

資料庫系統導論期末Project

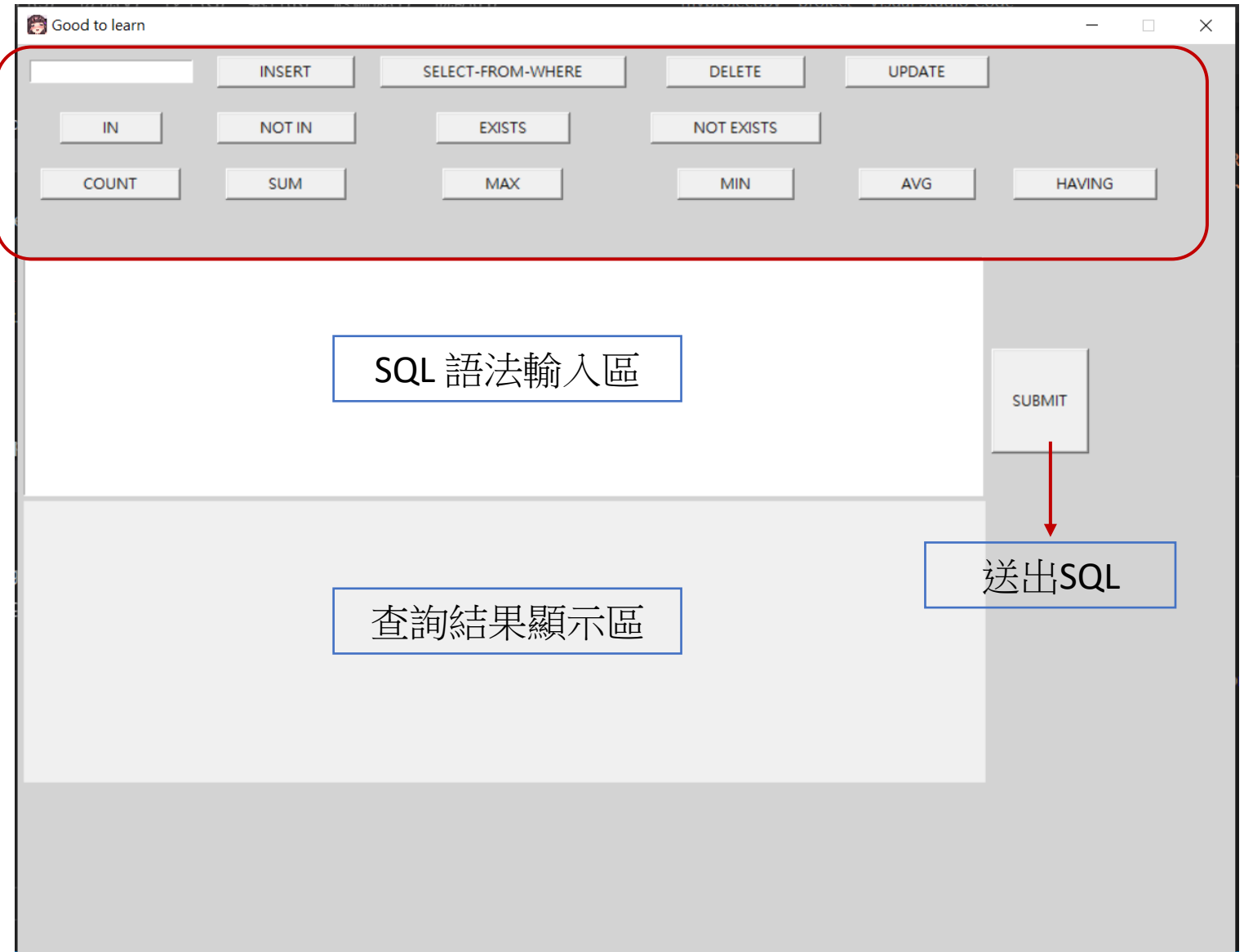
F74076213 吳定洋

介面截圖與使用說明

系統架構與環境

- 作業系統：Window10
- 系統資料庫：SQLite3
- GUI：Tkinter

各個Button



各Button細項說明

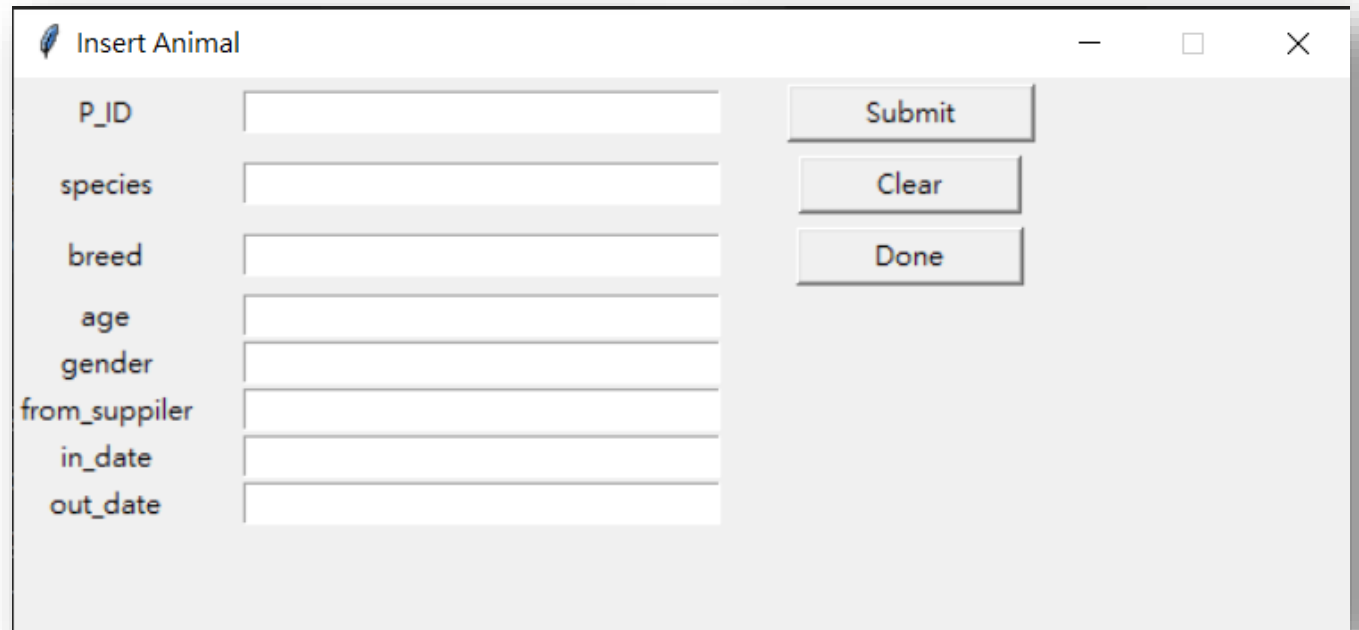
