

CHAPTER 06

# C# 程式基礎入門

HORAZON

手遊程式設計

# 章節目標 (第二階段啟動)

從現在開始，我們進入 Phase 2：程式開發階段。

- 建立第一支 C# Script (腳本)
- 認識程式編輯器 (Visual Studio)
- 理解 Unity 腳本結構 (Start vs Update)
- 學會使用 Console (控制台) 除錯

# 1. 為什麼要寫程式？

雖然 Unity 提供了很多組件 (Component)，但它們的功能是固定的。

- **客製化邏輯**：如果我想「收集 10 個金幣就進化」，這沒有現成的組件。
- **變數管理**：紀錄分數、血量、彈藥量。
- **事件溝通**：當 A 碰到 B，要叫 C 做 D 事情。

C# (C Sharp) 是 Unity 使用的語言，它是微軟開發的主流語言，強大且嚴謹。

## 2. 建立你的第一個腳本

1. 到 Project 視窗，進入 `Scripts` 資料夾 (好習慣)。
2. 右鍵 -> `Create` -> `C# Script`。
3. 立刻命名！輸入 `HelloWorld` 然後按 Enter。

**注意：檔案名稱必須跟裡面的 Class 名稱完全一樣！如果建立後才改檔名，程式會報錯。**

# 程式命名規則

為了避免錯誤，請遵守工程師的命名規範：

-  **PascalCase (大駝峰)**：單字首字母大寫。例如 `PlayerController`, `GameManager`。
-  **不要用中文**：雖然可用，但易出相容性問題。
-  **不要有空格**：電腦會讀錯。
-  **不要數字開頭**：`1stScript` 是錯的。

### 3. 認識編輯器

雙擊剛剛建立的腳本，會開啟 Visual Studio (或 VS Code)。

- **Intellisense (智慧感知) :**

打程式時會跳出選單讓你選，這是寫程式的好幫手。

- 如果你的程式碼全是白色的 (沒有顏色區分)，代表 Intellisense 沒運作。

(請檢查 Unity -> Preferences -> External Tools)。

## 4. 腳本結構拆解

打開腳本，你會看到這幾行預設代碼：

```
using UnityEngine; // 1. 引用函式庫 (工具箱)

// 2. 類別宣告 (一定要繼承 MonoBehaviour 才能掛在物件上)
public class HelloWorld : MonoBehaviour
{
    // 3. Start 事件：遊戲開始時執行一次
    void Start()
    {

    }

    // 4. Update 事件：每一格畫面執行一次 (約每秒 60 次)
    void Update()
    {

    }
}
```

## 5. 第一個指令：DEBUG.LOG

我們要叫電腦講話。在 `Start()` 的大括號 `{ }` 中間輸入：

```
void Start()
{
    Debug.Log("Hello World! 這是我的第一支程式");
}
```

- `Debug.Log`：叫 Unity 的控制台印出訊息。
- `()`：放參數的地方 (這裡是文字)。
- `""`：字串 (文字) 必需被雙引號包起來。
- `;`：分號。每一行指令結束都要加！(很重要)

## 6. 讓程式運作

只有寫完程式，它是不會動的。腳本只是「藍圖」。

1. 在編譯器按 **Ctrl + S** 存檔 (看檔名旁的星號消失)。
2. 回到 Unity Editor。
3. 選取場景上的一個物件 (例如 Player，或新增一個 Empty Object)。
4. 將腳本拖曳到該物件的 Inspector 上。

現在，這個物件擁有了這個腳本的大腦。

## 7. 驗證結果

1. 按下 Unity 上方的 Play (▶) 按鈕。
2. 切換視窗到左下角的 Console (或是 Window -> General -> Console)。
3. 你應該會看到一行字：

Hello World! 這是我的第一支程式

恭喜！你已經成功跟電腦對話了。

## 8. 實驗：START VS UPDATE

讓我們體驗一下 Update 的威力。修改程式碼：

```
void Update()
{
    Debug.Log("我正在持續運作..." + Time.time);
}
```

1. 存檔 -> 回 Unity -> Play。
2. 你會發現 Console 裡的訊息瘋狂跳動！
3. 因為 Update 每一秒鐘會執行約 60 次 (依電腦效能而定)。

結論：初始設定放 Start，持續偵測 (如按鍵、移動) 放 Update。

今天開啟了程式設計的大門：

- **建立腳本**：注意命名規則。
- **掛載腳本**：Script 也是一種 Component，要掛在物件上才會跑。
- **Debug.Log**：這是最基本的除錯工具。
- **Start / Update**：理解 Unity 的生命週期。

下一章，我們要學習如何用程式**控制變數**，讓遊戲數據活起來！