何寫出易懂的程式碼：跟你想的不一樣

建立日期 2013-04-30 09:43      最近更新在 2013-04-30 10:25作者是 謝良奇／編譯

或許你常聽到寫出易懂程式碼的重要性。開發者對於如何撰寫淺顯易懂的程式碼都很有想法，尤其越資深的開發者，意見越多。不過，你真的曾經停下敲鍵盤的手，想過如何真的撰寫易懂的程式碼嗎？

標準答案是

你或許會同意，不論使用的程式語言為何，下列項目有助於程式碼的易讀性：

1. 好的變數、方法、類別名稱
2. 變數、類別與方法都擁有單一目的
3. 一致性的縮排和格式樣式
4. 減少程式碼的巢狀層級

除此之外還有很多關於程式碼易讀性的標準答案與普遍公認的做法，我對這些沒什麼反對意見。

不過我倒是希望你對可讀性有更深入的瞭解。

詞彙和讀者經驗

我可以在看到一份程式碼的兩秒鐘裡，告訴你它寫得好不好以及容易理解與否。同時，我可以拿我寫得最好、有高度易讀性的程式碼範例，給一位新手程式設計者看，讓他們看不出，與他們看過的其他程式碼之間有何不同。

而對他們而言，與他們看過的其他程式碼相較，他們看不出兩者有何不同。

儘管我的程式碼中變數名稱具備高度描述性，僅有少量參數且名稱簡短的方法，一次只做一件事，程式碼清楚地依照功能加以組織分段。但是這些新手們一點也不覺得這樣做，比起他們未經任何結構思考的程式，有任何易於閱讀的地方。

事實上，我經常聽到有關的抱怨，抱怨我的程式碼方法太多，難以閱讀，變數名稱太長，容易讓人混淆。

這一切基本上都源自於老手與新手閱讀程式碼方式的不同

有經驗的開發者閱讀程式碼時，不會留意程式語言本身的詞彙，而是專注在程式碼表達的概念上，是程式碼的目的，而非如何步驟過程。

新手與經驗較少的開發者完全不一樣。當經驗較少的開發者閱讀程式碼時，會試著去了解程式碼的結構。新手則是把大部份重心放在語言的詞彙，而非語言試著要傳達的東西。

對他們而言，像 NumberOfCoins 這樣的長變數名稱毫無描述性，該變數的人格化表示加上過長的名稱，只會誤導他們，以為該變數除了表示一個整數之外，還包含了別的東西。他們寧願看到像 X 或 Number 這樣的變數名，因為光記住整數代表什麼就夠他們受的了。

有經驗的開發者完全不管整數或字串什麼的。他們希望瞭解的是，變數在方法或系統的邏輯語境下，所代表的意義，而非變數形態或變數如何運作。