

16. Interface

從語法上來看，Interface(介面)與類別非常相似，都可以定義相關的欄位(fields)及方法(methods)。請先參考以下的interface範例：

```
interface rule
{

}
```

不過和類別相比仍有以下三點不同：

1. 介面不可以用以產生物件實體，只能被加以實作；
2. 其欄位都必須宣告為具有初始值的靜態常數，例如 `public static final int x=5;`
3. 其方法只能進行原型的宣告，而不能提供實作。

物件在執行階段時，可以呼叫執行定義在其所屬類別裡的方法，例如一個Student類別的物件可以呼叫執行其isPass()與showInfo()等方法。

有時，我們會需要規範類別該有的行為或提供特定的操作，此時就可以使用Interface(介面)來「規範」類別行為的方法，不過介面僅定義「該有哪些行為(函式、方法)？」，具體的實作則是由選擇套用「介面」的類別負責提供。

interface定義語法

```
interface interfaceName {
```

```
    //field declarations
    [DataType
    variableName[=value]?[,variableName[=value]?]
    *;]*
```

```
    method declarations [returnType methodName(parameters)
    { method implementations
```

```
        statements
    }]*
```

```
}
```

不知道讀者有沒有換過燈泡的經驗？當一個燈泡壞掉的時候，我們只要依據燈泡底座的規格購買替換的燈泡即可，例如最常見的E27規格規範了螺口燈頭直徑為27mm，至於顏色、亮度等則不在規範之列。各個廠商可以自行決定生產不同顏色、亮度、材質等的燈泡，消費者只要確定其規格為E27就可以使用。Java語言的Interface就如同「產品的規格定義」，我們可用以規範類別的規格。

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: 251668

Permanent link:

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

Last update: **2025/09/04 13:27**

