

輔仁大學 108 年高教深耕計畫

【程式設計融入課程補助計畫】授課成效報告

基本資料

開課學院	全人教育課程中心	開課系/組	
學年度/學期	107 學年度 / 第 2 學期	學制別	大學 <input checked="" type="checkbox"/> 日間部 <input type="checkbox"/> 進修部
課程名稱	生命科學與人類生活	上課時間	星期 <u>一</u> , <u>15:40 ~ 17:30</u>
開課代碼	DNTN831282	修課人數	35
授課教師	藍清隆	聯絡電話	XXXXXXXXXX (研究室分機)
電郵信箱	002318@gapp.fju.edu.tw		

整體教學設計

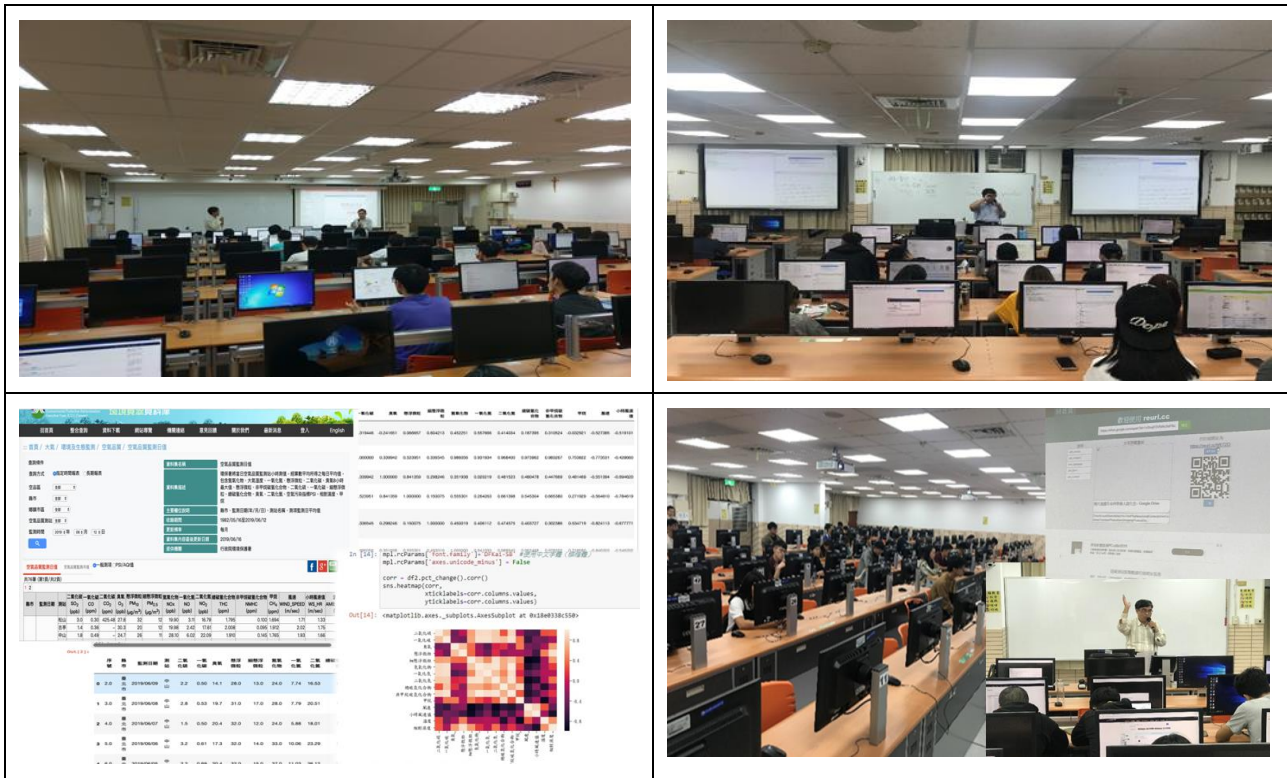
跨域特色	<p>本案第二次在「生命科學與人類生活」通識課程中融入 12 小時 Python 程式設計學習活動，以深化原課程單元”大數據與環境”的探索學習。</p> <p>大數據分析與應用是近年來國際發展的重要趨勢，前述課程引導學習者探索大數據解析氣候變遷與公眾健康的實務。透過模擬城市或國家層級的未來氣候變遷衝擊和調適，混搭大數據(big data mashups)被期待能真正探索和了解 21 世紀”邪惡的環境和健康問題”。鑑於推動大數據應用必須深化學習者運用開放資料的資訊素養，本案再次邀請曹祥雲老師協助授課，引發學習者程式設計的熱情，動手抓取、並使用政府開放資料進行微型的環境影響評估。</p> <p>本案的學習經驗期盼引導學習者能成為「citizen data scientist(公民數據師)」；也能在其專業領域進而探索資料分析領域五個關鍵職涯。</p>
