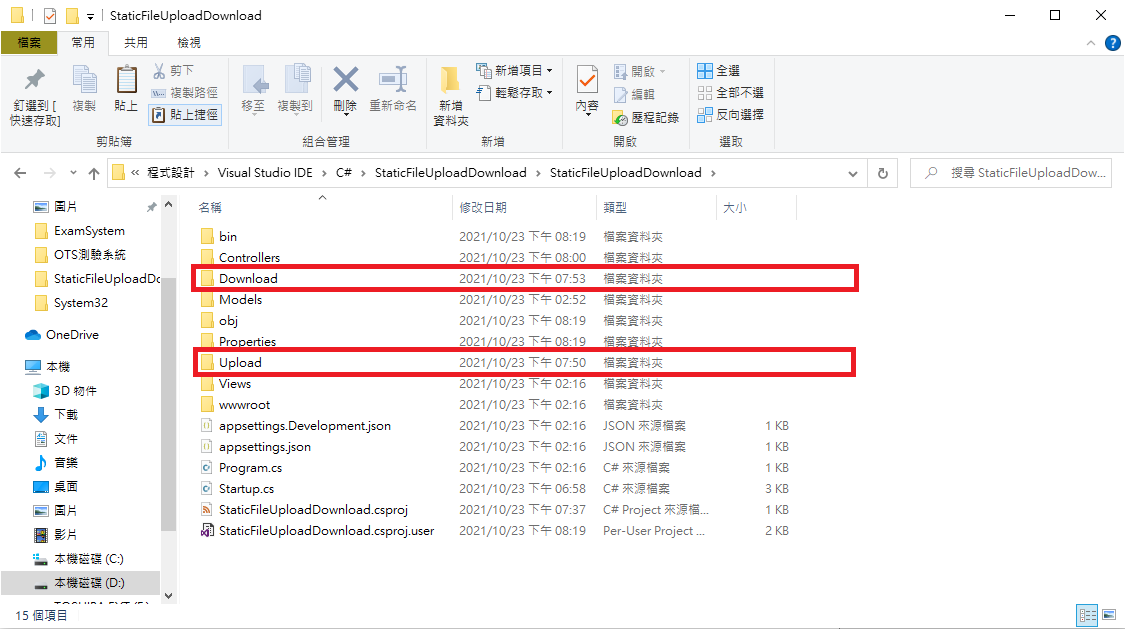
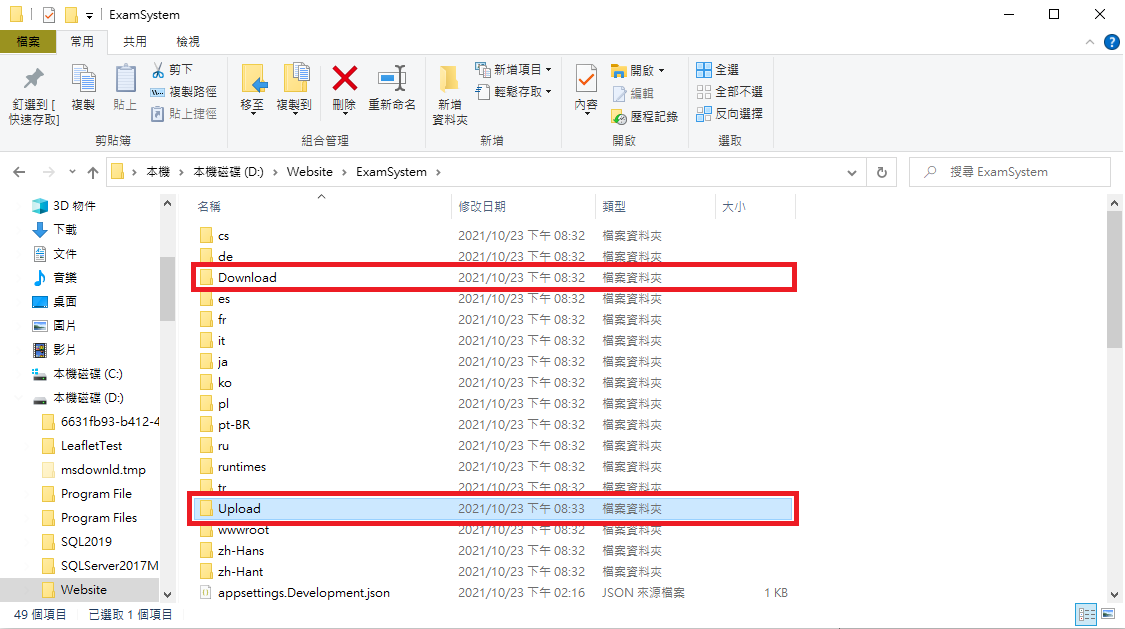
# 發布注意事項

