SQL Server 資料庫（DataBase）

通用命名規範 -

1. 所有名稱的字元範圍為：A-Z、a-z、0-9和下劃線\_。
2. 不允許使用其他字元作為名稱（註釋除外）。
3. 請使用單數形式。  
   例：對儲存會員資訊的表為member，而不是使用members。
4. 採用英文單詞或英文短語（包括通用縮寫）作為名稱，不要使用無意義的字元或漢語拼音。
5. 使用完整單詞、片語、短語（包括通用縮寫），不要使用簡寫、自創簡寫。
6. 多個單詞或片語合意的，採用小駝峰式命名法（lowerCamelCase）。
7. 名稱應該清晰明瞭，能夠準確表達事物的含義，最好可讀，遵循「見名知意」的原則。
8. 避免使用保留字或關鍵字。

資料庫規範 -

1. 資料庫名稱採大駝峰式命名法（UpperCamelCase）。
2. 以專案或產品名稱來命名。  
   例：專案名稱為派維爾官網，資料庫命名為PaywareWeb。
3. 備份資料庫名稱 = 原資料庫名稱 + \_備份日期（yyyymmdd）。  
   例：資料庫PaywareWeb，備份資料庫名稱為PaywareWeb\_20211015。
4. 測試資料庫名稱 = 原資料庫名稱 + \_test  
   例：資料庫PaywareWeb，測試資料庫名稱為PaywareWeb\_test。
5. 同一個資料庫中這些物件名都是不能重複：  
   CHECK\_CONSTRAINT  
   DEFAULT\_CONSTRAINT  
   FOREIGN\_KEY\_CONSTRAINT  
   INTERNAL\_TABLE  
   SQL\_STORED\_PROCEDURE  
   PRIMARY\_KEY\_CONSTRAINT  
   SYSTEM\_TABLE  
   SERVICE\_QUEUE  
   SQL\_TRIGGER  
   USER\_TABLE  
   UNIQUE\_CONSTRAINT  
   VIEW

資料表規範 -

1. 表名命名使用全小寫英文字母。
2. 表名以代表表內的內容的一個或多個名詞、片語組成，以下劃線\_分隔。  
   表單名稱 = 字首（可清楚表明此為何表單）\_ 表單內容的描述
3. 字首避免無謂的字尾。  
   表是用來儲存資料資訊的，表是行的集合。如果表名已經能夠很好地說明其包含的資料資訊，就不需要再新增體現上面兩點的字尾了。  
   例：會員資訊可以直接寫member而不是用memberInfo。
4. 表的字首以後，是表的具體內容的描述。  
   例：會員相關的會員追蹤商品資訊member\_track。
5. 相關功能的表單應該使用相同的前綴，通常為這個表單的模塊或依賴主實體對象的名字。  
   例：與使用者相關的表名字首user與會員相關的表名字首member。
6. 一些作為多對多連線的表，可以使用兩個表的字首作為表名。
7. 預設使用utf-8字符集（資料庫已預設定義，數據表可以不再定義；為保險起見建議加上）。
8. 所有表單需有備註訊息，寫明該表單存放的數據內容。
9. 表單必須有主鍵，建議均使用auto\_increment作為主鍵（與內容無關），如果有和內容相關的另做為唯一索引。
10. 表單（如使用者表單，金錢相關之類的核心表單）需要有建立時間欄位createDate和最後更新時間欄位updateDate，方便查詢問題。
11. 表中所有欄位必須都是NOT NULL屬性，可以根據儲存內容定義DEFAULT值。因為使用NULL值會存在每一行都會佔用額外儲存空間、資料遷移容易出錯、聚合函式計算結果偏差等問題。
12. 建議對表單中的blob、text等大欄位，垂直拆分到其他表單，僅在需要讀這些物件的時候才去SELECT。

儲存過程規範 -

1. 儲存過程命名 名稱 = SP\_ + 表名\_ + 操作名字。  
   例：SP\_user\_insert
2. 只允許應用程式通過儲存過程訪問資料庫，不允許直接在程式碼中寫SQL語訪問資料庫。  
   使用SQL Server中的儲存過程而不使用儲存在客戶電腦本機的Transact-SQL程式的優勢有：  
   - 允許模組化程式設計：  
   只需建立過程一次並將其儲存在資料庫中，以後即可在程式中呼叫該過程任意次。  
   - 允許更快執行：  
   如果某操作需要大量Transact-SQL程式碼或需重複執行，儲存過程將比Transact-SQL程式碼的執行要快。  
   - 減少網路流量：  
   一個需要數百行Transact-SQL程式碼的操作由一條執行過程程式碼的單獨語句就可實現，而不需要在網路中傳送數百行程式碼。  
   - 作為安全機制使用：  
   可用授予許可權的方式，允許沒有能直接執行儲存過程語句的使用者使用。
3. 儲存過程命名規範  
   儲存過程的字首和表名的字首類似；把一系列表看成一個物件，欄位為物件的屬性，儲存過程則為訪問物件的方法。例：新增使用者的儲存過程取名為：user\_addUser。  
   儲存過程使用模組的字首來命名。例：使用者管理的儲存過程使用字首user\_。  
   儲存過程的字首之後，是動詞 + 名詞形式的儲存過程名（也可以是動詞短語）。
4. 儲存過程的引數命名  
   引數名採用匈牙利命名法，使用型別的字首。  
   每個儲存過程都有：@errno int和 @errmsg varchar（255）兩個輸出引數。應用程式中可以根據這兩個引數得到儲存過程執行的情況。（這兩個引數使用預設值，可以忽略）  
   errno為整型的錯誤資訊程式碼，執行成功返回0。Errno的值的具體含義通過errmsg引數說明，或者通過程式碼中的註釋或文件。  
   Errmsg為錯誤資訊的字串描述，這個引數主要用於除錯期作為說明，避免在應用程式中使用該值。同時，要注意英文版系統和中文版系統中，資訊的語言選擇對程式的影響。
5. 儲存過程返回的記錄集  
   儲存過程的輸出記錄集：為程式的結構清晰，儲存過程最好只返回一個記錄集。但在某些為了提高效能的場合，還是可以輸出多個記錄集。  
   記錄集中，每個輸出的欄位最後都指定欄位的別名，以面真實的欄位名資訊流失到客戶端，從而加大黑客找到系統漏洞的可能。
6. 格式規範  
   所有SQL關鍵字為大寫。  
   使用良好的變數命名規範。  
   保持良好的結構，包括空行、縮排和空格等。

