



# Live07-HW02-Restaurant pizza SQL

ติดตั้ง library กี่ จำเป็น

```
##Connect to PostgreSQL server
library(RPostgreSQL)
library(tidyverse)
```

สร้าง Connection โดยนำข้อมูลมาจาก database ที่สร้างบน server

<https://www.elephantsql.com/>

**Details**

Server	Active Plan	
Region		
Created at		
User & Default database	<input type="button" value="Reset"/>	
Password	***	<input type="button" value="Rotate password"/>
URL		
Current database size		
Max database size		



Tiny Turtle

### API access

Control your instance via our [HTTP API](#).

API Key

## Connect กับ RStudio

```
con <- dbConnect(
  PostgreSQL(), ##เพื่อบอกโปรแกรมว่า Host เราเป็น PostgreSQL หรืออาจเป็น MySQL
  host = "*****.db.elephantsql.com", ##เหมือน ip address ของ database
  dbname = "*****",
  user = "*****",
  password = "*****",
  port = *****
)
```

```
> con <- dbConnect(
+   PostgreSQL(), ##เพื่อบอกโปรแกรมว่า Host เราเป็น PostgreSQL หรืออาจเป็น MySQL เป็นต้น
+   host = "floppy.db.elephantsql.com", ##เหมือน ipaddress ของ database ที่เราสร้างขึ้น
+   dbname = "objibjpn",
+   user = "objibjpn",
+   password = "DdF82ydwP-ue0cAwYY6zFDKfBvIeg2U3",
+   port = 5432
+ )
> con
<PostgreSQLConnection>
```

เช็ค Table ที่มีอยู่ใน data base ใช้ `dbListTables()`

จากตัวอย่าง มีการเพิ่ม table เข้าไปโดยใช้

```
dbWriteTable(#data base, "ชื่อtable", object ที่เก็บ table)
```

```
dbWriteTable(con, "products", products)
> dbListTables(con)
[1] "products"
```

เราสามารถนำข้อมูลมาใช้งานโดยใช้

```
dbGetQuery(ชื่อ database, "select คอลัมน์1, คอลัมน์2 from ชื่อtable")
```

```
> df <- dbGetQuery(con, "select id, product_name from products"
> df
  id      product_name
1 1          chocolate
2 2          pineapple
3 3 samsung galaxy S23
```

## HW02 - restaurant pizza SQL

### Create 3-5 data frames => write table into server

นำข้อมูลจากร้านพิซซ่า ใน SQL เข้าไปใส่ใน database แล้วลองจัดการข้อมูล

- ▼ เปรียบเทียบกับการสร้าง table ด้วย SQL (ความเดินบทที่แล้ว)

```
--customers table
CREATE TABLE customers(
  cus_id text,
  firstname text,
  lastname text,
  cus_gender text,
  age int
);
INSERT INTO customers VALUES
  ("1", "Antony", "Raerae", "Male", 38),
  ("2", "Jenny", "Jonas", "Female", 32),
  ("3", "Stella", "Kim", "Female", 21),
  ("4", "Krixi", "Jungle", "Female", 19),
  ("5", "Dokja", "Yu", "Male", 26),
```

```

("6", "Yorn", "Sang", "Male", 47),
("7", "Yusuke", "Fujisaki", "Male", 41),
("8", "Eren", "Yeager" , "Male", 35);
.mode box
SELECT * FROM customers;

```

## 1. สร้าง data frames ของร้านพิซซ่า ใน R สร้างด้วย tribble

**Table 1 : customers**

```

customers <- tribble(
  ~cus_id, ~firstname, ~lastnmae, ~cus_gender, ~age,
  "1", "Antony", "Raerae", "Male", 38,
  "2", "Jenny", "Jonas", "Female", 32,
  "3", "Stella", "Kim", "Female", 21,
  "4", "Kixi", "Jungle", "Female", 19,
  "5", "Dokja", "Yu", "Male", 26,
  "6", "Yorn", "Sang", "Male", 47,
  "7", "Yusuke", "Fujisaki", "Male", 41,
  "8", "Eren", "Yeager" , "Male", 35)

```

	# A tibble: 8 × 5				
	cus_id	firstname	lastnmae	cus_gender	age
	<chr>	<chr>	<chr>	<chr>	<dbl>
1	1	Antony	Raerae	Male	38
2	2	Jenny	Jonas	Female	32
3	3	Stella	Kim	Female	21
4	4	Kixi	Jungle	Female	19
5	5	Dokja	Yu	Male	26
6	6	Yorn	Sang	Male	47
7	7	Yusuke	Fujisaki	Male	41
8	8	Eren	Yeager	Male	35

**Table 2 : menus**

```

menus <- tribble(
  ~menu_id, ~menu_name, ~menu_price
  1, "Peperoni", 250,
  2, "Hawaiian", 250,
  3, "Seafood", 400,
  4, "Vegetarian", 300,
  5, "Meat Lovers", 350,
  6, "Salad", 150,
  7, "Spaghetti", 250,
  8, "Steak", 800,
  9, "Hamburger", 150,
  10, "Ice cream", 150
)

```

```

# A tibble: 10 × 3
  menu_id menu_name   menu_price
    <dbl> <chr>           <dbl>
1       1 Peperoni        250
2       2 Hawaiian        250
3       3 Seafood          400
4       4 Vegetarian      300
5       5 Meat Lovers     350
6       6 Salad            150
7       7 Spaghetti        250
8       8 Steak            800
9       9 Hamburger        150
10      10 Ice cream       150