程式語言	<input checked="" type="checkbox"/> Python <input type="checkbox"/> APP Inventor 2 <input type="checkbox"/> R <input type="checkbox"/> Javascript <input type="checkbox"/> 其他 _____
教學目標	<ul style="list-style-type: none"> • 知識面目標 (期望學習者透過課程能習得哪些知識): <ul style="list-style-type: none"> -能理解氣候變遷的基礎、歷史趨勢與未來變化。 -能理解從氣候變遷到疾病和健康的途徑。 -能理解解決問題的運算思維。 -能理解開放資料與運用國內/國外開放資料。 -能理解 21 世紀學習。 • 學科專業技能目標 (期望學習者透過課程能展現哪些學科專業技能): <ul style="list-style-type: none"> -能提升 21 世紀技能(如批判思考及問題解決、ICT 等)。 • 程式設計技能目標 (期望學習者透過課程能展現那些程式設計技能): <ul style="list-style-type: none"> -由從做中學，能使用 Python 語言抓取、分析與顯示政府開放資料。 • 態度面目標 (期望學習者修習完課程後能有哪些態度轉變): <ul style="list-style-type: none"> -能樂於 21 世紀學習、體會生命運作的奧妙。 -能關心且積極參與公民科學計畫。
作業設計	<p>小組 PBL 合作作業：<input checked="" type="checkbox"/> WebQuest 網站 <input checked="" type="checkbox"/> 簡報(口頭報告) <u>1</u> 次</p> <p>程式設計(小組或個人)：<u>1</u> 次</p> <p>期末學習心得分享：<u>1</u> 次</p>

