

## 第二次作業-PostgreSQL

姓名：林雨臻

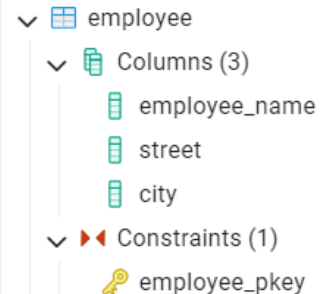
學號：11127138

班級：資訊二甲

### 1. 建立四個工作表實例：

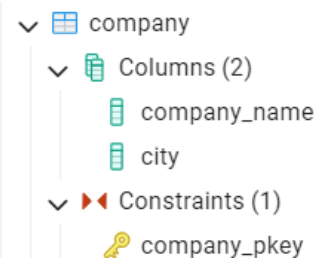
#### I. employee(employee\_name, street, city)

```
1 CREATE TABLE employee (  
2     employee_name varchar(255) PRIMARY KEY,  
3     -- employee_name varchar(255) UNIQUE NOT NULL, -- 與上行同義  
4     street varchar(255),  
5     city varchar(255)  
6 );
```



#### II. company(company\_name, city)

```
8 CREATE TABLE company (  
9     company_name varchar(255) PRIMARY KEY,  
10    city varchar(255)  
11 );
```



#### III. works(employee\_name, company\_name, salary)

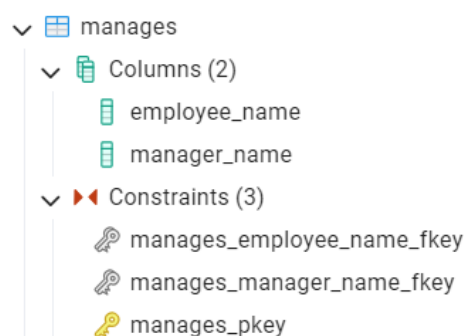
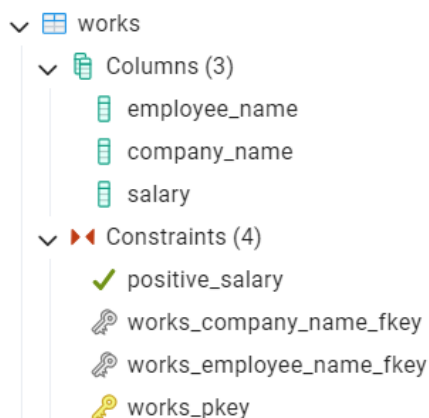
```
12 CREATE TABLE works (  
13     employee_name varchar(255) references employee(employee_name),  
14     company_name varchar(255) references company(company_name),  
15     -- salary integer DEFAULT 0 CHECK (salary >= 0),  
16     -- CONSTRAINT positive_salary 命名錯誤，使錯誤發生時，知道是哪個錯誤發生了  
17     salary integer DEFAULT 0 CONSTRAINT positive_salary CHECK (salary >= 0),  
18  
19     PRIMARY KEY(employee_name, company_name)  
20 );
```

#### IV. manages(employee\_name, manager\_name)

```
22 CREATE TABLE manages (  
23     employee_name varchar(255) PRIMARY KEY references employee(employee_name),  
24     manager_name varchar(255) references employee(employee_name)  
25 );
```

存在四個 table，且其 columns 與 constraints 和指令相符。

因為 PRIMARY KEY 本身的限制，其意義等同 UNIQUE NOT NULL。



插入資料內容進入 table 中。

因為在創建 table 時，限制了 works 的外鍵接向了 employee、company，manages 的外鍵接向了 employee，因此要先將 employee 和 company 的資料 INSERT 入 table 中。