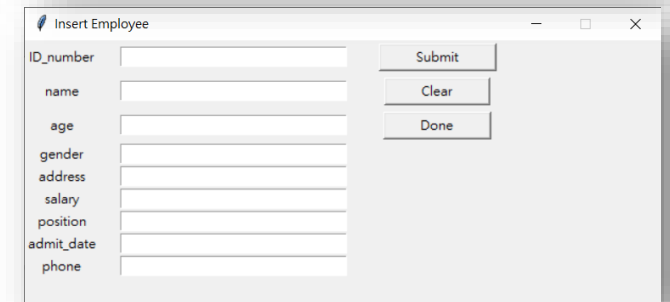
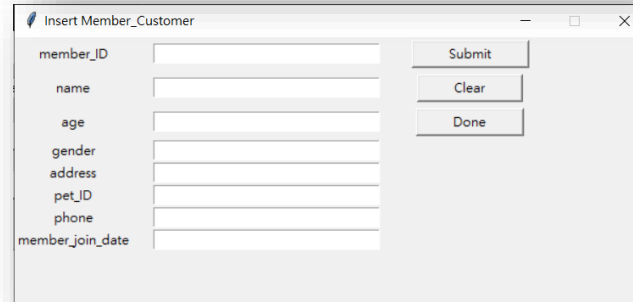
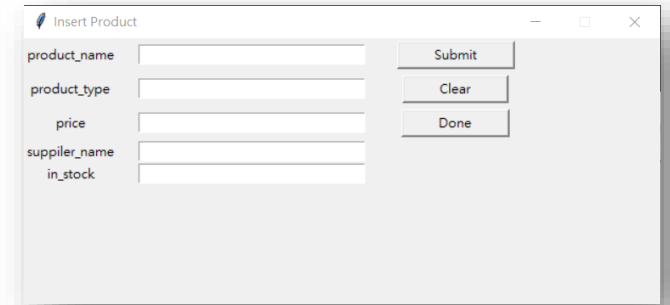
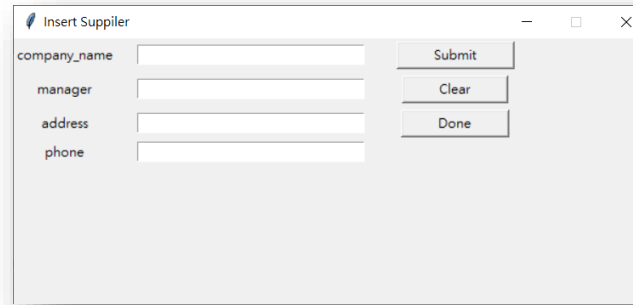
INSERT




