



B

Dev-C++ 的使用

B-1 Dev-C++ 的基礎

程式語言的「開發環境」(Development Environment) 是一組工具程式用來建立、編譯和維護程式語言建立的應用程式。目前的高階程式語言大都擁有整合開發環境，稱為「IDE」(Integrated Development Environment)，在同一個應用程式就可以編輯、編譯和執行特定語言的應用程式。

B-1-1 Dev-C++ 整合開發環境

Dev-C++ 是 Bloodshed Software 公司的產品，一套功能強大的 C/C++ 語言整合開發環境，可以開發 Windows 和主控台應用程式 (Console Applications)，所謂主控台應用程式就是在 MS-DOS 或命令提示字元執行的文字模式應用程式。

》》》》 GCC 編譯程式

Dev-C++ 本身是使用 Borland Delphi 設計的應用程式整合開發環境，使用 MinGW (Minimalist GNU for Windows，網址：<http://www.mingw.org/>) 的 C/C++ 編譯程式，這是 UNIX 系統 GCC 編譯程式的 Windows 版本。

》》》》 下載 Dev-C++

Dev-C++ 是一套免費軟體，Bloodshed 軟體公司提供完整的 CD 系統光碟，可以向該公司網站訂購，在下載網頁可以免費下載 Dev-C++，其網址如下所示：

■ <http://www.bloodshed.net/dev/devcpp.html>

請捲動網頁到 Downloads 區段，按下載的超連結，就可以下載 Dev-C++ 5.0 版。

B-1-2 Dev-C++ 5.0 版的安裝

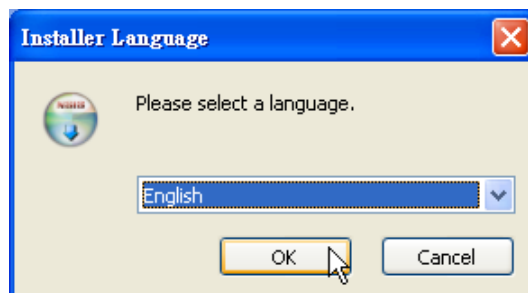
Dev-C++ 5.0 版系統需求是 32MB 記憶體，作業系統為 Windows 95 以上版本，因為 Dev-C++ 需要 MSVCRT.DLL 才能將 C/C++ 程式編譯成執行檔案，Windows 95 需要 OSR2 版本才能安裝且正確執行 Dev-C++。

在書附光碟提供 Dev-C++ 5.0 beta 9 版本，其安裝步驟如下所示：

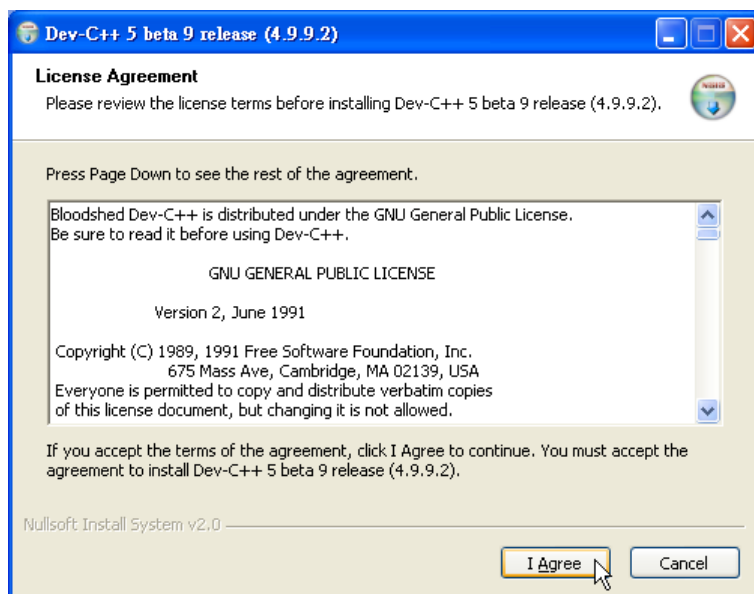
Step 1：請執行安裝程式【devcpp-4.9.9.2_setup.exe】安裝 Dev-C++，可以看到一個警告訊息，如下圖所示：