使用 INSERT INTO [TABLE NAME] ( [COLUMN1], [COLUMN2], ...[COLUMN3] ) VALUES ( ) ;  
將資料加入。

```
1 INSERT INTO employee (employee_name, street, city) VALUES
2 ('John Doe', '123 Main St', 'New York'),
3 ('Jane Smith', '456 Elm St', 'Los Angeles'),
4 ('Michael Johnson', '789 Oak St', 'Chicago'),
5 ('Emily Davis', '101 Pine St', 'Houston'),
6 ('Christopher Wilson', '222 Maple St', 'San Francisco'),
7 ('Jessica Taylor', '333 Cedar St', 'Boston'),
8 ('Daniel Anderson', '444 Walnut St', 'Seattle'),
9 ('Sarah Martinez', '555 Birch St', 'Miami'),
10 ('David Thompson', '666 Ash St', 'Denver'),
11 ('Jennifer Rodriguez', '777 Elm St', 'Atlanta'),
12 ('James Garcia', '888 Pine St', 'Dallas'),
13 ('Linda Hernandez', '999 Oak St', 'New York'),
14 ('William Martinez', '111 Maple St', 'Philadelphia'),
15 ('Patricia Gonzalez', '222 Cedar St', 'Detroit'),
16 ('Richard Perez', '123 Main St', 'New York'),
17 ('Susan Ramirez', '444 Birch St', 'Minneapolis'),
18 ('Daniel Lewis', '555 Ash St', 'Tampa'),
19 ('Maria Walker', '666 Elm St', 'Portland'),
20 ('Charles King', '777 Pine St', 'Orlando'),
21 ('Karen Wright', '888 Oak St', 'Austin');

1 INSERT INTO company (company_name, city) VALUES
2 ('First Bank', 'New York'),
3 ('Land Bank', 'Los Angeles'),
4 ('Prime Technologies', 'San Diego'),
5 ('Blue Sky Inc', 'New York'),
6 ('GreenTech', 'New York'),
7 ('Red Rock Solutions', 'Orlando'),
8 ('Dynamic Innovations', 'Austin'),
9 ('Tech Innovations', 'Los Angeles');
```

新增 works、manages 內的資料。

```
1 INSERT INTO works (employee_name, company_name, salary) VALUES
2 ('John Doe', 'First Bank', 23651),
3 ('Jane Smith', 'First Bank', 41783),
4 ('Michael Johnson', 'Land Bank', 32896),
5 ('Emily Davis', 'Prime Technologies', 45972),
6 ('Christopher Wilson', 'Blue Sky Inc', 18745),
7 ('Jessica Taylor', 'Tech Innovations', 39624),
8 ('Daniel Anderson', 'Blue Sky Inc', 26893),
9 ('Sarah Martinez', 'GreenTech', 12875),
10 ('David Thompson', 'Red Rock Solutions', 32984),
11 ('Jennifer Rodriguez', 'Red Rock Solutions', 48521),
12 ('James Garcia', 'Land Bank', 17236),
13 ('Linda Hernandez', 'Prime Technologies', 32459),
14 ('William Martinez', 'Tech Innovations', 40218),
15 ('Patricia Gonzalez', 'Tech Innovations', 36742),
16 ('Richard Perez', 'First Bank', 29385),
17 ('Susan Ramirez', 'Prime Technologies', 29624),
18 ('Daniel Lewis', 'Blue Sky Inc', 40875),
19 ('Maria Walker', 'GreenTech', 24896),
20 ('Charles King', 'Tech Innovations', 37291),
21 ('Karen Wright', 'Dynamic Innovations', 15642);
```



```
1 INSERT INTO manages (employee_name, manager_name) VALUES
2 ('John Doe', 'Jane Smith'),
3 ('Jane Smith', null),
4 ('Michael Johnson', null),
5 ('Emily Davis', null),
6 ('Christopher Wilson', 'Daniel Lewis'),
7 ('Daniel Anderson', 'Daniel Lewis'),
8 ('Sarah Martinez', 'Maria Walker'),
9 ('Jennifer Rodriguez', null),
10 ('James Garcia', 'Michael Johnson'),
11 ('Linda Hernandez', 'Emily Davis'),
12 ('William Martinez', 'Jennifer Rodriguez'),
13 ('Patricia Gonzalez', 'Charles King'),
14 ('Richard Perez', 'John Doe'),
15 ('Susan Ramirez', 'Emily Davis'),
16 ('Daniel Lewis', null),
17 ('Maria Walker', null),
18 ('Charles King', 'Jennifer Rodriguez'),
19 ('Karen Wright', null),
20 ('Jessica Taylor', 'William Martinez'),
21 ('David Thompson', 'Jennifer Rodriguez')
```

