

110年公務人員特種考試司法人員、法務部調查局
調查人員、海岸巡防人員、移民行政人員考試及110年
未具擬任職務任用資格者取得法官遴選資格考試試題

考 試 別：調查人員
等 別：三等考試
類 科 組：資訊科學組
科 目：資料庫應用
考 試 時 間：2 小時

座 號：_____

※注意：(一)禁止使用電子計算器。

(二)不必抄題，作答時請將試題題號及答案依照順序寫在試卷上，於本試題上作答者，不予計分。

(三)本科目除專門名詞或數理公式外，應使用本國文字作答。

一、有一銀行委予資料庫設計，做完系統分析後，得到如下需求 (Requirements)：

- (1)一銀行(BANK)，有銀行代碼(Code)，銀行名稱(Name)和地址(Addr)，銀行代碼用以區別不同的銀行。
- (2)一銀行有許多分行(BANK-BRANCH)，每一分行有地址(BranchAddr)和分行代碼(BranchNo)，分行代碼用以區別不同的分行，一分行不能單獨存在，一定要隸屬於某一銀行，但一銀行可以沒有分行(例如籌備中)。
- (3)一分行可以有許多的帳戶(ACCOUNT)，每一帳戶有帳戶編號(AcctNo)，餘額(Balance)，和帳號種類(Type)，帳戶編號用以區別不同的帳戶，一帳戶必須隸屬於某一分行，但分行可以沒有帳戶。
- (4)一客戶(CUSTOMER)可以擁有多個帳戶(ACCOUNT)，每一客戶有客戶代碼(SSN)，名字(CustName)，手機號碼(Phone)和地址(CustAddr)，客戶代碼用以區別不同的客戶；一帳戶(ACCOUNT)也可以由多個客戶(CUSTOMER)所擁有，一帳戶必須要有客戶所擁有，但客戶可以沒有帳戶。
- (5)一分行可以有許多的貸款(LOAN)，每一筆貸款有貸款編號(LoanNo)，種類(LoanType)和貸款金額(Amount)，每一貸款必須隸屬於某一分行，但分行可以沒有貸款。
- (6)一客戶可以有多筆貸款，一筆貸款也可以由多個客戶來共同借貸，一貸款必須隸屬於一客戶，但客戶可以沒有貸款。

請畫出此銀行資料庫的實體關係圖(ER Diagram)，此實體關係圖裡應包括有強實體型態(Strong Entity Type)與弱實體型態(Weak Entity Type)的設計，並請標示各項限制(Constraints)。(25分)

二、給予一關聯綱要(Relation Schema)EMP-PROJ(SSN, Pnumber, Hours, Ename, Pname, Plocation)，主鍵為(SSN, Pnumber)，此關聯綱要記錄員工的識別碼與名字，與所參與的計畫名稱及計畫所在位置，一個員工可參與多個計畫，每一參與的計畫會記錄參與的時數，此外，此關聯綱要給予一組

功能依附性 (Functional Dependencies) : $\{\{SSN, Pnumber\} \rightarrow \{Hours\}, \{SSN\} \rightarrow \{Ename\}, \{Pnumber\} \rightarrow \{Pname\}, \{Pnumber\} \rightarrow \{Plocation\}\}$, EMP-PROJ 關聯綱要有更改異常 (Update Anomalies) 問題嗎？試舉一例說明插入更改異常 (Insertion Anomalies) , EMP-PROJ 關聯綱要是 2NF 格式嗎？如不是，請將 EMP-PROJ 關聯綱要分割 (Decompose) 到 2NF 的關聯綱要，並論述分割的原理。(25 分)

三、在資料庫處理，交易處理的資料庫存取指令 (Access Operations) 包括那些？何謂交易 (Transactions) ？何謂行程 (Schedules) ？一行程由多個交易組成，一行程的執行會產生那些問題？在資料庫管理系統 (DBMS) 的並行控制和回復機制要強制交易處理達到那些特性 (Desirable Properties of Transactions) ？請論述之。(25 分)

四、給予下列四個行程 A、B、C、D，依優先序圖形 (Precedence Graph) ，請論述那一行程不具序列性 (Serializability)。(25 分)

Schedule A		Schedule B	
T ₁	T ₂	T ₁	T ₂
<i>Time</i>	read_item(Y); Y:=Y-N; write_item(Y); read_item(X); X:=X+N; write_item(X); ↓	<i>Time</i>	read_item(Y); Y:=Y-M; write_item(Y); read_item(X); X:=X+N; write_item(X); ↓
	read_item(Y); Y:=Y+M; write_item(Y);		read_item(Y); Y:=Y-N; write_item(Y); read_item(X); X:=X+N; write_item(X);
Schedule C		Schedule D	
T ₁	T ₂	T ₁	T ₂
<i>Time</i>	read_item(Y); Y:=Y-N; write_item(Y); ↓	<i>Time</i>	read_item(Y); Y:=Y-N; ↓
	read_item(Y); Y:=Y+M; write_item(Y); ↓		write_item(Y); read_item(X); ↓
read_item(X); X:=X+N; write_item(X);		X:=X+N; write_item(X);	write_item(Y);