

15. Overloading

15.1 Overloading

Overloading(多載)，指得是在一個類別中，定義多個「名稱相同」但「引數個數不同」或是「引數型態不同」的method。Java語言可以依據所傳入的參數來判斷，使用者所要呼叫的是哪一個版本。請參考下面的例子：

```
class NumberBox
{
    int value;

    NumberBox()
    {
        value = 0;
    }

    int addIntValue(int a)
    {
        return value+=a;
    }

    int addFloatValue(float a)
    {
        return value+=(int)a;
    }

    int addDoubleValue(double a)
    {
        return value+=(int)a;
    }

    int addCharValue(char a)
    {
        return value+=(int)a;
    }
}
```

這個類別被設計成一個可存放一個整數並可進行相關操作的「數字盒」，其中addXXXValue()用以將現有的數值加上一個特定的數值。但這種做法，將使得我們在使用該類別的物件時，必須針對所要增加的數值

之型態，呼叫不同的method來加以執行。但若採用以下的方法：

```
class NumberBox
{
    int value;

    NumberBox()
    {
        value = 0;
    }

    int addValue(int a)
    {
        return value+a;
    }

    int addValue(float a)
    {
        return value+=((int)a);
    }

    int addValue(double a)
    {
        return value+=((int)a);
    }

    int addValue(char a)
    {
        return value+=((int)a);
    }
}
```

如此一來，當需要增加數值時，只需呼叫addValue()即可，它可以視所傳入的資料之型態的參異，自行判斷找到適當的「版本」來加以執行，這就叫做「Overloading(多載)」。

15.2 STUScoreMan: 課程的學生管理需求

現在讓我們在將注意力放回到STUScoreMan專案上，我們來增加一門課程可以管理其修課學生的成績這項需求。

From:

<https://junwu.nptu.edu.tw/dokuwiki/> - Jun Wu的教學網頁

國立屏東大學資訊工程學系

CSIE, NPTU

Total: 252809



Permanent link:

<https://junwu.nptu.edu.tw/dokuwiki/doku.php?id=java:overloading>

Last update: **2019/07/02 15:01**