新增完資料後各 TABLE 的表格圖：

employee :

	employee_name [PK] character varying (255) 	street character varying (255) 	city character varying (255) 
1	John Doe	123 Main St	New York
2	Jane Smith	456 Elm St	Los Angeles
3	Michael Johnson	789 Oak St	Chicago
4	Emily Davis	101 Pine St	Houston
5	Christopher Wilson	222 Maple St	San Francisco
6	Jessica Taylor	333 Cedar St	Boston
7	Daniel Anderson	444 Walnut St	Seattle
8	Sarah Martinez	555 Birch St	Miami
9	David Thompson	666 Ash St	Denver
10	Jennifer Rodriguez	777 Elm St	Atlanta
11	James Garcia	888 Pine St	Dallas
12	Linda Hernandez	999 Oak St	New York
13	William Martinez	111 Maple St	Philadelphia
14	Patricia Gonzalez	222 Cedar St	Detroit
15	Richard Perez	123 Main St	New York
16	Susan Ramirez	444 Birch St	Minneapolis
17	Daniel Lewis	555 Ash St	Tampa
18	Maria Walker	666 Elm St	Portland
19	Charles King	777 Pine St	Orlando
20	Karen Wright	888 Oak St	Austin



company :

	company_name [PK] character varying (255) 	city character varying (255) 
1	First Bank	New York
2	Land Bank	Los Angeles
3	Prime Technologies	San Diego
4	Blue Sky Inc	New York
5	GreenTech	New York
6	Red Rock Solutions	Orlando
7	Dynamic Innovations	Austin
8	Tech Innovations	Los Angeles

works :

	employee_name [PK] character varying (255) 	company_name [PK] character varying (255) 	salary integer 
1	John Doe	First Bank	23651
2	Jane Smith	First Bank	41783
3	Michael Johnson	Land Bank	32896
4	Emily Davis	Prime Technologies	45972
5	Christopher Wilson	Blue Sky Inc	18745
6	Jessica Taylor	Tech Innovations	39624
7	Daniel Anderson	Blue Sky Inc	26893
8	Sarah Martinez	GreenTech	12875
9	David Thompson	Red Rock Solutions	32984
10	Jennifer Rodriguez	Red Rock Solutions	48521
11	James Garcia	Land Bank	17236
12	Linda Hernandez	Prime Technologies	32459
13	William Martinez	Tech Innovations	40218
14	Patricia Gonzalez	Tech Innovations	36742
15	Richard Perez	First Bank	29385
16	Susan Ramirez	Prime Technologies	29624
17	Daniel Lewis	Blue Sky Inc	40875
18	Maria Walker	GreenTech	24896
19	Charles King	Tech Innovations	37291
20	Karen Wright	Dynamic Innovations	15642

**manages :**

	employee_name [PK] character varying (255) 	manager_name character varying (255) 
1	John Doe	Jane Smith
2	Jane Smith	[null]
3	Michael Johnson	[null]
4	Emily Davis	[null]
5	Christopher Wilson	Daniel Lewis
6	Daniel Anderson	Daniel Lewis
7	Sarah Martinez	Maria Walker
8	Jennifer Rodriguez	[null]
9	James Garcia	Michael Johnson
10	Linda Hernandez	Emily Davis
11	William Martinez	[null]
12	Patricia Gonzalez	Charles King
13	Richard Perez	John Doe
14	Susan Ramirez	Emily Davis
15	Daniel Lewis	[null]
16	Maria Walker	[null]
17	Charles King	William Martinez
18	Karen Wright	[null]
19	Jessica Taylor	William Martinez
20	David Thompson	Jennifer Rodriguez

註：可以在 manages 的限制中檢查 employee\_name 和 manager\_name 是否為同公司。

## 2. 驗證 5-1~5-8 中的五個 SQL 查詢

### I. Exercise5-1

- A. Find all employees in the database who do not work for First Bank. (Assumes that an employee may **work for zero or more than one companies.**)