Step 2：訊息指出需要先解除舊版 Dev-C++ 後，才能安裝新版本，如果有安裝之前版本，請先解除安裝後，按【確定】鈕選擇安裝的語言。



Step 3：上述選擇是指安裝程式顯示的語言，因為目前並不支援中文，請選【English】，按【OK】鈕，可以看到軟體授權書。



Step 4：在捲動視窗閱讀授權書後，請按【I Agree】鈕同意授權，即可選擇安裝種類。

Step 5：請選【Full】完整安裝，按【Next】鈕可以選擇安裝路徑。

Step 6：預設安裝路徑是「C:\Dev-Cpp」，不用更改，按【Install】鈕，可以開始複製檔案和安裝 Dev-C++。

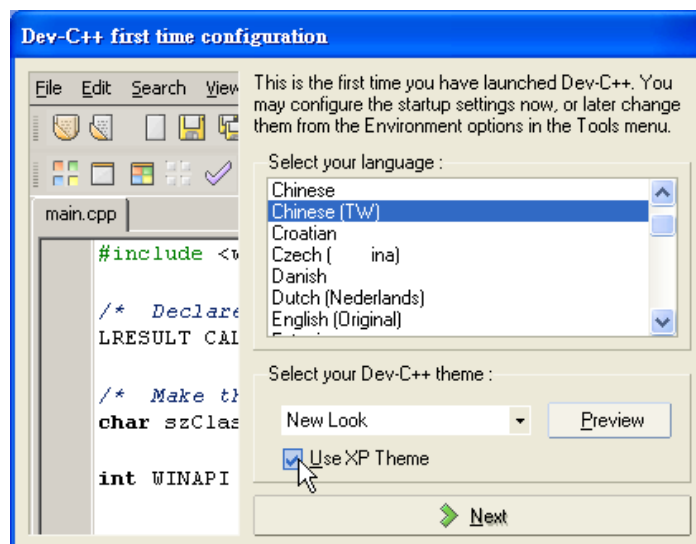
Step 7：在複製檔案的過程中，可能會看到一個訊息視窗，如下圖所示：



Step 8：訊息指出是否讓 Dev-C++ 安裝成給電腦所有使用者使用，按【是】鈕讓所有使用者使用，按【否】鈕只讓目前使用者使用。

Step 9：等到完成檔案複製和安裝後，按【Finish】鈕完成 Dev-C++ 5.0 版的安裝。

安裝程式最後一步驟預設會第 1 次啟動 Dev-C++ 5，可以看到訊息指出此為測試版本，如有任何程式錯誤，請通知 Bloodshed Software 公司網站，按【確定】鈕，可以看到 Dev-C++ 環境設定的精靈畫面。



在上方選中文【Chinese(TW)】，如為 XP 使用者，請在下方勾選【Use XP Theme】，按【Next】鈕可以最佳化 Dev-C++ 設定，不用更改，請直接按二次【Next】鈕開始環境設定，請稍等一下。

在完成設定後，按【OK】鈕即可啟動 Dev-C++，可以看到「每日提示」視窗，按【關閉】鈕進入 Dev-C++ 執行畫面。請執行「檔案」→「結束程式」指令結束 Dev-C++。

B-2 Dev-C++ 的使用

在安裝好 Dev-C++ 整合開發環境後，我們就可以啟動 Dev-C++ 來開發 C 應用程式，應用程式開發的基本步驟，如下所示：

Step 1：啟動 Dev-C++，新增原始碼檔案。

Step 2：輸入程式碼，建立和儲存 C 原始程式碼檔案，副檔名為.c。

Step 3：執行「執行」→「編譯並執行」指令，編譯並執行 C 程式。

》》》 步驟一：建立 C 程式檔案

Dev-C++ 是一套整合開發環境，在啟動 Dev-C++ 後，就可以編輯、編譯和執行 C 應用程式。建立 C 語言程式檔案的步驟，如下所示：

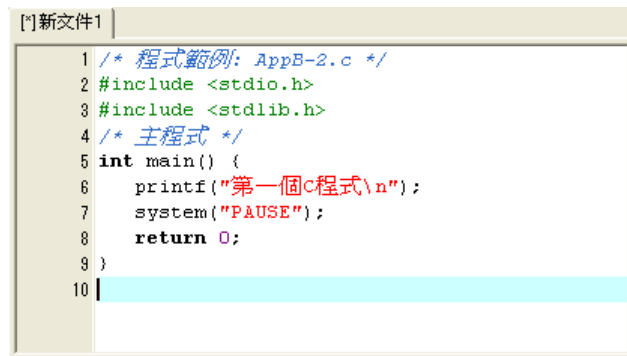
Step 1：請執行「開始」→「所有程式」→「BloodshedDev-C++」→「Dev-C++」指令啟動 Dev-C++。

