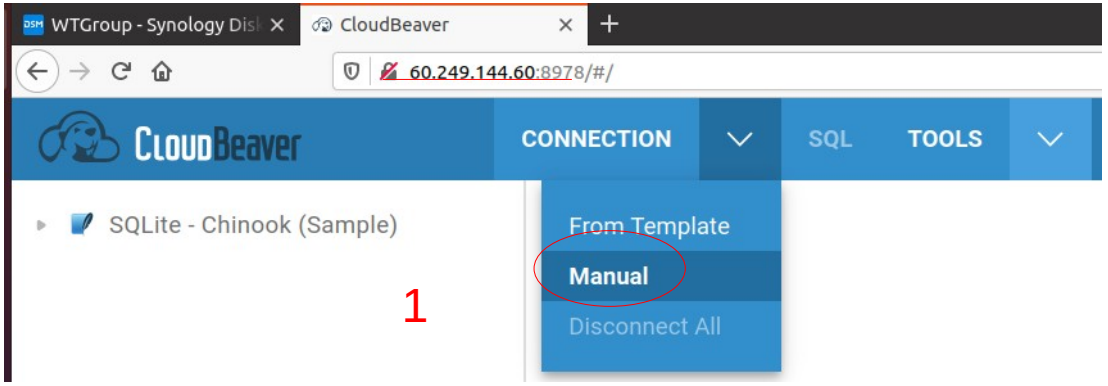
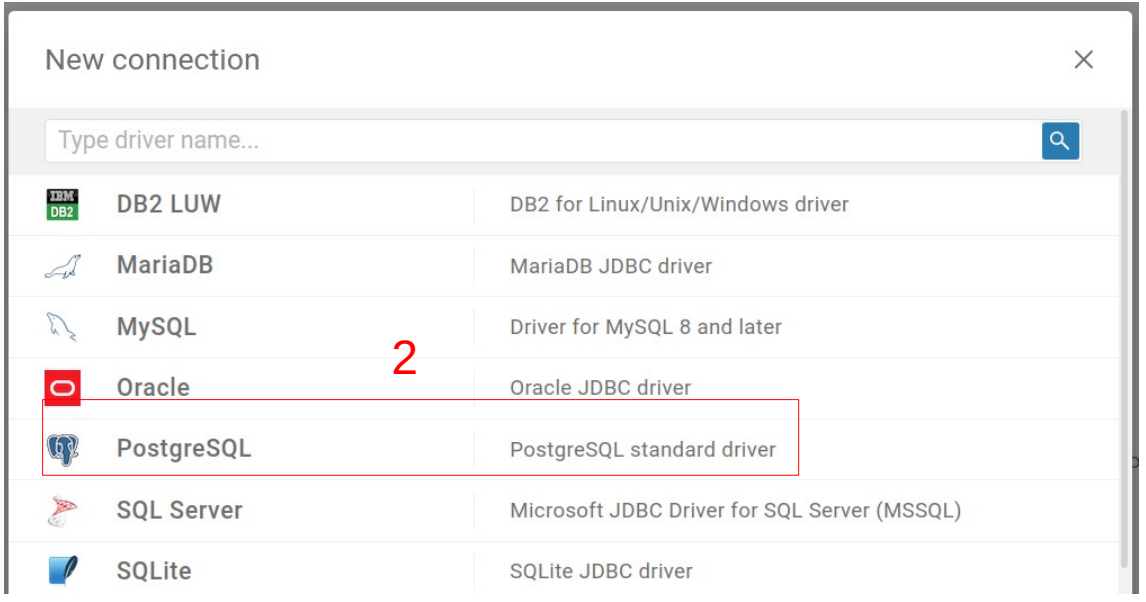
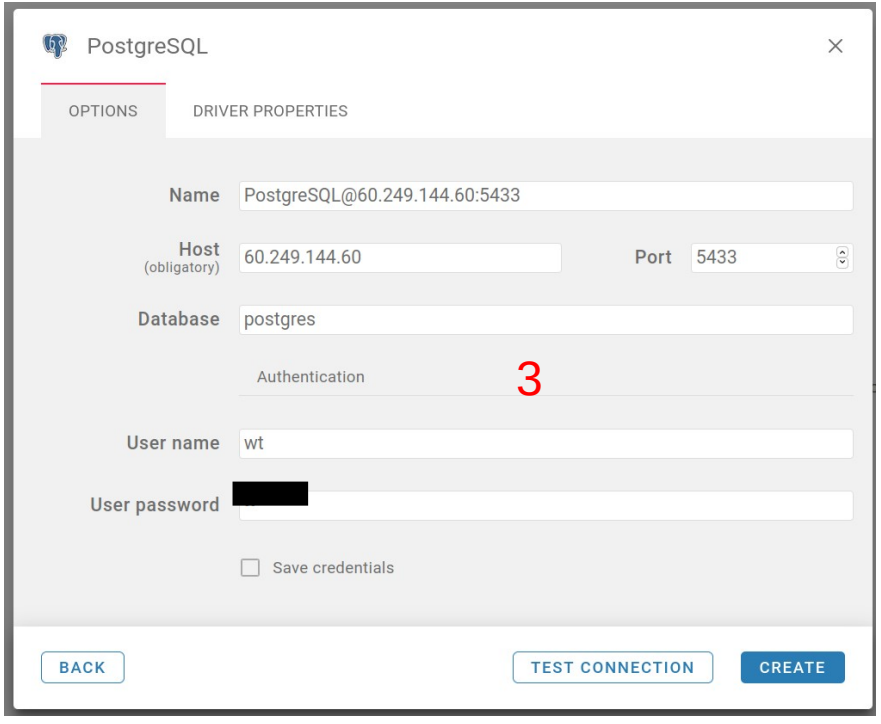
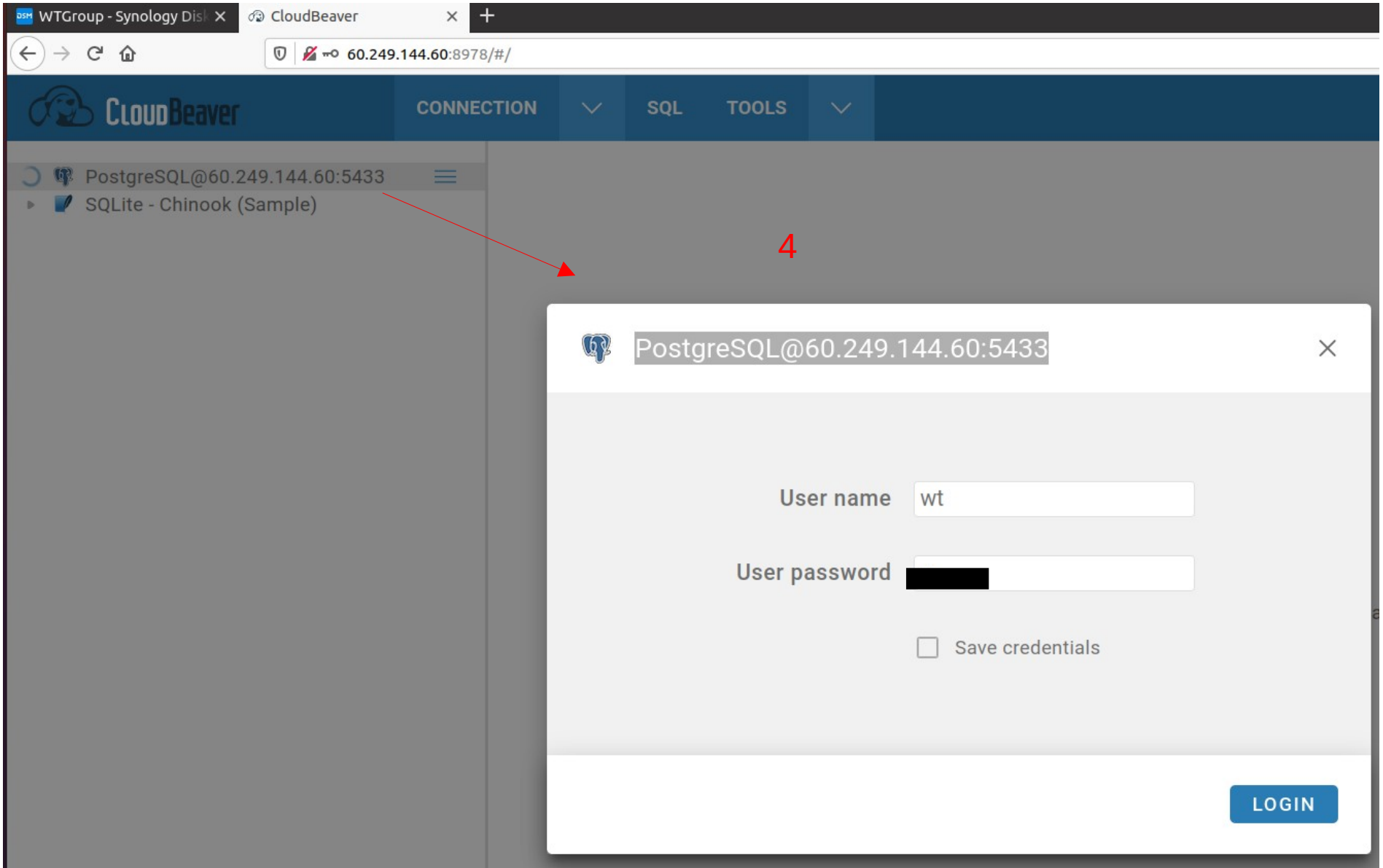
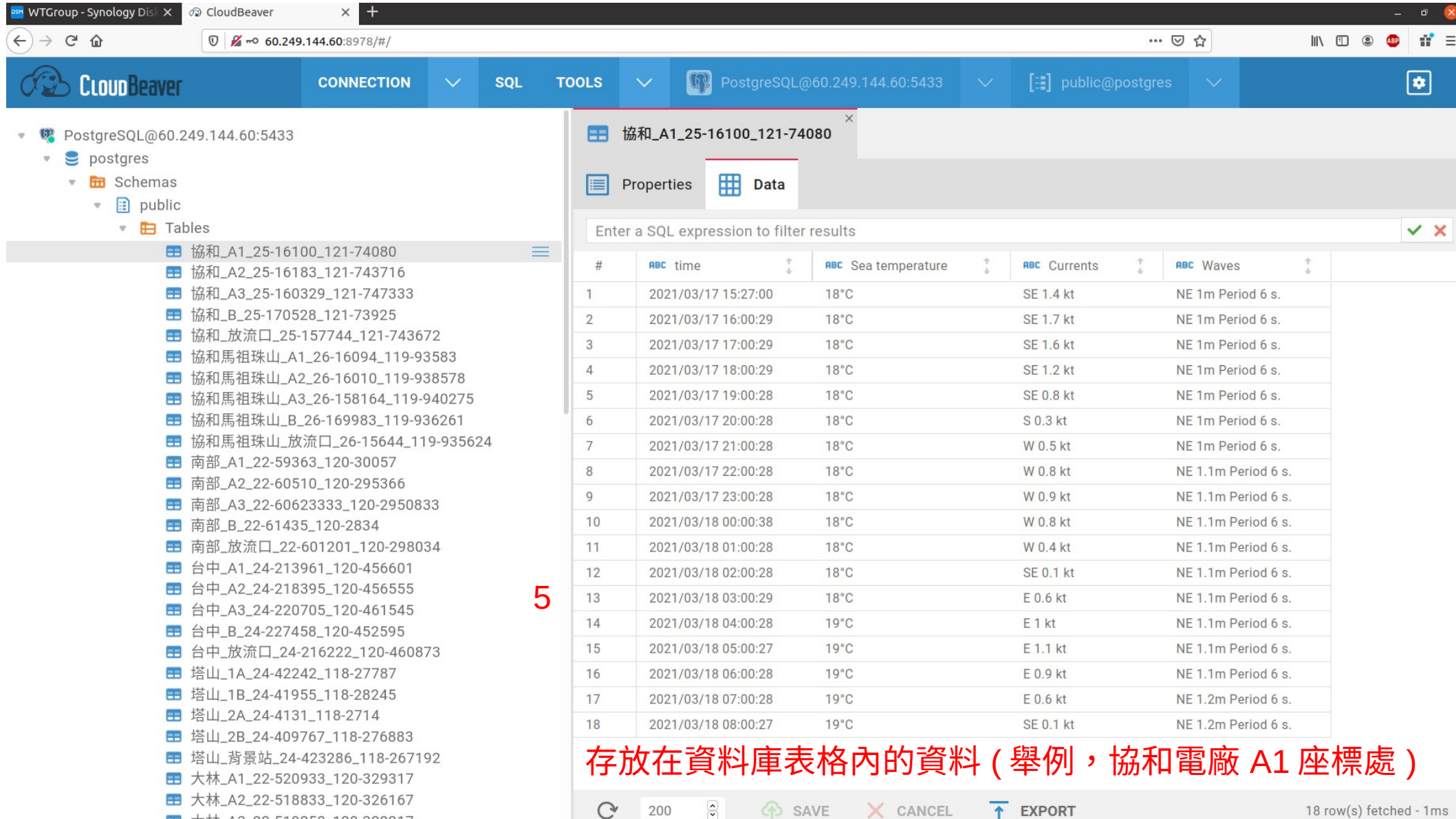
使用 CloudBeaver 資料庫管理程式 (DBeaver 雲端版)








WTGroup - Synology Dis
CloudBeaver
60.249.144.60:8978/#/
CloudBeaver
CONNECTION
SQL
TOOLS
PostgreSQL@60.249.144.60:5433
SQLite - Chinook (Sample)
4
PostgreSQL@60.249.144.60:5433
User name
wt
User password
Save credentials
LOGIN


WTGroup - Synology Dis
CloudBeaver
60.249.144.60:8978/#/
CloudBeaver
