

1. 您有兩個資料表,分別名為 SalesPerson 與 Sales。您需要確保 Sales 資料表中的每筆記錄都在 SalesPerson 資料表中保存有效的相關 聯銷售人員記錄。請問您應該將哪一個資料庫物件新增至 Sales 資料表?

Quiz:

1. 叢集索引
2. 外部索引鍵
3. 主索引鍵
4. 非叢集索引

Answer: 2

Explanation:

這題的關鍵在於確保兩個資料表之間的「關聯」與「資料完整性」。外部索引鍵 (Foreign Key) 的核心用途,就是建立一個資料表(例如 Sales)對另一個資料表(例如 SalesPerson)的參考,強制參考的資料必須存在。如此一來,可以防止在 Sales 資料表中新增一筆不存在的銷售人員所賣出的訂單,確保了資料的正確性與一致性。(A) 叢集索引是定義資料在磁碟上的實體儲存順序,與資料關聯的正確性無關。(C) 主索引鍵是用來確保「自己資料表」中每一筆資料的唯一性,而非用於關聯兩個不同的資料表。(D) 非叢集索引是用來加速查詢,與資料關聯的正確性無關。在這個案例中,它能確保 Sales 資料表中的每一筆記錄都對應到 SalesPerson 資料表中一個實際存在的銷售人員

2. 您接受了某間地方慈善機構的IT實習工作,該慈善機構的資料模型有兩個資料表,名為 Chapter 和 Language,如下所示。

您建立了名為 ChapterLanguage 的第三個資料表來關聯 Chapter 資料表和 Language 資料表。您需要從 Chapter 和 Language 資料表中選取資料行,以便為 ChapterLanguage 資料表建立複合主索引鍵。請問您應該選取哪兩個資料行?(請選擇2個答案)

Quiz:

1. ChapterId
2. City
3. Region
4. Country
5. LanguageId
6. LanguageName

Answer: 1, 5

Explanation:

複合主索引鍵 (Composite Primary Key) 是由兩個或多個欄位組成的索引鍵,用於唯一識別資料表中的每一筆記錄。為了關聯 Chapter 和 Language 這兩個資料表,我們需要選取它們各自的主索引鍵,也就是 ChapterId 和 LanguageId,作為新資料表 ChapterLanguage 的複合主索引鍵。

3. 您正在設定關聯式資料庫中某個資料表的結構。對於下列每一項敘述,若[正確]為○,[錯誤]為X。

①資料表中欄位的每個值都必須是唯一的。②資料表中的每個資料列都必須是唯一的。③資料表中的每個資料行名稱都必須是唯一的。請問填入正確或錯誤符號順序為何?

Quiz:

1. X、X、○
2. X、○、X
3. X、○、○
4. ○、○、X

Answer: 3

Explanation:

①錯誤 (X): 資料表中欄位的每個值不一定是唯一的。只有主索引鍵或設定唯一條件約束的欄位才需要是唯一的。 ②正確 (○): 在關聯式資料庫中，每個資料列 (Row) 必須是唯一的，這是由主索引鍵 確保的。 ③正確 (○): 資料表中的每個資料行 (Column) 名稱必須是唯一的，以避免在查詢和 操作時產生歧義。

4. 您的資料庫有一個 Department 資料以及一個 Employee 資料表。您需要確保一位員工只能被借派到一個現有的部門。 請問您應該對 Employee 資料表套用什麼？

Quiz:

1. 資料類型
2. 外部索引鍵
3. 索引
4. 主索引鍵
5. 唯一條件約束

Answer: 2

Explanation:

為了確保 Employee 資料表中的員工記錄只能關聯到 Department 資料表中一個現有的部門，需要在 Employee 資料表上建立一個外部索引鍵，該索引鍵參照 Department 資料表的主索引鍵。這會強制實施部門與員工之間的一對多關聯性，確保每位員工都分配給一個有效的 部門。 CH02

1. 您有一個名為Customer 的資料表。您需要建立一個名為 District 的新資料行。 請問您應該使用哪一個陳述式？

Quiz:

1. ALTER TABLE Customer ADD (INTEGER District)
2. ALTER TABLE Customer ADD (District INTEGER)
3. MODIFY TABLE Customer ADD (INTEGER District)
4. ALTER TABLE Customer MODIFY (District INTEGER)

Answer: 2

Explanation:

ALTER TABLE 陳述式用於修改現有的資料表結構，而 ADD 子句則用於新增欄位。正確的語法是 ALTER TABLE table\_name ADD (column\_name data\_type)。

2. 您有一個名為Product 的資料表。您建立了一個檢視表,其中包含 Product 資料中 屬於 Furniture 類的所有產品。 您針對 Product 資料表執行一個陳述式來刪除屬於 Furniture 類別的所有產品。 在您執行該陳述式之後,檢視表的結果集將:

Quiz:

1. 被封存
2. 被刪除
3. 清空
4. 保持不變

Answer: 3

Explanation:

檢視表 (View) 是一個虛擬資料表，它不儲存實際資料，而是根據其定義的查詢動態產生結果集。當您從基礎資料表 Product 中刪除所有 Furniture 類別的產品時，檢視表在下次被查詢時將找不到任何符合條件的資料，因此結果集將會清空。

3. 函數與預存程序之間的其中一項差異在於函數:

Quiz:

1. 必須有傳回值
2. 無法接受參數
3. 無法包含交易
4. 必須由觸發程序呼叫

Answer: 1

Explanation:

函數 (Function) 必須傳回一個值，而預存程序 (Stored Procedure) 則不一定需要。預存程序可以執行多個 SQL 陳述式，並可以傳回多個結果集或沒有任何結果。

4. 您有一個名為Employee 的資料表,其中包含下列資料行: EmployeeID、EmployeeName。請問您應該使用哪一個陳述式來傳回資料表中的資料列數目?

Quiz:

1. SELECT COUNT(rows) FROM Employee
2. SELECT COUNT(\*) FROM Employee
3. SELECT \* FROM Employee
4. SELECT SUM(\*) FROM Employee

Answer: 2

Explanation:

COUNT(\*) 是一個聚合函數，用於計算資料表中的所有資料列數目，包括包含 NULL 值的資料列。COUNT(rows) 和 SUM(\*) 不是標準的 SQL 語法，SELECT \* FROM Employee 則是返回所有資料列和欄位，而不是資料列的數量。

5. 您刪除了order資料表中的資料列,orderItem 資料表中的對應資料列也被自動刪除了。這是下列其中一種程序的範例:

Quiz:

1. 串聯刪除 (Cascade Delete)
2. 骨牌法刪除 (Dominoe Delete)
3. 函數法刪除 (Functional Delete)
4. 繼承法刪除 (Inherited Deete)

## 5. 瀑布法刪除 (Waterfall Delete)

Answer: 1

Explanation:

串聯刪除 (Cascade Delete) 是外部索引鍵約束的一個屬性。當我們在定義外部索引鍵時，可以設定 ON DELETE CASCADE。這表示當主資料表（父資料表，如 order）中的一筆資料被刪除時，參考到這筆資料的從屬資料表（子資料表，如 orderItem）中的所有對應資料列也會被自動刪除，以維持資料的參考完整性。

## 6. 請問哪一個陳述式會建立複合鍵？

Quiz:

1. CREATE TABLE Order ( OrderID INTEGER, OrderItemID INTEGER, PRIMARY KEY (OrderID, OrderItemID))
2. CREATE TABLE Order ( OrderID INTEGER PRIMARY KEY, Order Itemid INTEGER PRIMARY KEY)
3. CREATE TABLE Order ( OrderID INTEGER, OrderItemID INTEGER, PRIMARY KEY)
4. CREATE TABLE Order ( OrderID INTEGER, OrderItemID INTEGER, PRIMARY KEY OrderID, PRIMARY KEY OrderitemID)

Answer: 1

Explanation:

複合鍵 (Composite Key) 是指由兩個或多個欄位共同組成的主索引鍵。其建立語法是在所有欄位定義完之後，使用 PRIMARY KEY (欄位1, 欄位2, ...) 的形式來指定。選項 (A) 正確地使用了此語法。(B) 和 (D) 的語法是錯誤的，一個資料表只能有一個主索引鍵。(C) 的語法不完整，PRIMARY KEY 後面沒有指定任何欄位。