Step 2：執行「檔案」→「開新檔案」→「原始碼」指令，可以新增名為新文件 1 的程式檔案，看到【新文件 1】標籤的編輯視窗。

》》》 步驟二：編輯和儲存程式碼

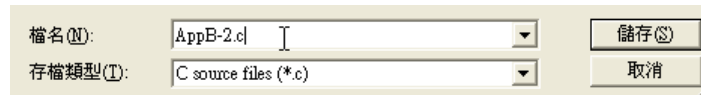
在新增程式碼檔案後，就可以編輯 C 程式碼，其步驟如下所示：

Step 3：請直接在 Dev-C++ 的編輯視窗輸入 C 語言程式碼，如下圖所示：



```
1 /* 程式範例: AppB-2.c */
2 #include <stdio.h>
3 #include <stdlib.h>
4 /* 主程式 */
5 int main() {
6     printf("第一個C程式\n");
7     system("PAUSE");
8     return 0;
9 }
10
```

Step 4：在輸入程式碼後，請執行「檔案」→「儲存」指令，可以看到「儲存檔案」對話方塊。



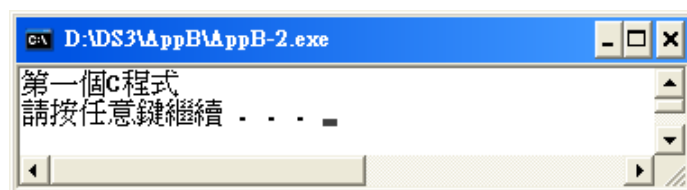
Step 5：請切換到「D:\DS3\AppB」路徑，在【檔名】欄輸入 C 程式檔案的名稱【AppB-2.c】，【存檔類型】欄選【C source(*.c)】，按【儲存】鈕儲存程式檔案，可以看到視窗上方標籤改為檔案名稱。

》》》》 步驟三：編譯和執行 C 應用程式

在建立和儲存 C 程式檔案後，Dev-C++ 即可進行編譯和執行 C 應用程式，其步驟如下所示：

Step 6：請執行「執行」→「編譯並執行」指令或按 **F9** 鍵，可以看到目前編譯進度的訊息視窗。

Step 7：如果程式沒有錯誤，編譯成功後，就會開啟「命令提示字元」視窗顯示執行結果，如下圖所示：



在上述視窗顯示執行結果的字串，正常的視窗色彩是黑底白字，筆者為了印刷美觀考量，已更改視窗色彩成為白底黑字。請按任意鍵結束執行且關閉視窗。

B-3 Dev-C++ 的專案管理

Dev-C++ 的「專案」(Projects) 是用來幫助程式設計者開發和維護 C/C++ 應用程式，專案通常是使用在多個程式檔案的大型應用程式，其內容包含應用程式檔案和相關檔案清單，程式檔案所在路徑和相關編譯設定等資訊。

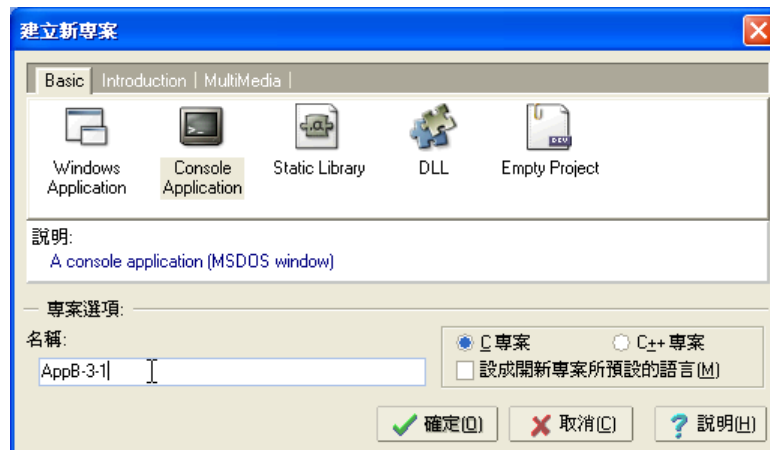
例如：C 語言的模組是由.h 標頭檔和.c 的程式檔案組成，換句話說，我們可以使用 Dev-C++ 的專案來管理模組的程式檔案。