為了滿足螢光筆部分的題意，加入一筆新的資料。

使 Karen Wright 同時替 'Dynamic Innovations' 與 'First Bank' 工作。

```
1 INSERT INTO works (employee_name, company_name, salary) VALUES
2 ('Karen Wright', 'First Bank', 20642);
```

```
7 SELECT DISTINCT employee_name
8 FROM employee
9 WHERE employee_name
10 NOT IN (SELECT employee_name
11         FROM works
12         WHERE company_name='First Bank');
```


```

17 SELECT employee_name
18 FROM employee
19 WHERE employee_name
20 NOT IN (SELECT employee_name
21         FROM works
22         WHERE company_name='First Bank');


```

因為在 employee 中可能有多筆重複的值，可使用 DISTINCT 去重，但因為在建立 employee 時有設定 employee\_name 為 primary key 因此在這可省略。



查詢結果(不在 First Bank 工作的員工)：

	employee_name [PK] character varying (255) 
1	Michael Johnson
2	Emily Davis
3	Christopher Wilson
4	Jessica Taylor
5	Daniel Anderson
6	Sarah Martinez
7	David Thompson
8	Jennifer Rodriguez
9	James Garcia
10	Linda Hernandez
11	William Martinez
12	Patricia Gonzalez
13	Susan Ramirez
14	Daniel Lewis
15	Maria Walker
16	Charles King

驗證：

	employee_name [PK] character varying (255) 
1	Jane Smith
2	John Doe
3	Karen Wright
4	Richard Perez

(左圖) 在 First Bank 工作的員工

	employee_name [PK] character varying (255) 	company_name [PK] character varying (255) 
1	John Doe	First Bank
2	Jane Smith	First Bank
3	Michael Johnson	Land Bank
4	Emily Davis	Prime Technologies
5	Christopher Wilson	Blue Sky Inc
6	Jessica Taylor	Tech Innovations
7	Daniel Anderson	Blue Sky Inc
8	Sarah Martinez	GreenTech
9	David Thompson	Red Rock Solutions
10	Jennifer Rodriguez	Red Rock Solutions
11	James Garcia	Land Bank
12	Linda Hernandez	Prime Technologies
13	William Martinez	Tech Innovations
14	Patricia Gonzalez	Tech Innovations
15	Richard Perez	First Bank
16	Susan Ramirez	Prime Technologies
17	Daniel Lewis	Blue Sky Inc
18	Maria Walker	GreenTech
19	Charles King	Tech Innovations
20	Karen Wright	Dynamic Innovations
21	Karen Wright	First Bank

(右圖) works 總資料




- B. Find all employees who earn more than the average salary of all employees of their companies. (Assumes that each employee works for at most one company.)”

為滿足題意，刪除剛剛加入的新資料。

```
48 DELETE FROM works
49 WHERE employee_name = 'Karen Wright' AND company_name = 'First Bank';

55 SELECT employee_name, company_name, salary
56 FROM works as w1
57 WHERE salary > (SELECT AVG(salary)
58                 FROM works as w2
59                 WHERE w1.company_name = w2.company_name);
```

查詢結果：

	employee_name [PK] character varying (255) 	company_name [PK] character varying (255) 	salary integer 
1	Jane Smith	First Bank	41783
2	Michael Johnson	Land Bank	32896
3	Emily Davis	Prime Technologies	45972
4	Jessica Taylor	Tech Innovations	39624
5	Jennifer Rodriguez	Red Rock Solutions	48521
6	William Martinez	Tech Innovations	40218
7	Daniel Lewis	Blue Sky Inc	40875
8	Maria Walker	GreenTech	24896

