

自我省思與改進

撰寫者：李秉軒

任教領域/科目：科技領域/資訊科技 教學單元：Python 程式設計的概念

日期：113 年 10 月 16 日第 5 節 地點：電腦教室 A

針對自我省思，可以針對課程設計與教學、班級經營與輔導進行兩部分撰寫

課程設計與教學

1.增加動手操作與探索的環節：考量到學生在語法錯誤上的困擾，除了老師示範之外，可以增加一些引導式的練習，例如提供半完成的程式碼讓學生填空，或是設計一些小挑戰，讓學生在「試誤」中學習並逐步熟悉 Python 語法。

2.預留錯誤排除與解說時間：課程後半段學生在撰寫程式時遇到語法錯誤是常見現象。建議老師可以將錯誤訊息整理歸納，在課堂上統一講解常見的錯誤類型及其解決方法，而不是等到學生求助時才個別指導。這樣不僅能提高除錯效率，也能培養學生獨立解決問題的能力。

3.提供語法查詢資源與工具：為了幫助學生克服語法上的困難，老師可以提供一些常用的 Python 語法速查表、線上教學資源連結，或是推薦學生使用具備自動完成和錯誤提示功能的整合開發環境 (IDE)，讓學生在練習時能更方便地查詢和修正錯誤。

4.設計循序漸進的練習題：「個人學期成績計算」對初學者而言可能有些複雜。建議老師可以將練習題拆解成幾個更小的步驟，例如先練習判斷單一條件，再逐步增加多重條件判斷，讓學生能逐步掌握選擇結構的應用，降低學習門檻。

班級經營與輔導

1.建立共同學習的氛圍：可以鼓勵學生在遇到問題時，先嘗試與鄰近同學討論，透過同儕互助的方式解決問題。這不僅能減輕老師個別指導的壓力，也能培養學生的團隊合作能力。

2.鼓勵學生勇於嘗試與提問：許多學生可能因為害怕犯錯或覺得問題太簡單而不敢提問。可以營造一個開放、鼓勵的學習環境，讓學生知道犯錯是學習的必經過程，並鼓勵他們勇敢地提出疑問。

3.提供差異化輔導：觀察到部分學生在語法上需要較多協助，可以考慮在課堂上設計一些小組活動，讓學習能力較強的學生擔任小老師，協助其他同學解決問題，同時老師也能將更多時間投入到需要個別輔導的學生身上。

4.善用回饋單或線上表單：課程結束後，可以設計一份簡單的回饋單或線上表單，讓學生匿名填寫他們在課程中遇到的困難、對課程的建議等。這有助於老師更全面地了解學生的學習狀況，並作為未來課程改進的參考。

5.肯定學生的努力與進步：即使學生程式無法完全正確執行，也應肯定他們在思考和嘗試過程中所付出的努力。適時的鼓勵和肯定，能增強學生的學習動機和自信心，讓他們對程式設計保持熱情。