CONNECTION
SQL
TOOLS
PostgreSQL@60.249.144.60:5433
public@postgres
PostgreSQL@60.249.144.60:5433
postgres
Schemas
public
Tables
協和_A1_25-16100_121-74080
協和_A2_25-16183_121-743716
協和_A3_25-160329_121-747333
協和_B_25-170528_121-73925
協和_放流口_25-157744_121-743672
協和馬祖珠山_A1_26-16094_119-93583
協和馬祖珠山_A2_26-16010_119-938578
協和馬祖珠山_A3_26-158164_119-940275
協和馬祖珠山_B_26-169983_119-936261
協和馬祖珠山_放流口_26-15644_119-935624
南部_A1_22-59363_120-30057
南部_A2_22-60510_120-295366
南部_A3_22-60623333_120-2950833
南部_B_22-61435_120-2834
南部_放流口_22-601201_120-298034
台中_A1_24-213961_120-456601
台中_A2_24-218395_120-456555
台中_A3_24-220705_120-461545
台中_B_24-227458_120-452595
台中_放流口_24-216222_120-460873
塔山_1A_24-42242_118-27787
塔山_1B_24-41955_118-28245
塔山_2A_24-4131_118-2714
塔山_2B_24-409767_118-276883
塔山_背景站_24-423286_118-267192
大林_A1_22-520933_120-329317
大林_A2_22-518833_120-326167
協和_A1_25-16100_121-74080
Properties
Data
Enter a SQL expression to filter results
ABC time ABC Sea temperature ABC Currents ABC Waves
1 2021/03/17 15:27:00 18°C SE 1.4 kt NE 1m Period 6 s.
2 2021/03/17 16:00:29 18°C SE 1.7 kt NE 1m Period 6 s.
3 2021/03/17 17:00:29 18°C SE 1.6 kt NE 1m Period 6 s.
4 2021/03/17 18:00:29 18°C SE 1.2 kt NE 1m Period 6 s.
5 2021/03/17 19:00:28 18°C SE 0.8 kt NE 1m Period 6 s.
6 2021/03/17 20:00:28 18°C S 0.3 kt NE 1m Period 6 s.
7 2021/03/17 21:00:28 18°C W 0.5 kt NE 1m Period 6 s.
8 2021/03/17 22:00:28 18°C W 0.8 kt NE 1.1m Period 6 s.
9 2021/03/17 23:00:28 18°C W 0.9 kt NE 1.1m Period 6 s.
10 2021/03/18 00:00:38 18°C W 0.8 kt NE 1.1m Period 6 s.
11 2021/03/18 01:00:28 18°C W 0.4 kt NE 1.1m Period 6 s.
12 2021/03/18 02:00:28 18°C SE 0.1 kt NE 1.1m Period 6 s.