輸入要查找的Table

隨著輸入不同的Table name會跳出不同的insert視窗，可以從此輸入需要的attributes。

- Submit：寫入資料庫
- Clear：清空Entry
- Done：結束



SELECT-FROM-WHERE

 SELECT-WHERE-FROM

SELECT

欲獲取的attribute

FROM

Table name

WHERE

輸入條件(可空)

Submit

DELETE

 DELETE

FROM

Table name

WHERE

輸入條件

Submit

UPDATE

 UPDATE

UPDATE

Table name

SET

欲更改的attribute

WHERE

輸入條件

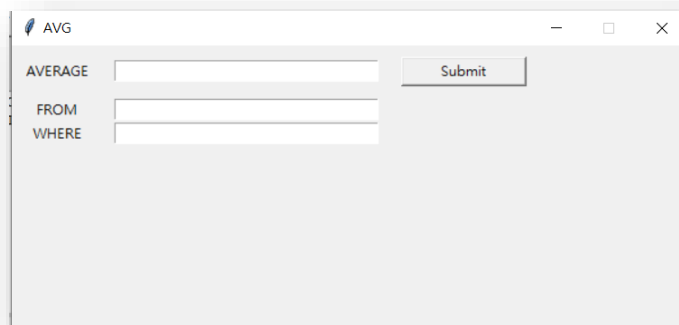
Submit

Table
column1 = 'value1', ...
some_column = 'some_value', ...

COUNT & SUM & MAX & MIN & AVG

簡單的Aggregate

- Aggregate function：顧名思義
(例如MAX就是用來找出在此Table中
特定條件下，此attribute的最大值。)
- FROM：Table name
- WHERE：條件(可空)