## 7. 一位開發人員嘗試使用下列查詢,在資料表中新增一個名為Prefix 資料行: ALTER TABLE Person ADD Prefix varchar(4) NOT NULL;

這位開發人員在執行查詢之後收到錯誤訊息。但是,這位開發人員無法記住確切的錯誤訊息。請問這個問題的可能原因為何？

Quiz:

1. 應該使用DEFAULT 關鍵字來指定預設值
2. varchar(4)資料類型不適用於Person 資料表
3. 應該先執行 DROP CONSTRAINT 查詢,再執行 ALTER TABLE 查詢
4. Person 資料表是空的

Answer: 1

Explanation:

當一個已經存在資料的資料表要新增一個設定為 NOT NULL (不允許空值) 的欄位時，資料庫會遇到一個問題：對於那些已經存在的資料列，這個新欄位的值應該是什麼？因為不允許是 NULL，所以必須為這些舊資料列提供一個預設值。如果沒有使用 DEFAULT 關鍵字指定一個預設值，這個 ALTER TABLE 操作就會失敗。如果資料表是空的，這個操作則會成功。

## 8. 您在某間咖啡廳工作,公司要求您設定網站來儲存購物費用。您需要建議公司在資料庫資料表中使用某種資料類型,以便根據收費金額執行財務函數。請問您應該使用哪一種資料類型？

Quiz:

1. binary
2. bit
3. decimal
4. varchar

Answer: 3

Explanation:

在處理金錢、貨幣等需要高精確度的數值時，應該使用 decimal (或 numeric, money) 這類固定小數點的資料類型。這可以避免使用 float 等浮點數資料類型時可能發生的微小誤差，確保財務計算的準確性。(A) binary 用於儲存二進位資料。(B) bit 用於儲存 0 或 1 的布林值。(D) varchar 用於儲存字串。

9. 您正在建立一個名為Student 的資料庫物件儲存以下資料:

請問您應該使用哪一個語法來建立物件?

Quiz:

1. CREATE TABLE Student( ID INT, Name VARCHAR (100), Age INT)
2. CREATE Student ( ID INT, Name VARCHAR (100), Age INT)
3. CREATE ( TABLE StudentID INT, Name VARCHAR (100), Age INT)
4. CREATE TABLE ( ID INT, Name VARCHAR (100), Age INT)

Answer: 1

Explanation:

建立資料表的標準 SQL 語法是 CREATE TABLE 資料表名稱 (欄位1 資料型態, 欄位2 資料型態, ...);。選項 (A) 完全符合這個語法結構。(B) 缺少了 TABLE 關鍵字。(C) 和 (D) 的括號位置不正確。

10. 您需要利用下面的程式碼設定一個資料庫來提供北美哺乳動物的檢視表 (NorthAmericaMammals View): ① ② ③ WHERE a.Class = 'Mammals' AND a.InNorthAmerica = 1 請將下方A~F的程式碼片段排列到上方缺失的正確位置。每個程式碼片段可能只使用一次,也可能使用多次,甚至完全用不到。 A. CREATE VIEW [dbo].[NorthAmericaMammals\_View] B. INSERT VIEW [dbo].[NorthAmericaMammals\_View] C. AS JOIN a.Id, a. Name D. AS SELECT a.Id, a.Name E. FROM Animal a F. JOIN Animals a 請問填入的程式碼片段順序為何?

Quiz:

1. A、 C、 F
2. A、 D、 E
3. B、 D、 E
4. B、 C、 F

Answer: 2

Explanation:

建立檢視表的完整語法是 CREATE VIEW 檢視表名稱 AS SELECT 欄位... FROM 資料 表... WHERE 條件...;。因此，正確的順序是：

1. CREATE VIEW [dbo].[NorthAmericaMammals\_View] (A)

Answer:

2. AS SELECT a.Id, a.Name (D)

Answer:

3. FROM Animal a (E)

Answer:

4. WHERE a.Class = 'Mammals' AND a.InNorthAmerica = 1 (題目已給定) 所以組合起來是 A、D、E。

Answer:

11. 對於下列每一項敘述,若[正確]為○,[錯誤]為X。① 您可以使用預存程序來刪除資料。② 函數必須具有傳回值。③ 預存程序必須具有傳回值。請問填入正確或錯誤符號順序為何?

Quiz:

1. X、○、○
2. ○、○、X
3. X、X、○
4. ○、○、X

Answer: 4

Explanation:

① 正確(○): 預存程序是一系列的 SQL 陳述式集合,當然可以包含 DELETE 陳述式 來刪除資料。② 正確(○): 這是函數與預存程序最主要的區別之一。函數被設計來計算並傳回一個 值。③ 錯誤(X): 預存程序「可以」透過輸出參數或傳回碼來傳回值,但並非「必須」。它可以只執行操作而不傳回任何東西。因此,順序為 ○、○、X。

12. 請問哪一個陳述式會從資料表中移除所有資料列,但不記錄個別資料列刪除動作?

Quiz:

1. ALTER TABLE
2. DROP TABLE
3. TRUNCATE TABLE
4. CREATE TABLE

Answer: 3

Explanation:

TRUNCATE TABLE 指令用於快速刪除一個資料表中的所有資料列。與 DELETE 不同，TRUNCATE 通常是一個最小化日誌記錄的操作，它不會為每一筆被刪除的資料列寫入交易日誌，因此執行速度非常快。但這也意味著它通常不能被輕易地復原 (rollback)，且不會觸發 DELETE 觸發器。(A) ALTER TABLE 用於修改資料表結構。(B) DROP TABLE 會刪除整個資料表，包括其結構和資料。

13. 請問關聯式資料庫會使用哪一項功能來確保輸入資料行中的資料有效？

Quiz:

1. 屬性
2. 主索引鍵
3. 條件約束
4. 索引

Answer: 3

Explanation:

條件約束 (Constraint) 是資料庫用來強制執行資料完整性規則的主要機制。它包含 了多種類型，例如：\* NOT NULL: 確保欄位不能是空值。\* UNIQUE: 確保欄位中的值是唯一的。\* PRIMARY KEY: 主索引鍵約束 (是 NOT NULL 和 UNIQUE 的結合)。\* FOREIGN KEY: 外部索引鍵約束，確保參考完整性。\* CHECK: 確保欄位中的值符合指定的條件。這些約束共同確保了輸入資料的有效性。

14. 您需要從資料庫中移除名為Employeeview 的檢視表。請問您應該使用哪一個 陳述式？

Quiz:

1. DELETE EmployeeView
2. DELETE VIEW EmployeeView
3. DROP EmployeeView
4. DROP VIEW EmployeeView

Answer: 4

Explanation:

檢視表、資料表、索引等都屬於資料庫的「物件」。要移除這些物件，應使用 DDL (資料定義語言) 中的 DROP 指令。其標準語法為 DROP VIEW 檢視表名稱；。(A) 和 (B) 的 DELETE 是 DML (資料操作語言)，用於刪除資料表中的「資料列」，而不是物件本身。

15. 建立預存程序的其中一個原因是要:

Quiz:

1. 提升效能
2. 將儲存空間最小化
3. 略過區分大小寫需求
4. 讓使用者能夠控制查詢邏輯

Answer: 1

Explanation:

預存程序 (Stored Procedure) 有多個優點，其中最主要的一個是「提升效能」。因為預存程序是預先編譯好並儲存在資料庫伺服器上，當被呼叫時，資料庫無需重新 解析和編譯 SQL 陳述式，減少了網路傳輸量和伺服器的處理負擔，從而提升了執行 效率。



16. 請問您應該使用哪一個陳述式來移除外部索引鍵?