驗證：

建立一個 VIEW 後將 works 和 employee\_name NATURAL JOIN，並輸出

```
47 CREATE VIEW employee_salary AS
48     (SELECT company_name, AVG(salary)
49      FROM works
50      GROUP BY company_name)

52 SELECT *
53 FROM employee_salary
54 NATURAL JOIN works;
```



	company_name character varying (255) 🔒	avg numeric 🔒	employee_name character varying (255) 🔒	salary integer 🔒
1	First Bank	31606.333333333333	John Doe	23651
2	First Bank	31606.333333333333	Jane Smith	41783
3	Land Bank	25066.000000000000	Michael Johnson	32896
4	Prime Technologies	36018.333333333333	Emily Davis	45972
5	Blue Sky Inc	28837.666666666667	Christopher Wilson	18745
6	Tech Innovations	38468.750000000000	Jessica Taylor	39624
7	Blue Sky Inc	28837.666666666667	Daniel Anderson	26893
8	GreenTech	18885.500000000000	Sarah Martinez	12875
9	Red Rock Solutions	40752.500000000000	David Thompson	32984
10	Red Rock Solutions	40752.500000000000	Jennifer Rodriguez	48521
11	Land Bank	25066.000000000000	James Garcia	17236
12	Prime Technologies	36018.333333333333	Linda Hernandez	32459
13	Tech Innovations	38468.750000000000	William Martinez	40218
14	Tech Innovations	38468.750000000000	Patricia Gonzalez	36742
15	First Bank	31606.333333333333	Richard Perez	29385
16	Prime Technologies	36018.333333333333	Susan Ramirez	29624
17	Blue Sky Inc	28837.666666666667	Daniel Lewis	40875
18	GreenTech	18885.500000000000	Maria Walker	24896
19	Tech Innovations	38468.750000000000	Charles King	37291
20	Dynamic Innovations	15642.000000000000	Karen Wright	15642

## II. Exercise5-2

A. Find all employees in the database who earn more than each employee of Land Bank.

```

12 SELECT employee_name
13 FROM works
14 WHERE salary > ALL (SELECT salary
15                      FROM works
16                      WHERE company_name='Land Bank');

```

查詢結果：

	employee_name character varying (255) 🔒
1	Jane Smith
2	Emily Davis
3	Jessica Taylor
4	David Thompson
5	Jennifer Rodriguez
6	William Martinez
7	Patricia Gonzalez
8	Daniel Lewis
9	Charles King

驗證：

所有在 Land Bank 中工作員工的薪水。

	employee_name [PK] character varying (255)	company_name [PK] character varying (255)	salary integer
1	Michael Johnson	Land Bank	32896
2	James Garcia	Land Bank	17236

查詢剛剛搜尋結果的員工的薪水(下方左圖)。

比對上圖與下方左圖這兩個表格，可知查詢結果中所得所有員工的薪水皆大於 Land Bank 中所有員工的薪水。

再比對剩下的員工(右圖)，可知剩餘員工薪水皆小於或等於 Land Bank 員工的薪水，查詢結果正確。

	employee_name character varying (255)	salary integer
1	Jane Smith	41783
2	Emily Davis	45972
3	Jessica Taylor	39624
4	David Thompson	32984
5	Jennifer Rodriguez	48521
6	William Martinez	40218
7	Patricia Gonzalez	36742
8	Daniel Lewis	40875
9	Charles King	37291

	employee_name character varying (255)	salary integer
1	John Doe	23651
2	Michael Johnson	32896
3	Christopher Wilson	18745
4	Daniel Anderson	26893
5	Sarah Martinez	12875
6	James Garcia	17236
7	Linda Hernandez	32459
8	Richard Perez	29385
9	Susan Ramirez	29624
10	Maria Walker	24896
11	Karen Wright	15642

(左圖) 搜尋結果的員工的薪水

(右圖)剩餘的員工的薪水

B. Find the company that has the most employees.

```
33 SELECT company_name
34 FROM works
35 GROUP BY company_name
36 HAVING COUNT (DISTINCT employee_name) >= ALL (SELECT COUNT(employee_name)
37 FROM works
38 GROUP BY company_name);
```

