

輔仁大學 107 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	外語學院 / 進修部	開課系所	進修部英國語文學系
學年度/學期	106 學年度 / 第 2 學期	學制別	☒ 大學 ☐ 研究所 ☐ 其他
課程名稱	英語語言學概論	上課時間	星期_四_, 18:40~21:20
開課單位	英國語文學系進修學士班	修課人數	50
授課教師	樂麗琪	聯絡電話	(手機) 0936-156-545 (研究室分機) 3722
電郵信箱	yueh@mail.fju.edu.tw		

● **整體教學設計**

跨域特色	將「運算思維認識」與「PYTHON 程式設計」融入「英語語言學概論」課程設計與教學活動中，探索語言學學科專業的資訊跨域整合實踐應用。使學生透過運算程式分析、理解和處理自然語言，從中發覺語言結構的規律性，明瞭培養資訊能力的重要性，進而能產生自主學習動機，能自我強化邏輯思考及語言分析能力，擴展學科專業研究能力。
程式語言	☒ Python ☐ APP Inventor 2 ☐ R ☐ Javascript ☒ 其他 <u>語料庫</u>
教學目標	• 知識面目標 (期望學習者透過課程能習得哪些知識): 1. 瞭解人文學科與程式語言運用的關聯性與發展趨勢。 2. 認識 Python 程式應用的相關趨勢。 3. 理解 Python 程式語言、簡易 python 程式撰寫與改寫應用。 • 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): 1. 認識語料庫及運用 2. 運用 Python 程式進行語意學之語料分析。 3. 運用 Python 程式進行社會語言學之語料分析。 • 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): 1. 學習和使用 NLTK 語料庫處理自然語言。 2. 利用 Python 第三方函式庫製表編程。 • 態度面目標 (期望學習者修習完課程後能有哪些態度轉變): 1. 明瞭運算思維邏輯能力在進行自然語言分析處理的重要性 2. 嘗試跨領域學習、強化資訊素養。 3. 培養自主發想研究計劃能力。
作業	個人報告: ☒書面 ☐簡報 <u>3</u> 次 小組報告: ☒書面 ☒簡報 <u>6</u> 次

設計

討論：8 次

✓ 程式設計(個人)：2 次 (含個人報告)

✓ 程式設計(小組)：2 次 (含小組報告)

✓ 討論：3 次

□其他 Google Survey 學前、學習回饋問卷 3 次

➤ 教師版 1：【程式設計融入課程：英語語言學概論】學前問卷 (附件一)

➤ https://docs.google.com/forms/d/e/1FAIpQLSejcRmHEQKXvNanDHW79LoHrS9rUZezppqj71_avebK4jr0ww/viewform



The screenshot shows a Google Form titled "輔仁大學107年度高教深耕計畫【程式設計融入課程：英語語言學概論】學前問卷". The form is in Chinese and includes a greeting, a brief introduction to the course, and a request for feedback. It contains two input fields: "電子郵件地址" (Email address) and "中文及英文姓名" (Chinese and English name). The form is displayed on a web browser with a URL that includes the form ID.

➤ 教師版 2：【程式設計融入課程：英語語言學概論】學習回饋單 (附件二)

➤ <https://docs.google.com/forms/d/e/1FAIpQLScTNaWN4gtHQLqA4TaS8FFVHroE1lkFkj8Kuj6KivHtgmbmew/viewform?c=0&w=1>



The screenshot shows a Google Form titled "輔仁大學107年度高教深耕計畫【程式設計融入課程：英語語言學概論】學習回饋單". The form is in Chinese and includes a greeting, a brief introduction to the course, and a request for feedback. It contains two input fields: "電子郵件地址" (Email address) and "中文及英文姓名" (Chinese and English name). The form is displayed on a web browser with a URL that includes the form ID.

➤ 學校版：107 年度教學創新試辦計畫課程融入程式設計之學習自評問卷

➤ <https://docs.google.com/forms/d/e/1FAIpQLSfomvE77sr-nrGwfS2KSUIqsi6IL9PO0N5ND8TfNgKOkuG7SA/viewform>

	 107年度教學創新試辦計畫課程融入程式設計之學習自評問卷 *必填 電子郵件地址 * 你的電子郵件 本問卷請修課學生填寫之目的，在於希望了解學習者在參與「程式設計融入課程」的學習後，對於程式設計學習的認知與感受，並根據回覆意見作為未來規劃程式設計相關課程的參考。 以下請填寫您的基本資料，讓我們對您的學習背景有所認識...
評量設計	➤ 形成性評量之規劃： ✓ 隨堂練習：Kahoot 互動網分組答題、Codecademy 單元練習、編程遊戲學習網站 The Hour of Code (https://code.org/learn): 藉由遊戲闖關+影片導引系列練習，學習程式設計概念 ✓ 線上測驗：Python 程式理解和應用 ➤ 總結性評量之規劃： 期末專題報告：各組合作學習運用 python 程式進行語言學專題研究
學習輔助資源	➤ 線上資源：☒Codecademy ☒Coursera ☐Code school ☒其他 <u>Linguistic Corpora、The Hour of Code (https://code.org/learn)</u> ➤ 實體資源：☒專題演講 ☒其他 <u>課程簡報資料及課程實錄影音檔</u>
參考與延伸學習資料	• 你的未來履歷表 用 AI 挑戰 300 萬年薪 https://www.gvm.com.tw/article.html?id=43043 • 不是工程師，也可以作文本分析 https://buzzorange.com/techorange/2017/03/03/rmakeyoueasy/ • 用 Python 做自然語言處理的八個工具 https://kknews.cc/zh-tw/tech/anygk6.html • NLTK 簡單文本分析 https://blog.csdn.net/zzulp/article/details/77150129 • NLTK 中研院 Sinica Treebank http://www.hankcs.com/program/python/nltk-chinese-corpus-sinica_treebank.html

- 中央研究院數位典藏國家型科技計畫：中文斷詞系統 jibea
<http://ckipsvr.iis.sinica.edu.tw/>
- 如何用 Python 做情感分析？
<http://blog.sciencenet.cn/blog-377709-1062960.html>
- Sentiment Analysis with Python NLTK Text Classification
<http://text-processing.com/demo/sentiment/>
- 斯坦福大學自然語言處理第七課 “情感分析（Sentiment Analysis）”
<https://reurl.cc/3VRX>
- Why tech needs the humanities_Eric Berridge_TED
<https://www.youtube.com/watch?v=F2XPF6rQ6fs>

● 教學設計

➤ 實施方式：LPWDP (Lectures → Practices → Workshop → Discussion → Projects)