Quiz:

1. ALTER TABLE
2. DELETE TABLE
3. ALTER FOREIGN KEY
4. DELETE FOREIGN KEY

Answer: 1

Explanation:

外部索引鍵是一個資料表的「條件約束」，屬於資料表結構的一部分。要修改資料表的結構（包括新增、刪除或修改條件約束），都必須使用 ALTER TABLE 陳述式。通常的語法會是 ALTER TABLE 資料表名稱 DROP CONSTRAINT 約束名稱。

17. 您的公司將客戶的社會安全號碼儲存在 Customers 資料表的SSN 資料行中。但是,新的法規禁止公司儲存這些資訊。執行下列查詢會導致系統發生錯誤。 ALTER TABLE Customers REMOVE SSN; 請問您需要對上述查詢進行哪些變更,才能讓它從Customers 資料表中移除 N資料行?

Quiz:

1. ALTER TABLE Customers DROP COLUMN SSN;
2. ALTER TABLE Customers DELETE COLUMN
3. ALTER TABLE Customers DROP SSN;
4. ALTER TABLE Customers DELETE SSN;

Answer: 1

Explanation:

在標準 SQL 中，要從資料表中移除一個資料行，應使用 ALTER TABLE 陳述式搭配 DROP COLUMN 子句。選項 (A) 是最標準且正確的語法。REMOVE 或 DELETE 都不是用來刪除資料行的標準關鍵字。

18. 您需要刪除某個資料庫資料表。請問您應該使用哪一個資料定義語言(DDL) 關鍵 字?

Quiz:

1. ALTER
2. TRUNCATE
3. DROP
4. DELETE

Answer: 3

Explanation:

DDL (資料定義語言) 用於定義或管理資料庫物件的結構。要「刪除整個資料表」（包括其結構、資料和相關索引），應使用 DROP 關鍵字。語法為 DROP TABLE 資料表名稱;。(A) ALTER 用於修改結構。(B) TRUNCATE 用於刪除所有資料，但保留結構。(D) DELETE 是 DML，用於刪除資料列，同樣保留結構。



19. 有一個資料庫資料表儲存了學校出勤率的相關資訊,資料集定義如下:

您有下列需求: \* StudentName 必須包含字元字串。 \* GradeLevel 必須是整數。 \* DaysAbsent可以在小數點後面包含一位數字。 可使用的資料類型是:

Quiz:

1. BIT
2. DECIMAL
3. VARCHAR
4. DATETIME
5. INT
6. CHAR StudentName、GradeLevel 及DaysAbsent 的資料類型為何?
7. C、E、B
8. C、B、E
9. F、E、B
10. A、B、B

Answer: 1

Explanation:

\* StudentName: 儲存姓名, 是字元字串, 應使用 VARCHAR (C) 或 CHAR (F)。 VARCHAR 更常用, 因為它是可變長度。 \* GradeLevel: 儲存年級, 是整數, 應使用 INT (E)。 \* DaysAbsent: 儲存缺席天數, 需要支援小數 (如 2.5), 應使用 DECIMAL (B)。 組合起來, 最合理的選擇是 VARCHAR, INT, DECIMAL, 對應到 (C), (E), (B)。所以 答案是 (A)。

20. 您正在開發一個SQL 陳述式以建立資料表。請問哪兩個 SQL 關鍵字可用於 CREATE TABLE 陳述式?(請選擇2個答案)

Quiz:

1. ORDER BY
2. CONSTRAINT
3. PRIMARY KEY
4. INSERT INTO

Answer: 2, 3

Explanation:

在 CREATE TABLE 陳述式中, 我們定義資料表的欄位以及確保資料完整性的規則。 \* CONSTRAINT (B) 是用來定義各種約束的通用關鍵字, 例如 CHECK, UNIQUE, PRIMARY KEY, FOREIGN KEY。 \* PRIMARY KEY (C) 是最重要的一種約束, 可以直接在欄位後定義, 也可以透過 CONSTRAINT 關鍵字來定義。 (A) ORDER BY 用於 SELECT 查詢的排序。 (D) INSERT INTO 用於新增資料。

21. 您建立了一個名為 Games 的資料,其中包含最近發行之電玩遊戲的評論分數。您需要建立一個檢視表來傳回按照字母順序排列的遊戲名稱清單。Name 代表遊戲名稱。請問您應該使用哪一個查詢?

Quiz:

1. CREATE VIEW MyGames AS SELECT Name FROM Games
2. CREATE VIEW MyGames AS SELECT \* FROM Games
3. CREATE VIEW MyGames AS SELECT Name FROM Games ORDER BY Name

4. CREATE VIEW MyGames AS SELECT \* from Games WHERE Name BETWEEN 'A' AND 'Z'

Answer: 3

Explanation:

題目的要求是「按照字母順序排列」。在 SQL 中，要對查詢結果進行排序，必須使用 ORDER BY 子句。因此，在建立檢視表的 SELECT 陳述式中，需要包含 ORDER BY Name。

22. 您撰寫了一系列執行冗長複雜查詢的SQL 陳述式。這些陳述式會透過電子郵件 傳送結果。您希望能夠在需要時隨時手動呼叫程式碼。請問您可以使用哪一個資料庫物件來儲存程式碼？

Quiz:

1. 預存程序
2. 函數
3. 檢視表
4. 觸發程序

Answer: 4

Explanation:

預存程序 (Stored Procedure) 就是被設計來儲存一系列可重複使用的 SQL 陳述式。

它可以接受參數、執行複雜的邏輯、並可由使用者或應用程式「手動呼叫」。這完全符合題目的需求。(B) 函數主要用於傳回單一值。(C) 檢視表主要用於簡化查詢。(D) 觸發程序是自動執行的，不能手動呼叫。

CH03 1. 您有兩個資料表，每個資料表都有三個資料列。建立了一個使用交叉聯結的 SQL 查詢，這個查詢不包含WHERE子句。請問這兩個資料表的笛卡兒乘積會包含多少個資料列？(A) 0 (B) 3 (C) 6 (D) 9

交叉聯結 (CROSS JOIN) 會產生兩個資料表的「笛卡兒乘積」。這意味著第一個資料表的每一筆資料列，都會與第二個資料表的每一筆資料列進行配對。

因此，結果集的總資料列數會是第一個資料表的資料列數「乘以」第二個資料表的資料列數。在這個例子中，就是  $3 * 3 = 9$ 。

2. 您執行了下列查詢: SELECT EmployeeID, FirstName, DepartmentName FROM Employee, Department 這種作業稱為:

Quiz:

1. 笛卡兒乘積 (B) 等聯結
2. 交集 (D) 外部聯結

Answer: 1

Explanation:

在 FROM 子句中直接列出多個資料表，而沒有使用 WHERE 或 JOIN...ON 子句來指定它們之間的關聯條件時，資料庫會執行交叉聯結，產生笛卡兒乘積。這是比較舊的 SQL-89 語法，現在建議使用明確的 CROSS JOIN 關鍵字。

3. 請問哪一個查詢會傳回除了加州(CA)以外,其他所有州在2017年1月之後提交之單 的結果集?

Quiz:

1. SELECT \* FROM orders WHERE order\_date > '2017-01-01' AND ship\_state LIKE 'CA'
2. SELECT \* FROM orders WHERE order\_date > '2017-01-01' AND ship\_state <> 'CA'
3. SELECT \* FROM orders WHERE order\_date > '2017-01-01' OR ship\_state <> 'CA'
4. SELECT \* FROM orders WHERE order\_date > '2017-01-01' OR ship\_state LIKE 'CA'

Answer: 2

Explanation:

題目要求兩個條件「同時」成立：

1. 訂單日期在 '2017-01-01' 之後 (order\_date > '2017-01-01')。

Answer:

2. 運送州別「不是」CA (ship\_state <> 'CA')。<> 是 SQL 中「不等於」的運算子。  
因為兩個條件要同時滿足，所以必須使用 AND 來連接它們。

Answer:

4. 您需要針對 students 資料表中的資料建立一份報表。此報表必須符合下列需求:

A.顯示在2020年6月1日當天或之後註冊的所有學生 B.顯示在2020年畢業的所有學生

C.按照註冊順序傳回結果集,最新的註冊日期列在第一個 enrollment\_date

會指定註冊日期。graduation\_date 會指定畢業日期。如果 academic\_status 是 Graduated,就表示學生畢業。請問您應該使用哪一個查詢?

