

# 第 6 章

## 條件述句

有時候，您要您的程式碼只在符合某個條件時才執行動作。一些特別的關鍵字就是用來達成這個目標的 [1.2]。

```
[1]
int age = 42;
if (age > 30)    // 符號 > 代表 "大於"
{
    NSLog(@"age is older than thirty."); // [1.4]
}
NSLog(@"Finished."); // [1.6]
```

第 [1.2] 行出現了 *if...* 指令，也被稱為條件指令。您可以認出大括號，而在大括號中的是只有當小括號中的邏輯算式結果為真（true）時您才要執行的所有程式碼。在這裡，如果條件 *age > 30* 成立，那麼在 [1.4] 行的字串就會被印出來。但是無論這個條件成立與否，在第 [1.6] 行的字串都會被印出來，因為它在 *if* 區塊的大括號之外。

我們可能也會提供另外一組指令是當條件不成立時所要執行的，這要使用 *if...else* 述句 [2]。

```
[2]
int age = 42;
if (age > 30)
{
    NSLog(@"age is older than thirty."); // [2.4]
}
else
{
    NSLog(@"age is not older thirty."); // [2.7]
}
NSLog(@"Finished."); // [1.6]
```

在述句 [2.7] 中的字串只有在條件不成立時才會被印出來，而這並不是範例 [2] 中的情況。

除了述句 [2.2] 中的大於符號之外，以下針對數字的比較運算子都可以任您運用。

== 等於

> 大於

< 小於  
>= 大於或等於  
<= 小於或等於  
!= 不等於

請特別留意等於運算子 - 它是兩個等號。它非常容易因為忘記而只輸入了一個等號。不幸的是，這剛好是指定運算子，因此會將這個變數指定為某個值。這是導致初學者寫出混亂與充滿臭蟲的程式碼的常見原因。現在，大聲的念出來：我絕對不會在測試是否相等的時候忘記用兩個等號！

比較運算子在您要重複一串述句幾次時很有用。這會是下一章的主題。但首先我們要先探討一些未來可能會用到的 if 述句性質。

讓我們仔細的看看進行比較這件事。一個比較運算子只會在兩個可能性中產生一個結果：不是 true（真）就是 false（假）。

在 Objective-C 中，true 與 false 分別被表示為 1 與 0。有一種名為 BOOL 的特殊資料型態，您可以用它來表示這些值。要表示 "true"，您可以寫成 1 或 YES。要表示 "false"，您可以寫成 0 或 NO。

```
[3]
int x = 3;
BOOL y;
y = (x == 4); // y 會是 0。
```

要同時檢查多個條件是可行的。如果多於一個條件都必須成立，請使用邏輯 AND，它以兩個 & 符號表示：&&。如果只要至少一個條件成立即可，請使用邏輯 OR，它以兩個 | 符號表示：||。

```
[4]
if ( (age >= 18) && (age < 65) )
{
    NSLog(@"Probably has to work for a living.");
}
```

要巢狀的組織條件述句也是可行的。只需要簡單的把一個條件述句放到另一個條件述句的大括號中即可。首先最外面的條件會先被檢驗，如果成立，裡面的下一個述句會接著檢驗，然後持續下去：

```
[5]
if (age >= 18)
{
```

```
if (age < 65)
{
    NSLog(@"Probably has to work for a living.");
}
}
```