本計畫從 python 初學者的角度出發，教師以融入「任務導向教學法」(Task-based Approach) 方式，將本計畫實踐於「語言學概論」課程中之「語意學」與「社會語言學」兩單元中，引導學生藉由

1. 觀摩 python 程式應用實例
2. 學習和使用 NLTK 語料庫處理自然語言
3. 解讀、應用、改寫 python 程式關鍵字功能
4. 利用 Python 第三方函式庫製表編程

能學會進行有效語料分析。因此，學習活動以專家演講、程式語言練習、工作坊方式進行，內容聚焦於 Python 程式基礎語法認識與改寫、NLTK 語料庫運用分析（語意分析、語篇分析），關鍵字計算、比對、字詞分析統計圖表運用等，輔導學生得以發展產出語言學專題研究量化報告。

週別	課程單元名稱	學習目標	教學設計重點
Week 4 3/22	Python 介紹 1	1. Python 程式初探 2. Python 套件認識: Anaconda, Jupyter	專家演講 (李孟霈 教育講師) Python 歷史 基本概念介紹、語法、功能 程式環境安裝與使用
Week 5 3/29	Python 介紹 2	1. Python 程式設計語法 2. 認識 NLTK 語言數據平台 3. Python 第三方函式庫介紹: Beautiful Soup、中文斷詞系統 jibea	專家演講 (李孟霈 教育講師) Python 基本程式語言練習 自學資源介紹 Codecademy PYTHON 練習
Week 8 4/19	NLTK & 語料庫 1	1. 認識語料庫資源 2. 練習 NLTK	專家演講 (胡翔喻 教育講師) Gutenberg Corpus

		3. 認識關鍵字、改寫 Python 程式	Brown Corpus Web and Chat Text 練習修改程式關鍵字
Week 9 4/26	NLTK & 語料庫 2	1. NLTK 進階語法(一) 2. 語料庫資源分析 3. Python 第三方函式庫應用: matplotlib 套件	專家演講 (胡翔喻 教育講師) Reuters Corpus Inaugural Address Corpus 語料分析
Week 11 5/10	Python 應用 1	1. NLTK 進階語法(二) 2. 網路爬蟲介紹、觀摩 3. Sentimental Analysis	實務演練工作坊 (李孟霈 程式教育講師) NLTK 語言學應用範例演示 1 語料庫關鍵字計算、比對、字詞分析製表 期末專題討論 Discussion 1
Week 12 5/17	Python 應用 2	1. 總複習 2. 專題研究討論	實務演練工作坊 (李孟霈 程式教育講師) Python 程式語言總複習 NLTK 語言學應用範例演示 2 期末專題討論 Discussion 2
Week 17 6/21	期末專題發表	電腦運算思維與程式設計融入語意學、社會語言學語料分析	教師主持 合作學習分組報告 學習回饋

➤ **Tronclass 教材呈現：**

Linguistics with Python

 0. 你的未來履歷表 用AI挑戰300萬年薪

 0. 【程式設計融入課程】學前問卷

 0. Learn Python --Intro.

 1. 0322python上課投影片 [查看檔案](#)

 1. 0322部分上課影片

 1. Codecademy 加分作業 [已結束](#) 繳交截止時間 2018.03.29 18:40

 2. 不是工程師，也可以作文本分析

 2. 0329python上課投影片 [查看檔案](#)

 2. 0329上課影片

 3. 用Python做自然語言處理的八個工具

 3. 4/19 python 教材 [查看檔案](#)

 3. NLTK簡單文本分析

 3. NLTK 中研院 Sinica Treebank

 3. 4/19 python 上課教材及課程錄影

 4. 4/26 Python [查看檔案](#)

 4. 文字運用頻率比較 [已結束](#) 繳交截止時間 2018.05.05 18:00

 5. 如何用Python做情感分析？

 5. Some projects of Sentiment Analysis

 5. 0510投影片

 5. 0510課程錄影part1

 5. 0510課程錄影part2

 Python 問題區 Any questions or difficulties? You can ask for help here!

 TOPICS for the Python Project

 Why tech needs the humanities_Eric Berridge_TED

[+ 新增學習活動](#)

