

#### 大綱

- ❷ 建立 Kinect 應用專案基本步驟
- ❷ Kinect 相關類別及成員介紹
- ▼ Kinect 感應器連線偵測



#### 建立 Kinect 應用專案

- 建立專案使用者介面





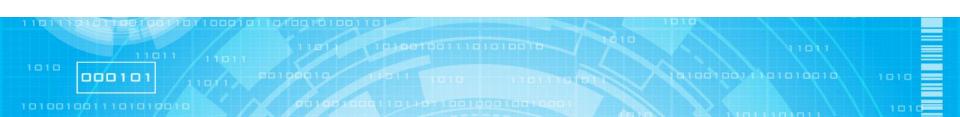


- ※ KinectSeneor: 代表一個 Kinect 感應器
  - Status: 屬性
  - KinectSensors: 屬性
  - ColorStream: 屬性
  - Start: 方法
  - Stop: 方法
  - ColorFrameReady: 事件
  - AllFramesRead: 事件





- ※ KinectSensorCollection: 包含所有連至電腦 Kinect 感應器的集合
  - StatusChanged: 事件
  - Count: 集合基底類別的屬性
- Status Changed Event Args: Kinect 感應器 狀態改變事件參數
  - Sensor: 參考至狀態改變的 Kinect 感應器
  - Status: 參考至狀態改變 Kinect 感應器目前狀態



- ✓ KinectStatus: 表示 Kinect 感應器狀態的列舉值,列舉內容如下
  - Connected
  - DeviceNotGenuine
  - DeviceNotSupported
  - Disconnedted
  - Error
  - Initializing
  - InsufficientBandwidth
  - NotPowered
  - NotReady



- - Format: 屬性
  - Enable: 方法
  - OpenNextFrame
  - FrameWidth
  - FrameHeight
  - FramePixelDataLength



- ② ColorImageFrame: 包含一個存放來自 Kinect 感應器 RGB 資料串流的暫存器
  - Format: 屬性
  - PixelDataLength: 屬性
  - CopyPixelDataTo: 方法
  - BytesPerPixel: 繼承自 ImageStream 類別的屬性
- - OpenColorImageFrame: 方法
- ❷ PixelFormats: 表示所支援的像素格式集合
  - Bgr32: 屬性

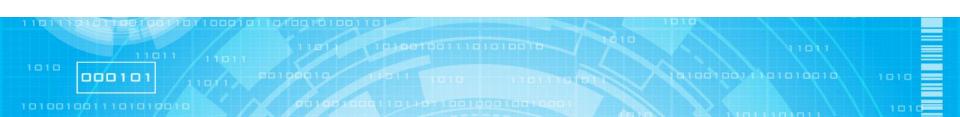


- ✔ Kinect 應用程式開發要從 KinectSensor 這個類別開始,這個類別實作的物件直接參考到連至電腦的 Kinect 感應器,透過它實作的物件就能存取來自 Kinect 感應器的影像與聲音資料。
- ※ Kinect SDK 允許同時有多個 Kinect 感應器 連至電腦。這些連到電腦的 Kinect 感應器可 以透過 KinectSensorCollection 類別,用集 合索引來存取。



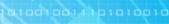


- 本範例為應用程式一開始執行時先偵測是否
   有 Kinect 感應器連至電腦
- Step 01: 啟動 Visual Studio 2017
- Step 03: 設定新專案類型為 Visual C# 的
   WPF 應用程式
- Step 04: 在專案中新增對應的 Kinect SDK 參考



- Step 05: 參考管理員中,選擇組件/擴充功能分類,加入 Microsoft.Kinect
- Microsoft.Kinect
- ❷ Step 07: 專案使用者介面建立,加入 Label 與 TextBox 控制項
- ♥ Step 08: 完成設計後的視窗定義 XAML 文





Step 09: 首先在原有的 using 敘述下增加對

Microsoft.Kinect 命名空間引用

using Microsoft.Kinect;

Step 10: 在類別層級,宣告用於參考

Kinect 的欄位

KinectSensor \_myKinect;

Step 11: 在 StatusChanged 事件中新增事件處理常式

KinectSensors.KinectSensors.StatusChanged += KinectSensors\_StatusChanged;



❷ Step 12: 建立事件處理常式並定義它的功能

```
private void KinectSensors_StatusChanged(object sender, StatusChangedEventArgs e)
      switch (e.Status)
        case KinectStatus.Initializing:
          StatusTxt.Text = "Initializing";
          break:
        case KinectStatus.Connected:
          StatusTxt.Text = "Connected";
          this. myKinect = KinectSensor.KinectSensors[0];
          break:
        case KinectStatus.Disconnected:
          StatusTxt.Text = "Disconnected";
          break;
        case KinectStatus.NotReady:
          StatusTxt.Text = "NotReady";
          break:
        case KinectStatus.Undefined:
          StatusTxt.Text = "Undefined";
          break:
        case KinectStatus.NotPowered:
          StatusTxt.Text = "NotPower":
          break;
        default:
          StatusTxt.Text = "Not Sure";
          break:
```



#### Step 13: 定義執行時 Loaded 事件

```
private void Window_Loaded_1(object sender, RoutedEventArgs e)
      if (KinectSensor.KinectSensors.Count == 0)
        MessageBox.Show("請將 Kinect 接上電腦");
      else if (KinectSensor.KinectSensors[0].Status == KinectStatus.Connected)
        this._myKinect = KinectSensor.KinectSensors[0];
```





#### Kinect 感應器連線偵測測試