查詢結果：

	company_name character varying (255)
1	Tech Innovations

驗證：

輸出所有公司的總員工數，確認 Tech Innovations 的確為擁有最多員工的公司。

```
28 SELECT company_name, COUNT(employee_name)
29 FROM works
30 GROUP BY company_name;
```

	company_name character varying (255) 🔒	count bigint 🔒
1	Blue Sky Inc	3
2	Dynamic Innovations	1
3	First Bank	3
4	GreenTech	2
5	Land Bank	2
6	Prime Technologies	3
7	Red Rock Solutions	2
8	Tech Innovations	4

### III. Exercise5-3

A. Find all companies located in every city in where Land Bank is located.

```
4 SELECT company_name
5 FROM company as c1
6 WHERE city in(SELECT city
7               FROM company
8               WHERE company_name = 'Land Bank');
```

查詢結果：

	company_name [PK] character varying (255) ✎
1	Land Bank
2	Tech Innovations

驗證：

利用上方搜尋結果再增加顯示 column city，Land Bank 與 Tech Innovations 同樣都位於 Los Angeles，且位於 Los Angeles 的公司數也為 2。

	company_name [PK] character varying (255) ✎	city character varying (255) ✎
1	Land Bank	Los Angeles
2	Tech Innovations	Los Angeles

	city character varying (255) 🔒	count bigint 🔒
1	New York	3
2	Austin	1
3	Orlando	1
4	San Diego	1
5	Los Angeles	2

```
12 SELECT city, COUNT(company_name)
13 FROM company
14 GROUP BY city
```

#### IV. Exercise5-4

A. Find the company that has the smallest payroll.

```
4 SELECT company_name
5 FROM works
6 GROUP BY company_name
7 HAVING SUM(salary) <= ALL(SELECT SUM(salary)
8                             FROM works
9                             GROUP BY company_name);
```

查詢結果：

	company_name character varying (255) 🔒
1	Dynamic Innovations

驗證：

利用上方搜尋結果再增加顯示 column SUM(salary), 可知 Dynamic Innovations 的員工薪水總額為 15642。

	company_name character varying (255) 🔒	sum bigint 🔒
1	Dynamic Innovations	15642

依照 company\_name 分組 works 後計算每間公司員工的薪水總額，重命名為 total\_salary 後再依照薪水總額高低由大到小降序輸出每列。

輸出表格後可知 Dynamic Innovations 的員工薪水總額為 15642，與上方數據一致，且在所有 8 間公司中擁有最低的薪水總額。

```
12 SELECT company_name, SUM(salary) AS total_salary
13 FROM works
14 GROUP BY company_name
15 ORDER BY total_salary DESC
```

	company_name character varying (255) 🔒	total_salary bigint 🔒
1	Tech Innovations	153875
2	Prime Technologies	108055
3	First Bank	94819
4	Blue Sky Inc	86513
5	Red Rock Solutions	81505
6	Land Bank	50132
7	GreenTech	37771
8	Dynamic Innovations	15642

- B. Find those companies whose employees earn a higher salary, on average, than the average salary at First bank.

```
23 SELECT company_name
24 FROM works
25 GROUP BY company_name
26 HAVING AVG(salary) > (SELECT AVG(salary)
27                        FROM works
28                        WHERE company_name = 'First Bank');
```

查詢結果：

	company_name character varying (255) 🔒
1	Prime Technologies
2	Red Rock Solutions
3	Tech Innovations

驗證：

```
31 SELECT company_name, AVG(salary) AS avg_salary
32 FROM works
33 GROUP BY company_name
34 ORDER BY avg_salary DESC
```

輸出所有公司的 company\_name 與 AVG(salary)，降序排序可看出 Red Rock Solutions、Tech Innovations、Prime Technologies 的員工平均薪水皆大於 First Bank 的員工平均薪水。

	company_name character varying (255) 🔒	avg_salary numeric 🔒
1	Red Rock Solutions	40752.500000000000
2	Tech Innovations	38468.750000000000
3	Prime Technologies	36018.333333333333
4	First Bank	31606.333333333333
5	Blue Sky Inc	28837.666666666667
6	Land Bank	25066.000000000000
7	GreenTech	18885.500000000000
8	Dynamic Innovations	15642.000000000000