13 2021/03/18 03:00:29 18°C E 0.6 kt NE 1.1m Period 6 s.
14 2021/03/18 04:00:28 19°C E 1 kt NE 1.1m Period 6 s.
15 2021/03/18 05:00:27 19°C E 1.1 kt NE 1.1m Period 6 s.
16 2021/03/18 06:00:28 19°C E 0.9 kt NE 1.1m Period 6 s.
17 2021/03/18 07:00:28 19°C E 0.6 kt NE 1.2m Period 6 s.
18 2021/03/18 08:00:27 19°C SE 0.1 kt NE 1.2m Period 6 s.
存放在資料庫表格內的資料（舉例，協和電廠 A1 座標處）
200
SAVE
CANCEL
EXPORT
18 row(s) fetched - 1ms

新增帳號
(只讀)

\set 是 postgres 終端界面下設定環境變數的指令

移除帳號

```
root@4f2dd13b097d:/# psql -U postgres
psql (10.16 (Debian 10.16-1.pgdg90+1))
Type "help" for help.

postgres=# \set DB_USER wt
postgres=# \set DB_NAME postgres
postgres=#
postgres=# CREATE ROLE :DB_USER WITH LOGIN PASSWORD '[redacted]
postgres-# NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION VALID UNTIL 'infinity';
CREATE ROLE
postgres=# GRANT CONNECT ON DATABASE :DB_NAME TO :DB_USER;
GRANT
postgres=# GRANT USAGE ON SCHEMA public TO :DB_USER;
GRANT
postgres=# GRANT SELECT ON ALL TABLES IN SCHEMA public TO :DB_USER;
GRANT
postgres=# GRANT SELECT ON ALL SEQUENCES IN SCHEMA public TO :DB_USER;
GRANT
postgres=# \du
                                   List of roles
 Role name |                         Attributes                         | Member of
-----------+------------------------------------------------------------+-----------
 postgres  | Superuser, Create role, Create DB, Replication, Bypass RLS | {}
 wt        | Password valid until infinity                              | {}

postgres=# REASSIGN OWNED BY :DB_USER TO postgres;
REASSIGN OWNED
postgres=# DROP OWNED BY :DB_USER;
DROP OWNED
postgres=# DROP USER :DB_USER;
DROP ROLE
postgres=# \du
                                   List of roles
 Role name |                         Attributes                         | Member of
-----------+------------------------------------------------------------+-----------
 postgres  | Superuser, Create role, Create DB, Replication, Bypass RLS | {}

postgres=#
```

新增的存取資料庫 python 程式碼

```python
import glob

#---- db ----#
from sqlalchemy import create_engine
from sqlalchemy import Table, Column, String, MetaData

db_string = "postgres://postgres:[redacted]0.249.144.60:5433/postgres"
db = create_engine(db_string)

def postgres_test():
    try:
        with db.connect() as conn:
            pass
        return True
    except:
        return False

database_on = postgres_test()
#print('test connect:', database_on)
meta = MetaData(db)

#---- 初始化 ----#
```

初始化

```python
    #with open(folders + 'test.csv', 'a', newline='', encoding='utf-8') as csvfile:
    with open(folders + filename_csv, 'a', newline='', encoding='utf-8') as f_a:
        #fieldnames = ['time', '海洋溫度', '洋流', '波浪']
        fieldnames = ['time', 'Sea temperature', 'Currents', 'Waves'] # docker
        writer = csv.DictWriter(f_a, fieldnames=fieldnames)
        #writer.writeheader() # 第一次寫入檔案時使用
        writer.writerow(record)

#---- db ----#
if database_on:
    try:
        with db.connect() as conn:
            filename_table= gps_name+'_'+gps_id+'_'+latitude+'_'+longitude

            windy_table = Table(filename_table, meta,
                                Column(fieldnames[0], String),
                                Column(fieldnames[1], String),
                                Column(fieldnames[2], String),
                                Column(fieldnames[3], String))

            windy_table.create(checkfirst=True)
            insert_statement = windy_table.insert().values(**record)
            conn.execute(insert_statement)
    except Exception as e:
        print('database error:',e)

#---- 主要程式 ----#
```

對表格插入一筆資料