B-3-1 新增 Dev-C++ 的專案

Dev-C++ 的專案是一個檔案，其副檔名為.dev，在此檔案記錄應用程式的相關設定和專案所屬的檔案清單。Dev-C++ 可以使用專案檔載入、儲存、編譯和執行 C/C++ 應用程式。

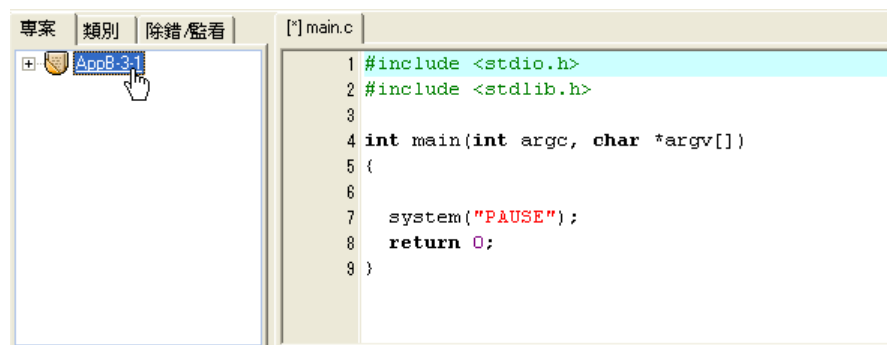
現在筆者準備在 Dev-C++ 建立新專案 ChB-3-1.dev，然後將第 1-4 節的模組檔案 Ch1-4.h 和 Ch1-4.c 加入專案，新增主程式 ChB-3-1.c 呼叫模組函數 cmpresult()，其步驟如下所示：

Step 1：請啟動 Dev-C++ 執行「檔案」→「開新檔案」→「專案」指令，可以看到「建立新專案」對話方塊。



Step 2：在【Basic】標籤選【Console Application】，在右下方框選【C 專案】，【名稱】欄輸入專案名稱【AppB-3-1】，按【確定】鈕，可以看到「儲存檔案」對話方塊來建立新專案。

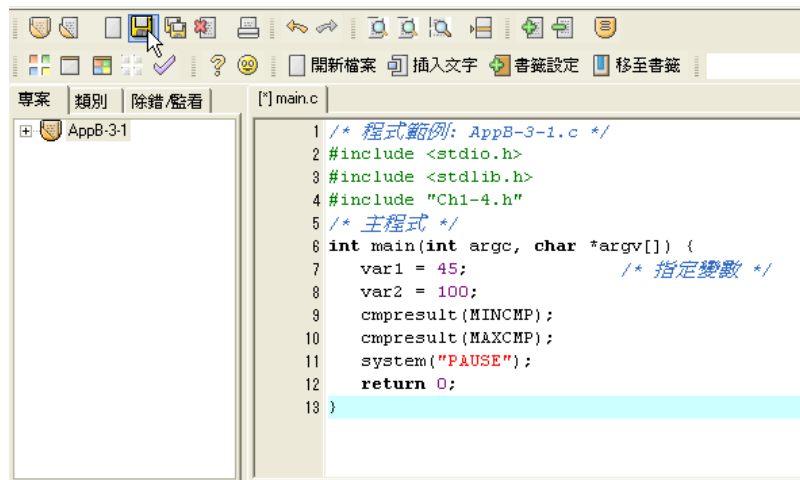
Step 3：在【儲存於】欄切換到專案儲存的路徑，以筆者的電腦為例是「D:\DS3\AppB」，在【檔名】欄是與專案名稱同名的【AppB-3-1】（附檔名為.dev），按【儲存】鈕儲存專案，就可以在「專案/類別瀏覽」視窗看到新增的專案 AppB-3-1。



上述 Dev-C++ 新增的 AppB-3-1 專案預設建立名為 main.c 的 C 程式檔案（儲存時可以更改檔名），在右邊的編輯視窗是預設的程式內容，內含 C 程式的基本架構，含括<stdio.h>和<stdlib.h>標頭檔（<stdlib.h>標頭檔是因為呼叫 system() 函數），主程式 main() 的傳回值為 int 整數，記得嗎！函數如果沒有指定傳回值型態，預設是整數 int。