FINAL PROJECT

1. Python Project

 Python Project [已結束](#) 繳交截止時間 2018.06.21 17:00

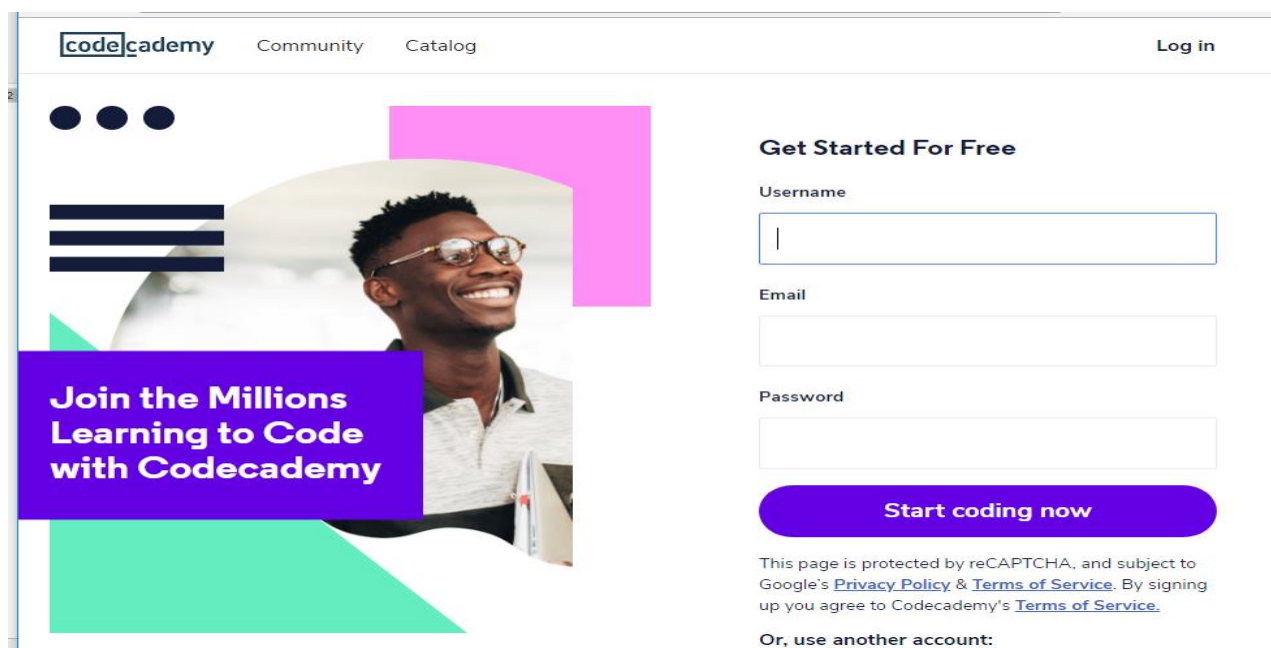
 深耕計畫PYTHON單元學習回饋

 Python project Revision [已結束](#) 繳交截止時間 2018.06.28 21:30

 線上問卷調查: FJCU 教學創新試辦計畫課程融入程式設計之學習自評

➤ 學生學習評量成果：

1. Codecademy 作業練習

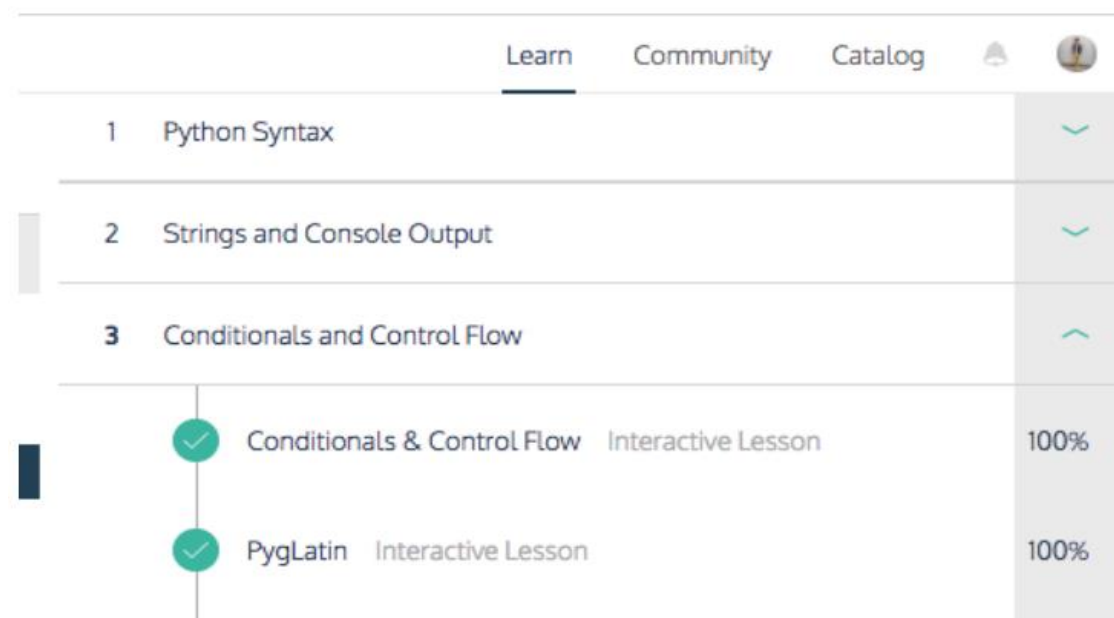


The screenshot shows the Codecademy homepage. On the left, there's a hero section with a smiling man and the text "Join the Millions Learning to Code with Codecademy". On the right, there's a "Get Started For Free" section with input fields for Username, Email, and Password, and a "Start coding now" button. Below the button, there's a reCAPTCHA notice and a link to "Or, use another account:".

✓ 作業敘述：

在 Codecademy Python 課程專區完成至少 3 單元。將完成進度條連同 Facebook 大頭照截圖上傳至作業繳交區即可加分。

截圖範例：



The screenshot shows the 'Learn' section of the Codecademy interface. It displays a progress bar for the 'Python Syntax' course. The progress bar is divided into three sections: '1 Python Syntax', '2 Strings and Console Output', and '3 Conditionals and Control Flow'. The first two sections are completed, indicated by green checkmarks. The third section is currently active and shows a progress bar. Below the progress bar, there are two lessons listed: 'Conditionals & Control Flow Interactive Lesson' and 'PygLatin Interactive Lesson', both of which are completed, indicated by green checkmarks and a 100% completion status.

Section	Lesson	Completion Status
1	Python Syntax	Completed
2	Strings and Console Output	Completed
3	Conditionals and Control Flow	In Progress
	Conditionals & Control Flow Interactive Lesson	100%
	PygLatin Interactive Lesson	100%

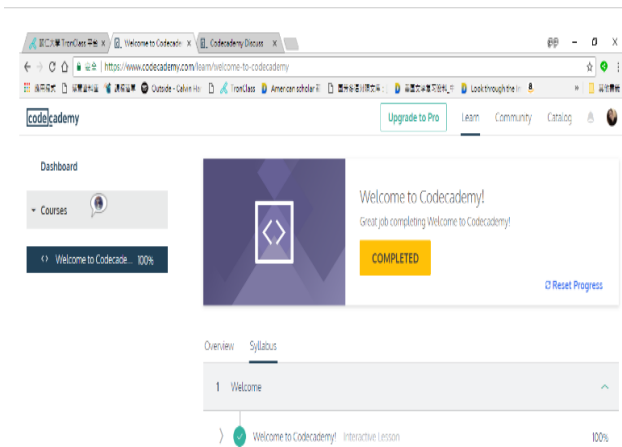
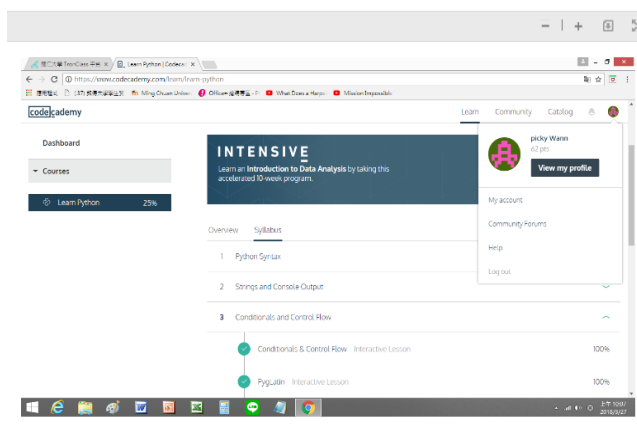
✓ 作業繳交狀況：

作業成績統計



✓ 學生繳交練習頁面

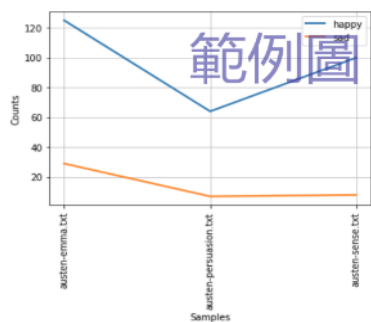
python作業.png



2. 語料庫分析：關鍵文字運用頻率比較圖表練習

✓ 作業敘述：

在Gutenberg 的語料庫中，找出 'austen-emma.txt', 'austen-persuasion.txt', 'austen-sense.txt' 各個文檔中，'happy' 跟 'sad' 出現的頻率，並透過 plot() 來畫圖。預期輸出結果如下