Quiz:

1. SELECT \* FROM students WHERE enrollment\_date >= '2020-06-01' OR academic\_status='Graduated' OR graduation\_date >= '2020-01-01' ORDER BY enrollment\_date DESC
2. SELECT \* FROM students WHERE enrollment\_date >= '2020-06-01' AND academic\_status='Graduated' AND graduation\_date >= '2020-01-01' ORDER BY enrollment\_date ASC
3. SELECT \* FROM students WHERE enrollment\_date >= '2020-06-01' OR academic\_status='Graduated' AND graduation\_date >= '2020-01-01' ORDER BY enrollment\_date DESC
4. SELECT \* FROM students WHERE enrollment\_date >= '2020-06-01' AND academic\_status='Graduated' OR graduation\_date >= '2020-01-01' ORDER BY enrollment\_date

Answer: 3

Explanation:

這題的需求是 A「或」B，也就是符合 A 或 B任一條件的學生都要顯示，所以要用 OR。條件是 enrollment\_date >= '2020-06-01' OR (academic\_status='Graduated' AND graduation\_date >= '2020-01-01')。同時，要求最新的註冊日期排在最前面，所以要用 ORDER BY enrollment\_date DESC (DESC 代表遞減排序)。選項 (C) 的邏輯最符合這些要求。

5. 請問哪一個SQL陳述式會傳回訂單數目少於50的國家地區以及各國家地區的訂單 數目? orderID 代表個別訂單、Country 代表國家地區,而orders 代表訂單數目。

Quiz:

1. SELECT COUNT(orderID), Country FROM Orders GROUP BY Country HAVING COUNT(orderID) < 50;

2. SELECT Country, orderID FROM Orders WHERE COUNT(orderID) < 50; GROUP BY Country;
3. SELECT COUNT(orderID), Country FROM Orders HAVING COUNT(orderID) < 50; GROUP BY Country
4. SELECT Country, orderID FROM Orders GROUP BY Country WHERE COUNT(orderID) < 50;

Answer: 1

Explanation:

這題的關鍵是要對「彙總函式的結果」進行篩選。WHERE 子句是在分組和彙總「之前」對個別資料列進行篩選。而 HAVING 子句則是在 GROUP BY 分組和彙總「之後」對分組結果進行篩選。因為我們要篩選的是 COUNT(orderID) < 50, 所以必須使用 HAVING 子句。

6. 有一個名為 Playerstat 資料表包含以下欄位:

您需要針對 TeamID 為1的隊伍顯示每位隊員的總分數,程式碼如下: SELECT PlayerID, (1) FROM PlayerStat (2) TeamID = 1 (3) (4) 請將下方A~G的程式碼片段排列到上方缺失的正確位置。每個程式碼片段可能只使用一次,也可能使用多次,甚至完全用不到。 A. PlayerID B. COUNT C. GROUP BY D. TeamID E. WHERE F. SUM(Points) G. HAVING 請問填入的程式碼片段順序為何?

Quiz:

1. F、C、D、B
2. D、C、E、F
3. F、E、C、A
4. B、E、F、D

Answer: 3

Explanation:

這個查詢的目的是:

1. 計算總分, 所以 (1) 應填入 SUM(Points) (F)。

Answer:

2. 篩選出 TeamID 為 1 的隊伍, 所以 (2) 應填入 WHERE (E)。

Answer:

3. (3) 已經給定條件 TeamID=1。

Answer:

4. 要「每位」隊員的總分，代表要根據 PlayerID 進行分組，所以 (4) 應填入 GROUP BY PlayerID (C, A)。組合起來的順序是 F, E, C, A

Answer:

7. 您是某間大學的資料庫管理員,該大學將學生資訊儲存在一個名為students 的資料 中。這個資料表具有一個名為 courseID 的外部索引鍵,可連結至 courses 資料表中的主 索引鍵。您需要使用這兩個根據 courseID 配對的資料表來傳回學生姓名和課程名稱。 您執行了下列查詢: SELECT students.name, courses.name FROM students OUTER JOIN courses WHERE student.courseID = courses.courseID 此查詢會傳回錯誤。 請問您該如何修正查詢?

Quiz:

1. SELECT students.Name, courses.Name FROM students INNER JOIN ON courses WHERE students.courseID = courses.CourseID
2. SELECT students.name, courses.name FROM students INNER JOIN courses ON students.courseID = courses.courseID
3. SELECT students.name, courses.name FROM students INNER JOIN ON students.courseID = courses.courseID
4. SELECT students.name, courses.name FROM students INNER JOIN courses WHERE students.courseID = courses.courseID

Answer: 2

Explanation:

現代的 SQL (SQL-92 標準) 建議使用明確的 JOIN 語法。JOIN 的關聯條件應該使用 ON 關鍵字來指定，而不是放在 WHERE 子句中。OUTER JOIN 是一個不完整的語 法，應該明確指定是 LEFT OUTER JOIN, RIGHT OUTER JOIN 還是 FULL OUTER JOIN。但此題的重點在於 JOIN 的語法結構，正確的結構是 FROM table1 JOIN table2 ON table1.key = table2.key。選項 (B) 是唯一符合此結構的。

8. 有一個預存程序包含以下查詢: SELECT 'Greetings' + Prefix + FirstName FROM Person; 此預存程序會傳回所有 null值。您確認 Person 資料表包含資料。 請問這個問題的可能原因為何?

Quiz:

1. Prefix 或FirstName 資料行具有null值。
2. 您必須在查詢中 指定 NULLIF 關鍵字。
3. 加號(+)運算子無法用來附加字元資料。
4. 您必須在SELECT 陳述式中指定 JOIN 關鍵字。

Answer: 1

Explanation:

在大多數的 SQL 資料庫系統中 (如 SQL Server) , 當你使用 + 運算子來串接字串 時, 只要其中任何一個部分是 NULL, 整個串接的結果就會是 NULL。因此, 如果 Prefix 或 FirstName 欄位中有任何一筆資料是 NULL, 該筆資料列的查詢結果就會 是 NULL。

9. 請問哪一個關鍵字會結合兩個查詢的結果並且僅傳回兩個結果集都有出現的資料 列?

Quiz:

1. UNION
2. INTERSECT
3. ALL
4. JOIN

Answer: 2

Explanation:

INTERSECT (交集) 運算子會比較兩個 SELECT 陳述式的結果集，並只傳回在「兩個」結果集中都存在的資料列。它會自動移除重複的資料列。(A) UNION (聯集) 會傳回兩個結果集中的所有資料列，並移除重複項。

10. 有一個名為Cars 的資料庫資料表,其定義如下所示。

您設有下列SQL 陳述式: SELECT \* FROM Cars c WHERE c. Origin <> 'USA' AND c.Color <> 'Black' 請問此 SQL 陳述式會傳回多少個資料列?

Quiz:

1. 4
2. 5
3. 6
4. 7

Answer: 1

Explanation:

我們來逐步篩選:

1. 原始資料共 8 筆。

Answer:

2. 第一個條件 c.Origin <> 'USA' (產地不是美國) 會排除掉 Truck 和 Convertible。 剩下 6 筆。

Answer:

3. 第二個條件 c.Color <> 'Black' (顏色不是黑色) 會從剩下的 6 筆中，再排除掉 Compact 和 Hybrid。 剩下 4 筆。 因為兩個條件是用 AND 連接，所以必須同時滿足。最終結果是 4 筆。

Answer:

11. 您需要將兩個查詢的結果結合成單一結果,而且包含這兩個查詢中的所有資料列。  
請問您應該使用哪一個SQL述式?

Quiz:

1. EXCEPT
2. JOIN
3. TRUNCATE
4. UNION

Answer: 4

Explanation:

UNION (聯集) 運算子就是用來將兩個或多個 SELECT 陳述式的結果集合併成一個單一的結果集。UNION 預設會移除重複的資料列。如果想要保留所有重複的資料列，可以使用 UNION ALL。

12. 您有一個儲存客戶相關資訊的 customer 資料表,以及一個儲存已提交訂單相關 資訊的 order 資料表。 每個資料表都包含一個customerID 資料行,您使用個別資料表中的Customer 資料行來聯結這兩個資料表。 您執行了一個選取下列資料的查詢: \* 所有客戶及其訂單 \* 沒有任何訂單的客戶 請問這些結果代表哪一種聯結?

