資料庫管理（113-1）
作業三

作業設計：孔令傑
國立臺灣大學資訊管理學系

繳交作業時，請至 NTU COOL 下載本作業題目的地方上傳一個 PDF 檔。在生成這個 PDF 檔時，可以用打字的也可以用手寫的，但不管怎樣，請務必注意繳交的文件的專業程度（通常透過排版、文字圖片表格方程式的清晰程度、用字遣詞等面向呈現），如果專業程度不夠，也會被酌量扣分。每位學生都要上傳自己寫的解答。不接受紙本繳交。可以用中文或英文作答。這份作業的截止時間是 ***10 月 16 日早上 08:00:00***，遲交在 12 小時內者扣 10 分、在 12 到 24 小時內者扣 20 分、超過 24 小時的這份作業將得不到分數。

相關規定與提醒

1. 關於上網查詢與 AI 工具：

   任何一份作業都可以被用任何方式完成，包括上網搜尋和使用各種 AI 工具。如果你想用，請留意以下幾件事。首先，抄襲還是不被允許的，如果我們發現抄襲（包括抄襲網路上的答案，或是抄襲同學的答案），都還是會給予嚴厲的懲罰（視情節輕重而定，通常是該份作業算零分，或者不予通過這門課）。只要沒有抄襲，那我們就只根據你交上來的答案的品質給分，不論你是自己想出來的，還是有利用 AI 工具。如果某甲善用了 AI 工具後寫得很好，某乙自己努力寫但寫得不好，那某甲會得到比較高分。其次，如大家所知，AI 工具給的答案可能會錯，也可能不合適。使用 AI 工具是學生的自由，但確認 AI 工具的答案是否合適、是否需要調整則是學生的任務。請務必自行確認答案的正確性與合適性。最後還是想提醒大家，學到多少東西都是自己的，如果一時困難用 AI 工具度過難關那是無妨，但之後建議還是花時間把東西學起來，對自己比較好。

2. 關於「專業」:

   在我們這門課及許多課程中，都需要繳交各種報告。一份報告如果要達成他的效果（例如成功募資、完成溝通的任務等等），除了需要好的內容，也需要「專業」，而顯得專業通常需要「長得好看」以及「看起來用心」，這在沒有標準答案的任務上更是如此。有鑑於此，在這門課的作業和專案，我們都會要求報告的格式和美觀，並且納入評分標準。為此，我們提供報告格式參考指南「DB_reportFormatGuideline.pdf」，上面列舉了一份格式良好的報告的最低標準。在作業一我們會請助教就違反參考指南的地方標出來讓大家知道（我們理解那份指南並不是最完美的，但如果完全沒有標準，同學們容易無所適從，所以我們還是設計一份當標準），但不會扣分，只是提醒大家；從作業二起就會有部分分數是報告專業度分數。之所以要求這些不是想要找大家麻煩，而是大家離出社會也不是太遠了，確實應該要開始被要求報告的可讀性和易讀性，所以我們願意花一些時間要求大家，但不會刁難大家，也請大家理解和盡力嘗試了。

3. 關於「批改」：

   如課程大綱所述，為了不要累死助教，每次作業可能只有部分題目會被批改和給予回饋，但每一題的參考解答都會在作業截止後公佈。如果有一題沒被批改，那所有有寫那一題的學生都會得到那一題的全部分數，但沒寫或期限前沒交作業的自然就不會拿到那一題的分數。最後，請注意是「可能」，換言之也有可能是所有題目都被批改。

題目

在本次作業中，有 10 分是依照大家交上來的報告的專業度給分（換言之，格式和美觀再怎麼糟糕，最多也只會扣 10 分）。其餘分數則按照以下題目的正確性和合適性給分。

1. （30 分；一題 5 分）上課時多次使用的 `COMPANY` 資料庫的資料，被放在 COOL 的「頁面」中的 XLSX 檔和 PostgreSQL 資料庫備份檔（兩個檔案中裝有同一份資料）。針對這個資料庫，請幫以下任務撰寫合適的 SQL 指令，每題一個，並附上針對附件的資料執行該指令後得到的查詢結果。

   (a) 針對公司裡所有在跨越兩個以上地點的部門工作的每一位職員，列出他的 SSN、生日，以及他登記在公司的親屬人數。

   (b) 針對公司裡有複數位員工的每一個部門，列出其部門編號、部門名稱、部門內所有員工到 2024/10/1 為止的平均年齡（算實歲且無條件捨去到整數位，例如 2022/10/2 出生的人是 1 歲、2022/10/1 出生的人是 2 歲）。

   (c) 針對公司裡參與人數第二多的專案，列出其專案編號、專案名稱、專案參與人的在該專案的平均工時。如果有 n 個專案的參與人數都是第二多，你的搜尋結果就應該有 n 筆資料。在找「第二多」的時候，假設有數個專案的參與人數並列第一多，則那些專案都算排名第一，而非只有其中一個是第一。舉例來說，如果專案有五個，分別是 A、B、C、D、E，其參與人數分別是 10、8、10、12、12，則人數排名第二的專案是 A 和 C，既不是 D 也不是 E。

