

## 測試案例

### 案例設計與撰寫所使用之策略：黑箱測試

#### 如何測試輸入:

針對使用者在課程新增與搜尋方面可能輸入的資訊來當作測試的資料，在搜尋部分，輸入值不外乎是使用者輸入正確的資料（課程名正確無誤）、錯誤的資料（有可能使用者亂打、資料不存在）這兩種結果，在新增部分，延續上述兩個再加上有可能新增時資料庫已有該課程資料，所以會額外再寫一個測試。

#### 等價劃分:

以使用者可能輸入的課程名稱做為輸入判斷。

合法範圍:課程字數為 2~10。

非法範圍:少於 2 個字，大於 10 個字，中文以外的字元。

案例編號	案例簡述	前置條件	測試步驟	預期結果	測試結果
Course_01	使用者輸入課程名稱成功搜尋到課程相關資訊。	客戶端設備正常及資料庫資料建置完整，且正常運作中。	使用者選擇搜尋功能，輸入課程名稱“軟體工程導論”。	成功連結後端資料庫並且回傳相對應的開課老師資訊與評價。	通過：程式執行符合預期。
Course_02	使用者輸入空值，資料庫中查無資料並跳出警告資訊。	客戶端設備正常及資料庫資料建置完整，且正常運作中。	使用者選擇搜尋功能，輸入“”。	跳出訊息：查無資料。	通過：程式執行符合預期。
Course_03	使用者選擇新增課程資訊並輸入課程名稱，將資料新增至資料庫中。	客戶端設備正常及資料庫資料建置完整，且正常運作中。	使用者選擇新增課程資訊，輸入課程名稱“語言資料科學基石”。	成功新增至後端資料庫並且回傳相訊息:新增課程資料成功。	通過：程式執行符合預期。
Course_04	使用者選擇新增課程資訊並輸入課程名稱，但資料中已有該課程的資料。	客戶端設備正常及資料庫資料建置完整，且正常運作中。	使用者選擇新增課程資訊，輸入課程名稱“軟體工程導論”。	跳出訊息：已有該課程的資訊	通過：程式執行符合預期。
Course_05	使用者選擇新增課程資訊並輸入不合法的資料。	客戶端設備正常及資料庫資料建置完整，且正常運作中。	使用者選擇新增課程資訊，輸入“”。	跳出訊息：所輸入的資訊有誤。	通過：程式執行符合預期。