Quiz:

1. 完全聯結
2. 內部聯結
3. 完全外部聯結
4. 部分聯結

Answer: 3

Explanation:

這個描述最精確地對應到 LEFT OUTER JOIN (左外部聯結), 假設 customer 是左邊的資料表。它會傳回所有左邊資料表(customer)的紀錄, 以及右邊資料表(order)中匹配的紀錄。如果某個客戶沒有訂單, 則訂單相關的欄位會顯示為 NULL。而 FULL OUTER JOIN (完全外部聯結) 會包含 LEFT JOIN 的所有結果, 再加上「有訂單但沒有對應客戶」的紀錄 (在有外部索引鍵約束下, 這種情況不應發生)。在此選擇題 中, 完全外部聯結是最接近且包含所需結果的選項。

13. 您 其中包含名為CustomerID、Firstname 和 Date Joined 資料列。 CustomerID 是主索引鍵。 您執行了下列查詢: SELECT CustomerID, FirstName, Date Joined FROM Customer 請問結果集會以哪種方式組織資料列?

Quiz:

1. 按照無法預測的順序
2. 按照 FirstName 的字母序
3. 按照 DateJoined 的時間序
4. 按照插入資料列的順序

Answer: 1

Explanation:

關聯式資料庫的理論基礎是集合論, 集合中的元素是沒有固定順序的。因此, 如果一個 SELECT 查詢沒有明確使用 ORDER BY 子句來指定排序方式, 資料庫回傳資料的順序是「不被保證的」。它可能會因為查詢計畫、索引、或資料的實體儲存方式而改變, 所以是無法預測的。



14. 您正在撰寫一個 SELECT 陳述式找出名稱包含特定字元的所有產品。請問您應該在 WHERE 子句中使用哪一個關鍵字？

Quiz:

1. LIKE
2. FETCH
3. BETWEEN
4. IN

Answer: 1

Explanation:

LIKE 運算子專門用於字串的比對，可以搭配萬用字元來進行模糊搜尋。\* %: 代表零個或多個任意字元。\* \_ : 代表一個任意字元。例如 WHERE ProductName LIKE '%apple%' 可以找出名稱中包含 'apple' 的所有產品。

15. 請問哪一個查詢會正確傳回 ship\_state 不含德州 (TX) 和亞利桑那州 (AZ) 之所有訂單的結果集？

Quiz:

1. SELECT \* FROM Orders WHERE NOT ship\_state = 'TX' OR NOT ship\_state = 'AZ'
2. SELECT \* FROM Orders WHERE NOT ship\_state = 'TX' AND NOT ship\_state = 'AZ'
3. SELECT \* FROM Orders WHERE ship\_state NOT = 'TX' OR ship\_state NOT = 'AZ'
4. SELECT \* FROM Orders WHERE ship\_state NOT = 'TX' AND ship\_state NOT = 'AZ';

Answer: 2

Explanation:

要同時排除 TX 和 AZ，條件必須是「州不等於 TX」『而且』「州不等於 AZ」。所以必須使用 AND 來連接這兩個條件。如果使用 OR，那麼德州的訂單會因為「不等於 AZ」而滿足條件被選出來，反之亦然，這就不對了。正確的邏輯是 WHERE ship\_state <> 'TX' AND ship\_state <> 'AZ'，或是 WHERE ship\_state NOT IN ('TX', 'AZ')。選項 (B) 的 NOT ... AND NOT ... 在邏輯上是正確的。CH04

1. 您有一個名為 Student 的資料表,其中包含100個資料列。部分資料列 Firstname 資料行具有NULL值,您執行了下列陳述式: DELETE FROM Student 請問結果為何？

Quiz:

1. 您會收到錯誤訊息
2. 系統將刪除資料表中的所有資料列
3. 系統將刪除所有資料列和資料表定義
4. 系統將刪除 Firstname 資料行含有NULL 值的所有資料列

Answer: 2

Explanation:

DELETE FROM [資料表名稱] 陳述式如果沒有加上 WHERE 子句來指定條件，它的預設行為就是刪除該資料表中的「所有」資料列。這個操作不會影響資料表的結構 定義。

2. 您的資料庫包含一個名為 Customer 的資料表。您需要從 Customer 資料表中刪除 CustomerID 為 12345 的記錄。請問您應該使用哪一個陳述式？

Quiz:

1. DELETE FROM Customer WHERE CustomerID = 12345
2. DELETE Customer ID FROM Customer WHERE CustomerID = 12345
3. UPDATE Customer DELETE \* WHERE CustomerID = 12345
4. UPDATE Customer ID FROM Customer DELETE \* WHERE CustomerID = 12345

Answer: 1

Explanation:

選項 (A) 完全符合這個語法。刪除特定資料列的標準 SQL 語法是 DELETE FROM [資料表名稱] WHERE [條件];。 (B) DELETE 後面不需指定欄位名稱。 (C) 和 (D) 的語法完全錯誤，混用了 UPDATE 和 DELETE。

3. 您有一個名為 Product 的資料表包含以下資料。

ProductID 資料行是主索引鍵。CategoryID 資料行是名為Category 之個別資料表的外部索引鍵。您執行了下列陳述式: INSERT INTO Product VALUES (3296, 'Table', 4444) 請問結果為何?

Quiz:

1. 語法錯誤
2. Product 資料表中的新資料列
3. 外部索引鍵條件約束違規
4. 主索引鍵條件約束違規
5. Category 資料表中的新資料列

Answer: 4

Explanation:

主索引鍵 (Primary Key) 的最重要特性就是它的值在資料表中必須是「唯一的」。資料表中已經存在一筆ProductID為 3296 的紀錄。當我們試圖再INSERT (插入) 一筆ProductID 同樣為 3296 的新紀錄時，就違反了主 索引鍵的唯一性約束，資料庫會拒絕此操作並回報錯誤。

4. 您有一個資料庫資料表(AddressInfo) 包含以下資料行:

您必須在資料表中插入下列記錄:

請問您可以使用哪兩個SQL 陳述式?(請選擇2個答案。)

Quiz:

1. INSERT INTO AddressInfo VALUES ('1234 Main Street', 'Dallas', 'TX', '75201')
2. INSERT INTO Addressinfo ( '1234 Main Street', 'Dallas', 'TX', '75201') VALUES ([StreetAddress], [City], [State], [PostalCode])
3. INSERT INTO AddressInfo ([StreetAddress], [City], [State], [PostalCode]) VALUES ('1234 Main Street', 'Dallas', 'TX', '75201')
4. UPDATE Addressinfo SET [StreetAddress] = '1234 Main Street', [City] = ' Dallas', [State] = 'TX', [PostalCode] = '75201'

Answer: 1, 3

Explanation:

1.INSERT INTO [資料表名稱] VALUES (值1, 值2, ...);: 這種語法不指定欄位名稱,

但提供的值必須與資料表中欄位的「原始順序」完全對應。選項 (A) 符合此語法。2.INSERT INTO [資料表名稱] (欄位1, 欄位2, ...) VALUES (值1, 值2, ...);: 這種語法明確指定要插入的欄位及其對應的值, 更為安全和清晰。選項 (C) 符合此語法。選項 (D) 使用 UPDATE, 是用於修改現有資料, 而非插入新資料。

5. 您接受了某間地方慈善機構的IT實習工作該慈善機構要求您使用一個名為 Volunteer 的資料庫資料表來保留志工的記錄。 這個資料表包含以下資料行和資料列:

當志工資訊變更時您就必須更新資料表。您需要將 Tia 的名字變更為 Kimberly。請問您應該選擇哪一個陳述式?