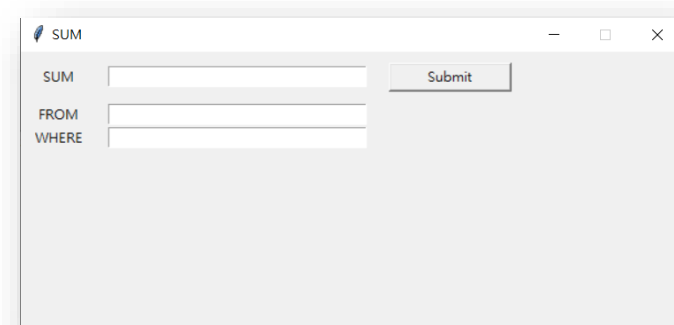
AVG

AVERAGE

FROM

WHERE

Submit



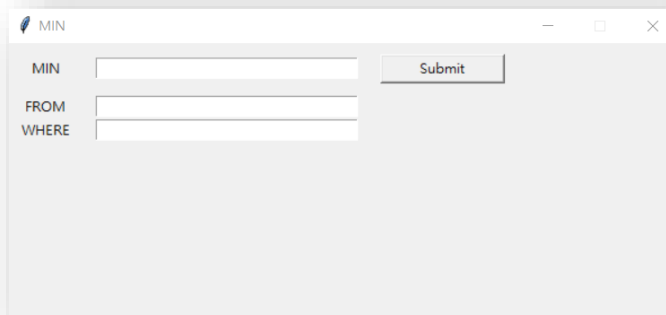
SUM

SUM

FROM

WHERE

Submit



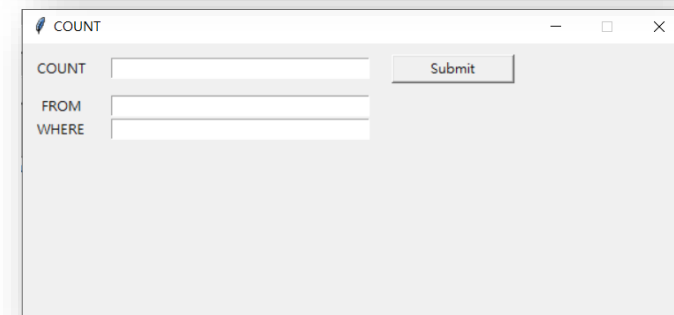
MIN

MIN

FROM

WHERE

Submit



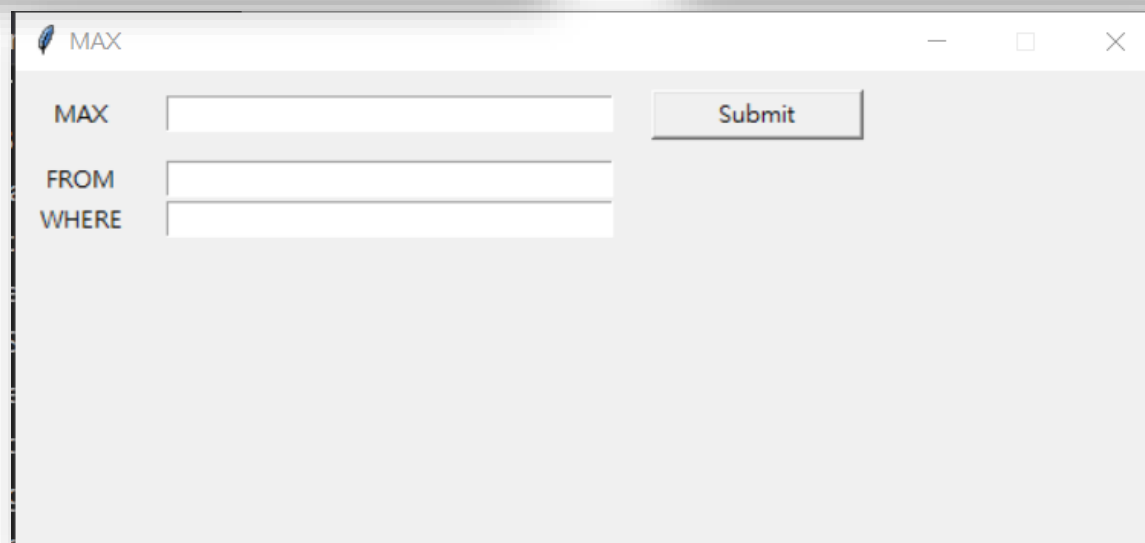
COUNT

COUNT

FROM

WHERE

Submit



MAX

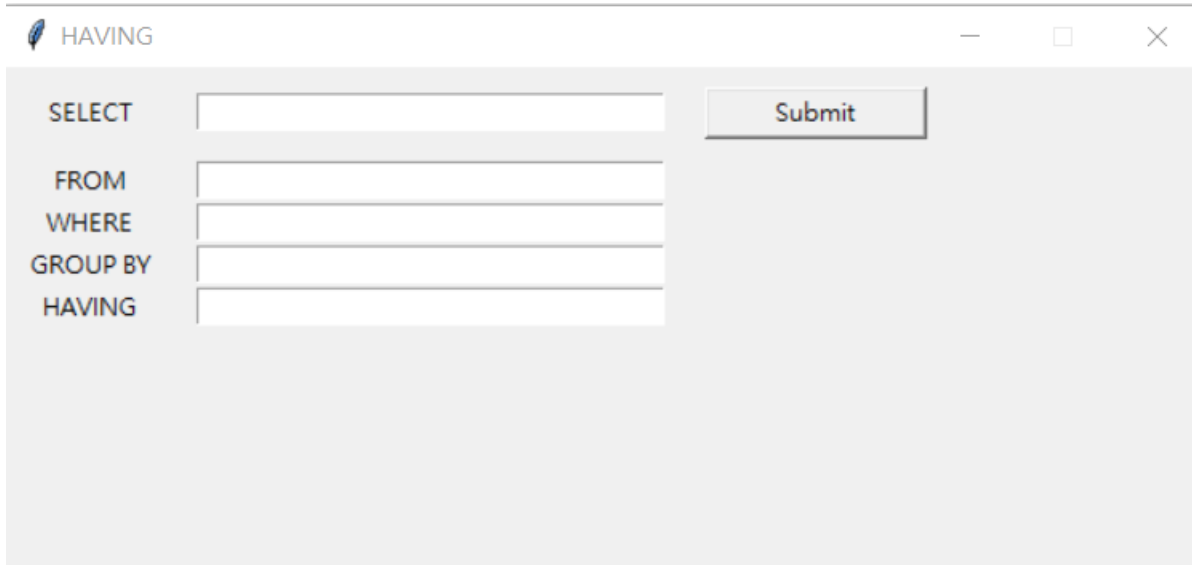
MAX

FROM

WHERE

Submit

HAVING



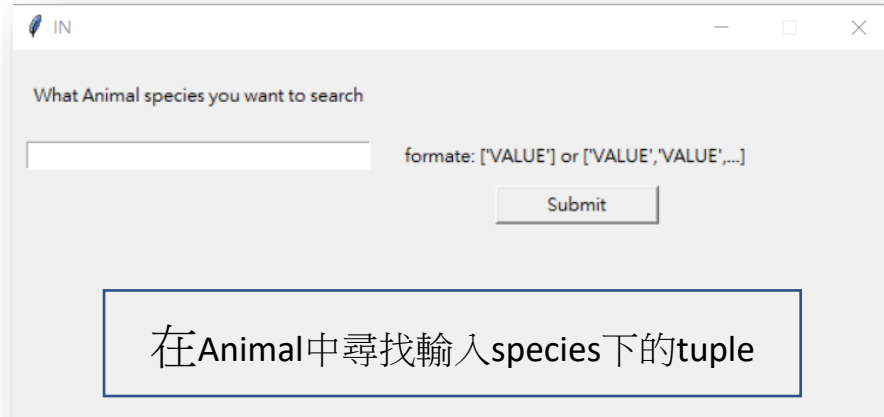
The screenshot shows a web application window titled "HAVING" with a feather icon. The interface contains a SQL query builder with the following components:

- Labels on the left: SELECT, FROM, WHERE, GROUP BY, and HAVING.
- Input fields on the right corresponding to each label.
- A "Submit" button located to the right of the SELECT input field.

- SELECT : "AttributeA" 和 Aggregate function
- FROM : Table name
- WHERE : 條件 (可空)
- GROUP BY : AttributeA"
- HAVING : Aggregate function 條件

IN & NOT IN

因為Nested queries的SQL寫起來較複雜，所以我寫比較死，限縮條件



What Animal species you want to search

format: ['VALUE'] or ['VALUE','VALUE',...]