此次作業要記得安裝 matplotlib 的畫圖套件

Mac 的話在終端機(terminal) 輸入：python -mpip install -U matplotlib

Windows 的話在小黑窗(cmd) 輸入：python -mpip install -U matplotlib

作業繳交方式：

請附上程式碼與plot()畫圖結果之截圖

範例程式碼：

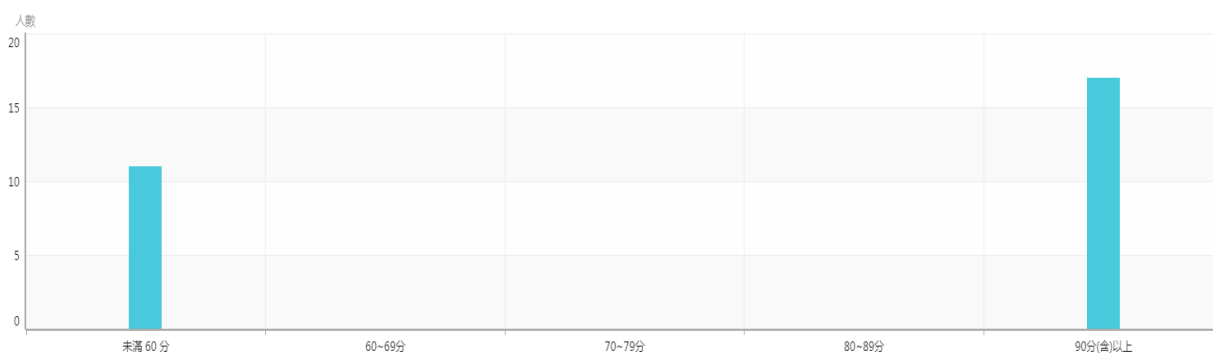
```
import nltk
from nltk.corpus import gutenberg
shakespeare = ['shakespeare-caesar.txt', 'shakespeare-hamlet.txt', 'shakespeare-macbeth.txt']
cfd = nltk.ConditionalFreqDist(
    (target, fileid)
    for fileid in shakespeare
    for w in gutenberg.words(fileid)
    for target in ['her', 'his']
    if w.lower().startswith(target))
cfd.plot()
```

✓ 作業繳交狀況：

文字運用頻率比較

作業類型 作業批改 成績統計

作業成績統計



3. python 遊戲教學網站 / The Hour of Code (<https://code.org/learn>)



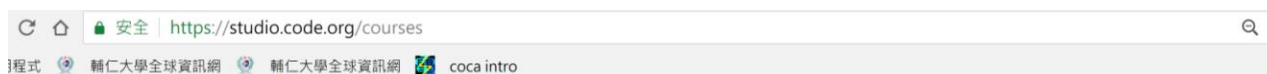
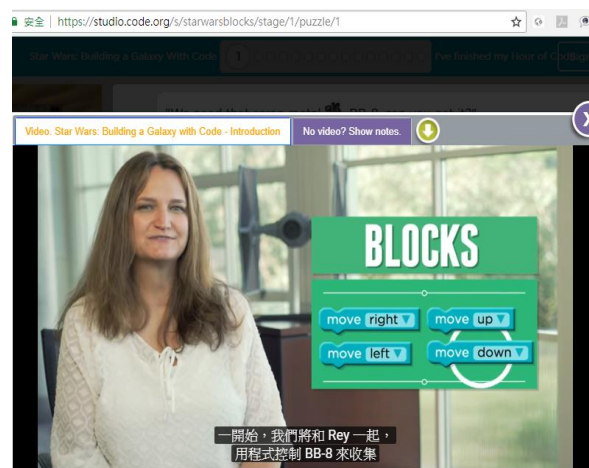
Star Wars: Building a Galaxy with Code

Code.org
Grades 2+ | Blocks, JavaScript

Learn to program droids, and create your own Star Wars game in a galaxy far, far away.

Start

More resources	Teacher notes
Short link	https://hourofcode.com/star-wars
Student experience	Beginner
Classroom technology	All modern browsers, Android tablet, iPad, Android phone, iPhone
Topics	Computer Science only



計算機科學基礎

Start learning an introduction to computer science with these 20 hour courses for all ages.

[檢視我最近的課程](#)

課程 1
給學齡前幼童的計算機科學介紹。
年齡 4-6

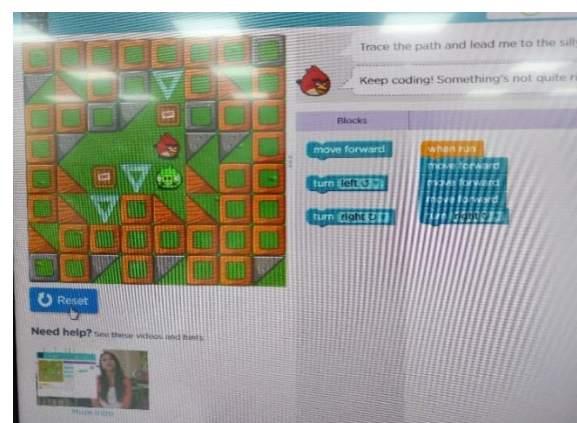
課程 2
給有閱讀能力學生的計算機科學介紹。
年齡 6+ (需閱讀能力)

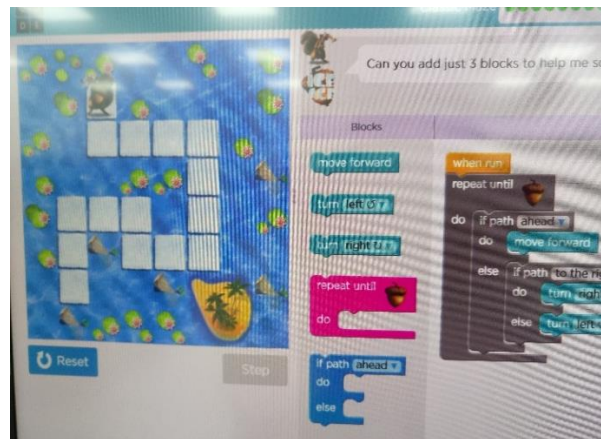
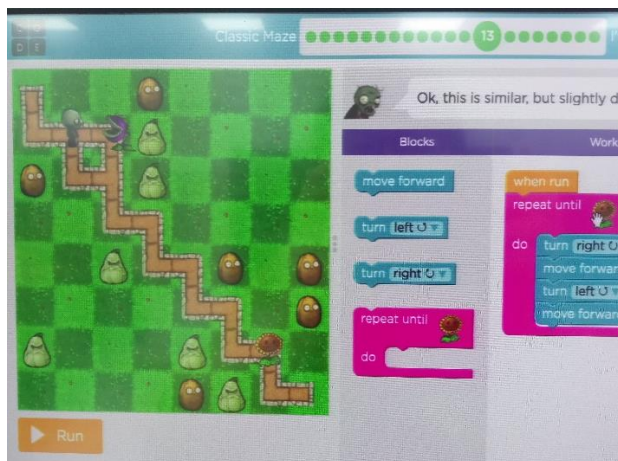
課程 3
課程 3 是課程 2 的延續課程，你可以更深入學習程式設計，用編程來寫出互動遊戲和故事。
年齡 8-18