```

**Table 3 : orders**

```

orders <- tribble(
  ~order_id, ~order_date, ~menu_id, ~cus_id
  1, "2023-09-01", 1, "1",
  2, "2023-09-01", 9, "1",
  3, "2023-09-02", 3, "8",
  4, "2023-09-02", 10, "8",
  5, "2023-09-04", 2, "3",
  6, "2023-09-05", 3, "3",
  7, "2023-09-17", 6, "5",
  8, "2023-09-23", 8, "4",
  9, "2023-09-24", 8, "7",
  10, "2023-09-24", 7, "7",
  11, "2023-09-24", 10, "7",
  12, "2023-09-30", 8, "6"
)
orders

```

```

# A tibble: 12 × 4
  order_id order_date menu_id cus_id
    <dbl> <chr>       <dbl> <chr>
1       1 2023-09-01     1 1
2       2 2023-09-01     9 1
3       3 2023-09-02     3 8
4       4 2023-09-02    10 8
5       5 2023-09-04     2 3
6       6 2023-09-05     3 3
7       7 2023-09-17     6 5
8       8 2023-09-23     8 4
9       9 2023-09-24     8 7
10      10 2023-09-24    7 7
11      11 2023-09-24    10 7
12      12 2023-09-30    8 6

```

## 2. ใส่ data frame ลงใน data base ของเราด้วย `dbWriteTable()`

เมื่อขึ้นว่า TRUE แสดงว่าการใส่ข้อมูลสำเร็จ

```
dbWriteTable(con, "customers", cus  
dbWriteTable(con, "menus", menus)  
dbWriteTable(con, "orders", orders)  
  
> dbwriteTable(con, "customers", customers)  
[1] TRUE  
> dbwriteTable(con, "menus", menus)  
[1] TRUE  
> dbwriteTable(con, "orders", orders)  
[1] TRUE
```

## 3. ตรวจสอบ table ใน data base ด้วย `dbListTables()`

```
> dbListTables(con)  
[1] "products" "customers" "menus" "orders"
```

## 4. ลบ table ที่ไม่เกี่ยวกับร้านพิซซ่าออก ด้วย `dbRemoveTable()`

เมื่อขึ้นว่า TRUE แสดงว่าการลบข้อมูลสำเร็จ

```
dbRemoveTable(con, "produ  
> dbRemoveTable(con, "pro  
[1] TRUE  
  
> dbListTables(con)  
[1] "products" "customers" "menus" "orders"  
> dbRemoveTable(con, "products")  
[1] TRUE  
> dbListTables(con)  
[1] "customers" "menus" "orders"
```

## 5. ทดลองจัดการข้อมูล โดยดึง table มาใช้ด้วย `dbGetQuery()`

```
dbGetQuery(con, "select * from customers")
```

row.names	cus_id	firstname	lastname	cus_gender	age
1	1	Antony	Rae	Male	38
2	2	Jenny	Jonas	Female	32
3	3	Stella	Kim	Female	21
4	4	Krixi	Jungle	Female	19
5	5	Dokja	Yu	Male	26
6	6	Yorn	Sang	Male	47
7	7	Yusuke	Fujisaki	Male	41
8	8	Eren	Yeager	Male	35

ต้องการเอา row name ออก และเก็บใน data frame ใหม่ เพื่อใช้งาน

```

df_customers <- dbGetQuery(con, "select cus_id, firstname, lastname, cus_gender, age from customers")
df_menus <- dbGetQuery(con, "select menu_id, menu_name, menu_price from menus")
df_orders <- dbGetQuery(con, "select order_id, order_date, menu_id, cus_id from orders")

> df_customers <- dbGetQuery(con, "select cus_id, firstname, lastname, cus_gender, age from customers")
> df_customers
  cus_id firstname lastname cus_gender age
1      1    Antony   Raerae     Male  38
2      2     Jenny    Jonas   Female  32
3      3    Stella     Kim   Female  21
4      4    Krixi   Jungle   Female  19
5      5     Dokja      Yu     Male  26
6      6      Yorn     Sang     Male  47
7      7  Yusuke  Fujisaki     Male  41
8      8      Eren    Yeager     Male  35

> df_menus <- dbGetQuery(con, "select menu_id, menu_name, menu_price from menus")
> df_menus
  menu_id menu_name menu_price
1        1  Peperoni       250
2        2  Hawaiian       250
3        3   Seafood       400
4        4 Vegetarian      300
5        5 Meat Lovers     350
6        6      Salad       150
7        7  Spaghetti      250
8        8      Steak       800
9        9 Hamburger      150
10       10  Ice cream     150
> df_orders <- dbGetQuery(con, "select order_id, order_date, menu_id, cus_id from orders")

> df_orders <- dbGetQuery(con, "select order_id, order_date, menu_id, cus_id from orders")
> df_orders
  order_id order_date menu_id cus_id
1         1 2023-09-01      1     1
2         2 2023-09-01      9     1
3         3 2023-09-02      3     8
4         4 2023-09-02     10     8
5         5 2023-09-04      2     3
6         6 2023-09-05      3     3
7         7 2023-09-17      6     5
8         8 2023-09-23      8     4
9         9 2023-09-24      8     7
10        10 2023-09-24     7     7
11        11 2023-09-24     10     7
12        12 2023-09-30      8     6

```

## 6. Disconnection

```

dbDisconnect(con)
> dbDisconnect(con)
[1] TRUE

```

