

好的變數命名習慣

精神：

- (1)增進溝通：讓程式碼自己解釋，其他人只要看 code，就能很快的了解這個程式的功能。
- (2)釐清：讓人能快速的理解程式的邏輯。
- (3)一致性：將變數系統化能協助未來功能的增減

規則：

(1) 編譯器規定（基本要求）：

- 1. 不能以數字開頭， 也不能以中文作為變數名稱，另外大寫和小寫是不一樣的。
- 2. 而字元長度通常有一個限制，像 VC Compiler 是 255 個字元。
- 3. 在同一個 name space 中， 不能有兩個變數使用同樣的名字。
- 4. 保留字作為語法的關鍵字不能當變數名稱。 電腦會搞不懂我們寫的是指令還是變數。

(2) 自我要求：

- 1. 變數名稱和存放在其內資料的特性要相符。例如：有一變數存放現在是西元某某年，我們可以將它取名為 `currentYear`。
- 2. 通常保留全部是大寫的名稱作為常數（或者是常數變數） 例如：`PI`, `NUMBER_OF_DAYS_PER_YEAR...` 第
- 3. 2 第一個字母大寫的名稱，例如：`Register`，常保留下來作為自定資料型態的名稱。
- 4. 避免奇怪的縮寫，避免別人看不懂自己的 code。
- 5. Never use the word “User” in your code：因為 user 的描述過於廣泛，從字面上無法直接推斷這個變數的用途，就跟用 `a, b, c` 差不多。

常見命名風格：

- (1) 匈牙利命名法(Hungarian notation)：在 Microsoft 中的軟體常用。將資料型態寫在變數名稱的前面， 例如 `intNum`(整數)、`strName`(字串)。
- (2) 駝峰式命名法(Camel-Case)：單字與單字之間不用任何符號隔開，而是利用大小寫區別， 例如 `firstName`。有些人習慣首字大寫如 `FirstName`。它的高低起伏就像駱駝一樣。