

## 物件繼承與委派

繼承 (Inherits) 是物件導向程式設計中非常重要的機制。繼承是從既有的類別來建立新的類別，繼承後的類別稱為子類別 (Child Class) 或衍生類別 (Derived Class)，而被繼承的類別稱為父類別 (Parent Class) 或基底類別 (Base Class)。

委派 (Delegate) 是將程序或函數當做參數傳遞給另外一個程序，它其實是將指標指向實際執行程序的位址。

所謂泛型 (Generics)，是指可利同一組相同的程式碼，就能對不同資料型別執行相同的功能。泛型允許設計者以未定型別的參數來建立類別、程序等，待實際要使用該泛型時，再指定參數的型別。

### 學習重點

---

- 繼承
- 覆寫
- 多載
- 遮蔽
- 介面與實作
- 委派
- 部份類別
- 泛型

## 7.1 繼承

繼承 (Inherits) 是物件導向程式設計中非常重要的機制。繼承是從既有的類別來建立新的類別，繼承後的類別稱為子類別 (Child Class) 或衍生類別 (Derived Class)，而被繼承的類別稱為父類別 (Parent Class) 或基底類別 (Base Class)。

### 7.1.1 建立子類別

要完成這個動作，只要在子類別中加入 Inherits 關鍵字即可，Inherits 陳述式必須是類別陳述式 (Public Class) 之後的第一個非註解陳述式。

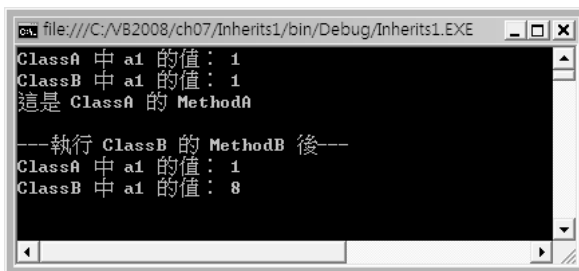
```
Public Class 父類別名稱
    程式碼
End Class
Public Class 子類別名稱
    Inherits 父類別名稱
    程式碼
End Class
```

子類別會繼承父類別中宣告為 Public 及 Protected 的成員 (包括屬性、方法和事件)，但不會繼承宣告為 Private 的成員。

### 範例：建立子類別

定義兩個類別 (父、子)：第一個類別 ClassA (父) 是具有一個方法和三個欄位的基底類別，第二個類別 ClassB (子) 從基底類別 ClassA 繼承方法 MethodA() 和非 Private 欄位 a1、a3，並新增 b1 欄位和方法 MethodB()。(Inherits1.sln)

執行結果



```
file:///C:/VB2008/ch07/Inherits1/bin/Debug/Inherits1.EXE
ClassA 中 a1 的值： 1
ClassB 中 a1 的值： 1
這是 ClassA 的 MethodA
---執行 ClassB 的 MethodB 後---
ClassA 中 a1 的值： 1
ClassB 中 a1 的值： 8
```

(可以開啟 <C:\VB2008\ch07\完成參考\Inherits1\Inherits1.sln> 專案觀看結果)

### 動手實作

1. 新增專案：新增 **主控台應用程式** 專案，專案名稱為「Inherits1」。
2. 輸入下列程式碼。

```

2 Public Class ClassA '父類別 A
3     Public a1 As Integer = 1
4     Private a2 As Integer = 2
5     Protected a3 As Integer = 3
6     Sub MethodA() '宣告 Public 方法
7         Console.WriteLine("這是 ClassA 的 MethodA")
8     End Sub
9 End Class
10 Public Class ClassB '子類別
11     Inherits ClassA '繼承 ClassA 父類別
12     Public b1 As Integer '宣告新的 Public b1
13     Sub MethodB() '宣告新的 Public 方法
14         a1 = 8 '子類別可以繼承父類別中 Public、Protected 成員
15         a3 = 7
16         'a2 = 5 '但不會繼承 Private 的成員，所以 a2=5 會出現錯誤
17     End Sub
18 End Class
19
20 Sub Main()
21     Dim ObjectA As New ClassA
22     Dim ObjectB As New ClassB
23     Console.WriteLine("ClassA 中 a1 的值： {0}", ObjectA.a1)
24     'ObjectA.a1=1
25     Console.WriteLine("ClassB 中 a1 的值： {0}", ObjectB.a1)
26     'ObjectB.a1=1
27     Console.WriteLine("ClassB 中 a1 的值： {0}", ObjectB.a3) '會
28     產生錯誤
29     ObjectB.MethodA() '顯示「這是 ClassA 的 MethodA」
30     Console.WriteLine()
31     ObjectB.MethodB()
32     Console.WriteLine("---執行 ClassB 的 MethodB 後---")
33     Console.WriteLine("ClassA 中 a1 的值： {0}", ObjectA.a1)
34     'ObjectA.a1=1
35     Console.WriteLine("ClassB 中 a1 的值： {0}", ObjectB.a1)
36     'ObjectB.a1=8
37     Console.Read()
38 End Sub

