

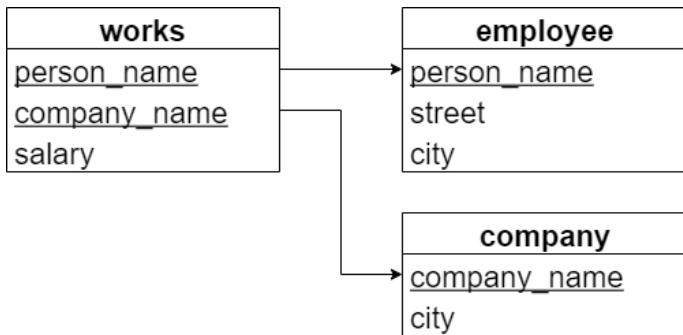
employee (person_name, street, city)
company (company_name, city)
works (person_name, company_name, salary)

Figure 1. An Employee database.

A simple Employee database is shown in Figure 1. Please answer Questions 1-3 based on this schema.

1. (20%) The attribute *person_name* is defined as the primary key of the *employee* relation for simplicity(假設沒有兩個人同名), and the attribute *company_name* is defined as the primary key of the *company* relation. The *works* relation describes which company(s) a person works for and the associated salary(s). Please determine what the best primary key of the *works* relation should be based on the following two assumptions, respectively. Note that you need to explain your answer.
 - (a) An employee can work for only one company in this database.
 - (b) An employee might work for many companies in this database.
 - (a) 訂{ *person_name* }為 primary key :
若限定一個 *employee* 只能為一家 *company* 工作，則每個 *person_name* 在 relation *works* 中不會重複，具有唯一性，故以此做為 primary key。
 - (b) 訂{ *person_name*, *company_name* }為 primary key :
若一個 *employee* 可以為多家 *company* 工作，則每個 *person_name* 在 relation *works* 中會重複不具有唯一性，而一家 *company* 會有多個 *employee*，*company_name* 在 relation *works* 中也會重複不具有唯一性，只有在{ *person_name*, *company_name* }作為 primary key 才有不重複性和唯一性。
- 兩個小題一小題各 10 分，其中 primary key 找錯扣 10 分，解釋 primary key 的原因不充分扣 2~4 分。
2. (20%) Suppose that the primary key of the *works* relation is (*person_name*, *company_name*).
 - (a) Identify the two appropriate foreign keys of the *works* relation.
 - (b) Construct a schema diagram for this database.
 - (a) *person_name*，從 *works* 參照至 *employee* 表格
company_name，從 *works* 參照至 *company* 表格

(b)



- (a) 找錯 foreign key 一個扣 5 分，未寫參照的表格1個扣 2 分
(b) 完全未畫 schema diagram 扣 8-10 分，未依課本規則畫表扣 2 分，表格內少寫/多加屬性、未寫表格名稱、未將表格內屬性表示 primary key 或標示錯誤一個表格扣 1 分，未標 foreign key 的 reference 關係、標示方向錯誤、標示不明確一條扣 2 分

employee (person_name, street, city)
company (company_name, city)
works (person_name, company_name, salary)

3. (40%) Suppose that the primary key of the *works* relation is the attribute *person_name*. Give an expression in the relational algebra for each of the following queries, respectively.

- (a) Find the name of each employee who lives in the city “Miami”.

解法 1:

$$\Pi_{\text{person_name}} (\sigma_{\text{city} = \text{"Miami"}} (\text{employee}))$$

符號、格式錯扣 1 分， Π 欄位錯扣 2 分， σ 限制式錯扣 3 分，資料表錯扣 3 分

- (b) Find the name of each employee who lives in the city “Miami” and whose salary is greater than \$100000.

解法 1 :

$$\Pi_{\text{employee.person_name}} (\sigma_{\text{employee.person_name} = \text{works.person_name} \wedge \text{works.salary} > 100000} (\text{employee} \times \text{works}))$$

解法 2 :

$$\Pi_{\text{employee.person_name}} (\sigma_{\text{works.salary} > 100000 \wedge \text{employee.city} = \text{"Miami}}} (\text{employee} \bowtie_{\text{employee.person_name} = \text{works.person_name}} \text{works})$$

Π 欄位錯扣 2 分， σ 限制式錯扣 3 分，join 錯扣 4 分，資料表錯扣 1 分

- (c) Find the name and the city of residence of each employee who works for “FirstBank”.

解法 1:

$$\Pi_{\text{works.person_name}, \text{employee.city}} (\sigma_{\text{works.company_name} = "FirstBank" \wedge \text{employee.person_name} = \text{works.person_name}} (\text{employee} \times \text{works}))$$

解法 2:

$$\Pi_{\text{works.person_name}, \text{employee.city}} (\sigma_{\text{works.company_name} = "FirstBank"} (\text{employee} \bowtie_{\text{employee.person_name} = \text{works.person_name}} \text{works}))$$

Π 欄位錯扣 1~2 分， σ 限制式錯扣 3 分，join 錯扣 4 分，資料表錯扣 1 分

- (d) Find the name of each employee who does NOT work for “FirstBank”.

解法1:

$$\Pi_{\text{person_name}}(\text{works}) - \Pi_{\text{person_name}}(\sigma_{\text{company_name} = "FirstBank"} (\text{works}))$$

或

$$\Pi_{\text{person_name}}(\text{employee}) - \Pi_{\text{person_name}}(\sigma_{\text{company_name} = "FirstBank"} (\text{works}))$$

解法2:

$$\Pi_{\text{person_name}} (\sigma_{\text{company_name} \neq "FirstBank"} (\text{works}))$$

解法3:

$$\Pi_{\text{person_name}} (\sigma_{\neg(\text{company_name} = "FirstBank")} (\text{works}))$$

Π 欄位錯扣 2 分， σ 限制式或差集錯扣 6 分，資料表錯扣 2 分

4. (10%) Explain the concept of physical data independence and its importance in database systems.

- 實體資料獨立性（Physical Data Independence）是指在不改變邏輯層綱要的情況下，能夠修改資料的實體儲存方式。
- 它在資料庫系統的重要性是因為應用程式依賴於邏輯層綱要，若能達到實體資料獨立性，就能確保在變更實體層綱要時，不會對系統的其他部分產生重大影響

少解釋概念或重要性一項扣4分，說明不完整或錯誤扣1~2分

5. (10%) 請問Oracle資料庫軟體的最新版本為何？並請詳述它三個最主要的特色或優點。

- 目前Oracle資料庫轉體的最新版本為 Oracle Database 23ai。
- 特色有下列幾點：
 - AI Vector Search：資料庫原生支援AI向量的型別在資料庫創建或儲存，並且可以進行向量相似度查詢。
 - JSON Relational Duality：同一份資料可同時以 JSON 文件形式與關聯式表格形

式操作。

- Hybrid Partitioned Tables：系統支援同時使用區間分割和清單分割。

寫錯或沒寫資料庫軟體版本扣2分，沒列出新版本特色少一個扣2分，沒說明新特色少一個扣1分