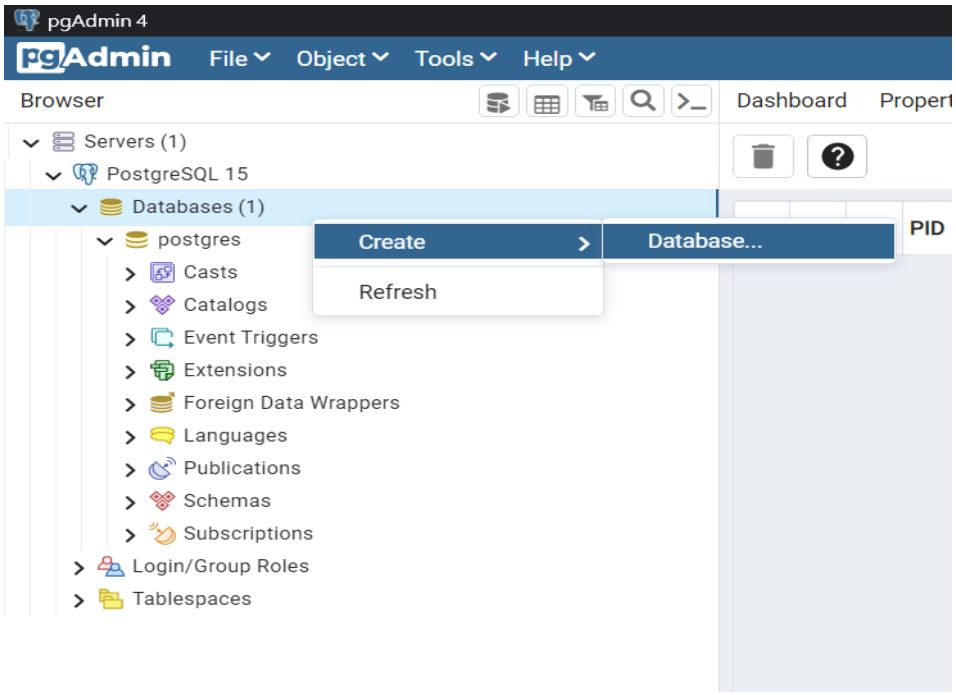
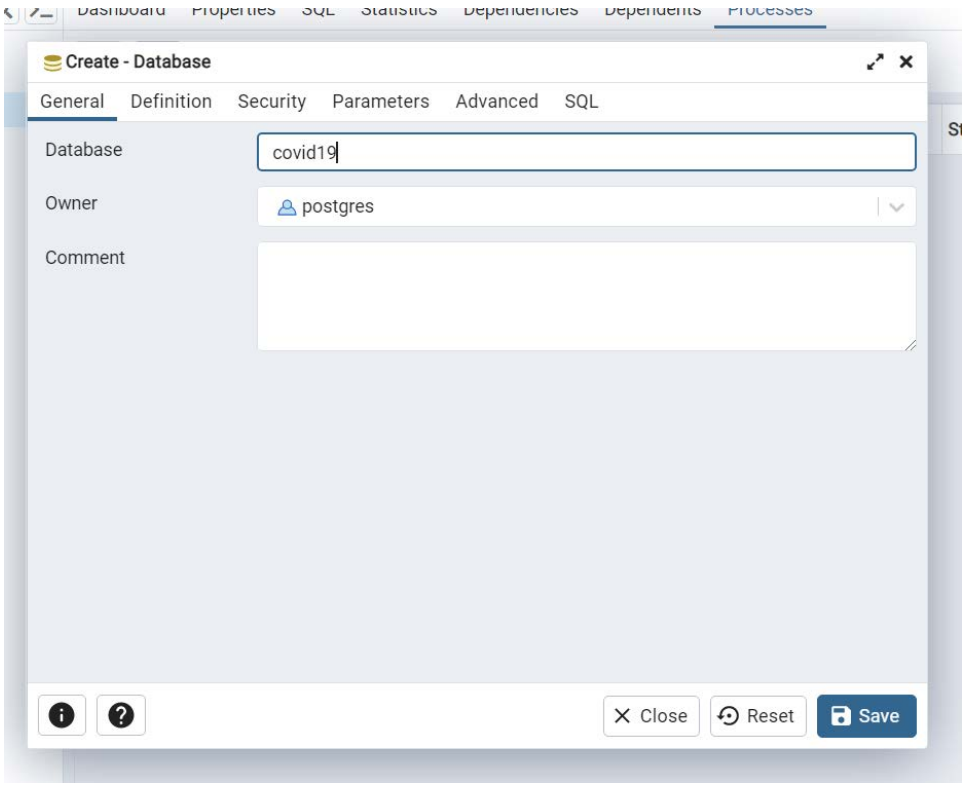
Database HW1