```

## 3. 儲存專案及執行。

## 程式說明

- 2-9 建立父類別。
- 4 Private 成員 a2 僅能在 ClassA 類別本身使用。
- 5 a3 因為是 Protected 型態，僅可在 ClassA 類別本身及其子類別 ClassB 中使用。
- 10-18 建立子類別。
- 11 設定 ClassB 繼承 ClassA。
- 14-15 a1 及 a3 是繼承自 ClassA 的欄位。
- 16 無法繼承 Private 的成員，ClassB 沒有 a2 欄位，所以此列設定 a2 的值會出現錯誤。
- 23 顯示 ClassA 的 a1 欄位值。
- 24 顯示 ClassB 的 a1 欄位值。注意：ClassB 並未宣告 a1 欄位，而是繼承自 ClassA，所以與 ClassA 的 a1 欄位值相同。
- 25 a3 是 Protected 型態，無法在 Main() 函式中使用，所以執行此列會產生錯誤。
- 26 MethodA 也是繼承自 ClassA。
- 28 MethodB 會改變 a1 及 a3 的欄位值。
- 30 執行 MethodB 改變的是 ClassB 中 a1 的欄位值，所以 ClassA 的 a1 欄位值不變。
- 31 執行 MethodB 後 ClassB 中 a1 的欄位值成為新值。

雖然 ClassB 繼承 ClassA，但 ClassA 中的 Private 成員 a2 僅可在 ClassA 類別本身使用，並未被繼承，所以在 ClassB 中使用 a2 將會產生錯誤。ClassA 中的 Protected 成員 a3 可在 ClassA 及繼承的 ClassB 類別中使用，但在 ClassA 及繼承的 ClassB 類別以外的地方使用將會產生錯誤。

類別成員若未加修飾詞，其存取範圍依成員而異：類別的欄位，若省略修飾詞宣告，欄位視為 Private；類別的程序或函式，若省略修飾詞宣告，程序或函式視為 Public。

例如：

```
Public Class Car
    Dim Speed As Integer
    Sub Turbo()
        程式碼
    End Sub
End Class
```

Speed 存取範圍為 Private，無法被繼承；而 Turbo 方法存取範圍為 Public，繼承的子類別可使用此方法。

## 7.1.2 子類別的應用

繼承最大的好處是可以由已存在的類別來擴充自己所需的功能，原來就設計好的部分可以直接使用，設計者只需專心於擴充的部分即可。例如：若已存在一個計算長方形面積的類別，現在需要計算三角形面積，由於兩者的計算資料都是寬與高，可以直接繼承此長方形面積類別，新增一個計算三角形面積的公式即可。完成的新類別，不但可以計算三角形面積，也可以計算長方形面積，真是一舉兩得。

### 範例：計算面積

先定義 Rectangle 類別，並宣告公用欄位 Width 及 Protected 欄位 Height，再定義 GetArea() 方法來計算矩形面積。然後再定義 Triangle 類別，Triangle 類別繼承 Rectangle 類別並增加一個計算三角形面積的方法 GetArea2()。(Inherits2.sln)

執行結果



(可以開啟 <C:\VB2008\ch07\完成參考\Inherits2\Inherits2.sln> 專案觀看結果)

動手實作

1. 新增專案：新增 **主控台應用程式** 專案，專案名稱為「Inherits2」。
2. 輸入下列程式碼。

```

2 Class Rectangle    '定義 Rectangle 父類別
3     Public Width As Integer    '公用變數 Width
4     Protected Height As Integer    'Protected 變數 Height
5     Public Function GetArea() As Integer '計算矩形面積
6         GetArea = Width * Height
7     End Function