課程 4
學生在完成課程 2 和課程 3 後，才可以進入課程 4 學習更複雜的編程概念，像是迴圈和帶參數的函式。
年齡 9-18

速成課程
在速成課程的課程 2-4 中學習基本的計算機科學。
年齡 10-18

不插電的教學活動
如果你沒有電腦，不妨試試這些不插電教學活動。
年齡 4+





4. 線上測驗：程式理解和應用

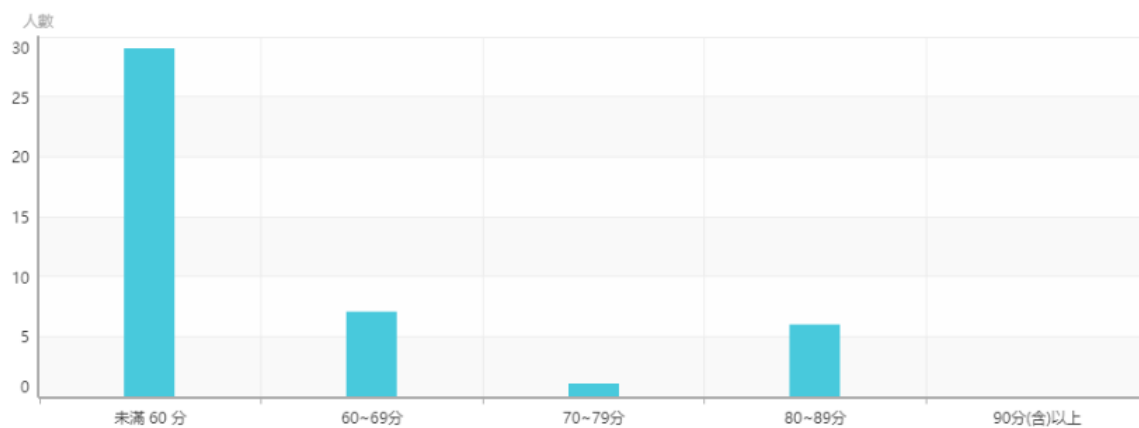
python quiz

[編輯](#) [匯出](#) [下載學生計](#)

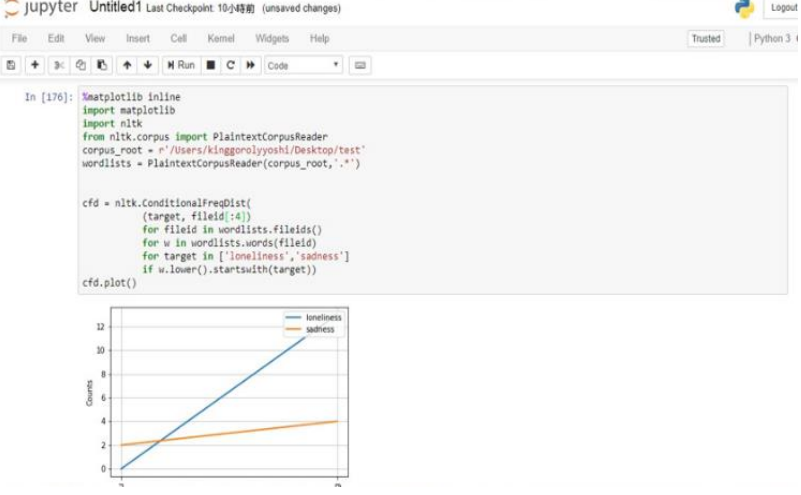
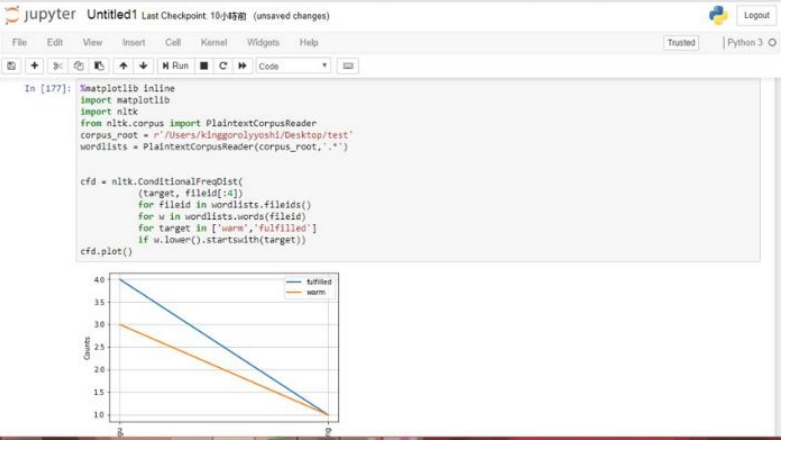
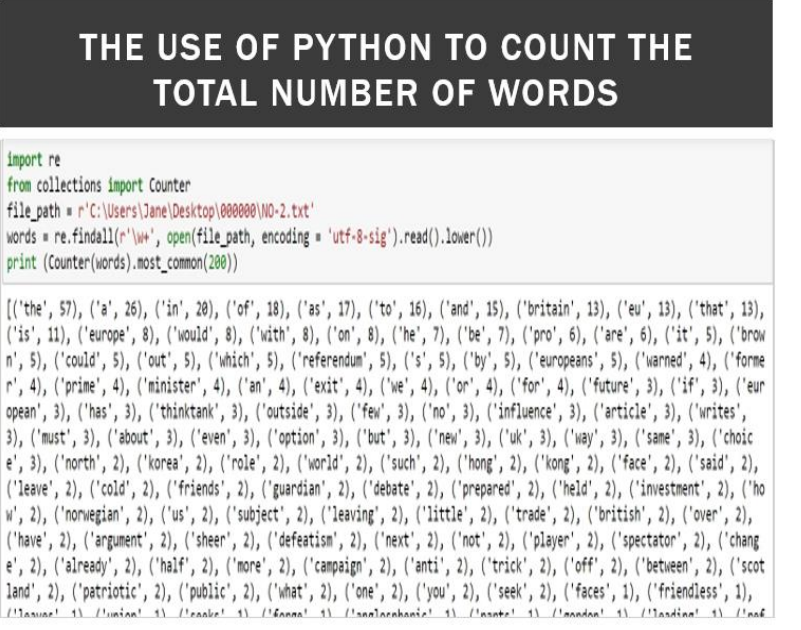
[基本資訊](#) [答題結果](#) [成績統計](#) [題目統計](#)