蔡師睿 110550093

1. The process of creating the “covid19” databases.

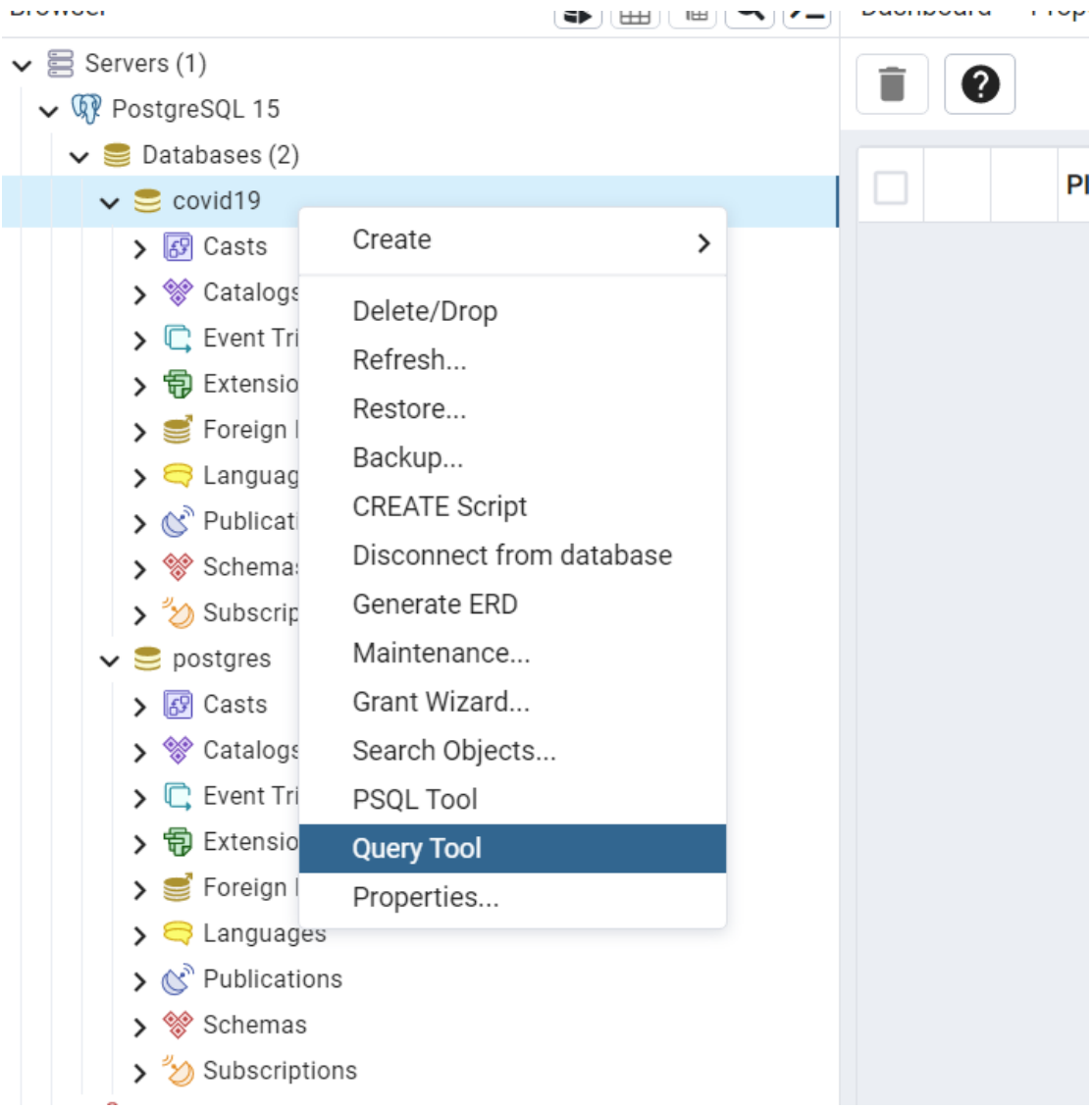
右鍵點擊 Database，移到 create 按下 Database…，輸入名稱並按下 Save。