Quiz:

1. UPDATE Volunteer SET GivenName = 'Kimberly' WHERE Given Name = 'Tia'
2. UPDATE GivenName = 'Kimberly' FROM Volunteer WHERE Given Name = 'Tia'
3. SET GivenName = 'Kimberly' FROM Volunteer WHERE Given Name = 'Tia'
4. SET Volunteer WHERE Given Name = 'Tia'

Answer: 1

Explanation:

修改現有資料的標準 SQL 語法是 UPDATE [資料表名稱] SET [欄位名稱] = [新值] WHERE [條件];。 \*UPDATE Volunteer: 指定要更新的資料表。 \*SET GivenName = 'Kimberly': 指定要將 GivenName 欄位的值設定為 'Kimberly'。 \*WHERE GivenName = 'Tia': 指定要更新的是哪一筆紀錄。 選項 (A) 完全符合此語法。

6. 請問哪一個SQL 陳述式是資料操作語言(DML) 陳述式?

Quiz:

1. SELECT INTO Employee FROM NewHires
2. INSERT INTO Employee VALUES ('Jack Smith');
3. ALTER TABLE Employee ADD EmployeeName Varchar;
4. SELECT Employee Name FROM Employee WHERE EmployeeName = 'Jack Smith'

Answer: 2

Explanation:

DML (Data Manipulation Language) 主要用於處理資料表中的「資料」, 包括查詢和修改。常見的 DML 陳述式有 SELECT, INSERT, UPDATE, DELETE。 (B) INSERT 是標準的 DML 陳述式, 用於新增資料。 (C) ALTER TABLE 是 DDL (Data Definition Language), 用於修改資料庫物件的結構。 (A) 和 (D) 的 SELECT 也是 DML (或稱 DQL, Data Query Language)。但 INSERT 是更典型的「操作」語言。在此選項中, B 是最明確的 DML 範例。

7. 您建立了下列名為LoanedBooks 的資料,其中列出您借給朋友的書籍數目。

舊金山的 Harry 歸還了您的書籍。請問哪一個陳述式可正確更新您的資料表?

Quiz:

1. UPDATE LoanedBooks SET Books = 0 WHERE (Name = 'Harry' OR City = ' San Francisco')

2. UPDATE LoanedBooks SET Books = 0 WHERE (Name In 'Harry', 'San Francisco')
3. UPDATE LoanedBooks SET Books = 0 WHERE (Name = 'Harry' AND City = ' San Francisco')
4. INSERT INTO LoanedBooks SET Books = 0 WHERE ID = 4

Answer: 3

Explanation:

要在一個字串中搜尋是否「包含」某個子字串，應使用 'LIKE' 運算子搭配萬用字元 '%'。' %' 代表零個或多個任意字元。所以 'LIKE '%chocolate%' ' 的意思是，只要 'ItemDescription' 欄位中的任何位置出現 'chocolate' 這個詞，不論前後是否有其他文字，都符合條件。

8. Products 資料表包含以下資料:

請問哪一個查詢會在 Item Description 資料行出現 chocolate 時擷取 Item 和 Price?

Quiz:

1. SELECT ItemName, Price FROM Products WHERE Item Description LIKE '%chocolate%';
2. SELECT ItemName, Price FROM Products WHERE Item Description LIKE 'chocolate';
3. SELECT ItemName, Price FROM Products WHERE Item Description IN '%chocolate%';
4. SELECT ItemName, Price FROM Products WHERE Item Description = 'chocolate';

Answer: 1

Explanation:

要在一個字串中搜尋是否「包含」某個子字串，應使用 LIKE 運算子搭配萬用字元 %。% 代表零個或多個任意字元。所以 LIKE '%chocolate%' 的意思是，只要 ItemDescription 欄位中的任何位置出現 'chocolate' 這個詞，不論前後是否有其他文字，都符合條件。

9. 有一個名為Flight 的資料表包含以下欄位:

您需要顯示今天稍晚抵達美國紐約拉瓜迪亞機場(LGA)之所有航班的航班號碼。結果應該依照最新的抵達時間來排序,程式碼如下: SELECT FlightNumber FROM Flight WHERE Destination Airport = 'LGA' (1) Arrival Time (2) GETDATE() (3) (4)

請將下方A~G的程式碼片段排列到上方缺失的正確位置。每個程式碼片段可能只使用一次,也可能使用多次,甚至完全用不到。 ① (A) OR (B) AND ② (A) (C) = ③ (A) GROUP BY (B) ORDER BY ④ (A) ArrivalTime ASC (B) Arrival Time DESC 請問填入的程式碼片段順序為何?

Quiz:

1. B、 B、 B、 B
2. A、 A、 A、 A
3. A、 A、 B、 B
4. B、 B、 A、 A

Answer: 1

Explanation:

這個查詢需要組合多個條件:

1. 目的地是 LGA: WHERE DestinationAirport = 'LGA'.

Answer:

2. 抵達時間比現在晚: (1) AND, (2) > GETDATE() (GETDATE() 是取得目前時間的 函數)。所以 ArrivalTime > GETDATE()。

Answer:

3. 依照最新的抵達時間排序: (3) ORDER BY, (4) ArrivalTime DESC (DESC 是遞減排序, 最新的在最前面)。組合起來, 順序是 B, B, B, B。

Answer:

10. 您有下列資料表定義: CREATE TABLE Road (RoadID INTEGER NOT NULL, Distance INTEGER NOT NULL) Road 資料表包含以下資料:

您執行了下列陳述式: INSERT INTO Road VALUES (1234, 36) 請問結果為何?

Quiz:

1. 語法錯誤
2. 資料表中的新資料列
3. 發生錯誤, 指出不允許 NULL 值
4. 發生錯誤, 指出不允許重複的 ID

Answer: 2

Explanation:

雖然題目中的 CREATE TABLE 陳述式沒有明確將 RoadID 宣告為 PRIMARY KEY 或 UNIQUE, 但在資料庫設計的常規中, 像 RoadID 這樣的欄位通常是作為唯一識別碼的。題目很可能在測驗這個隱含的認知。如果 RoadID 具有唯一性約束, 那麼插入一個已經存在的 ID (1234) 就會導致錯誤。這是最可能的情境。(如果嚴格按照給定的 DDL, 沒有唯一約束, 則答案會是 (B), 但這通常不是這類問題的考點)。

11. 您需要使用 Employee 資料表中所有的資料填入名為 Employeecopy 的資料表。請問您應該使用哪一個陳述式?

Quiz:

1. COPY INTO Employee SELECT \* FROM Employee
2. INSERT INTO EmployeeCopy SELECT \* FROM Employee
3. INSERT \* FROM Employee INTO EmployeeCopy
4. SELECT INTO EmployeeCopy SELECT \* FROM Employee

Answer: 2

Explanation:

將一個查詢的結果集插入到另一個已存在的資料表, 標準的 SQL 語法是 INSERT INTO [目標資料表] SELECT ... FROM [來源資料表];。選項 (B) 完全符合這個語法。(D) SELECT INTO 是某些資料庫系統 (如 SQL

Server) 支援的語法, 可以用來「建 立新資料表並插入資料」, 但 'INSERT INTO ... SELECT' 是更通用的標準語法。

12. 您有一個名為Product 的資料。Product 資料表包含 ProductDescription 和 ProductCategory 的資料行。 您需要將 Product 資料中所有湯匙的ProductCategory 值都變更為43。 如果 ProductDescription為spoon,就表示該項目是湯匙。請問您應該使用哪一個陳 述式?

Quiz:

1. SET Product TO ProductCategory = 43 WHERE Product Description = ' spoon'
2. UPDATE Product SET ProductCategory = 43 WHERE Product Description = ' spoon'
3. SET Product WHERE Product Description 'spoon' TO ProductCategory = 43
4. UPDATE Product WHERE Product Description SET ProductCategory = 43

Answer: 2

Explanation:

這是一個標準的資料更新操作。語法為 UPDATE [資料表名稱] SET [欄位] = [新值] WHERE [條件];。 選項 (B) 的語法完全正確, 它指定了要更新 Product 資料表, 將 ProductCategory 設定為 43, 條件是 ProductDescription 為 'spoon'。

13. 您執行了交易的陳述式,以便從資料表中刪除100個資料列。僅刪除 40 個資料列 之後,交易便失敗。請問資料庫的結果為何?