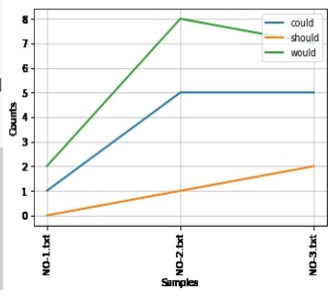
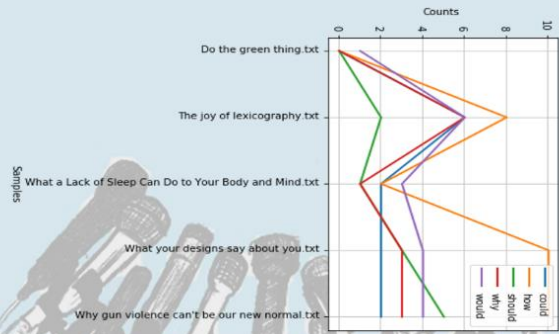
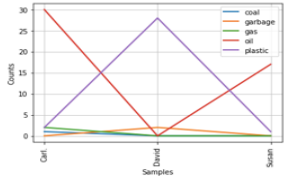
線上測驗成績統計

平均成績 48.4 分	最高成績 85 分	最低成績 0 分	平均嘗試次數 1.7 次
-----------------------	---------------------	--------------------	------------------------



➤ 期末專題報告

組別	期末小組專題報告主題	程式設計部分截圖
1	Different Genders' Aspects and Key Words Used in Speeches about Emotional Control	 The first screenshot shows a Jupyter Notebook cell with the following code: <pre> In [176]: %matplotlib inline import matplotlib import nltk from nltk.corpus import PlaintextCorpusReader corpus_root = r'/Users/kingorolyoshi/Desktop/test' wordlists = PlaintextCorpusReader(corpus_root, ".") cfd = nltk.ConditionalFreqDist((target, fileid[:4]) for fileid in wordlists.files() for w in wordlists.words(fileid) for target in ['loneliness', 'sadness'] if w.lower().startswith(target)) cfd.plot() </pre> The plot shows the frequency of words starting with 'loneliness' and 'sadness'. The 'loneliness' series (blue line) shows a higher frequency than the 'sadness' series (orange line).  The second screenshot shows a Jupyter Notebook cell with the following code: <pre> In [177]: %matplotlib inline import matplotlib import nltk from nltk.corpus import PlaintextCorpusReader corpus_root = r'/Users/kingorolyoshi/Desktop/test' wordlists = PlaintextCorpusReader(corpus_root, ".") cfd = nltk.ConditionalFreqDist((target, fileid[:4]) for fileid in wordlists.files() for w in wordlists.words(fileid) for target in ['warm', 'fulfilled'] if w.lower().startswith(target)) cfd.plot() </pre> The plot shows the frequency of words starting with 'fulfilled' and 'warm'. The 'fulfilled' series (blue line) shows a higher frequency than the 'warm' series (orange line).
2	Britain-divorced from the European Union: An Attitude Study	 The code snippet shows a Python script that counts the total number of words in a file: <pre> import re from collections import Counter file_path = r'C:\Users\Jame\Desktop\000000\NO-2.txt' words = re.findall(r'\w+', open(file_path, encoding = 'utf-8-sig').read()).lower() print(Counter(words).most_common(200)) </pre> The output shows the 200 most common words in the file, including 'the', 'a', 'in', 'of', 'as', 'to', 'and', 'britain', 'eu', 'that', 'is', 'europe', 'would', 'with', 'on', 'he', 'be', 'pro', 'are', 'it', 'brow', 'n', 'could', 'out', 'which', 'referendum', 's', 'by', 'europeans', 'warned', 'forme', 'r', 'prime', 'minister', 'an', 'exit', 'we', 'or', 'for', 'future', 'if', 'eur', 'opean', 'has', 'thinktank', 'outside', 'few', 'no', 'influence', 'article', 'writes', 'must', 'about', 'even', 'option', 'but', 'new', 'uk', 'way', 'same', 'choic', 'e', 'north', 'korea', 'role', 'world', 'such', 'hong', 'kong', 'face', 'said', 'leave', 'cold', 'friends', 'guardian', 'debate', 'prepared', 'held', 'investment', 'ho', 'w', 'norwegian', 'us', 'subject', 'leaving', 'little', 'trade', 'british', 'over', 'have', 'argument', 'sheer', 'defeatism', 'next', 'not', 'player', 'spectator', 'chang', 'e', 'already', 'half', 'more', 'campaign', 'anti', 'trick', 'off', 'between', 'scot', 'land', 'patriotic', 'public', 'what', 'one', 'you', 'seek', 'faces', 'friendless', 'leaves', 'union', 'banks', 'fence', 'unapologetic', 'nests', 'london', 'ladies', 'leaf'.

		PICK OUT THE WORD WITH HIGH FREQUENCY OR OBVIOUS POSITION <pre> %matplotlib inline import matplotlib import nltk from nltk.corpus import PlaintextCorpusReader corpus_root = r'C:\Users\Jane\Desktop\000000\no' wordlists = PlaintextCorpusReader(corpus_root, '.*') cfd = nltk.ConditionalFreqDist((target, fileid[:20]) for fileid in wordlists.fileids() for w in wordlists.words(fileid) for target in ['could', 'would', 'should'] if w.lower().startswith(target)) cfd.plot() </pre> 
3	The Analysis of Persuasive Speeches	About speeches standpoints -----Positive, Neutral, Negative? speech 1: What your designs say about you? Analyze Sentiment Language: english Enter text We are today talking about moral persuasion, what it moral and temporal, in trying to change people's behaviors by using technology and using design? And I don't know what you expect, but when I was thinking about that issue, I early on realized what I'm not able to give you an answer. I'm not able to tell you what is moral or temporal, because we're Enter up to 50000 characters Analyze Sentiment Analysis Results The text is pos. The final sentiment is determined by looking at the classification probabilities below: Subjectivity - neutral: 0.2 - polar: 0.8 Polarity - pos: 0.6 - neg: 0.4 To compare persuasive/ natural/unpersuasive speeches 
4	Environment_Ocean Pollution	Pollution <pre> %matplotlib inline import matplotlib import nltk from nltk.corpus import PlaintextCorpusReader corpus_root = r'Users\cloria\Downloads\cloria' cloria = PlaintextCorpusReader(corpus_root, '.*') cfd = nltk.ConditionalFreqDist((target, fileid[:5]) for fileid in cloria.fileids() for w in cloria.words(fileid) for target in ['garbage', 'oil', 'plastic', 'gas', 'coal'] if w.lower().startswith(target)) cfd.plot() </pre> 