```

```
8 End Class
9 Public Class Triangle '子類別
10     Inherits Rectangle '繼承 Rectangle 父類別
11     Sub New()
12         Width = 5 '子類別可繼承父類別中 Public、Protected 的成員
13         Height = 6
14     End Sub
15     Public Function GetArea2() '計算三角形面積
16         GetArea2 = (Width * Height) / 2
17     End Function
18 End Class
19
20 Sub Main()
21     Dim obj1 As New Triangle
22     Console.WriteLine("矩形面積 = " & obj1.GetArea()) '矩形面積=30
23     Console.WriteLine("-----")
24     Console.WriteLine("三角形面積 = " & obj1.GetArea2()) '三角形面
        積=15
25     Console.Read()
26 End Sub
```

### 3. 儲存專案及執行。

#### 程式說明

- 2-8 計算長方形面積的類別。
- 5-7 計算長方形面積的方法。
- 9-18 計算三角形面積的類別。
- 10 宣告繼承 Rectangle 類別。
- 11-14 在建立物件時設定寬與高的初值。
- 13 由於 Height 是 Protected 欄位，無法在 Main() 函式中設定，所以要在  
此處設定欄位值。
- 15-17 計算三角形面積的方法。
- 21 建立 Triangle 類別的物件。
- 22 顯示長方形面積。
- 24 顯示三角形面積。

## 7.1.3 MustInherit 及 NotInheritable 修飾詞

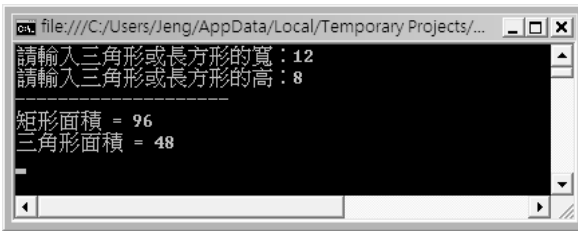
當父類別被子類別繼承後，要執行父類別中的方法可以使用父類別的物件，也可以使用子類別的物件。如果要保護父類別不再被使用，可以將父類別宣告為 **MustInherit**，則此父類別將不可以直接以 **New** 的方式來建立物件實體，此時若要使用父類別中的成員，只能在被繼承之後再於子類別中使用。

## 範例：動態計算面積

與前一範例相同，此處的寬與高可由使用者自行輸入，然後會計算長方形及三角形的面積。

(MustInherits1.sln)

執行結果



(可以開啟 <C:\VB2008\ch07\完成參考\MustInherits1\MustInherits1.sln> 專案觀看結果)

動手實作

1. 新增專案：新增 **主控台應用程式** 專案，專案名稱為「MustInherits1」。
2. 輸入下列程式碼。

```

2 MustInherit Class Rectangle    '定義 Rectangle 父類別要被繼承後才能
    使用
3     Public Width As Integer
4     Public Height As Integer
5     Public Function GetArea() As Integer '計算矩形面積
6         GetArea = Width * Height
7     End Function
8 End Class
9 Public Class Triangle    '子類別
10    Inherits Rectangle    '繼承 Rectangle 父類別
11    Public Function GetArea2()    '計算三角形面積
12        GetArea2 = (Width * Height) / 2
13    End Function
14 End Class
15
16 Sub Main()
17     Dim obj1 As New Triangle
18     Console.WriteLine("請輸入三角形或長方形的寬：")
19     obj1.Width = Console.ReadLine()
20     Console.WriteLine("請輸入三角形或長方形的高：")
21     obj1.Height = Console.ReadLine()
22     Console.WriteLine("-----")

```

```
23 Console.WriteLine("矩形面積 = " & obj1.GetArea())
24 Console.WriteLine("三角形面積 = " & obj1.GetArea2())
25 Console.ReadLine()
26 ' 因為 Rectangle 要被繼承後才能使用，所以下一行的宣告將出現錯誤
27 ' Dim b As New Rectangle
28 End Sub
```

### 3. 儲存專案及執行。

#### 程式說明

```
2 使用 MustInherit 宣告類別。
4 為了讓使用者自行輸入寬與高的數值，所以兩者都要設為 Public 欄位，
   Protected 欄位無法在 Main() 函式中使用。
17 使用 Triangle 子類別建立物件。
18-19 使用者自行輸入寬的數值。
20-21 使用者自行輸入高的數值。
23 可在子類別物件中執行父類別的 GetArea() 方法。
27 如果使用 New 來建立父類別會產生錯誤訊息。
```

