

## PostgreSQL practice

這次作業要利用 postgresQL 與 pgadmin 練習 5，在練習時使用 docker 開啟 postgresQL 與 pgadmin

### PostgreSQL docker

```
docker run --name {name}-e POSTGRES_PASSWORD={password}-p 5432:5432 -d postgres
```

### pgadmin docker

```
docker run --name pgadmin-container -p 5050:80 -e PGADMIN_DEFAULT_EMAIL={account} -e PGADMIN_DEFAULT_PASSWORD={password} -d dpage/pgadmin4
```

並且使用 python 進行撰寫查詢與 command line 的操作

連線到 postgresQL

```
connect = psycopg2.connect(  
    host= postgresQL IP address,  
    port=postgresQL port,  
    database=name of database,  
    user=name,  
    password=password  
)
```

分別做了 5-1~5-4 以及 5-8，每一題都做了 view 方便後面直接呼叫

以下是各題的 SQL 查詢

### 5-1 a

```
sql.execute("""  
    SELECT employee_name, company_name  
    FROM works  
    WHERE company_name != 'First-Bank'  
    """)
```

5-1 b

```
sql.execute("""
    CREATE OR REPLACE VIEW question_5_1_b AS
    SELECT w1.employee_name, company_name, salary
    FROM works w1
    WHERE w1.salary > (
        SELECT AVG(w2.salary) as avg
        FROM works w2
        WHERE w1.company_name = w2.company_name
    )
    """)
```

5-2 a

```
sql.execute("""
    CREATE OR REPLACE VIEW question_5_2_a AS
    SELECT w1.employee_name, w1.salary
    FROM works w1
    WHERE w1.salary > ALL (
        SELECT w2.salary
        FROM works w2
        WHERE w2.company_name = 'Land-Bank'
    )
    """)
```

5-2 b

```
sql.execute("""
    CREATE OR REPLACE VIEW question_5_2_b AS
    SELECT company_name, COUNT(employee_name) as employee_count
    FROM works
    GROUP BY company_name
    ORDER BY employee_count DESC
    LIMIT 1
    """)
```

5-3

```
sql.execute("""
    CREATE OR REPLACE VIEW question_5_3 AS
    SELECT DISTINCT company_name
    FROM company AS c1
    WHERE NOT EXISTS (
        SELECT city
        FROM company
        WHERE company_name = 'Land-Bank'
        EXCEPT
        SELECT city
        FROM company AS c2
        WHERE c1.company_name = c2.company_name
    )
    """)
```

5-4 a

```
sql.execute("""
    CREATE OR REPLACE VIEW question_5_4_a AS
    SELECT company_name, SUM(salary) as total_payroll
    FROM works
    GROUP BY company_name
    ORDER BY total_payroll ASC
    LIMIT 1
    """)
```

5-4 b

```
sql.execute("""
    CREATE OR REPLACE VIEW question_5_4_b AS
    SELECT company_name, AVG(salary) as avg_salary
    FROM works
    GROUP BY company_name
    HAVING AVG(salary) > (
        SELECT AVG(salary)
        FROM works
        WHERE company_name = 'First-Bank'
    )
    """)
```

5-8

```
sql.execute("""
    CREATE OR REPLACE VIEW question_5_8 AS
    SELECT m.manager_name, AVG(w.salary) as avg_salary
    FROM manages m
    JOIN works w ON m.employee_name = w.employee_name
    GROUP BY m.manager_name;
    """)
```

在 command line 中使用者決定要哪一題，以及哪一小題

```
=====postgresql practice=====
0 : EXIT
=====
1 : 5-1
2 : 5-2
3 : 5-3
4 : 5-4
5 : 5-8
=====
6 : Help
Enter a command : █
```

ex.

```
=====postgresql practice=====
0 : EXIT
=====
1 : 5-1
2 : 5-2
3 : 5-3
4 : 5-4
5 : 5-8
=====
6 : Help
Enter a command : 1
Which part do you want to do? (a or b) : a

People who don't work in First bank :
Jane Smith is work in Land-Bank
Emily Johnson is work in Third-Bank
Sarah Davis is work in Land-Bank
Laura White is work in Land-Bank
```