	■其他：線上討論版、TronClass-支援互動 App
評量設計	<ul style="list-style-type: none"> • 形成性評量之規劃：線上測驗(TBL 式) • 總結性評量之規劃：小組 PBL 作業成果(使用 rubric 的自評與小組互評)
學習輔助資源	線上資源： <input type="checkbox"/> Codecademy <input checked="" type="checkbox"/> Coursera <input type="checkbox"/> Code school <input checked="" type="checkbox"/> edX <input checked="" type="checkbox"/> FutureLearn ■其他：TronClass 線上課程(網址) 實體資源： <input checked="" type="checkbox"/> 專題演講(視訊) <input type="checkbox"/> 其他
參考與延伸閱讀資料	MOOCs: Climate change policy and public health 、 Protecting public health in a changing climate: A primer for city, local, and regional action (Coursera), The health effects of climate change (edX), Python basics for data science (edX), Big data and the environment (FutureLearn) 。

教學設計

	日期	課程單元名稱	學習目標	教學設計重點
		Anaconda 安裝/運作環境操作 Python 基本語法	1)能理解學習程式設計的重要性 2)能理解 Python 程式開發環境與基本語法	10 分鐘學程式設計 (第 3-7 週破冰/熱身教學活動)
1	04-08	運算思維 Python 套件安裝 政府開放資料取得	1)能理解運算思維概念與步驟 2)能執行 Python 套件安裝 3)能理解/取得政府開放資料	解決問題(使用政府的開放資料進行微型的環境影響評估)的運算思維實作 PTT parser
2	04/15	Python 雲端開發環境 /資料擷取	1)能理解 Python 雲端開發環境 (Google Colaboratory/Jupyter notebook) 2)能運用 Python 擷取網頁資料	實作擷取 EPA 空氣品質監測網即時網頁資料 實作擷取 EPA 環境資源資料庫歷年監測資料
3	04/22	Python 開放資料的分析/可視化	1)能理解資料的擷取/轉換/分析	實作 EPA 環境資源資料庫歷年監測資料轉換/分析/可視化
4	04/29		2)能理解資料的可視化	
5	05/06		3)能運用 numpy/pandas 分析開放資料 2)能運用 matplotlib/seaborn 進行開放資料的可視化	
6	05/13	人工智慧簡介	1)能理解人工智慧 2)能理解 CNN/RNN/LSTM	

課堂活動剪影



授課心得感想

一)學習活動之規劃與實施

在本校 107 年高教深耕計畫的補助經費支持下，「生命科學與人類生活」通識課程順利融入 12 小時 Python 程式設計學習活動(參考 107-1 程式設計融入課程[授課成效報告](#))。本案再次申請補助經費以邀請校外曹祥雲老師繼續協助授課，合力建置執行 107 年度計畫案所發現的部分問題之初步解決方案，並評估其成效。有關前述課程的教學目標、教學設計與教學活動不在此報告中贅述，若有興趣瞭解則請參考 107-1 程式設計融入課程[授課成效報告](#)、[107-2 學習活動清單](#)章 1-章 3。

近年來各國政府推動大數據(big data)應用蔚為風潮，大數據應用於環境數據分析(environmental data analytics)可由 MOOCs [Big data and the environment](#)、[Citizen science: From data to action](#) 的解說管窺一斑。推動大數據應用必須深化學習者發掘、存取、使用與分享開放資料(open data)的素養。本案主持人能掌握其資料分析面的應用、卻缺乏其網站抓取與分析資料面的知能，因此再次邀請曹老師協助透過教學活動引導學習者：1)理解運算思維(computational thinking)、認識環保署空氣監測網；經由合作學習動手以 Python 擷取環保署[空氣品質監測站即時值](#)、環保署環境資源資料庫[空氣品質監測日值](#)等，運用 numpy/pandas 套件進行資料分析(data analysis)、使用 matplotlib/seaborn 套件進行數據可視化(data visualization)。2)認識人工智慧(artificial intelligence)，CNN、RNN 神經網路架構與長短期記憶模型(LSTM)，透過圖像識別、自然語言處理、文字產生等案例揭開人工智慧的面紗。配合六週課程(如上教學設計的說明；參考課程[雲端共享資源](#)與課程解說視訊 [01](#)、[02](#)、[03](#)、[04](#)、[05](#)、[06](#))，本案安排「10 分鐘學程式設計」實體教室破冰/熱身教學活動，也提供完整的線上學習資源包括一小時玩程式(Hour of code)、認識運算思維、為什麼要學習 Python、建置雲端或桌端 Python 開發環境、使用 Jupyter notebook 學習 Python 基礎程式設計語法、認識開放資料與網頁資料擷取與分析：使用 Python 等([107-2 學習活動](#)

[清單](#)章 4)。

兩位教師均適時鼓勵學習者能在專業領域上進而探索資料分析領域五個關鍵職涯---資料工程師/軟體工程師、資料分析師、資料科學家、領域專家。有鑑於過去兩年台灣正快速跟上全球的人工智慧(artificial intelligence)浪潮，也以案例揭開人工智慧的神秘面紗，鼓勵學習者不能無視或畏懼人工智慧帶來的挑戰，學會面對、應用人工智慧。

二) 反思與心得

往年選修筆者開授的通識課程之學生數都在 50-70 位間，本課程自 107-1 學期融入 12 小時 Python 程式設計學習活動(66 位選修學生)，本學期(107-2)選修學生數卻減少至 35 位，是否與融入程式設計教學有關仍待觀察！