若類別不想被繼承，可為類別加上修飾詞 `NotInheritable`，該類別就無法被繼承；如果程式中繼承該類別，則會產生錯誤。例如：

```
NotInheritable Class Triangle
```

類別被繼承後，父類別中的 `Public` 及 `Protected` 成員均會被繼承，同時也可以對父類別中不足的功能作擴充或調整。當然子類別在進行擴充或調整時也要考慮原來以父類別撰寫的程式碼，是否仍舊相容原有父類別，同時又能提供子類別更新更強的功能以供未來的開發。

要完成這樣的機制，常用的方式有多載 (Overloads)、覆寫 (Overrides) 和遮蔽 (Shadows)。



## 7.2 多載

多載 (Overloads) 又稱同名異式，即具有相同名稱但不同參數的程序或函式。如此一來，就可以使用相同的程序或函式名稱來完成多樣的功能，只要傳遞的參數不同即可。

### 7.2.1 不同參數的多載

大部分多載實程序是以參數來區分，製作多載程序的方法是在程序或函式前加上 `Overloads` 關鍵字。父類別和子類別中最多可以省略一個 `Overloads` 關鍵字。

例如：

---

```
Public Overloads Sub Show(ByVal x As Integer, ByVal y As Integer) '
    多載 Show() 函式
```

---

#### 範例：不同參數的多載

父類別 `Employee` 可用 `Show()` 方法顯示本薪，子類別 `Manager` 繼承 `Employee` 父類別，並且多載 `Show()` 函式可以顯示本薪和獎金。(Overloads1.sln)

執行結果



```
Employee 薪資 = 20000
-----
Manager 薪資<本薪+獎金> = 35000
```

(可以開啟 <C:\VB2008\ch07\完成參考\Overloads1\Overloads1.sln> 專案觀看結果)

動手實作

1. 新增專案：新增 **主控台應用程式** 專案，專案名稱為「Overloads1」。
2. 輸入下列程式碼。

---

```
2 Public Class Employee '定義 Employee 類別
3     Public Salary As Integer
4     Public Overloads Sub Show(ByVal x As Integer) '
5         '父類別和子類別中最多可以省略一個 Overloads 關鍵字
6         'Public Sub show(ByVal x As Integer) 亦可
```

---

```

7         Console.WriteLine("Employee 薪資 = {0}", x)
8         Console.WriteLine("-----")
9     End Sub
10 End Class
11 Public Class Manager '定義 Manager 類別
12     Inherits Employee '繼承 Employee 類別
13     Public Bonus As Integer
14     Public Overloads Sub Show(ByVal x As Integer, ByVal y As Integer)
15         '多載 Show() 函式
16         Console.WriteLine("Manager 薪資(本薪+獎金) = {0}", x + y)
17     End Sub
18 End Class
19 Sub Main()
20     Dim obj1 As New Employee
21     obj1.Salary = 20000 '本薪 20000 元
22     obj1.Show(obj1.Salary) '薪資=20000
23     Dim obj2 As New Manager
24     obj2.Salary = 30000 '本薪 30000 元
25     obj2.Bonus = 5000 '獎金 5000 元
26     obj2.Show(obj2.Salary, obj2.Bonus) '薪資(本薪+獎金)=35000
27     Console.ReadLine()
28 End Sub

```

### 3. 儲存專案及執行。

#### 程式說明

```

4     父類別中的多載函式 Show()，只有一個參數。
14    子類別中的多載函式 Show()，有兩個參數。
6     第 4 或 11 列其中一個 Overloads 關鍵字可以省略。要注意只能省略其中一
    個，若兩個都省略則會產生錯誤。
22    傳遞一個參數，所以使用 Employee 父類別中的 Show() 方法顯示本薪。
26    傳遞兩個參數，所以使用 Manager 子類別中的 Show() 方法顯示本薪。

```

## 7.2.2 相同參數的多載

有一種較特殊的情況是子類別的多載函式參數和父類別完全一樣，此時父類別將使用父類別的函式，而子類別則使用子類別中的函式。

### 範例：相同參數的多載

建立子類別和父類別使用相同參數的多載函式 Show()。(Overloads2.sln)