Submit

在Animal中尋找輸入species下的tuple

EXISTS & NOT EXISTS



Member Customer Pet ID

Submit

輸入Member Customer的P_ID，查找Animal table 中與之呼應的pet_ID，印出此tuple

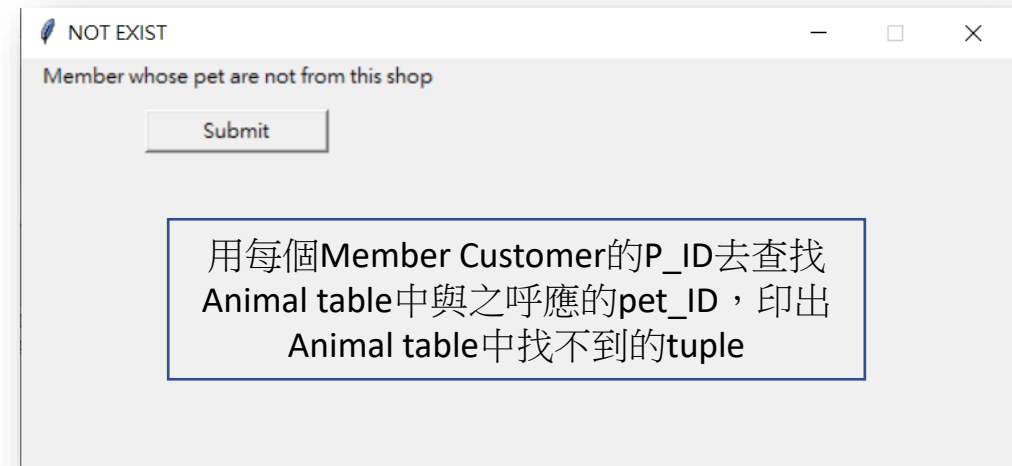


The Employee with the position you don't care

format: ['VALUE'] or ['VALUE','VALUE',...]

