(2),  
4     primary key(populationVaccinated)  
5 )
```

The 'Data Output' tab shows the message: 'CREATE TABLE' and 'Query returned successfully in 419 msec.' The status bar at the bottom indicates 'Total rows: 0 of 0' and 'Query complete 00:00:00.419'.

Create table “index”

The screenshot shows the pgAdmin interface with the 'index' table being created. The left sidebar shows the database structure: Servers (2) > PostgreSQL 15 > dbhw1 > Databases (3) > oxcgrt > Schemas. The main pane shows the SQL query:

```
1 create table "index"(  
2     Data_id int,  
3     StringencyIndex_Average real,  
4     GovernmentResponseIndex_Average real,  
5     ContainmentHealthIndex_Average real,  
6     EconomicSupportIndex real,  
7     primary key(Data_id),  
8     foreign key(StringencyIndex_Average) references Stringency(StringencyIndex_Average),  
9     foreign key(GovernmentResponseIndex_Average) references GovernmentResponse(GovernmentResponseIndex_Average),  
10    foreign key(ContainmentHealthIndex_Average) references ContainmentHealth(ContainmentHealthIndex_Average),  
11    foreign key(EconomicSupportIndex) references EconomicSupport(EconomicSupportIndex)  
12 )
```

The 'Data Output' tab shows the message: 'CREATE TABLE' and 'Query returned successfully in 248 msec.' The status bar at the bottom indicates 'Total rows: 0 of 0' and 'Query complete 00:00:00.248'.

Create table “stringency”

The screenshot shows the pgAdmin interface with the 'stringency' table being created. The left sidebar shows the database structure: Servers (2) > PostgreSQL 15 > dbhw1 > Databases (3) > oxcgrt > Schemas. The main pane shows the SQL query:

```
1 create table Stringency(  
2     StringencyIndex_Average real,  
3     StringencyIndex_Average_ForDisplay real,  
4     primary key(StringencyIndex_Average)  
5 )
```

The 'Data Output' tab shows the message: 'CREATE TABLE' and 'Query returned successfully in 267 msec.' The status bar at the bottom indicates 'Total rows: 0 of 0' and 'Query complete 00:00:00.267'.

Create table “containmentHealth”

The screenshot shows the pgAdmin interface with the 'Schemas' tree on the left. The 'Query' tab is active, displaying the following SQL code:

```
1 create table ContainmentHealth(  
2     ContainmentHealthIndex_Average real,  
3     ContainmentHealthIndex_Average_ForDisplay real,  
4     primary key (ContainmentHealthIndex_Average)  
5 )
```

The 'Messages' tab shows the execution result: 'CREATE TABLE' and 'Query returned successfully in 567 msec.' The status bar at the bottom indicates 'Total rows: 0 of 0' and 'Query complete 00:00:00.567'.

Create table “governmentResponse”

The screenshot shows the pgAdmin interface with the 'Schemas' tree on the left. The 'Query' tab is active, displaying the following SQL code:

```
1 create table GovernmentResponse(  
2     GovernmentResponseIndex_Average real,  
3     GovernmentResponseIndex_Average_ForDisplay real,  
4     primary key (GovernmentResponseIndex_Average)  
5 )
```

The 'Messages' tab shows the execution result: 'CREATE TABLE' and 'Query returned successfully in 354 msec.' The status bar at the bottom indicates 'Total rows: 0 of 0' and 'Query complete 00:00:00.354'.

Create table “economicSupport”

The screenshot shows the pgAdmin interface with the 'Schemas' tree on the left. The 'Query' tab is active, displaying the following SQL code:

```
1 create table EconomicSupport(  
2     EconomicSupportIndex real,  
3     EconomicSupportIndex_ForDisplay real,  
4     primary key (EconomicSupportIndex)  
5 )
```

The 'Messages' tab shows the execution result: 'CREATE TABLE' and 'Query returned successfully in 339 msec.' The status bar at the bottom indicates 'Total rows: 0 of 0' and 'Query complete 00:00:00.339'.

Import data

	PID	Type	Server	Object	Start Time	Status	Time Taken (sec)
<input type="checkbox"/>	10408	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.data	2022/12/4 下午7:40:...	Finished	6.04
<input type="checkbox"/>	19472	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.country_conti...	2022/12/4 下午7:40:...	Finished	1.73
<input type="checkbox"/>	21152	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.continent	2022/12/4 下午7:40:...	Finished	1.77
<input type="checkbox"/>	10256	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.country	2022/12/4 下午7:40:...	Finished	1.74
<input type="checkbox"/>	10796	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.jurisdiction	2022/12/4 下午7:39:...	Finished	1.74
<input type="checkbox"/>	8724	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.policy	2022/12/4 下午7:39:...	Finished	4.25
<input type="checkbox"/>	5836	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.people	2022/12/4 下午7:38:...	Finished	4.9
<input type="checkbox"/>	5860	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.index	2022/12/4 下午7:38:...	Finished	8.88
<input type="checkbox"/>	10880	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.majorityvaccin...	2022/12/4 下午7:37:...	Finished	2.18
<input type="checkbox"/>	19632	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.containmentth...	2022/12/4 下午7:37:...	Finished	2.05
<input type="checkbox"/>	19172	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.stringency	2022/12/4 下午7:37:...	Finished	2.27
<input type="checkbox"/>	18624	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.governmentre...	2022/12/4 下午7:36:...	Finished	2.06
<input type="checkbox"/>	19560	Import Data	dbhw1 (database-2.cwhx9u...	oxcgrt/public.economicssup...	2022/12/4 下午7:35:...	Finished	1.78

4. Normal form

Table (country, continent, country_continent, jurisdiction, policy, data, people, index, majorityVaccinated) satisfies 1. No repeating groups 2. No partial dependency 3. No transitive dependency 4. the left-hand side of every functional dependency is a candidate key. So, they are in BCNF.

Table (stringency, governmentResponse, containmentHealth, economicSupport) satisfies 1. No repeating groups 2. No partial dependency 3. No transitive dependency. So, they are in 3NF.

5. functional dependency

country:

{country_id} -> {country_name},
 {country_id} -> {country_code},
 {country_id} -> {country_number}

continent:

{continent_id} -> {continent_name},
 {continent_id} -> {continent_code}

country_continent:

{country_id, continent_id} -> {country_id, continent_id}

jurisdiction:

{country_name} -> {jurisdiction}

policy:

{data_id} -> {C1},

{data_id} -> {C2},

{data_id} -> {C3},

{data_id} -> {C4},

{data_id} -> {C5},

{data_id} -> {C6},

{data_id} -> {C7},

{data_id} -> {C8},

{data_id} -> {C1_flag},

{data_id} -> {C2_flag},

{data_id} -> {C3_flag},

{data_id} -> {C4_flag},

{data_id} -> {C5_flag},

{data_id} -> {C6_flag},

{data_id} -> {C7_flag},

{data_id} -> {E1},

{data_id} -> {E2},

{data_id} -> {E3},

{data_id} -> {E4},

{data_id} -> {E1_flag},

{data_id} -> {H1},

{data_id} -> {H2},

{data_id} -> {H3},

{data_id} -> {H4},

{data_id} -> {H5},

{data_id} -> {H6},

{data_id} -> {H7},

{data_id} -> {H8},

{data_id} -> {H1_flag},

{data_id} -> {H6_flag},

{data_id} -> {H7_flag},

{data_id} -> {H8_flag},
{data_id} -> {V1},
{data_id} -> {V2A},
{data_id} -> {V2B},
{data_id} -> {V2C},
{data_id} -> {V2D},
{data_id} -> {V2E},
{data_id} -> {V2F},
{data_id} -> {V2G},
{data_id} -> {V3},
{data_id} -> {V4}

data:

{data_id} -> {date},
{data_id} -> {country_id}

people:

{data_id} -> {confirmedCases},
{data_id} -> {confirmedDeaths},
{data_id} -> {populationVaccinated}

index:

{data_id} -> {stringencyIndex_average},
{data_id} -> {governmentResponseIndex_average},
{data_id} -> {containmentHealthIndex_average},
{data_id} -> {economicSupportIndex}

majorityVaccinated:

{populationVaccinated} -> {majorityVaccinated}

stringency:

{stringencyIndex_average} -> {stringencyIndex_average_ForDisplay},
{stringencyIndex_average_ForDisplay} -> {stringencyIndex_average}

governmentResponse:

```
{governmentResponseIndex_average} ->  
{governmentResponseIndex_average_ForDisplay},  
{governmentResponseIndex_average_ForDisplay} ->  
{governmentResponseIndex_average}
```

containmentHealth:

```
{containmentHealthIndex_average} ->  
{containmentHealthIndex_average_ForDisplay},  
{containmentHealthIndex_average_ForDisplay} ->  
{containmentHealthIndex_average}
```

economicSupport:

```
{economicSupportIndex} -> {economicSupportIndex_ForDisplay},  
{economicSupportIndex_ForDisplay} -> {economicSupportIndex}
```

6. output results of 4a

	country_name character varying (80)	continent_name character varying (50)	Date date
1	Argentina, Argentine ...	South America	2020-06-01
2	Australia, Commonwe...	Oceania	2021-06-01
3	Bahamas, Commonw...	North America	2022-06-01
4	China, People's Repub...	Asia	2022-06-01
5	Cuba, Republic of	North America	2020-06-01
6	Fiji, Republic of the Fij...	Oceania	2020-06-01
7	Honduras, Republic of	North America	2020-06-01
8	Ireland	Europe	2020-06-01
9	Iraq, Republic of	Asia	2020-06-01
10	Italy, Italian Republic	Europe	2021-06-01
11	Libyan Arab Jamahiriya	Africa	2020-06-01
12	Malta, Republic of	Europe	2020-06-01
13	Mauritius, Republic of	Africa	2021-06-01
14	Nepal, State of	Asia	2020-06-01
15	Nepal, State of	Asia	2021-06-01
16	Peru, Republic of	South America	2022-06-01
17	El Salvador, Republic of	North America	2020-06-01
18	Trinidad and Tobago, ...	North America	2021-06-01
19	Ukraine	Europe	2022-06-01
20	Venezuela, Bolivarian ...	South America	2021-06-01
Total rows: 22 of 22		Query complete 00:00:09.499	

	country_name character varying (80) 🔒	continent_name character varying (50) 🔒	Date date 🔒
3	Bahamas, Commonw...	North America	2022-06-01
4	China, People's Repub...	Asia	2022-06-01
5	Cuba, Republic of	North America	2020-06-01
6	Fiji, Republic of the Fij...	Oceania	2020-06-01
7	Honduras, Republic of	North America	2020-06-01
8	Ireland	Europe	2020-06-01
9	Iraq, Republic of	Asia	2020-06-01
10	Italy, Italian Republic	Europe	2021-06-01
11	Libyan Arab Jamahiriya	Africa	2020-06-01
12	Malta, Republic of	Europe	2020-06-01
13	Mauritius, Republic of	Africa	2021-06-01
14	Nepal, State of	Asia	2020-06-01
15	Nepal, State of	Asia	2021-06-01
16	Peru, Republic of	South America	2022-06-01
17	El Salvador, Republic of	North America	2020-06-01
18	Trinidad and Tobago, ...	North America	2021-06-01
19	Ukraine	Europe	2022-06-01
20	Venezuela, Bolivarian ...	South America	2021-06-01
21	Vanuatu, Republic of	Oceania	2022-06-01
22	Zimbabwe, Republic of	Africa	2022-06-01
Total rows: 22 of 22		Query complete 00:00:09.499	

7. output results of 4b

	country_name character varying (80)	continent_name character varying (50)	over_stringency_index double precision	Date date
1	Belarus, Republic of	Europe	194.46000480651855	2022-06-01
2	Bermuda	North America	80.09750080108643	2021-06-01
3	Barbados	North America	1101.7999572753906	2020-06-01
4	Brunei Darussalam	Asia	738.9199829101562	2020-06-01
5	Congo, Republic of the	Africa	713.0200042724609	2021-06-01
6	Dominica, Commonwealth of	North America	453.73999786376953	2022-06-01
7	Fiji, Republic of the Fiji Islan...	Oceania	985.1800384521484	2020-06-01
8	Gambia, Republic of the	Africa	1088.9199829101562	2020-06-01
9	Guyana, Co-operative Repub...	South America	38.08000040054321	2020-06-01
10	Guyana, Co-operative Repub...	South America	0.47932877177128885	2021-06-01
11	Kyrgyz Republic	Asia	505.5400085449219	2022-06-01
12	Kiribati, Republic of	Oceania	557.3400192260742	2022-06-01
13	Liechtenstein, Principality of	Europe	790.7199935913086	2020-06-01
14	Sierra Leone, Republic of	Africa	590.9399871826172	2022-06-01
15	San Marino, Republic of	Europe	330.5400085449219	2021-06-01
16	Suriname, Republic of	South America	1.1381678981635406	2022-06-01
17	Tajikistan, Republic of	Asia	453.73999786376953	2021-06-01
18	Tonga, Kingdom of	Oceania	661.0800170898438	2021-06-01

Total rows: 18 of 18

Query complete 00:00:38.502

8. Database auto-update (from the data provider's GitHub repo)

strategy and implementation

我不會 QQ