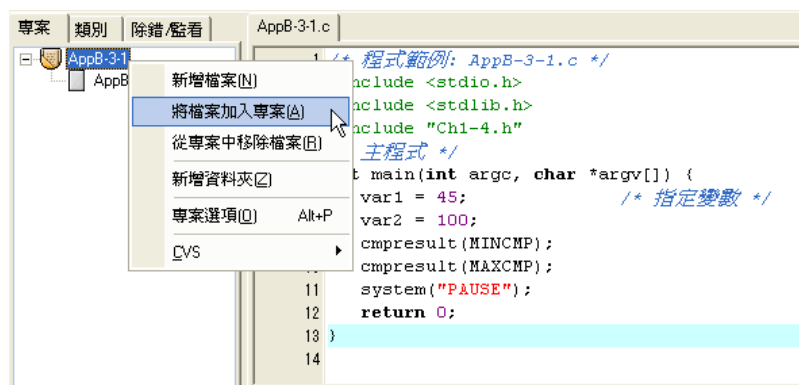
函數 system() 可以執行 MS-DOS 環境命令，PAUSE 命令的功能是暫停螢幕輸出，以避免關閉「命令提示字元」視窗，傳回值是傳回給作業系統，0 表示沒有錯誤，如果為非零值表示有錯誤產生。

Step 4：請在編輯視窗輸入程式碼，含括模組 "Ch1-4.h" 標頭檔，指定變數 var1 和 var2 值後，呼叫 cmpresult() 函數，如下圖所示：

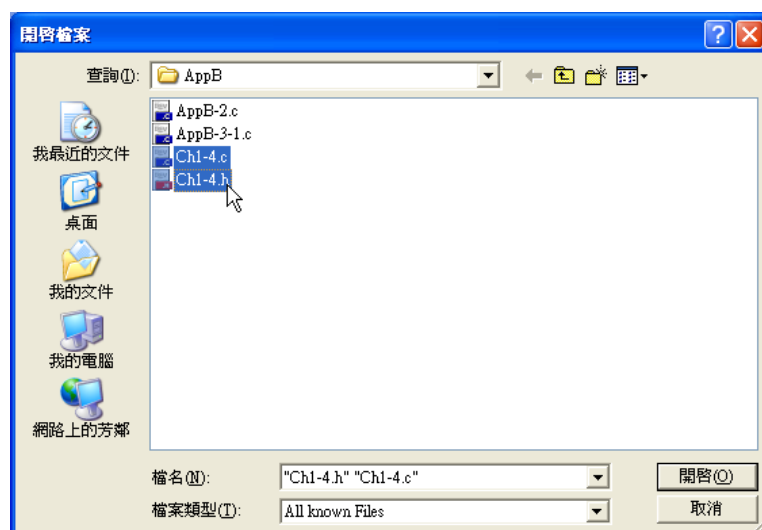


Step 5：按游標所在的【儲存】鈕，可以看到「儲存檔案」對話方塊。

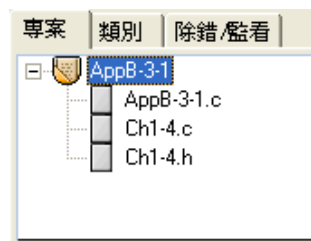
Step 6：在【檔名】欄輸入【AppB-3-1.c】，【存檔類型】為【C source (*.c)】，按【儲存】鈕儲存程式檔案，可以看到編輯視窗的標籤改為 AppB-3-1.c。



Step 7：在「專案/類別瀏覽」視窗展開【AppB-3-1】專案，可以看到新增的檔案 AppB-3-1.c，在【AppB-3-1】專案上執行【右】鍵快速功能表的「將檔案加入專案」指令，可以看到「開啟檔案」對話方塊。



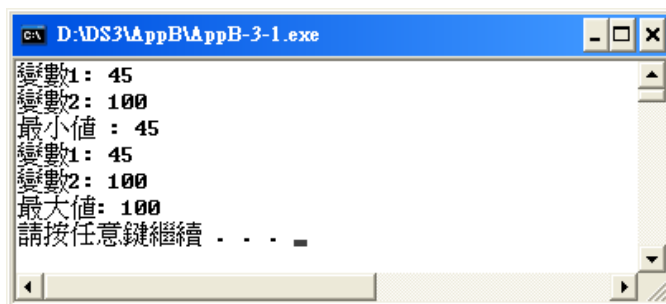
Step 8：請同時按 **Ctrl** 和 **Shift** 鍵和檔名選取多個加入專案的程式檔案，以此例是 Ch1-4.h 和 Ch1-4.c，按【開啟】鈕將程式檔案加入專案。



