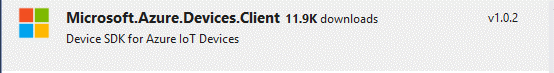
## 說明

* 本節會透過Web Job專案撰寫一隻程式，此程式會不斷地向IOT Hub詢問是否有Cloud-to-Device訊息，如果有，會將訊息儲存在APP\_Data下。FieldGateway的ReceiveCommand API會到APP\_DATA查詢Device ID的訊息並回傳給前端裝置。

## 步驟

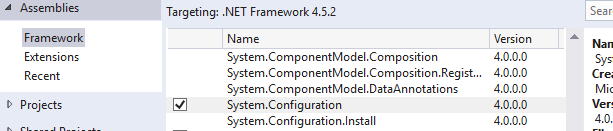
* 新增一個WebJob專案，並加入以下的NUGet參考







* 加入以下參考



* 在App.config加入Storage Account連線字串，用來儲存Azure Web Job Dashboard與Log資訊



* 修改Functions.cs如下；其中
  + iotHubOwnerConnectionString為IOT Hub的Owner權限之連線字串

public class Functions

{

/// <summary>

/// In this sample gateway, device information stored at \APP\_DATA\{deviceId}; This function get all device information from the folder

/// </summary>

/// <returns></returns>

public async static Task<string[]> GetAllDevicesAsync()

{

var connString = System.Configuration.ConfigurationManager.ConnectionStrings["iotHubOwnerConnectionString"];

RegistryManager mgr = RegistryManager.CreateFromConnectionString(connString.ConnectionString);

var hostname = connString.ConnectionString.Split(';')[0].Split('=')[1];

var devices = await mgr.GetDevicesAsync(1000);

return devices.Select(d => $"HostName={hostname};DeviceId={d.Id};SharedAccessKey={d.Authentication.SymmetricKey.PrimaryKey}").ToArray();

}

private static void WriteFile(string deviceId,string log)

{

var root = Environment.GetEnvironmentVariable("WEBROOT\_PATH");

var fn = System.IO.Path.Combine(root, $"App\_Data\\{deviceId}.txt");

if (!System.IO.File.Exists(fn))

{

using (var sw = File.CreateText(fn))

{

sw.Write(log);

}

}

else

{

File.AppendAllText(fn, log + Environment.NewLine);

}

}

public static async Task ReceiveC2DAsync()

{

while (true)

{

var conns = await GetAllDevicesAsync();

if (conns != null && conns.Length > 0)

{

foreach (var conn in conns)

{

Thread.Sleep(1000);

try

{

Console.WriteLine($"Checking [{conn}]...");

DeviceClient deviceClient = DeviceClient.CreateFromConnectionString(conn);

var cmd = await deviceClient.ReceiveAsync(TimeSpan.FromSeconds(1000));

if (cmd != null)

{

Console.WriteLine($"...Got message");

using (var sr = new StreamReader(cmd.GetBodyStream()))

{

var msg = sr.ReadToEnd();

var deviceId = conn.Split(';')[1].Split('=')[1];

WriteFile(deviceId, msg);

Console.WriteLine($"\*\*\* Received:{msg}");

}

await deviceClient.CompleteAsync(cmd);

}

else

{

Console.WriteLine("...no message");

}

}

catch (Exception exp)

{

Console.WriteLine($"Error handling {conn}");

Console.WriteLine($"Exception:{exp.Message}");

}

}

}

}

}

* 修改Program.cs，在main()中新增以下程式碼

var host = new JobHost();

Functions.ReceiveC2DAsync();

// The following code ensures that the WebJob will be running continuously

host.RunAndBlock();

* 將WebJob佈署到Web Host中