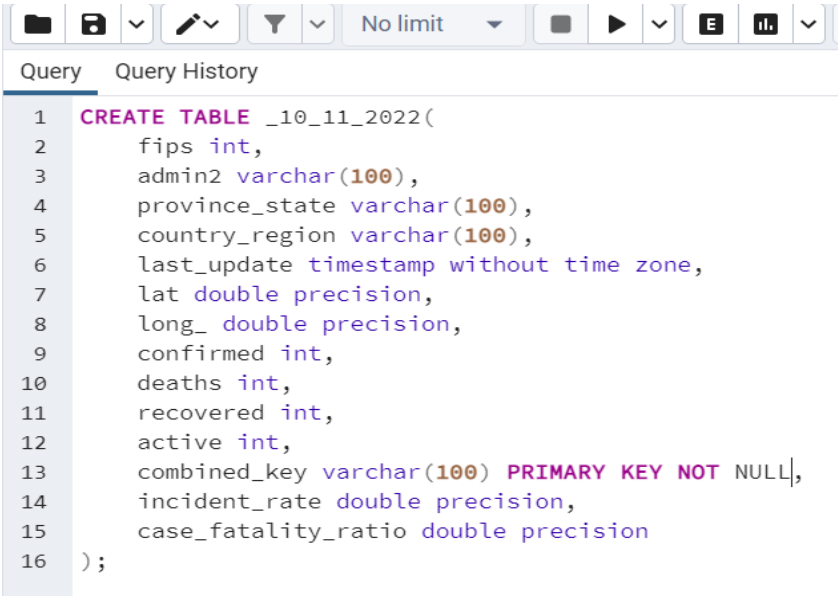
2. The process of importing three required .csv files into covid19 database.

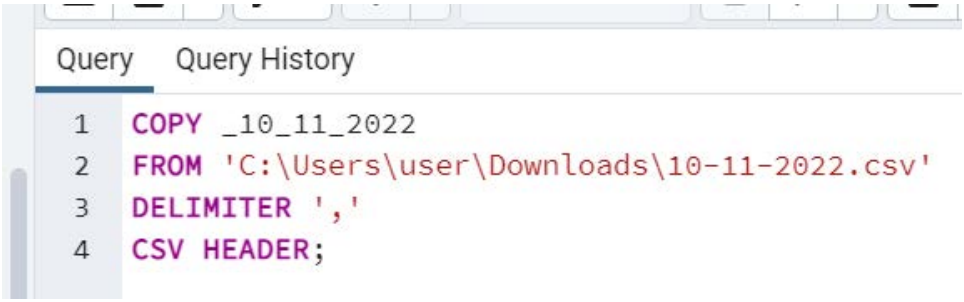
滑鼠右鍵點擊 covid19 並打開 Query Tool，之後輸入指令開始創建 table 和引入.csv 檔案。



a. Create table _10_11_2022 and import 10-11-2022.csv:

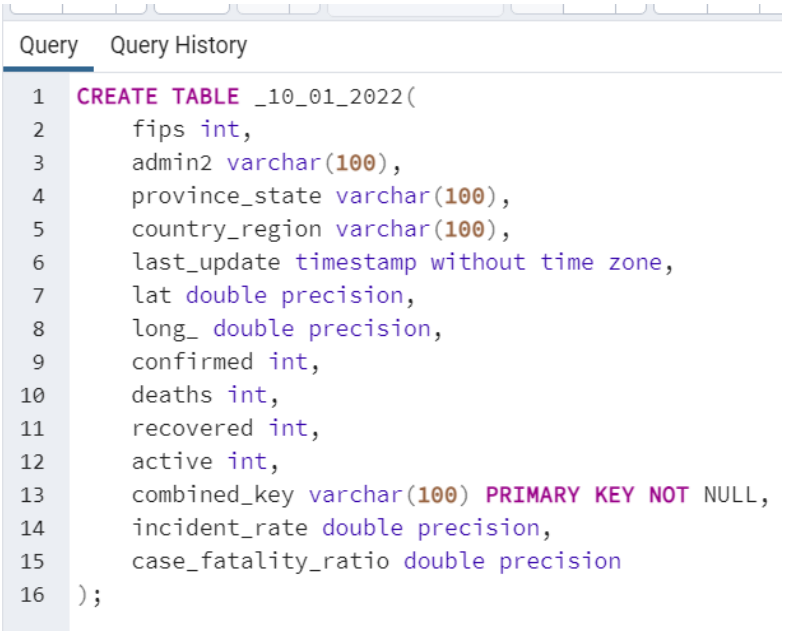
從 10-11-2022.csv 檔中發現每一筆資料的 combined_key 都不相同，且皆不為 NULL，因此我們將 combined_key 設為此 table 的 primary key。





b. Create table _10_01_2022 and import 10-01-2022.csv:

從 10-01-2022.csv 檔中發現每一筆資料的 combined_key 都不相同，且皆不為 NULL，因此我們將 combined_key 設為此 table 的 primary key。



Query Query History

```
COPY _10_01_2022
FROM 'C:\Users\user\Downloads\10-01-2022.csv'
DELIMITER ','
CSV HEADER;
```

c. Create table country_continent_code and import country-and-continent-codes-list-csv.csv:

因為 country-and-continent-codes-list-csv.csv 檔的每一筆資料中沒有任一個 column 皆為不同的值，因此在 create table 時加了一行_id 為 serial 當作 primary key。

Query Query History

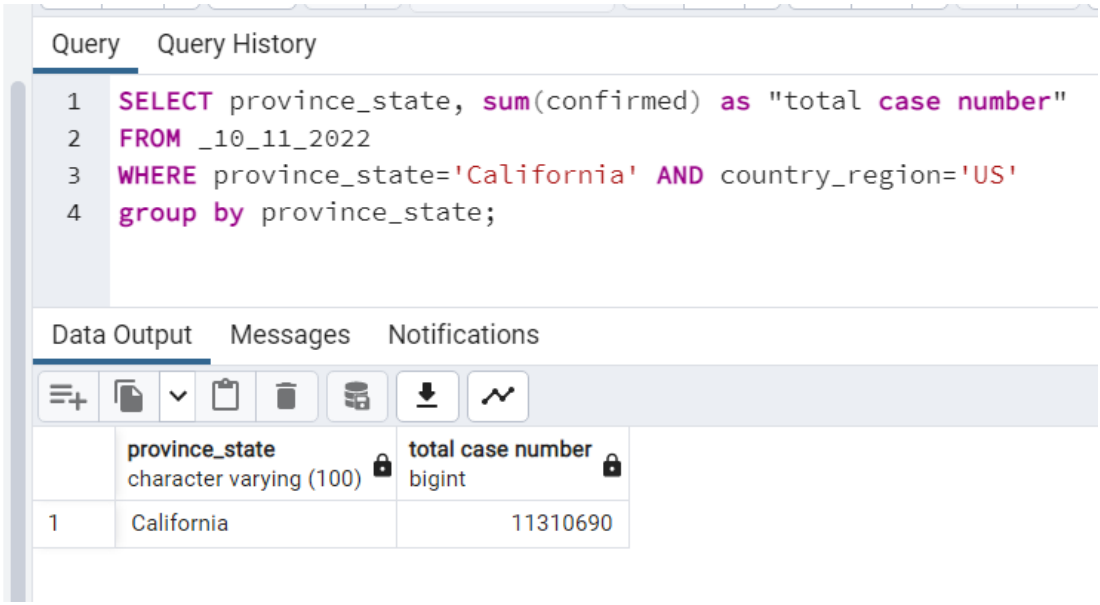
```
CREATE TABLE country_continent_code(
    _id serial PRIMARY KEY NOT NULL,
    continent_name varchar(100),
    continent_code varchar(10),
    country_name varchar (100),
    two_letter_country_code varchar(10),
    three_letter_country_code varchar(10),
    country_number int
);
```