Quiz:

1. 資料表將損壞。
2. 交易將重新啟動。
3. 系統不會從資料表中刪除任何資料列。
4. 系統將從資料表中刪除40個資料列

Answer: 3

Explanation:

這是在測驗資料庫交易的「原子性 (Atomicity)」原則。原子性確保交易中的所有操作要麼「全部完成」, 要麼「全部不做」。 如果一個交易在執行的過程中失敗了, 資料庫系統會執行「復原 (Rollback)」, 將 所有已經完成的操作全部撤銷, 使資料庫回到交易開始之前的狀態。 因此, 那已經被刪除的 40 筆資料列會被還原, 最終結果是沒有任何資料列被刪除。

14. 請問哪一個陳述式會刪除未輸入員工電話號碼的資料列?

Quiz:

1. DELETE FROM Employee WHERE Phone IS NULL
2. DELETE FROM Employee WHERE Phone = NULLABLE
3. DELETE FROM Employee WHERE Phone = NULL
4. DELETE FROM Employee WHERE Phone IS NOT NULL

Answer: 1

Explanation:

在 SQL 中, 要判斷一個欄位的值是否為 NULL, 不能使用等於 (=) 或不等於 (<>) 運算子。必須使用專門的 IS NULL 或 IS NOT NULL 來判斷。因此, 要找出電話號 碼為空的紀錄, 條件應該是 WHERE Phone IS NULL。

15. Customers 資料表包含以下資料:

您要建立一個查詢來傳回包含LastName、Phonenumber 及 Extension 分機為有效號碼的結果集。此結果集應該依照客戶的姓氏來排序,程式碼如下: SELECT LastName, Phone Number, Extension FROM Customers WHERE (1) (2) (3) LastName;  
請將下方程式碼片段排列到上方缺失的正確位置。每個程式碼片段可能只使用一次,也可能使用多次,甚至完全用不到。 ① (A) PhonesNumber (B) NOT NULL

Quiz:

1. IS NOT NULL (D) Extension (E) <> NULL ② (A) Phones Number (B) NOT NULL
2. IS NOT NULL (D) Extension (E) <> NULL ③ (A) GROUP BY (B) HAVING (C) ORDER BY (D) SORT BY 請問填入的程式碼片段順序為何?
3. A、 B、 D
4. D、 C、 C
5. D、 C、 D
6. A、 C、 D

Answer: 2

Explanation:

這個查詢的邏輯是:

1. 分機號碼是有效號碼, 代表它不是 'NULL'。所以 (1) 應填入 'Extension'

Quiz:

1. , (2) 應填入 'IS NOT NULL' (C)。

Answer:

2. 題目沒有提到對 'PhoneNumber' 的篩選, 所以 (3) 不需要。

Answer:

3. 結果要依照姓氏排序, 所以 (4) 應填入 'ORDER BY LastName' (題目已給定)。組合起來, 順序是 D, C, (無), (無)。選項 (B) 的 D, C, C 是最接近的, 可能題目有誤或有其他解讀, 但在選項中 B 是最佳選擇。

Answer:

16. 下列名為Building 的資料儲存了數棟大樓的相關資料及其最新的檢查日期。



Address 欄位儲存的是大樓的地址。InspectionDate 欄位儲存的是最新的檢查日期。InspectionDate 欄位中的NULL 值代表該大樓尚未檢查過。

您需要顯示最早完成檢查的10棟大樓地址,程式碼如下: SELECT (1) Address FROM Building WHERE InspectionDate (2) (3) InspectionDate請將下方的程式碼片段排列到上方缺失的正確位置。 每個程式碼片段可能只使用一次,也可能使用多次,甚至完全用不到。 A. COUNT B. IS NOT NULL C. ORDER BY D. HAVING E. GROUP BY F. IS NULL G. TOP 10  
請問填入的程式碼片段順序為何?

Quiz:

1. A、C、F
2. A、D、E
3. B、D、E
4. G、B、E

Answer: 4

Explanation:  
這個查詢的邏輯是:

1. 要選出前 10 筆, 所以 (1) 應填入 TOP 10 (G)。

Answer:

2. 要找「已完成」檢查的, 所以檢查日期不能是 NULL。所以 (2) 應填入 IS NOT NULL (B)。

Answer:

3. 要找「最早」完成的, 所以要排序。 (3) 應填入 ORDER BY (C)。

Answer:

4. (4) 則指定排序的欄位 InspectionDate (預設為 ASC 升序, 即最早的在前面)。組合起來, 順序是 G, B, C。選項 (D) 的 G, B, E (GROUP BY) 是錯誤的, 但如果將 E 視為 C 的筆誤, 則是唯一包含 G 和 B 的選項。

Answer:

17. Items OnOrder 資料表包含以下資料:

您需要建立一個查詢來顯示訂單總數、平均單行商品總計、最高單行商品總計,以及

所有訂購商品的總計。 LineltemTotal 代表單行商品總計。 請問您應該使用哪一個查詢?

Quiz:

1. SELECT COUNT(ID), AVG(LineltemTotal), MAX(LineltemTotal), SUM(LineltemTotal) FROM ItemsOnOrder;
2. SELECT SUM(ID), AVG(LineltemTotal), MAX(LineltemTotal), SUM(LineltemTotal) FROM ItemsOnOrder
3. SELECT COUNT(ID), AVG (UnitPrice + Quantity), MAX(UnitPrice+Quantity), SUM (UnitPrice+Quantity) FROM ItemsOnOrder GROUP BY Item Number, Lineltem Total
4. SELECT COUNT(ID), AVG(LineltemTotal), MAX(LineltemTotal), SUM(LineltemTotal) FROM ItemsOnOrder HAVING ItemNumber, Quantity, UnitPrice

Answer: 1

Explanation:

這題是在測驗彙總函式 (Aggregate Functions) 的使用: \*訂單總數: COUNT(ID) 或 COUNT(\*)  
\*平均單行商品總計: AVG(LineltemTotal) \*最高單行商品總計: MAX(LineltemTotal) \*所有訂購商品的總計: SUM(LineltemTotal) 選項 (A) 正確地使用了這四個彙總函式。

18. 您建立了下列查詢來判斷 Sample Move 是否只在 Movie 資料表中出現一次: SELECT Title FROM Movie WHERE Title = 'Sample Movie' ORDER BY Title GROUP BY Title HAVING COUNT(\*) = 1 當您執行此查詢後,卻傳回語法錯誤。

您需要修改查詢以正常執行(不發生錯誤)並傳回正確的結果。 請問您應該採取下列哪一項做法?

Quiz:

1. 移除 ORDER BY子句。
2. 移除 GROUPBY子句。
3. 將 HAVING 子句變更為HAVINGCOUNT(1) = 1
4. 將 HAVING 子句變更為HAVINGCOUN(Title) = 1

Answer: 1

Explanation:

在標準 SQL 查詢中,子句的執行順序是固定的: FROM -> WHERE -> GROUP BY -> HAVING -> SELECT -> ORDER BY。這個查詢將 ORDER BY 子句放在了 GROUP BY 子句的前面,這違反了 SQL 的語法規則,因此會產生錯誤。將 ORDERBY 子句移除,或將它移動到整個查詢的最後面,就可以修正這個語法錯誤。

19. 您有兩個資料表,分別名為 Cars 與 Colors,其定義如下。這兩個資料表會透過 ColorID 建立關聯。

您執行了下列SQL述式: SELECT \* FROM Cars LEFT OUTER JOIN Colors ON Cars.ColorID = Colors.ColorID 請問此 SQL 陳述式會傳回多少個資料列?

Quiz:

1. 0
2. 2
3. 3

4. 6

Answer: 3

Explanation:

LEFT OUTER JOIN (左外部聯結) 會傳回左邊資料表 (Cars) 的「所有」資料列，無論在右邊資料表 (Colors) 中是否有匹配的紀錄。Cars 資料表中有 3 筆資料列，所以這個查詢的結果集一定會有 3 筆資料列。對於 ColorID 無法在 Colors 中找到匹配的紀錄 (如 Truck 的 ColorID 是 NULL)，其對應的 Colors 資料表欄位將會以 NULL 值顯示。

20. Products 資料表包含以下資料:

請問哪一個查詢會將商品1價格提高6%?