課前問卷顯示 35 位選修學生中有 5 位學習過某一種程式語言的學生(占 20%)。有 6 位學生在開學一個月後就很少到課(不具名課前問卷，而無法供確認其中有無學習過某一種程式語言者)。程式課程教學期間使用 TronClass-支援「數字點名」抽點的缺課率在 7-30% 間。學期初學習者貼文認同程式設計教學活動；估計教學活動結束時至少有 1/3 的學習者缺課或到課而不積極投入學習，雖然符合預期指標(程式設計課程出席率達 2/3)，卻再次顯示需要建置補救教學機制、即時協助學習者處理學習障礙。

觀察 TronClass-支援討論版的發文數、造訪量，TronClass-支援互動 App (教學回饋)，TronClass-支援學習分析顯示學習者的程式設計單元平均參與度(造訪次數、停留時長、參考檔案查看次數/下載次數、線上連結查看次數、影音教材觀看次數、討論版造訪次數/發文次數/回覆次數、點名出席率)，結論(有學習過某一種程式語言的學習者和自主學習力強的學習者之程式設計單元平均參與度相對較高)與 107-1 學期類似。因此不在此報告中贅述，若有興趣瞭解則請參考 107-1 程式設計融入課程[授課成效報告](#)。

側面觀察在六週間學習者的學習行為也發現：有學習過某一種程式語言的學習者和自主學習力強的學習者之程式設計單元平均參與度相對較高、期末繳交程式設計作業率也相對較高(結論與 107-1 學期雷同)。程式設計作業顯然是學習者最大的難題！曹老師詳細介紹怎樣思維、怎樣取得/分析資料、怎樣呈現結果。原則上由 2-3 位同學組成一個小組完成程式設計作業，但鼓勵由 1 位同學獨自完成作業(會視其成果加分)。有 13 位同學上傳個人作業，1 組(2 位同學)上傳小組作業。本案只有 48%的學習者繳交程式設計作業，其中約 87%的學習者獨立完成，接近預期的目標(作業繳交率 50%；結果也與 107-1 學期類似)；但是繳交的作業則只有一件未達預設標準(而 107-1 學期則約有 20%的作業未達預設標準)！

本校積極培養非資訊科學主修學生的資訊力以期達成「107 年度 40%的學生曾修讀程式設計課程」的績效目標。可惜本學期選修學生中沒有在 107-1 學期選修過相關課程，無法供確認筆者的原先期待(107-2 學期的課程學習者之參與會有明顯的改變)！

筆者在 107-1 學期發現如右的問題：1)需要安排破冰學習活動以更引發學習者程式設計的熱情，2)需要提供用者友善性更高的 Python 程式編輯環境與相關學習物件，3)需要設計多個階段性小練習以更滿足學習者程式設計旅程的成就感，4)需要提供學習完整指引以更有效鷹架學習者完成程式設計作業的教學需求，5)需要介紹多個案例以更吸引學習者探

索程式設計的職涯，6)需要建置機制以更敏銳感知學習者程式設計的學習障礙、適時提供補救教學。第6項需求則顯然已經超出筆者的能力範圍！至於第2項需求，即使本課程已經選用資料科學領域廣受喜愛的 Jupyter notebook 環境，然而不免還是有學習者認為友善性不夠而望門興嘆！第1、3、4、5項在107-2學期已經提供初步解決方案，然而還是有30-45%的學習者缺課或到課而不積極投入學習！事實上也符合筆者在通識課堂中屢次目睹這一代的大學生學習意願低落、上課不專心的觀察。

許多專業領域都有開放資料庫；隨著知識經濟與雲端應用時代來臨，歐美主要國家及聯合國、世界銀行等國際組織近年來更大力提倡及推動政府開放資料做為國家發展之重要策略(參考 edX MOOC：[Open government](#))。筆者仍深信本案可以提供程式設計訓練略嫌不足的同儕參考，在其專業課程中嘗試融入 Python 程式設計學習活動，引導學習者認識運算思維、開放資料與 Python 語言、善用雲端開發環境如 [Google Colaboratory](#)、實作使用 Jupyter notebook 學習 Python 基礎程式設計/ Python 網頁資料擷取與分析等。