Submit

在Employee中尋找不屬於輸入position下的tuple

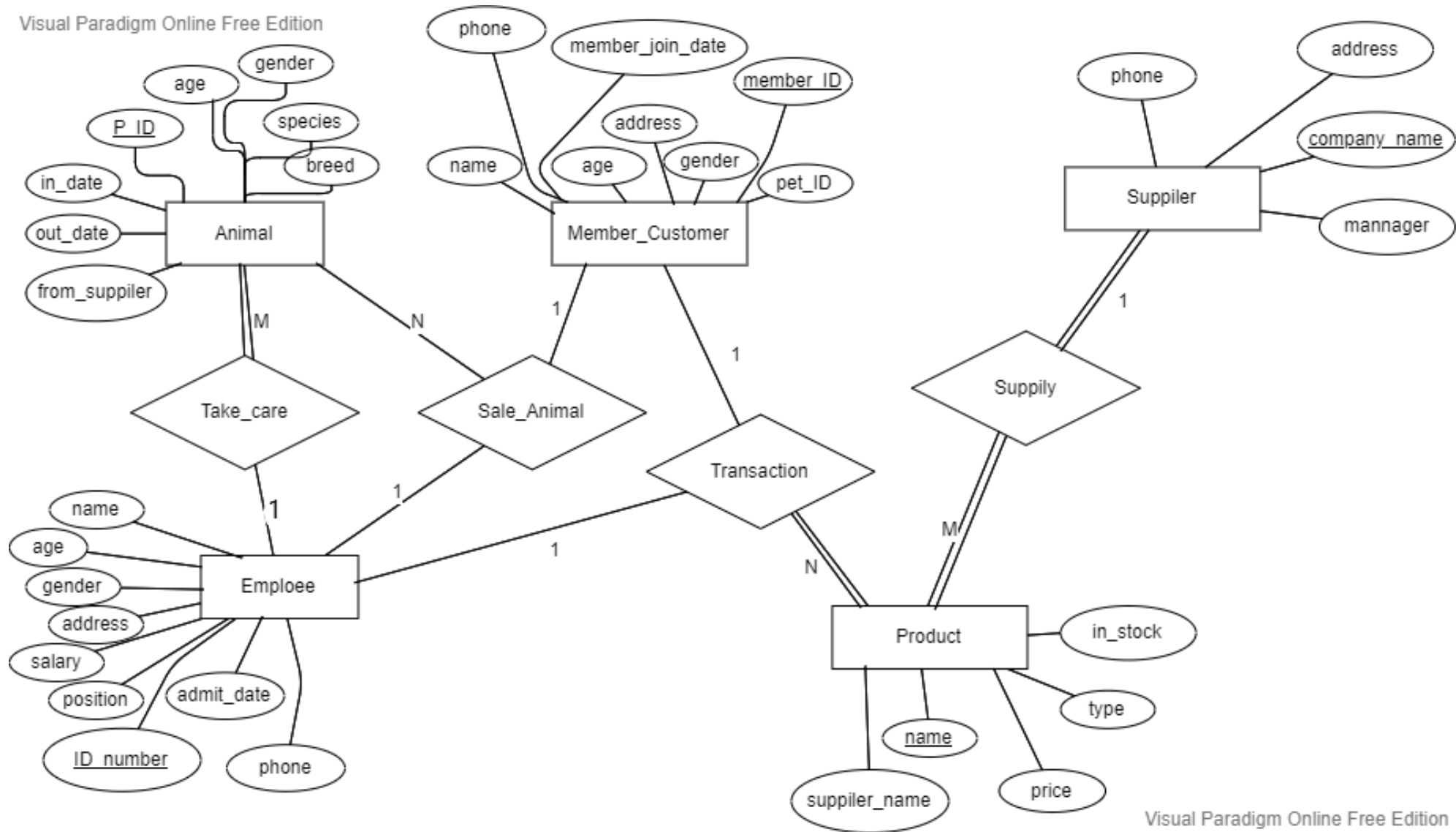


Member whose pet are not from this shop

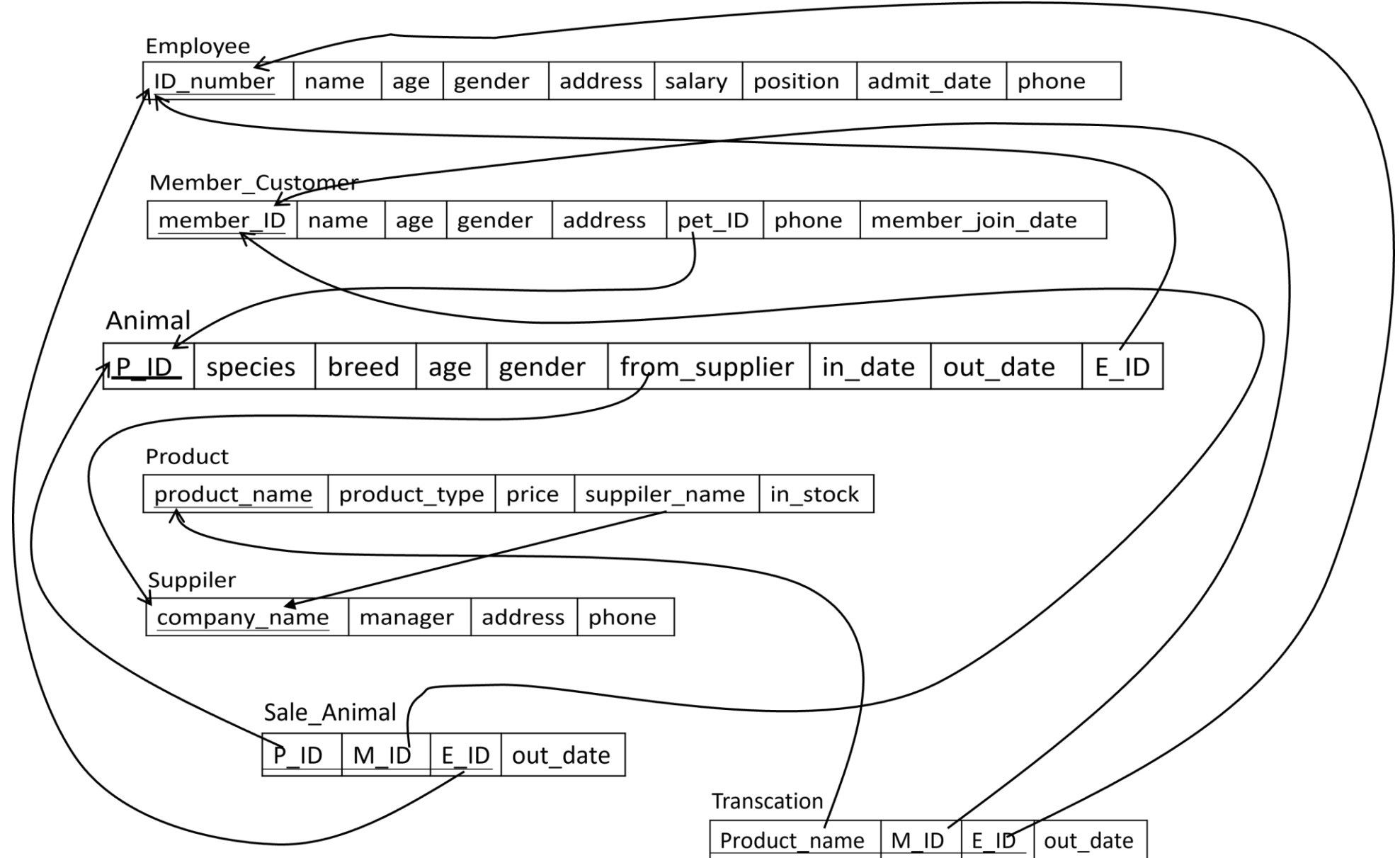
Submit

用每個Member Customer的P_ID去查找Animal table中與之呼應的pet_ID，印出Animal table中找不到的tuple

Petshop ER diagram



Petshop Relation Schema



資料庫說明

Animal table - 用來記錄每隻petshop動物的table

Attributes -

- P_ID: 寵物編號。為**Primary key**。
- species: 物種。例如：狗、貓、烏龜、蛇...
- breed: 品種。例如：馬爾濟斯、柴犬、拉不拉多...
- age: 動物年齡。(可NULL)
- gender: 動物性別。
- from_supple: 來自哪家供應商。
- in_date: 進入此petshop日期。
- out_date: 離開此petshop日期。(可NULL)
- E_ID: 因 Take_care Relation 而從Employee中獲得的Attribute，代表員工ID。

Employee table – 用來記錄每位petshop員工的table

Attributes -

- ID_number: 員工編號。為**Primary key**。
- name: 名字。
- age: 年齡。
- gender: 性別。
- address: 住家地址。
- salary: 每月薪水。
- position: 職位。例如：店長、店員...
- admit_date: 錄取日期。
- phone: 連絡電話。

資料庫說明(cont.)

Member_Customer table - 用來記錄每位petshop會員顧客的table

Attributes -

- member_ID: 會員編號。為Primary key。
- name: 名字。
- age: 年齡。
- gender: 性別。
- address: 聯絡地址。(可NULL)
- pet_ID: 本petshop之P_ID (寵物編號)。(可NULL)
- phone: 連絡電話。
- member_join_date: 加入此petshop會員日期。

Product table – 用來記錄每項petshop商品的table

Attributes -

- product_name: 商品名稱。為Primary key。
- product_type: 商品類型。例如：食物、玩具...
- price: 價錢。
- supplier_name: 供應商公司名稱。
- in_stock: 在本petshop中的庫存。

Supplier table – 用來記錄每個供應商的table

Attributes -

- company_name: 供應商公司名稱。為Primary key。
- manager: 供應商管理者名字。
- address: 供應商地址。
- phone: 供應商電話。

資料庫說明(cont.)