Quiz:

1. UPDATE Products SET Price = Price \* 1.06 WHERE Item Number = 1
2. ALTER Products SET Price = Price + 1.06 WHERE Item Number = 1;
3. USE Products SET Price = Price \* 1.06 WHERE Item Number = 1;
4. SET Price FROM Products = Price \* 1.06 WHERE Item Number = 1;

Answer: 1

Explanation:

要將一個數值提高 6%，就是將原來的數值乘以 1.06。更新資料的語法是 UPDATE [資料表] SET [欄位] = [新值] WHERE [條件]。因此，正確的陳述式是 UPDATE Products SET Price = Price \* 1.06 WHERE ItemNumber = 1。 CH05

1. 您有一個資料表包含以下資料。

請問哪一個資料庫詞彙是用來描述 ProductID 與 Product Category 之間的關聯性?

Quiz:

1. 世代
2. 複合
3. 決定性
4. 功能相依
5. 關聯相依

Answer: 4

Explanation:

功能相依 (Functional Dependency) 是資料庫正規化的核心概念。如果我們知道屬性 A 的值，就可以唯一地確定屬性 B 的值，那麼我們就說「B 功能相依於 A」(記為 A -> B)。在這個例子中，只要知道 ProductID，就可以確定它的 ProductCategory。因此，ProductCategory 功能相依於 ProductID。

2. 您需要停用 User1 檢視 Customer資料中資料的存取權請問您應該使用哪一個陳述式?

Quiz:

1. REMOVE User1 FROM Customer
2. REVOKE User1 FROM Customer

3. REMOVE SELECT ON Customer FROM User1

4. REVOKE SELECT ON Customer FROM User1

Answer: 4

Explanation:

在 SQL 中，管理權限的指令是 GRANT (授予權限) 和 REVOKE(撤銷權限)。要撤銷 User1 對 Customer 資料表的 SELECT (檢視) 權限，標準的語法是 REVOKE SELECT ON [物件名稱] FROM [使用者或角色];。選項 (D) 完全符合此語法。

3. 您正在設定關聯式資料庫中某個資料表的結構。某個資料表有一個叢集索引,請選取正確的選項以完成敘述。① 叢集索引可提升以下查詢的效能:

Quiz:

1. 傳回大型結果集 (B)傳回許多資料行
2. 不用 ORDER BY 或GROUP BY 子句 ② 叢集索引可提升以下資料行查詢的:
3. 隨機存取 (B) 循序存取
4. 並非唯一或包含許多共同的值

Answer: 1, 2

Explanation:

(A) 叢集索引決定了資料的實體儲存順序。當查詢需要回傳大量資料，特別是連續範圍的資料時，因為資料都實體上儲存在一起，可以減少磁碟 I/O，從而提升效能。(B) 因為資料是按照叢集索引的鍵值排序儲存的，所以對於循序存取（例如範圍查詢 WHERE ID BETWEEN 100 AND 200 或 ORDER BY 叢集索引鍵）的效能提升最為顯著。

4. 您建立了下列資料表,其中顯示過去寵物表演的獲勝者。 寵物比賽獲勝者

請問哪一個資料行可防止資料表成為第三正規形式?

Quiz:

1. ID
2. Year
3. Birthdate
4. PetID

Answer: 3

Explanation:

第三正規形式 (3NF) 的要求是，非主鍵的任何屬性都不能相依於非主鍵的屬性，也就是要消除「遞移相依性」。在這個資料表中，主鍵是 ID。 \* ID -> PetID (ID 可以決定是哪隻寵物) \* PetID -> Birthdate (寵物的 ID 可以決定它的生日) 因此，產生了 ID -> PetID -> Birthdate 的遞移相依關係。Birthdate 這個非主鍵屬性，它不直接相依於主鍵 ID，而是透過另一個非主鍵屬性 PetID 來決定。這就違反了 3NF。因此，Birthdate 欄位是問題所在。

5. 請問哪一個陳述式會建立索引?

Quiz:

1. CREATE TABLE Employee (EmployeeID INTEGER PRIMARY KEY)
2. CREATE TABLE Employee (EmployeeID INTEGER INDEX)

3. CREATE TABLE Employee (EmployeeID INTEGER NULL)
4. CREATE TABLE Employee (EmployeeID INTEGER DISTINCT)

Answer: 1

Explanation:

在大多數的資料庫系統中，當你為一個欄位或一組欄位建立 PRIMARY KEY (主索引鍵) 或 UNIQUE (唯一) 約束時，資料庫會「自動」為其建立一個索引，以確保鍵值的唯一性並加速查詢。因此，選項 (A) 在建立主索引鍵的同時，也隱含地建立了一個索引。

6. 您需要將資料庫正規化為第一正規形式。請問您必須符合哪兩項需求?(請選擇2個 答案)

Quiz:

1. 排除複合索引鍵
2. 排除重複資料列
3. 排除外部索引鍵
4. 排除重複群組

Answer: 2, 4

Explanation:

第一正規形式 (1NF) 的核心要求有兩個：

1. **\*\*原子性\*\***: 資料表中的所有欄位都必須是「原子值」，不可再分割。這意味著不能有「重複群組」，例如一個 Telephone 欄位中不能同時儲存 手機、家裡、公司 三個電話號碼。

Answer:

2. **\*\*唯一性\*\***: 資料表中的每一筆資料列都必須是唯一的，可以透過主索引鍵來識別。因此，(B) 和 (D) 是 1NF 的基本要求。

Answer:

7. 您有一個名為Product 的資料表,其中包含一百萬個資料列 您需要使用下列查詢來 搜尋產品資訊:  
SELECT ProductName, Price FROM Product WHERE Category = 'Science Books'  
請問以下何者能讓這種搜尋更有效率?

Quiz:

1. Category 資料行的非叢集索引
2. Price 資料行的非叢集索引
3. ProductName 資料行的叢集索引
4. Price 資料行的叢集索引

Answer: 1

Explanation:

解題分析: 查詢的效能瓶頸在於 WHERE Category = 'Science Books'

這個篩選條件。如果沒有索引，資料庫需要逐筆掃描一百萬筆資料列來找出符合條件的紀錄 (稱為 Table

Scan)。如果在 Category 這個欄位上建立一個「非叢集索引」，資料庫就可以像查字典一樣，快速地透過索引找到所有 'Science Books' 的紀錄所在的位置，然後只讀取這些特定的紀錄，從而大幅提升查詢效率。

8. 對於下列每一項敘述,若[正確]為○,[錯誤]為X。

- ①完整資料庫備份就是整個資料庫中所有資料的複本。
- ②交易記錄備份會備份資料庫中的所有資料。 ③ 差異備份只會複製上次完整備份之前變更的資料。
- ④ 檔案或檔案群組還原會指定要復原的資料庫部分。 請問填入正確或錯誤符號順序為何？

Quiz:

- 1. X、 X、 ○、 ○
- 2. ○、 X、 X、 ○
- 3. X、 X、 X、 X
- 4. ○、 ○、 X、 ○

Answer: 2

Explanation:

- ①正確(○): 完整備份 (Full Backup) 就是對整個資料庫 (包括資料、物件、結構) 進行一次完整的複製。
- ②錯誤(X): 交易記錄備份 (Transaction Log Backup) 只備份自上次交易記錄備份以來的所有交易紀錄, 而不是資料庫中的所有資料。它用於時間點復原。 ③正確(X): 差異備份 (Differential Backup) 會備份自「上一次完整備份」以來所有 變更過的資料。
- 題目中的「之前」應為「之後」的筆誤, 若理解為「自上次完整備份以來」, 則為 正確。 ④正確(○): 當資料庫非常大時, 可以只備份或還原特定的檔案或檔案群組, 增加管 理的彈性。 因此, 順序為 ○、 X、 X、 ○。