Import csv 檔時，因為 table 的 column 跟.csv 檔的不完全相同，因此需指定要 copy 的 column。

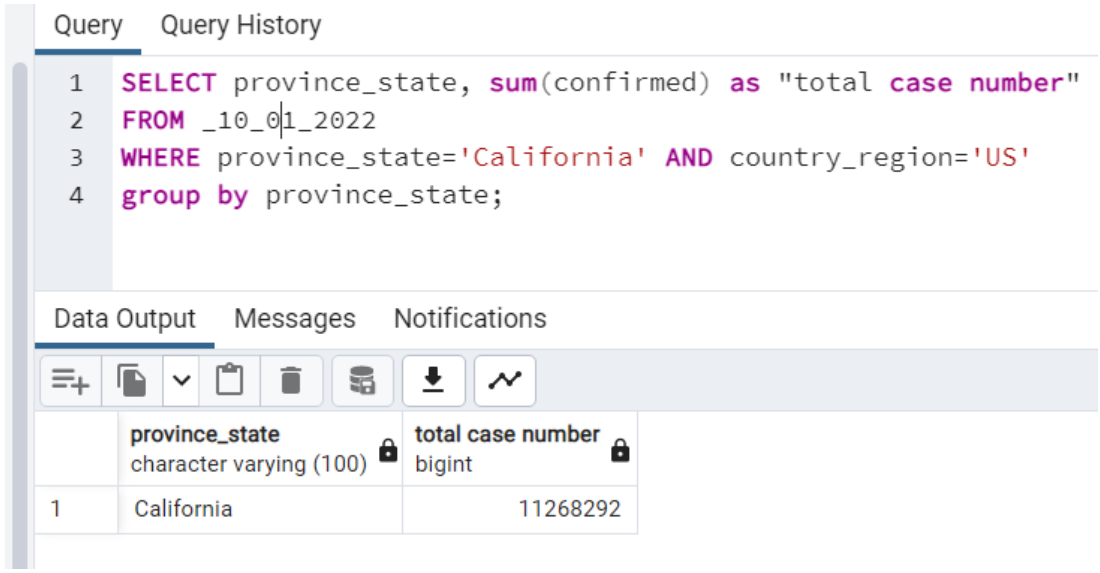
Query Query History

```
COPY country_continent_code(
    continent_name,
    continent_code,
    country_name,
    two_letter_country_code,
    three_letter_country_code,
    country_number)
FROM 'C:\Users\user\Downloads\country-and-continent-codes-list-csv.csv'
DELIMITER ','
CSV HEADER;
```

3. **The SQL statements and output results of extracting the total case number in California, US on 2022-10-11**



4. **The SQL statements and output results of extracting the total case number in California, US on 2022-10-01**