## 執行結果

```

file:///C:/VB2008/ch07/Overloads2/bin/Debug/Overloads2...
Employee 薪資 = 20000
-----
Manager 薪資<本薪+獎金> = 35000

```

(可以開啟 <C:\VB2008\ch07\完成參考\Overloads2\Overloads2.sln> 專案觀看結果)

## 動手實作

1. 新增專案：新增 **主控台應用程式** 專案，專案名稱為「Overloads2」。
2. 輸入下列程式碼。

```

2 Public Class Employee '定義 Employee 類別
3     Public Salary As Integer
4     Public Sub Show(ByVal x As Integer)
5         Console.WriteLine("Employee 薪資 = {0}", x)
6         Console.WriteLine("-----")
7     End Sub
8 End Class
9 Public Class Manager '定義 Manager 類別
10    Inherits Employee '繼承 Employee 類別
11    Public Bonus As Integer
12    Sub New()
13        Bonus = 5000 '獎金 5000 元
14    End Sub
15    Public Overloads Sub Show(ByVal x As Integer) '多載 Show() 函
        式,且參數完全相同
16        Console.WriteLine("Manager 薪資(本薪+獎金) = {0}", x +
            Bonus)
17    End Sub
18 End Class
19
20 Sub Main()
21    Dim obj1 As New Employee
22    obj1.Salary = 20000 '本薪 20000 元
23    obj1.Show(obj1.Salary) '薪資=20000
24    Dim obj2 As New Manager
25    obj2.Salary = 30000 '本薪 30000 元
26    obj2.Show(obj2.Salary) '薪資(本薪+獎金)=35000
27    Console.ReadLine()
28 End Sub

```

### 3. 儲存專案及執行。

#### 程式說明

- 4 父類別中的多載函式 Show()，省略 Overloads 關鍵字，有一個參數。
- 5 父類別中的多載函式 Show() 只顯示薪資。
- 12-14 建立物件時設定獎金數值。
- 15 子類別中的多載函式 Show()，也是一個參數。
- 16 子類別中的多載函式 Show() 顯示薪資和獎金。
- 23 由 Employee 類別建立的物件，使用 Employee 類別中的 Show() 方法顯示本薪。
- 26 由 Manager 類別建立的物件，使用 Manager 類別中的 Show() 方法顯示本薪和獎金。

## 7.2.3 保留父類別的多載程序

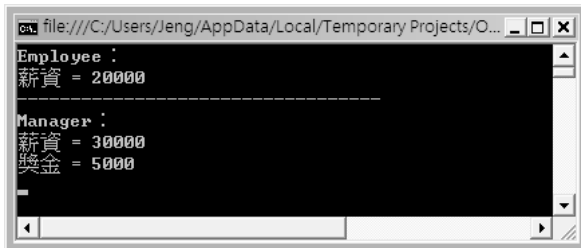
如果子類別的多載程序會使用到原有父類別多載程序功能，所以想保留原有父類別的多載程序，並在子類別中擴充新的功能，而不必重新撰寫父類別就具有的功能，如此可以快速建立功能強大的多載程序。

要保留父類別的多載程序，只要在子類別的多載程序中使用 `MyBase` 代替父類別名稱，即可在子類別中使用父類別的多載程序。

### 範例：保留父類別的多載

父類別中的多載程序 `Show()` 可以顯示薪資，子類別中的多載程序 `Show()` 除了顯示薪資外還要顯示獎金。於子類別的多載程序中使用 `MyBase` 可以執行父類別的 `Show()` 多載程序，再撰寫顯示獎金部分即可。(Overloads3.sln)

#### 執行結果



```
file:///C:/Users/Jeng/AppData/Local/Temporary Projects/O...
Employee :
薪資 = 20000
-----
Manager :
薪資 = 30000
獎金 = 5000
```

(可以開啟 <C:\VB2008\ch07\完成參考\Overloads3\Overloads3.sln> 專案觀看結果)