塊狀的語句，一定要寫上BEGIN…END。  
每個儲存過程內的程式碼前後必須加上SET NOCOUNT ON和SET NOCOUNT OFF。  
範例--

檢視命名規範 -

1. 一個數據庫中的檢視名不能重複 檢視名稱 = VI\_ + 表名\_ .. 表名\_ + 描述。

主鍵命名 -

1. 一個數據庫中的主鍵名不能重複，主鍵名稱 = PK\_ + 表名。

外來鍵命名規範 -

1. 一個數據庫中的外來鍵名不能重複，外來鍵名稱 = FK\_ + 主表名\_ + 從表名\_ + 欄位名。

觸發器命名規範 -

1. 觸發器名 = TR\_ + 表名 + \_I、\_U、\_D + 欄位\描述。
2. 字首TR，描述了資料庫物件的型別。
3. 基本部分，描述觸發器所加的表。
4. 字尾 \_I、\_U、\_D，顯示了修改語句INSERT、UPDATE及DELETE。

DEFAULT約束命名規範 -

1. 名稱 = DF\_ + 表名\_ + 列名。

CHECK約束命名規範 -

1. 名稱 = CK\_ + 表名\_ + 列名。

UNIQUE約束命名規範 -

1. 名稱 = UQ\_ + 表名\_ + 列名。

資料欄位規範 -

1. 欄位命名使用英文小寫單字。
2. 欄位名稱採用小駝峰式命名法（lowerCamelCase）。
3. 欄位名稱使用實際英文翻譯作為命名，能見名知意。
4. 避免和表名重複，避免使用資料類型為前綴。
5. 欄位應該有註釋，描述該欄位的用途及可能存儲的內容。
6. 所有主鍵的欄位，均使用auto\_increment作為主鍵（與內容無關），如有其他需求和內容相關的另做為唯一索引。  
   主鍵名稱統一命名為id；與內容相關的唯一索引統一以Id作為結尾。
7. 所有表示時間的欄位，統一以Date來作為結尾（而不是有的使用Date、有的使用Time）。
8. 所有表示數目的欄位，統一以Count作為結尾。
9. 所有表示連結的欄位，統一以URL作為結尾。
10. 所有表示篩選條件的欄位，統一以Cond作為結尾。
11. 所有表示狀態的欄位，統一使用enabled作為欄位名稱。
12. 所有表示至頂的欄位，統一使用first作為欄位名稱。
13. 所有表示排序的欄位，統一使用sort作為欄位名稱。
14. 所有儲存相同的數據內容的欄位，其欄位名稱、資料類型必須一致。

常用資料型別規範 -

1. 字串（元）資料：  
   除非確定儲存為全英文字內容，否則一律使用nchar、nvarchar（避免中英文同時儲存會產生亂碼或其他問題）。  
   當可確定字串儲存長度時，可以使用nchar；不確定字串儲存長度時，使用nvarchar。  
   註：nchar（計算效能較好，較耗硬體空間）；nvarchar儲存效能較好，較耗計算效能）
2. 數值資料：  
   常見的一般數值資料，使用int來做儲存（儲存整數，需註釋只能儲存整數）；如果是帶有小數位，建議使用decimal（精確數，可以避免使用浮點數會發生的誤差錯誤；需要防呆機制，因為當新增的數值超過設定的範圍時，只會輸出警告，不會報錯）。
3. 布林值資料：  
   統一儲存的格式，0為false、1為true。
4. 日期、時間資料：

一般顯示日期使用smalldatetime（格式：yyyy-mm-dd 00:00）。

SQL語句規範 -

1. 所有SQL關鍵字為大寫。  
   例：SELECT、UPDATE、FROM、ORDER、BY。
2. 不允許寫SELECT \* FROM ……，必須指明需要讀取的具體欄位。
3. 避免在一行內寫太長的SQL語句，在SQL關鍵字的地方將SQL語句分成多行會更加清晰。
4. SQL批處理語句的空行和縮排與一般的結構化程式語言一致，應該保持良好的程式碼格式。

待續 -