兩個學期整理出如下的以實務問題為核心，鼓勵學習者進行小組討論，期望強化學習者主動學習和問題解決的能力模組：

<p>01 運算思維與程式設計 案例學習：運算思維與微型環境數據分析 延伸學習：Big data and the environment</p>	<p>02 Python 雲端開發環境與基礎語法 案例學習：Google Colab 入門：1)Jupyter notebook 101, 2)Python 101</p>
<p>03 開放資料 案例學習：Open data in a day 延伸學習：Discovering open data</p>	
<p>05 資料擷取、轉換和載入 延伸學習：Getting and cleaning data</p>	<p>04 Python 網頁資料擷取 案例學習：擷取 EPA 空氣品質監測網即時網頁資料、EPA 環境資源資料庫歷年監測資料</p>
<p>07 Python 數據可視化 案例學習：運用 matplotlib/seaborn 可視化 EPA 環境資源資料庫歷年監測資料 延伸學習：Visualizing data with Python</p>	<p>06 Python 資料分析 案例學習：運用 numpy/pandas 分析 EPA 環境資源資料庫歷年監測資料 延伸學習：Python basics for data science</p>
	<p>08 Python 大數據分析與機械學習 延伸學習：AI Experiments (Google)</p>

108 學年度起，「生命科學與人類生活」通識課程融入的 Python 程式設計學習活動都會在 Google Colab 雲端環境進行，提供程式碼供學習者在 Jupyter notebook 編輯器中撰寫及執行程式。課程中雖然會簡單說明範例程式，只想促進學習者有機會接觸與了解運算思維及程式設計，只會引導學習者改寫部分程式碼(例如從「擷取環保署空氣品質監測網即時網頁資料」改寫成「擷取台北市 ubike 場站即時網頁資料」)，不期待學生在修課完畢後能具備獨立撰寫與執行程式的能力。

值得特別一提的是東海大學為非資訊相關科系學生建置創新的學習場域([PBL 教室](#))、發展創新的程式語言學習機制，確實有耳目一新的感覺([網站](#))、「啟動非資訊領域學生的程式設計創新教學」[發表會](#)音訊 [01/02](#))！目前筆者仍未自聯絡人李政雄教授收到發表會後新的進展資訊。該校開發出兩種[軟體工具](#)：「CT2Flow」能「---讓學生能夠將解題的運算思維邏輯以流程圖的方式並配合自然語言的說明來呈現，幫助學生把解題的過程與想法用流程圖來表示---」，「CT2Code」則「---基於 CT2Flow 的概念，更進一步結合程式語言，讓學生能夠把流程圖轉換成為可執行的程式碼，學習基礎程式語言設計---」。可惜筆者取得的「CT2Code」目前只能夠把流程圖轉換成為可執行的 Java 程式碼，等該新版軟體能夠把流程圖轉換成為可執行的 Python 程式碼時，將會再嘗試結合於課程中。

東海大學的 PBL 教室中講桌及教師座位在教室正中央，學生座位則採圓弧形排列圍繞。老師與同學距離不再有前後之分，同學彼此間也可以看到對方。教室正中央上方，設置環形電視牆(4 台數位面板組成)，老師可透過無線控制，隨時播放教材內容或現場直播學生的解題過程。分組討論時，學生則可利用教室四周的討論圓桌，透過牆上的數位面板，無線分享彼此觀點與做法。作為促進者和引導者的老師，亦可以隨時切入小組，播放輔助內容至數位面板上；同時亦可分享該組內容給其他小組，促進大家討論、激發想法(資訊引用自 <http://space.thu.edu.tw/detail/C/C114/2018/13/>)。筆者的初淺推想是：老師與學生善用手機透過 Google Colab 撰寫及執行程式，可某種程度模擬上述 PBL 教室的學習情境(想像共用不同的 Colab notebooks 分別就是上述教室中的中央電視牆、四周討論圓桌邊牆的數位面板，學生可以組內/組間分享彼此觀點與做法，老師也可以隨時切入小組促進討論/激發想法、或跨組分享該組內容而引導討論/激發想法)，或許也值得一試！