下圖為專案內的資料夾，紅框處兩個資料夾為檔案上傳與下載時儲存資料用的資料夾。



當發布至 IIS 伺服器的時候這兩個資料夾需另外建立，否則網站無法正常執行。



## [安全性考量](https://docs.microsoft.com/zh-tw/aspnet/core/mvc/models/file-uploads?view=aspnetcore-5.0)

1. 將檔案 Upload 到專用檔案上傳區域，最好是非系統磁片磁碟機。專用位置可讓您更輕鬆地對上傳的檔案強加安全性限制。停用檔案上傳位置的執行許可權。(對應方式是去停用執行權限，位於IIS管理員的站台的「處理常式對應」>「開啟功能」>「編輯功能權限」，不過預設就是停用，另外指令碼停用會導致網站無法執行，[參見](https://blog.miniasp.com/post/2010/08/04/IIS7-How-to-Turn-off-Execute-Permission))
2. 請勿將上傳的 檔案保存在 與應用程式相同的目錄樹狀結構中。(這部分就是建議存在 wwwroot 之上的資料夾)
3. 使用應用程式所決定的安全檔案名。 請勿使用使用者提供的檔案名或上傳檔案的不受信任檔案名。HTML 會在顯示不受信任的檔案名時進行編碼。 例如，記錄檔案名或在 UI 中顯示 (Razor 會自動以 HTML 編碼輸出) 。(這個部分有使用亂碼與原始副檔名，並產生新的文件進行儲存)
4. 只允許應用程式設計規格的核准副檔名。(使用List<string>去比對上傳的檔案的副檔名是否吻合應用程式的設計)(尚未實作)
5. 確認已在伺服器上執行用戶端檢查。用戶端檢查很容易規避。(看不出來寫這段要幹嘛？如果是伺服器端作檢查，可能是文章有列出的所有檢查方式，也可能[參考這篇文章](https://dotnetthoughts.net/file-upload-extension-validation-in-aspnet-core/)) (尚未實作)
6. 檢查上傳檔案的大小。 設定大小上限以防止大上傳。(參見[大小驗證的章節](https://docs.microsoft.com/zh-tw/aspnet/core/mvc/models/file-uploads?view=aspnetcore-5.0#size-validation)) (尚未實作)
7. 使用相同名稱的已上傳檔案覆寫檔案時，請先檢查資料庫或實體儲存體的檔案名，再上傳檔案。(目前會使用不同檔案名稱，如果開放的話相同檔案名稱重複上傳可以接受，不開放的話可以透過檢查資料庫)
8. 在儲存檔案之前，在上傳的內容上執行病毒/惡意程式碼掃描器。(可能是指[AMSI](https://docs.microsoft.com/zh-tw/windows/win32/amsi/how-amsi-helps)) (尚未實作)

# 備註

由於此範例並不牽涉到資料庫，故上傳與下載的資料夾分開，且資料下載的表格是固定三個檔案，如果使用到資料庫則可以上傳下載皆於同一個資料庫

# 變更筆記

目前還沒實作「多個檔案上傳」與「下載時修改檔名」可以再花時間研究。(2021/10/23 22:57)