## 動手實作

1. 新增專案：新增 **主控台應用程式** 專案，專案名稱為「Overloads3」。
2. 輸入下列程式碼。

```

2 Public Class Employee '定義 Employee 類別
3     Public Salary As Integer
4     Public Sub Show(ByVal x As Integer)
5         Console.WriteLine("薪資 = {0}", x)
6     End Sub
7 End Class
8 Public Class Manager '定義 Manager 類別
9     Inherits Employee '繼承 Employee 類別
10    Public Bonus As Integer
11    Public Overloads Sub Show(ByVal x As Integer, ByVal y As Integer)
12        '多載 Show() 方法
13        MyBase.Show(x) '呼叫父類別的 Show() 方法
14        Console.WriteLine("獎金 = {0}", y)
15    End Sub
16 End Class
17 Sub Main()
18     Dim obj1 As New Employee
19     Console.WriteLine("Employee : ")
20     obj1.Salary = 20000 '本薪 20000 元
21     obj1.Show(obj1.Salary) '薪資=20000
22     Console.WriteLine("-----")
23     Dim obj2 As New Manager
24     Console.WriteLine("Manager : ")
25     obj2.Salary = 30000 '本薪 30000 元
26     obj2.Bonus = 5000 '獎金 5000 元
27     obj2.Show(obj2.Salary, obj2.Bonus) '薪資=30000,獎金=5000
28     Console.ReadLine()
29 End Sub

```

3. 儲存專案及執行。

## 程式說明

- 12 子類別 Manager 使用 MyBase.Show(x) 執行父類別的 Show() 方法。
- 13 子類別 Manager 除了繼承父類別 Employee 的 Show() 方法，也可再增加新的功能：顯示獎金。
- 27 使用 Manager 類別中的 Show() 方法顯示本薪和獎金。

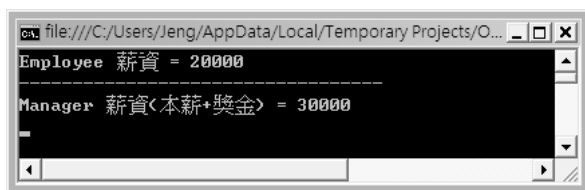
## 7.3 覆寫

覆寫 (Overrides) 是在子類別中以相同名稱取代原來父類別中的成員 (包括屬性及方法)，如果是覆寫的是「方法」，則子類別參數必須和原來父類別的參數完全相同。要完成覆寫必須在父類別加入 `Overridable` 關鍵字表示允許覆寫，同時也要在子類別加上 `Overrides` 進行覆寫。如果子類別中沒有加入 `Overrides` 關鍵字，或子類別中方法的參數和父類別中的參數不同，將會產生錯誤。

### 範例：覆寫父類別的「方法」

子類別以 `Overrides` 覆寫父類別的 `Show()` 方法。(Overrides1.sln)

執行結果



(可以開啟 <C:\VB2008\ch07\完成參考\Overrides1\Overrides1.sln> 專案觀看結果)

動手實作

1. 新增專案：新增 **主控台應用程式** 專案，專案名稱為「Overrides1」。
2. 輸入下列程式碼。

```
2 Public Class Employee '定義 Employee 類別
3     Public Salary As Integer
4     Public Overridable Sub Show(ByVal x As Integer)
5         Console.WriteLine("Employee 薪資 = {0}", x)
6         Console.WriteLine("-----")
7     End Sub
8 End Class
9 Public Class Manager '定義 Manager 類別
10    Inherits Employee '繼承 Employee 類別
11    Public Bonus As Integer
12    Public Overrides Sub Show(ByVal x As Integer) '覆寫 Show() 函
    式
13        Console.WriteLine("Manager 薪資(本薪+獎金) = {0}", x)
```

```
14     End Sub
15 End Class
16
17 Sub Main()
18     Dim obj1 As New Employee
19     obj1.Salary = 20000    '本薪 20000 元
20     obj1.Show(obj1.Salary) '薪資=20000
21     Dim obj2 As New Manager
22     obj2.Salary = 30000    '本薪 30000 元
23     obj2.Bonus = 5000     '獎金 5000 元
24     obj2.Show(obj2.Salary) '薪資(本薪+獎金)=35000
25     Console.Read()
26 End Sub
```

### 3. 儲存專案及執行。

#### 程式說明

```
4     父類別中以 Overridable 宣告 Show() 方法可以覆寫。
12    子類別中以 Overrides 覆寫 Show() 方法。
24    使用 Manager 類別中的 Show() 方法顯示本薪和獎金。
```