   特別說明：針對這一小題，我們推薦大家使用 `LIMIT` 和 `OFFSET` 語法，例如 `LIMIT 1 OFFSET 1` 這樣。建議大家自行上網學習如何使用這些語法，然後在這一題中試試看！

   (d) 針對公司裡有參與任何一個「參與人數第二多的專案」（定義如前一小題）的所有員工，列出他們的 SSN、姓氏、薪資是否達到 40000、所屬部門主管的 SSN、所屬部門主管的姓氏。針對第三個欄位，如果有達到（大於等於）則記錄為「yes」，否則則記錄為「no」。

   (e) 幫公司裡的每一位職員列出其 SSN，並計算其參與的專案數、每週時數總和，以及其參與的專案跨幾個地點。搜尋結果中的每一列應該是一位職員，應該有四欄。如果有職員在某個專案的每週工時是 NULL，則在搜尋時請略過那筆資料（就像是該

職員沒有參與那個專案）。每位職員都應該被列出。如果有一位職員沒有參與任何專案，則該職員的第二、三、四欄都應該要是 0。

特別說明：針對這一小題，我們推薦大家使用 `COALESCE` 語法，建議大家自行上網學習如何使用這樣的語法，然後在這一題中試試看！

(f) 針對沒有下屬的所有職員，列出他們的 SSN 以及參與的專案數。

2.（15 分；每小題 5 分）針對 9/25 課堂上做的線上教育個案，請先把四個資料表中的資料裝進一個 RDBMS，然後在某種 general-purpose 程式語言（例如 Python）中讀取該 RDBMS 中的這四張表的資料，執行以下分析任務：

(a) 我們想知道有哪些使用者在退訂後還是持續在作答（在該公司這是允許的，退訂後只是無法在拿到素養任務，但還是可以做一般題目），並且想看到這些使用者在退訂後不早於 2021/5/1 的所有作答紀錄。我們想列出這些作答紀錄在 `Answers` 資料表中的 `AnswerID`、`UserID`、`QuestionID`、`MissionID`、`IsCorrect`、`CostTime`、`CreatedAT` 欄位和 `Subscriptions` 資料表中的 `EndedAt` 欄位。

請複製你的完整程式碼（以在 Python 上用 `DuckDB` 為例，要包含匯入套件、連資料庫、把資料庫的內容放進 `pandas` 的 Data Frame、把 `pandas` 的 Data Frame 註冊到 `DuckDB` 的 connection 裡、對 `DuckDB` 的 connection 下 SQL、展示結果等等）到你的解答中，並且列出回傳結果的前五筆。

提示：如果老師和助教沒算錯，結果應該有 29248 筆答題記錄。

(b) 承上題，請把前一題得到的作答紀錄彙總起來，幫每位使用者計算他在前一題的作答紀錄中的平均每題所花時間和總答對率，然後將這兩個變數畫成一張散佈圖，橫軸是平均每題所花時間，縱軸是總答對率。在你的散佈圖上，請加一條垂直線代表平均每題所花時間的中位數，再加一條水平線代表總答對率的中位數。對於平均每題所花時間，只要畫 50 秒以內的點就好。

請複製貼上你的完整程式碼，但跟第一小題重複的部份請不要繳交（以在 Python 上用 DuckDB 為例，要包含對 DuckDB 的 connection 下 SQL、將結果繪製成散佈圖、展示結果等等）到你的解答中，並且一併繳交你畫出來的散佈圖。

提示：如果老師和助教沒算錯，結果應該有 342 位使用者。

(c) 承第一小題。為了瞭解 Imperative Programming 和 Declarative Programming 之間的差別，我們現在想比較 Python `pandas` 的兩種寫法。請使用 `pandas` 去完成以下兩個任務：

i. 請寫一段程式碼依序去 (1) merge `Answers` 和 `Subscriptions`、(2) 篩選出 `Answers.CreatedAt` 晚於 `Subscriptions.EndedAt` 且 `Answers.CreatedAt` 不早於 2021/5/1 的作答紀錄、(3) 挑出要留下的欄位、(4) 展示前五筆結果。

ii. 請寫一段程式碼依序去 (1) 在 `Answers` 中篩選出 `Answers.CreatedAt` 不早於 2021/5/1 的作答紀錄、(2) merge 篩選過的 `Answers` 和完整的 `Subscriptions`、

(3) 篩選出 `Answers.CreatedAt` 晚於 `Subscriptions.EndedAt` 的作答紀錄、(4) 挑出要留下的欄位、(5) 展示前五筆結果。