Sentimental Analysis

Carl Safina → **Negative**
The oil spill's unseen villains -- and victims

Sentiment Analysis with Python NLTK Text Classification

This is a demonstration of sentiment analysis using a NLTK 2.0.4 powered text classification process. It can tell you whether it thinks the text you enter below expresses **positive sentiment**, **negative sentiment**, or if it's neutral. Using hierarchical classification, neutrality is determined first, and sentiment polarity is determined second, but only if the text is not neutral.

Analyze Sentiment
Language
english
Enter text
This is the ocean as I used to know it. And I find that since I've been in the Gulf a couple of times, I really kind of am traumatized because whenever I look at the ocean now, no matter where I am, even where I know that none of the oil has gone, I sort of see slicks, and I'm finding that I'm very much haunted by it. But what I want to talk to you about today is a lot of things that try to put all of this
Enter up to 50000 characters
Analyze

Sentiment Analysis Results
The text is **neg**
The final sentiment is determined by looking at the classification probabilities below.
Subjectivity
• neutral: 0.2
• polar: 0.8
Polarity
• pos: 0.3
• neg: 0.7

5

Gender Equality: A Case study of Sociolinguistics

Michael Kimmel

Sentiment Analysis with Python NLTK Text Classification

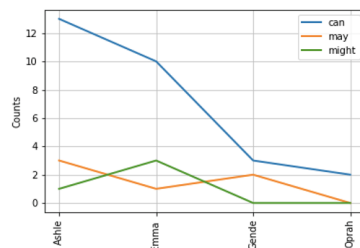
This is a demonstration of sentiment analysis using a NLTK 2.0.4 powered text classification process. It can tell you whether it thinks the text you enter below expresses **positive sentiment**, **negative sentiment**, or if it's neutral. Using hierarchical classification, neutrality is determined first, and sentiment polarity is determined second, but only if the text is not neutral.

Analyze Sentiment
Language
english
Enter text
I'm here to recruit men to support gender equality. (Cheers) Wait, wait, what? What do men have to do with gender equality? Gender equality is about women, right? I mean, the word gender is about women. Actually, I'm even here speaking as a middle class white man.
Enter up to 50000 characters
Analyze

Sentiment Analysis Results
The text is **pos**
The final sentiment is determined by looking at the classification probabilities below.
Subjectivity
• neutral: 0.5
• polar: 0.5
Polarity
• pos: 0.5
• neg: 0.5



Hedges – May / Might / Can

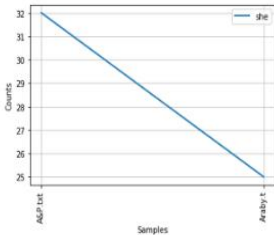
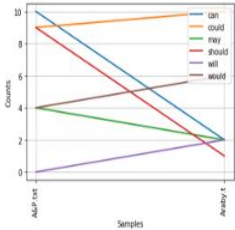


	Ashely	Gender	Oprah	Emma
May	3	2	0	1
Might	1	0	0	3
Can	13	3	2	10
Total	17	5	2	14

Female tend to use more hedges in speech. It is true both in Ashley's and Emma's, but not very obvious in Oprah Winfrey's.



6	The Attitude of Three American Presidents in Inauguration Speech	<pre>%matplotlib inline import matplotlib import nltk from nltk.corpus import PlaintextCorpusReader corpus_root = r'\\Users\\Marco\\Desktop\\Wordlist' Wordlist = PlaintextCorpusReader(corpus_root, '.*') cfd = nltk.ConditionalFreqDist((target, fileid[:5]) for fileid in Wordlist.fileids() for w in Wordlist.words(fileid) for target in ['justice', 'courage', 'citizen', 'freedom', 'democracy'] if w.lower().startswith(target)) cfd.plot()</pre>								
Sentimental Analysis - Positive - Bush -> (9) - Obama -> (13) - Trump -> (12) <pre>%matplotlib inline import matplotlib import nltk from nltk.corpus import PlaintextCorpusReader corpus_root = r'\\Users\\Marco\\Desktop\\Wordlist' Wordlist = PlaintextCorpusReader(corpus_root, '.*') cfd = nltk.ConditionalFreqDist((target, fileid[:5]) for fileid in Wordlist.fileids() for w in Wordlist.words(fileid) for target in ['good', 'better', 'great', 'strong'] if w.lower().startswith(target)) cfd.plot()</pre>										
7	The Analysis of Obama's speeches in Universities	Step1 The Length of a speech <table><tr><td>2009.1 Total 15264</td><td><pre>Thank you, God Bless You, and may God print (len(my_str)) #2009.1.txt < 15264</pre></td><td>2009.5 Total 20684</td><td><pre>If nothing else, that knowledge should gi May God bless you, and may God bless the print (len(my_str)) #2009.5.txt < 20684</pre></td></tr><tr><td>2010.5 Total 21679</td><td><pre>If you are willing, as past generations were willing Congratulations on your graduation, 2010. May God Thank you. print (len(my_str)) #2010.5.txt < 21679</pre></td><td>2010.11 Total 22833</td><td><pre>Sebagai penutup, saya mengucapkan kepada seluruh rakyat Thank you. print (len(my_str)) #2010.11.txt < 22833</pre></td></tr></table> Step5 Lexical category <pre>for tokens in words: tags.append(nltk.pos_tag(tokens)) tags Out[113]: [['Let', 'VB'), ('me', 'PRP'), ('begin', 'VB'), ('by', 'IN'), ('thanking', 'VBG'), ('George', 'NNP'), ('Mason', 'NNP'), ('University', 'NNP'), ('for', 'IN'), ('their', 'PRP\$'), ('extraordinary', 'JJ'), ('hospitality', 'NN'), ('and', 'CC'), ('to', 'TO'), ('thank', 'VB'), ('all', 'DT'), ('the', 'DT'), ('great', 'JJ'), ('friends', 'NNS'), ('the', 'DT'), ('governors', 'NNS'), ('the', 'DT')]</pre> <pre>tags Out[113]: [['Let', 'VB'), ('me', 'PRP'), ('begin', 'VB'), ('by', 'IN'), ('thanking', 'VBG'), ('George', 'NNP'), ('Mason', 'NNP'), ('University', 'NNP'), ('for', 'IN'), ('their', 'PRP\$'), ('extraordinary', 'JJ'), ('hospitality', 'NN'), ('and', 'CC'), ('to', 'TO'), ('thank', 'VB'), ('all', 'DT'), ('the', 'DT'), ('great', 'JJ'), ('friends', 'NNS'), ('the', 'DT'), ('governors', 'NNS'), ('the', 'DT')]</pre>	2009.1 Total 15264	<pre>Thank you, God Bless You, and may God print (len(my_str)) #2009.1.txt < 15264</pre>	2009.5 Total 20684	<pre>If nothing else, that knowledge should gi May God bless you, and may God bless the print (len(my_str)) #2009.5.txt < 20684</pre>	2010.5 Total 21679	<pre>If you are willing, as past generations were willing Congratulations on your graduation, 2010. May God Thank you. print (len(my_str)) #2010.5.txt < 21679</pre>	2010.11 Total 22833	<pre>Sebagai penutup, saya mengucapkan kepada seluruh rakyat Thank you. print (len(my_str)) #2010.11.txt < 22833</pre>
2009.1 Total 15264	<pre>Thank you, God Bless You, and may God print (len(my_str)) #2009.1.txt < 15264</pre>	2009.5 Total 20684	<pre>If nothing else, that knowledge should gi May God bless you, and may God bless the print (len(my_str)) #2009.5.txt < 20684</pre>							
2010.5 Total 21679	<pre>If you are willing, as past generations were willing Congratulations on your graduation, 2010. May God Thank you. print (len(my_str)) #2010.5.txt < 21679</pre>	2010.11 Total 22833	<pre>Sebagai penutup, saya mengucapkan kepada seluruh rakyat Thank you. print (len(my_str)) #2010.11.txt < 22833</pre>							

