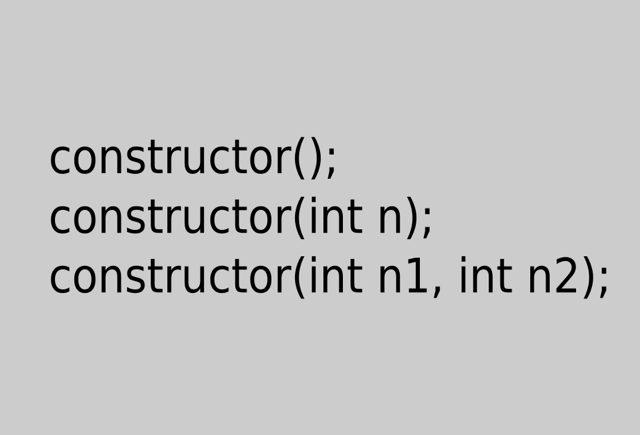
C++ 入門指南 - 建構子

<http://pydoing.blogspot.tw/2012/10/cpp-constructor.html>

以下C++檔案，請參考 t15\_4.cpp

函數 (function) 可以有多個參數 (parameter) 版本，建構子 (constructor) 亦同。多個參數版本的函數稱為多載函數 (overloaded function)  


函數只能有一個回傳值 (return value) ，卻可以有多個參數版本，這是因為很多時候都得處理不同的情況，不同情況需要的參數大多也不同。

由於 Demo 有兩個變數成員 *a* 與 *b* ，因此我們打算設置三種建構子的參數版本，分別是沒有參數、一個參數及兩個參數

|  |
| --- |
| Demo();  Demo(int n);  Demo(int n1, int n2); |

沒有參數的建構子直接將 *a* 與 *b* 都設定為整數 1 ，一個參數的建構子將 *a* 與 *b* 都設定為參數值，至於兩個參數的建構子， *a* 設定為第一個參數， *b* 設定為第二個參數

|  |
| --- |
| Demo::Demo() {      a = 1;      b = 1;  }    Demo::Demo(int n) {      a = n;      b = n;  }    Demo::Demo(int n1, int n2) {      a = n1;      b = n2;  } |

我們寫成完整範例如下

|  |  |
| --- | --- |
|  | #include <iostream>    using namespace std;    class Demo {  public:      Demo();      Demo(int n);      Demo(int n1, int n2);      void setA(int n);      void setB(int n);      int getA();      int getB();      int do\_something();    private:      int a;      int b;  };    Demo::Demo() {      a = 1;      b = 1;  }    Demo::Demo(int n) {      a = n;      b = n;  }    Demo::Demo(int n1, int n2) {      a = n1;      b = n2;  }    int Demo::do\_something() {      return getA() + getB();  }    void Demo::setA(int n) {      a = n;  }    void Demo::setB(int n) {      b = n;  }    int Demo::getA() {      return a;  }    int Demo::getB() {      return b;  }    int main(void) {      Demo t1;      Demo t2(11);      Demo t3(11, 22);        cout << endl;      cout << t1.do\_something() << endl;      cout << t2.do\_something() << endl;      cout << t3.do\_something() << endl;      cout << endl;        return 0;  } |

利用建構子建立物件 (object) 變數 (variable) ，宣告時加小括弧即可，需要留意沒有參數的建構子不需要加小括弧

|  |  |
| --- | --- |
|  | Demo t1;  Demo t2(11);  Demo t3(11, 22); |