Sale_Animal table – 因為Animal、Employee、Member_Customer 之三元Relation而延伸出的table

Attributes -

- P_ID: 寵物編號，由Animal的P_ID獲得。
- M_ID: 會員編號，由Member_Customer的member_ID獲得。
- E_ID: 員工編號，由Employee的ID_number獲得
- outdate: 賣出日期。

P_ID + M_ID + E_ID 三者合起來為**Primary key**。

Transcation table – 因為Product、Employee、Member_Customer 之三元Relation而延伸出的table

Attributes -

- Product_name: 商品名稱，由Product的product_name獲得。
- M_ID: 會員編號，由Member_Customer的member_ID獲得。
- E_ID: 員工編號，由Employee的ID_number獲得
- outdate: 賣出日期。

Product_name + M_ID + E_ID 三者合起來為**Primary key**。

資料庫說明(cont.)

Take_care Relation :

- Animal和Employee的Relation
- Employee (partial participate) , 沒有全部員工都參與此relation。
- Animal (total participate) , 全部動物皆參與此relation。
- 有1個Employee會照顧N個Animal

Suppile Relation :

- Suppiler和Product的Relation
- Suppiler (total participate) , 全部供應商都參與此relation。
- Product (total participate) , 全部商品都參與此relation。
- 1個Suppiler會提供N個Product

Transcation Relation :

- Member_Customer、Employee、Product的Relation
- Member_Customer (partial participate) , 沒有全部會員顧客都參與此relation。
- Employee (partial participate) , 沒有全部員工皆參與此relation。
- Product (total participate) , 全部商品都參與此relation。
- 有1個Employee會和1個Member_Customer 交易N個Product

Sale_Animal Relation :

- Member_Customer、Employee、Animal的Relation
- Member_Customer (partial participate) , 沒有全部會員顧客都參與此relation。
- Employee (partial participate) , 沒有全部員工皆參與此relation。
- Animal (partial participate) , 沒有全部動物都參與此relation。
- 有1個Employee會和1個Member_Customer 買賣N隻Animal