

委派

[Day8] 委派(delegate), Action<T> , 與Func<T,TResult> - iT 邦幫忙::一起幫忙解決難題, 拯救 IT 人的一天

什麼是委派 在一開始接觸到這個名詞時 我也是一頭霧水 讓我們來看看官方文件怎麼寫 委派是一種類型, 代表具有特定參數清單及傳回型別的方法參考。當您具現化委派時, 可以將其執行個體與任何具有>>...

 <https://ithelp.ithome.com.tw/articles/10234953>

=====

```
using System;          //引用系統類別
using System.Linq.Expressions;
using System.Text;

// Declare a delegate
// 委派 具名方法
// 委派時 會接收兩個值
public delegate void Del(int i, double j);

class MathClass
{
    // 建立一個不回傳值、只執行Console.WriteLine的函式
    public void MultiplyNumbers(int m, double n)
    {
        // 印出 兩數相乘
        Console.WriteLine(m * n);
    }

    // 主程式
    static void Main()
    {
        // 實體化
        MathClass m = new MathClass();

        // 原本應該寫成 m.MultiplyNumbers(i, 2);
        // 將 m.MultiplyNumbers; 委派 給 d, d代替m.MultiplyNumbers去接收值(i, 2)
        Del d = m.MultiplyNumbers;

        // Invoke the delegate object.
        Console.WriteLine("Invoking the delegate using 'MultiplyNumbers':");

        // for迴圈 i從1~5
        for (int i = 1; i <= 5; i++)
        {
            //如果不使用委派,應該要這樣寫 ↓
            //m.MultiplyNumbers(i, 2);

            // 使用委派d,用來代替m.MultiplyNumbers接收值
            d(i, 2);
        }

        // Keep the console window open in debug mode.
    }
}
```

```

        Console.WriteLine("Press any key to exit.");
        Console.ReadKey();

    }
}

```

```

Microsoft Visual Studio 偵錯主控台
Invoking the delegate using 'MultiplyNumbers':
2
4
6
8
10
Press any key to exit.

```

練習：

判斷奇數、偶數

```

using System;           //引用系統類別
using System.Linq.Expressions;
using System.Text;

// Declare a delegate
// 委派 具名方法
// 委派時 會接收一個值(該值資料型態是浮點數double)
public delegate void Del(double i);

class MathClass
{
    // 建立一個不回傳值、只執行判斷奇數、偶數的函式
    public void consider_Numbers(double n)
    {
        // 如果除以2, 餘數是0, 表示是偶數
        if (n % 2 == 0) {
            Console.WriteLine("偶數");
        }

        // 反之奇數
        else
        {
            Console.WriteLine("奇數");
        }
    }
}

// 主程式
static void Main()
{
    // 實體化
    MathClass m = new MathClass();

    // 原本應該寫成 m.consider_Numbers(num_in外部輸入一個數);
    // 將 m.consider_Numbers; 委派 給 d, d代替m.consider_Numbers去接收值(num_in)
}

```

```

Del d = m.consider_Numbers;

// Invoke the delegate object.
Console.WriteLine("Invoking the delegate using 'MultiplyNumbers':");

// cmd輸入提示訊息
Console.WriteLine("Enter a number:");

//從控制台讀取字串後進行換行
string num_in = Console.ReadLine();

// 字串 轉換成float數值
double num = float.Parse(num_in);

//如果不使用委派，應該要這樣寫 ↓
m.consider_Numbers(num);

// 使用委派d，用來代替m.consider_Numbers接收值
//d(num);

// Keep the console window open in debug mode.
Console.WriteLine("Press any key to exit.");
Console.ReadKey();

    }
}

```