

## CHAPTER 13

# 手機觸控操作 (MOBILE INPUT) HORIZON

手遊程式設計

# 章節目標

- 手機沒有鍵盤 (WASD)，如何操作？
- 實作 **虛擬按鈕** (UI Button) 控制跳躍
- 實作 **螢幕搖桿** 控制移動
- 使用 Unity Remote 進行手機連線測試

# 1. 策略分析

要在手機上玩，我們需要把實體按鍵轉換成虛擬訊號。

- **跳躍**：好處理，做一個大顆的 UI 按鈕，按下去就跳。
- **移動**：比較麻煩，我們需要分別處理「按住左」與「按住右」的訊號。

## 2. 實作跳躍按鈕

1. 在 Canvas 右下角建立一個 UI Button，圖案換成跳躍圖示。
2. 在 `PlayerController` 加入一個**公開方法** (Public Method)：

```
// 給 UI 呼叫的跳躍功能
public void OnJumpBtnPressed()
{
    // 直接執行原本的跳躍邏輯
    if (IsGrounded())
    {
        rb.AddForce(Vector2.up * jumpForce);
    }
}
```

3. 設定 Button 的 On Click () 事件，綁定這個 Function。

### 3. 實作移動邏輯 (變數分離)

原本我們直接寫 `Input.GetAxis`，現在要把輸入值變成變數。

```
float inputH = 0; // 統一的輸入值 (-1 ~ 1)

void Update()
{
    // 預設讀取鍵盤 (方便在電腦測試)
    inputH = Input.GetAxisRaw("Horizontal");

    // ... (後面的移動程式碼都不用動，只要把原本的變數換成 inputH)
}
```

## 4. 製作左右移動按鈕

我們需要在外掛兩個功能來「強制修改」 `inputH` 的值。

```
// 當按住左鍵時呼叫
public void MoveLeft()
{
    inputH = -1;
}

// 當按住右鍵時呼叫
public void MoveRight()
{
    inputH = 1;
}

// 當放開按鈕時呼叫 (歸零)
public void StopMove()
{
    inputH = 0;
}
```

## 5. UI EVENT TRIGGER (事件觸發器)

普通的 Button 只有「點擊瞬間」，無法偵測「長按」。

我們需要 **Event Trigger Component**。

1. 建立左方向按鈕 (UI Image)。
2. Add Component -> **Event Trigger**。
3. Add New Event Type -> **Pointer Down (按下)** -> 綁定 `MoveLeft`。
4. Add New Event Type -> **Pointer Up (放開)** -> 綁定 `StopMove`。

(右方向按鈕以此類推)

## 6. UNITY REMOTE (手機連線測試)

每次改完都要 Build 成 APK 手機會瘋掉。

1. 在手機下載 Unity Remote 5 App。
2. 用 USB 線連接手機與電腦。
3. Unity -> Edit -> Project Settings -> Editor -> Device 改為你的手機。
4. 按下 Play。
5. **手機畫面會同步顯示電腦畫面！** (雖然畫質會變差，但可測觸控)。

# 總結

現在你的遊戲已經不需要鍵盤了！

- 透過 **UI 事件** 來驅動角色。
- 這是一種最基礎但也最穩定的做法。
- 進階做法可以使用 "On-Screen Stick" (虛擬搖桿)，需要引用 Unity 新版 Input System，原理大同小異。

下一章，最後一哩路：**輸出 APK，分享給朋友！**