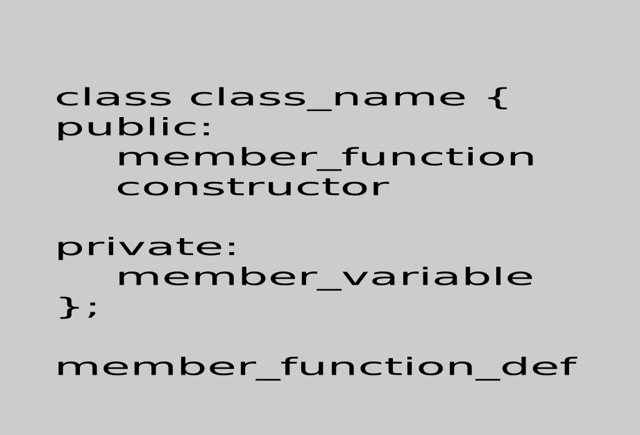
C++ 入門指南 - 類別

節錄自<http://pydoing.blogspot.tw/2012/10/cpp-class.html>

以下C++檔案，請參考 t15\_1.cpp

類別 (class) 用來設計自己需要的物件 (object) ，這是說，類別是物件的模板。 C++ 中設計類別使用關鍵字 (keyword) **class** ，後面接大括弧宣告 (declare) 類別的成員 (member)  
  
類別成員可以是資料 (data) 、函數 (function) 與建構子 (constructor) ，資料其實就是專屬於類別的變數 (variable) ，我們在這裡沿用 C++ 的習慣稱之為成員變數 (member variable) ，同樣的，函數也是專屬於類別的，稱之為成員函數 (member function) ，至於建構子屬於特別的成員函數，用來建立該類別物件的專屬函數。

因為建構子用來建立物件，所以建構子沒有回傳值，或著可以這麼想像，建構子預設回傳物件自己本身，因此無須宣告回傳值 (return value) 。

類別定義必須利用存取標籤 (access label) **public** 或 **private** 將成員歸類，屬於 **public** 的成員可以在程式中的任何地方存取，**private** 則只能在類別裡。

例題 (t15\_1.cpp)

#include <iostream>

 using namespace std;

 class Demo {

public:

    int a;

    int b;

    int do\_something();

}

int Demo::do\_something() {

    return a + b;

}

int main(void) {

    Demo t;

    t.a = 11;

    t.b = 22;

    cout << endl;

    cout << t.do\_something() << endl;

    cout << endl;

    return 0;

}

Demo 類別宣告三個 **public** 成員， *a* 、 *b* 為成員變數， *do\_something()* 為成員函數，記得， **class** 大括弧的最後要加上分號

|  |  |
| --- | --- |
|  | class Demo {  public:      int a;      int b;      int do\_something();  }; |

實作 Demo 類別的 *do\_something()* 成員函數

|  |  |
| --- | --- |
|  | int Demo::do\_something() {      return a + b;  } |

*do\_something()* 的工作很簡單，就是回傳兩個成員變數 *a* 與 *b* 的相加值。這裡須注意定義成員函數時，必須在**成員函數名稱前加上所屬類別名稱接兩個冒號**

|  |
| --- |
| Demo::do\_something() |

這是標明 *do\_something()* 是屬於 Demo 類別的，連續兩個冒號 :: 為作用域運算子 (scope operator) 。**沒有這樣寫的話，編譯器 (compiler) 會把 *do\_something()* 當成普通的函數處理**。  
  
  
下面的 main() 直接用小數點運算子設定 *a* 與 *b* ，然後印出呼叫 *do\_something()* 的結果

|  |  |
| --- | --- |
|  | int main(void) {      Demo t;      t.a = 11;      t.b = 22;        cout << endl;      cout << t.do\_something() << endl;      cout << endl;        return 0;  } |

編譯執行看看結果吧

