

CH. 6

邏輯判斷：猜數字遊戲

HORAZON

應用程式設計

本章目標

1. 製作「終極密碼」遊戲 (1~100)
2. 深入理解 Global Variables (全域變數)
3. 熟悉 比較運算子 (大於、小於、等於)
4. 學習 Notifier (對話框) 元件

遊戲規則

1. 電腦隨機選一個 1~100 的數字 (答案)。
2. 玩家輸入一個數字。
3. 電腦判斷：
 - 太大了！
 - 太小了！
 - 猜對了！

這需要用到 比較 (Compare) 與 流程控制 (Control)。

畫面設計

1. **標籤**：顯示提示 (例如 "請猜 1~100")。
2. **文字輸入盒**：讓玩家輸入數字。
3. **按鈕**：猜！
4. **按鈕**：重新開始 (重置遊戲)。
5. **Notifier (對話框)**：在 User Interface 裡找，拉進去 (它在畫面上看不到，會在下面)。

定義變數

我們需要兩個變數：

1. `answer` (答案)：存電腦選的數字。
2. `guess` (猜測)：存玩家輸入的數字 (其實直接用輸入盒的文字也可以，但用變數比較清楚)。

初始化全域變數 `answer = 0`

遊戲初始化 (GAME START)

我們寫一個 **程序 (Procedure)** 或者直接在 **Screen.Initialize** (畫面啟動時) 設定：

```
當 Screen1.Initialize  
    設 answer 為 (隨機整數 1 到 100)  
    設 Txt_Input.文字 為 ""  
    設 Lbl_Hint.文字 為 "請猜 1~100"
```

記得「重新開始」按鈕也要做一樣的事！

核心邏輯：判斷大小

當「猜」按鈕被按下：

```
如果 (Txt_Input.文字 > answer)
    呼叫 Notifier1.ShowAlert ("太大了!")
否則如果 (Txt_Input.文字 < answer)
    呼叫 Notifier1.ShowAlert ("太小了!")
否則 (=)
    呼叫 Notifier1.ShowAlert ("恭喜猜對了!")
```

- 比較積木：在 Math 抽屜裡的 `=`，可以下拉選 `>` 或 `<`。
- ShowAlert: 會像 Toast 一樣短暫顯示訊息。

進階挑戰：縮小範圍

真正的終極密碼會顯示範圍，例如原本 1-100，猜 50 太小，範圍會變 50-100。

這需要另外兩個變數：

- `min` (下限)：初始 1
- `max` (上限)：初始 100

當猜太小，就把 `min` 變成那個數字；當猜太大，就把 `max` 變成那個數字。

並且更新標籤顯示：`min + "~" + max`。

重點回顧

- 變數 是存狀態的關鍵 (答案、範圍)。
- 比較運算子 ($>$, $<$, $=$) 決定遊戲走向。
- Notifier 可以用來顯示簡單的提示訊息。

下一章：剪刀石頭布 (邏輯練習)