請在你的電腦上執行前述這兩段程式，並且記錄時間（例如用 Python 的 `time` 函式庫）。哪一個程式比較快？你認為是為什麼？合理嗎？請簡要說明你的理由。

3.（30 分；每小題 10 分）你擔任某公司的顧問，到公司服務的第一天，發現他們正想要更新公司的資訊系統，請你看看他們的資料庫設計。請協助他們完成以下任務：

(a) 公司需要所有同仁的出缺勤紀錄，對每一位同仁記錄每一天是否休假，可能為休整天、休上午或休下午，若休整天則同仁整天都不需要有打卡時間紀錄，若休上午則同仁當天理論上應該在 13:00 前打卡抵達公司、17:00 後打卡離開公司，若休下午則同仁當天理論上應該在 8:00 前打卡抵達公司、12:00 後打卡離開公司。每位同仁每次打卡的時間與類型也應該被記錄，包含精確到分的打卡時間以及打卡類型，而類型可能是「抵達公司」或「離開公司」。同仁有時候必須離開公司去執行業務，公司規定要離開公司就必須打卡，回到公司也必須打卡，因此一個同仁一天可能有很多次抵達公司和離開公司的時間。每次要離開公司都需要寫一個理由，是一段寫什麼都可以的字串；如果是單純下班，就可以不填。如果某次離開公司是要去拜訪客戶，必須在資料庫中記錄這次要拜訪的客戶編號，而一次離開公司如果要拜訪數個客戶，該同仁就得要提供全部要拜訪的客戶的資訊，而資料庫就要把這些客戶編號全部記錄下來，並且在未來可以查詢每個客戶的每次被拜訪是被連結到跟哪個同仁在哪一天的哪次離開。最後，一個客戶可以一次有多個同仁一起去拜訪，當然這所有同仁就都應該有相對應的打卡記錄。公司為此設計的 relational schema 如下，其中 `off_type` 可能的值有休整天、休上午、休下午或沒休假、`check_in_time`、`check_out_time`、`check_out_reason` 和 `customer_to_visit` 是 multi-valued attributes：

```
EMPLOYEE_CHECK(employee_id, work_date, off_type,
   check_in_time, check_out_time, check_out_reason,
   customer_to_visit)
```

請幫這家公司正規化這個設計。請簡要說明你的設計是否分別滿足 1NF、2NF、3NF、BCNF 和 4NF 的條件。

(b) 這家公司的其中一個部門提供諮詢與故障排除服務，顧客可以詢問跟產品有關的問題，由某位工程師顧問提供解答，並根據提供之服務的類型收取費用（視情況當然也可能費用是零）。公司為此設計的 relational schema 如下：

```
Consulting(EngineerID, CustomerID, ConsultingDate,
   ServiceType, Fee)
```

依序是工程師編號、顧客編號、諮詢日期、服務類型、費用。已知一個工程師在一天內對一個顧客只會做最多一次諮詢、每次諮詢只會提供一種類型的服務，且一種

類型的服務就會恰好有一個價格（不論工程師、顧客是誰，也不論發生在何時）。請問這個 relation 是 2NF 嗎？如果不是，請把它正規化到 2NF。請問你正規化後的結果是 3NF 嗎？如果不是，請把它正規化到 3NF。不論你做什麼，請簡要地說明原因。

(c) 這家公司跟供應商採購商品然後賣掉。一個供應商可能提供多個商品、不同供應商針對同一個商品給的價格可能不同、一個供應商針對一個商品在不同時期可能設定不同售價。資料庫中需要記錄每個產品的資訊，包括產品編號、產品名稱、產品尺寸（這家公司的產品都是方方正正的，所以產品尺寸只要記錄長、寬、高即可）、該產品可以跟哪個或哪些供應商採購、跟每個供應商採購該商品的單價，以及該單價的適用起始日（這個日期也可能是未來的日期，例如某個供應商宣布未來某一天起單價調整為若干的話，系統就可以儲存這樣的資訊）。歷史上所有的「供應商、產品、單價、單價適用起始日」關係都應該被完整記錄，以便公司可以隨時查詢過往某一天跟某個供應商買某個產品的單價是多少。請幫公司設計一個有合理正規化的 relational schema。如果有必要，可以有多個 relation。請簡要說明你的設計是否分別滿足 1NF、2NF、3NF、BCNF 和 4NF 的條件。

4.（15 分）在上課學到的知識之外，請自行上網查詢補充資訊或詢問 AI 工具，然後用自己的話簡要地回答以下三題。每小題限 300 字。

(a) 用 SQL 對資料庫下 query 以獲得結果，跟用 ORM 連結資料庫獲得結果，各有什麼好處壞處？

(b) 除了 SQL injection 以外，還有什麼常見的針對資料庫的攻擊？請舉出一個，簡要地說明這種攻擊手法以及常用的防範措施。

(c) 在開發資訊系統時，「將前端和後端切開」跟「將 client 端和 server 端切開」是一樣的還是不一樣？如果不一樣，差別為何？