## V. Exercise5-5

- A. Consider the employee database and give an expression in SQL.
- B. Modify the database so that the employee Emily Davis now lives in Taipei.

```
14 UPDATE employee
15 SET city = 'Taipei'
16 WHERE employee_name = 'Emily Davis'
```

查詢結果：

	employee_name [PK] character varying (255)	city character varying (255)
1	John Doe	New York
2	Jane Smith	Los Angeles
3	Michael Johnson	Chicago
4	Christopher Wilson	San Francisco
5	Jessica Taylor	Boston
6	Daniel Anderson	Seattle
7	Sarah Martinez	Miami
8	David Thompson	Denver
9	Jennifer Rodriguez	Atlanta
10	James Garcia	Dallas
11	Linda Hernandez	New York
12	William Martinez	Philadelphia
13	Patricia Gonzalez	Detroit
14	Richard Perez	New York
15	Susan Ramirez	Minneapolis
16	Daniel Lewis	Tampa
17	Maria Walker	Portland
18	Charles King	Orlando
19	Karen Wright	Austin
20	Emily Davis	Taipei

驗證：

對比最初插入資料後輸出的表格，可知 Emily Davis 的 city 從 Austin 被改為 Taipei。

C. Give all employees of First Bank a 10 percent raise.

```
20 SELECT *
21 FROM works
22 WHERE company_name = 'First Bank'
```

驗證：

以 John Doe 為例，加薪前 John Doe 的薪水是 23651，加薪後\*1.1 倍。

(圖一) 原本員工薪水數據。

	employee_name [PK] character varying (255)	company_name [PK] character varying (255)	salary integer
1	John Doe	First Bank	23651
2	Jane Smith	First Bank	41783
3	Richard Perez	First Bank	29385

(圖二) 加薪後員工薪水數據。

	employee_name [PK] character varying (255)	company_name [PK] character varying (255)	salary integer
1	John Doe	First Bank	26016
2	Jane Smith	First Bank	45961
3	Richard Perez	First Bank	32324

D. Give all managers of First Bank a 10 percent raise.

```
43 UPDATE works
44 SET salary = salary*1.1
45 WHERE employee_name in (SELECT manager_name
46                          FROM manages) AND company_name = 'First Bank'
```

驗證：

由(圖一-1)與(圖一-2)可知 First Bank 的經理有 John Doe、Jane Smith，Richard Perez 僅有員工身分，加薪 First Bank 的經理後，在(圖二)可看出，John Doe、Jane Smith 都有漲薪，而 Richard Perez 的薪水維持原樣。

(圖一-1) 原本經理薪水數據。

	employee_name [PK] character varying (255)	company_name [PK] character varying (255)	salary integer
1	John Doe	First Bank	26016
2	Jane Smith	First Bank	45961

(圖一-2) 原本員工薪水數據。

	employee_name [PK] character varying (255)	company_name [PK] character varying (255)	salary integer
1	John Doe	First Bank	26016
2	Jane Smith	First Bank	45961
3	Richard Perez	First Bank	32324

(圖二) 加薪後在 First Bank 工作的員工薪水數據。

	employee_name [PK] character varying (255)	company_name [PK] character varying (255)	salary integer
1	Richard Perez	First Bank	32324
2	John Doe	First Bank	28618
3	Jane Smith	First Bank	50557

E. Delete all tuples in the works relation for employees of Land Bank.

```
54 DELETE FROM works
55 WHERE company_name = 'Land Bank';
```

Data Output Messages Notifications

DELETE 2

Query returned successfully in 52 msec.