5. **The SQL statements and output results of extracting the newly diagnosed case number in California, US on 2022-10-11, compared with 2022-10-01, and returning the difference between them, in one SQL statement**

Query Query History

```
1  SELECT _old.province_state,
2  sum(_new.confirmed)-sum(_old.confirmed) as "new cases"
3  FROM _10_01_2022 as _old, _10_11_2022 as _new
4  WHERE _old.combined_key=_new.combined_key
5  and _old.province_state=_new.province_state
6  and _old.province_state='California'
7  GROUP BY _old.province_state;
```

Data Output Messages Notifications

	province_state character varying (100)	new cases bigint
1	California	42398

提取資料時若沒有加上讓兩組 table 的 primary key 相等的條件，得出的結果將會是錯誤的。

6. The SQL statements and output results of extracting the country names and total confirmed COVID cases with more than 20,000,000 total COVID-19 cases on 2022-10-11

Query Query History

```
1  CREATE VIEW total as
2      SELECT country_region,sum(confirmed) as "total confirmed cases"
3      FROM _10_11_2022
4      GROUP BY country_region;
5
6  SELECT *
7  FROM total
8  WHERE "total confirmed cases">20000000;
```

Data Output Messages Notifications

	country_region character varying (100)	total confirmed cases bigint
1	France	36187658
2	Korea, South	25025749
3	Italy	22896742
4	US	96783524
5	United Kingdom	23957457
6	Germany	34257916
7	India	44616235
8	Japan	21593704
9	Russia	20929929
10	Brazil	34731539

創造一個叫 total 的 view 紀錄每個國家的確診總數，再從 total 中提取我們所要的資料。

7. The SQL statements and output results of extracting the country names and total confirmed COVID cases with more than 20,000,000 total COVID-19 cases on 2022-10-11. Try to join the Country code and continents mapping table, and return only the data from countries in Asia

Query　Query History

```
SELECT continent_name, country_region, "total confirmed cases"
FROM (
    SELECT DISTINCT code.continent_name, total.country_region, total."total confirmed cases"
    FROM country_continent_code as code, total
    WHERE '%' || code.country_name || '%'
    like '%' || split_part(total.country_region, ',', 1) || '%'
)as "temp"
WHERE continent_name='Asia' and "total confirmed cases">20000000;
```

Data Output　Messages　Notifications

	continent_name character varying (100)	country_region character varying (100)	total confirmed cases bigint
1	Asia	India	44616235
2	Asia	Japan	21593704
3	Asia	Korea, South	25025749
4	Asia	Russia	20929929

從前一次建立好的 view total 中和 table country_continent_code 找到國家相對應的洲，並選取我們所要的資料。

8. The SQL statements and output results of extract the country names and newly diagnosed case number of countries with a newly diagnosed case number > 100,000. In descending order of newly diagnosed case numbers

從 table _10_01_2022 中找到各個國家的確診數，再從 view total 中找到相同的國家相減，得到從 10/1 到 10/11 新增的確診數，最後加上條件讓 output 的 data 確保新增確診數大於 100000 人，並以降冪形式排列。

Query | Query History

```sql
SELECT *
FROM (
    SELECT total_11.country_region,
    total_11."total confirmed cases"-total_01."total confirmed cases" as newly_diagnosed
    FROM total as total_11, (
        SELECT country_region, sum(confirmed) as "total confirmed cases"
        FROM _10_01_2022
        GROUP BY country_region
    )as total_01
    WHERE total_11.country_region=total_01.country_region
)as compare
WHERE newly_diagnosed>100000
ORDER BY newly_diagnosed DESC
```

Data Output | Messages | Notifications

	country_region character varying (100)	newly_diagnosed bigint
1	Germany	871687
2	France	579373
3	Taiwan*	440596
4	Italy	396396
5	US	386493
6	Japan	264185
7	Russia	212106
8	Korea, South	206138
9	Austria	129544
10	Greece	106302