Step 9：在【專案】標籤顯示專案的程式檔案清單，在檔案清單上按一下檔案名稱，就可以在右邊編輯視窗開啟檔案編輯程式碼。

Step 10：請執行「檔案」→「儲存所有檔案」指令儲存專案檔案。

Step 11：現在我們就可以編譯和執行專案，請執行「執行」→「編譯並執行」指令或按 **F9** 鍵，如果程式沒有錯誤，可以看到專案程式的執行結果，在「命令提示字元」視窗顯示執行結果，如下圖所示：



在上述視窗可以看到執行結果，按任意鍵結束執行且關閉此視窗。執行「檔案」→「關閉專案」指令可以關閉專案。

B-3-2 專案管理的基本操作

Dev-C++ 的專案管理除了新增存在的 C 程式檔案，我們還可以在專案的檔案清單新增、刪除程式檔案或重新命名檔案。

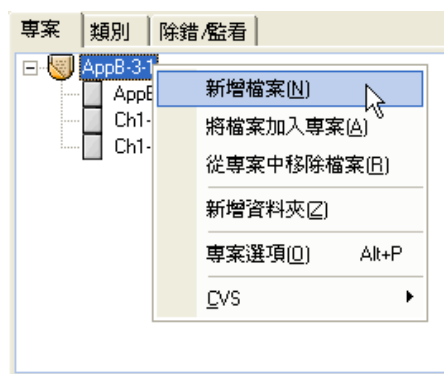
》》》》 開啟 Dev-C++ 專案

在啟動 Dev-C++ 後，請執行「檔案」→「開啟專案/一般檔案」指令，可以看到「開啟檔案」對話方塊。

在【檔案類型】選【dev-c++ project(*.dev)】類型，可以看到專案的檔案清單，選取後，按【開啟】鈕即可開啟存在的專案檔案。

》》》》 新增檔案

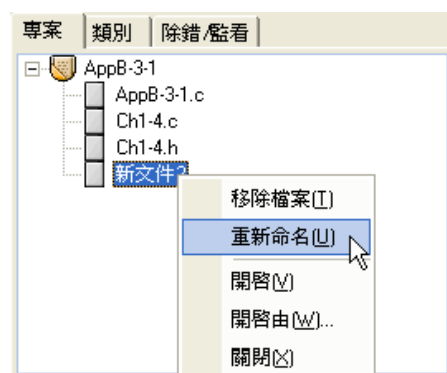
如果專案需要新增全新的程式檔案，例如：在 AppB-3-1 專案新增名為 AppB-3-2.c 的 C 程式檔案，請在專案名稱上按【右】鍵顯示快顯功能表，如下圖所示：



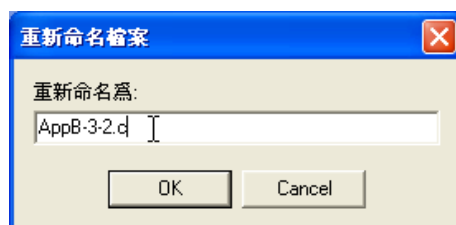
執行【新增檔案】指令，即可新增名為【新文件?】的程式檔案，同時在右邊開啟和顯示新文件的編輯視窗，在編輯視窗輸入程式碼，儲存檔案後，就可以新增專案的程式碼檔案。

》》》》 重新命名

如果對專案檔案清單的檔案名稱不滿意，我們可以直接重新命名檔案，請在檔案上按【右】鍵顯示快顯功能表，如下圖所示：



執行【重新命名】指令，可以看到「重新命名檔案」對話方塊。



在【重新命名為:】欄輸入新檔名，按【OK】鈕即可更改檔案名稱。

》》》》 移除檔案

如果專案需要移除檔案清單中的檔案，請在移除檔案上執行【右】鍵快顯功能表的【移除檔案】指令，就可以移除專案的檔案，請注意！此操作只是將檔案從專案檔案清單上移除，並不是刪除檔案。