驗證：

```
51 SELECT *
52 FROM works
53 WHERE company_name = 'Land Bank';
```

(圖一) 還沒執行 DELETE 前輸出所有在 Lank Bank 工作的員工。

	employee_name [PK] character varying (255)	company_name [PK] character varying (255)	salary integer
1	Michael Johnson	Land Bank	32896
2	James Garcia	Land Bank	17236

(圖二) 執行 DELETE 後輸出所有在 Lank Bank 工作的員工，並無輸出任何資料，可知在 Lank Bank 工作的員工皆已被刪除。

	employee_name [PK] character varying (255)	company_name [PK] character varying (255)	salary integer
--	---	--	-------------------

## VI. Exercise5-8

A. Consider the employee database.

Using SQL, define a view consisting of manager\_name and the average salary of all employees who work for that manager.

```

1 CREATE VIEW employee_avg_salary AS
2 SELECT m.manager_name, AVG(w.salary)
3 FROM works as w
4 JOIN manages AS m ON w.employee_name = m.employee_name
5 GROUP BY m.manager_name

```

驗證：

成功創建一個 view，輸出 employee\_avg\_salary。

```

11 SELECT *
12 FROM employee_avg_salary

```

	manager_name character varying (255)	avg numeric
1	[null]	38097.285714285714
2	Daniel Lewis	22819.000000000000
3	Emily Davis	31041.500000000000
4	John Doe	32324.000000000000
5	Charles King	36742.000000000000
6	Jennifer Rodriguez	32984.000000000000
7	William Martinez	38457.500000000000
8	Jane Smith	28618.000000000000
9	Maria Walker	12875.000000000000



### 3. 驗證表上兩種完整性驗證

#### I. referential integrity constraint，參考完整性約束，使用外鍵確保表格之間的關聯性。

創建 Table manages 時的 columns 與 constraint。

```
21 CREATE TABLE manages (  
22     employee_name varchar(255) references employee(employee_name),  
23     manager_name varchar(255) references employee(employee_name),  
24     PRIMARY KEY(employee_name)  
25 );
```

manages 連接了 employee(employee\_name)，加入黃乙家及他的上司郭怡楨，執行後因為黃乙家不在員工列表中，故看到錯誤訊息，驗證 referential integrity constraint。

```
19 INSERT INTO manages (employee_name, manager_name) VALUES  
20 ('Haung Yijia', 'Jenny');
```

Data Output Messages Notifications

ERROR: Key (employee\_name)=(Haung Yijia) is not present in table "employee".insert or update on table "manages" violates foreign key constraint "manages\_employee\_name\_fkey"

ERROR: insert or update on table "manages" violates foreign key constraint "manages\_employee\_name\_fkey"

SQL state: 23503

Detail: Key (employee\_name)=(Haung Yijia) is not present in table "employee".

#### II. check constraint using a subquery，檢查約束使用子查詢，檢查特定條件是否滿足。

創建 Table works 時的 columns 與 constraint。

```
13 CREATE TABLE works (  
14     employee_name varchar(255) references employee(employee_name),  
15     company_name varchar(255) references company(company_name),  
16     salary integer DEFAULT 0 CONSTRAINT positive_salary CHECK (salary >= 0),  
17  
18     PRIMARY KEY(employee_name, company_name)  
19 );
```

works 中有限制 salary 不能小於等於 0，下方加入新資料，其中陳乙菁的 salary 為負，執行後可以看到報錯訊息"positive\_salary"，驗證 check constraint using a subquery。

```
1 INSERT INTO employee (employee_name, street, city) VALUES  
2 ('Dandelion', '123', 'Chiayi'),  
3 ('Fiona', '123', 'Hsinchu'),  
4 ('Pearl', '123', 'Miaoli'),  
5 ('Jenny', '123', 'Hsinchu');  
  
8 INSERT INTO company (company_name, city) VALUES  
9 ('Chung Yuan', 'Taoyuan');  
  
24 INSERT INTO works (employee_name, company_name, salary) VALUES  
25 ('Dandelion', 'Chung Yuan', 70000),  
26 ('Fiona', 'Chung Yuan', -100),  
27 ('Pearl', 'Chung Yuan', 50000),  
28 ('Jenny', 'Chung Yuan', 1000000);
```

Data Output Messages Notifications

ERROR: Failing row contains (Fiona, Chung Yuan, -100).new row for relation "works" violates check constraint "positive\_salary"

ERROR: new row for relation "works" violates check constraint "positive\_salary"

SQL state: 23514

Detail: Failing row contains (Fiona, Chung Yuan, -100).