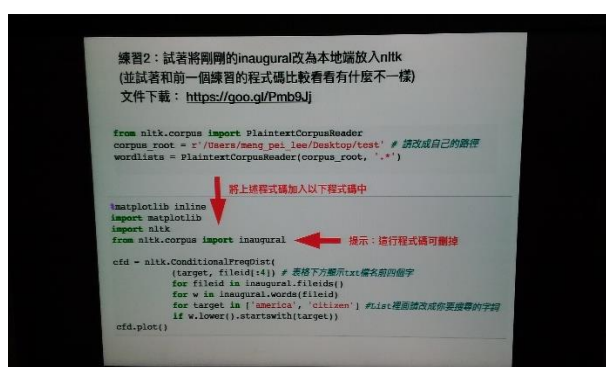
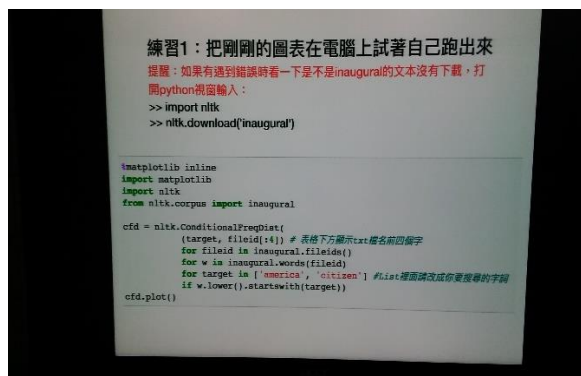
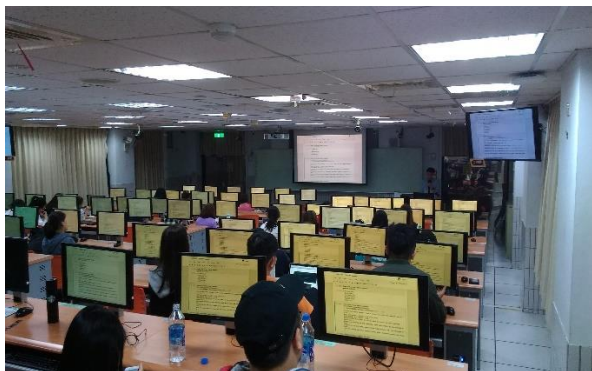
8	The Lyrics Analysis of the Three singers	Lovelorn- Back to December • Subjectivity neutral: 0.4 polar: 0.6 • Polarity pos: 0.2 neg: 0.8 **The text is neg.  https://goo.gl/KC9EuN Passionately in Love – And I Love Her • Subjectivity neutral: 0.1 polar: 0.9 • Polarity pos: 0.6 neg: 0.4 **The song is pos.  https://goo.gl/w8SoFu
9	Initiation Story- The Analysis of A&P and Araby by Python	<pre>In [35]: %matplotlib inline import matplotlib import nltk from nltk.corpus import PlaintextCorpusReader corpus_root = r'/Users/cindy/Desktop/python' wordlists = PlaintextCorpusReader(corpus_root, '.*') cfd = nltk.ConditionalFreqDist((target, fileid[:7]) for fileid in wordlists.fileids() for w in wordlists.words(fileid) for target in ['she']) if w.lower().startswith(target)) cfd.plot()</pre>  'she' <pre>In [33]: %matplotlib inline import matplotlib import nltk from nltk.corpus import PlaintextCorpusReader corpus_root = r'/Users/cindy/Desktop/python' wordlists = PlaintextCorpusReader(corpus_root, '.*') cfd = nltk.ConditionalFreqDist((target, fileid[:7]) for fileid in wordlists.fileids() for w in wordlists.words(fileid) for target in ['will', 'would', 'shall', 'should', 'can', 'could', 'may', 'might']) if w.lower().startswith(target)) cfd.plot()</pre>  Modal auxiliary verb 'will', 'would', 'shall', 'should', 'can', 'could', 'may', 'might'

● 課堂活動剪影

➤ Lecutres 課程剪影

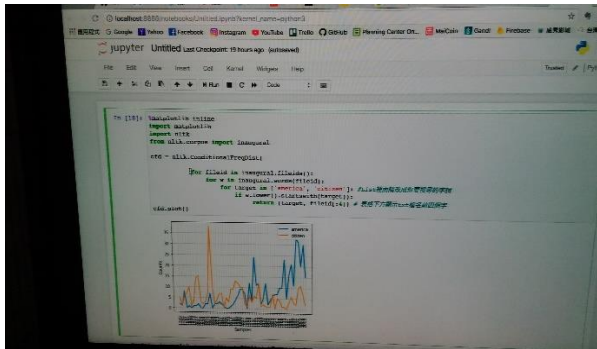


➤ Practices 程式語言練習

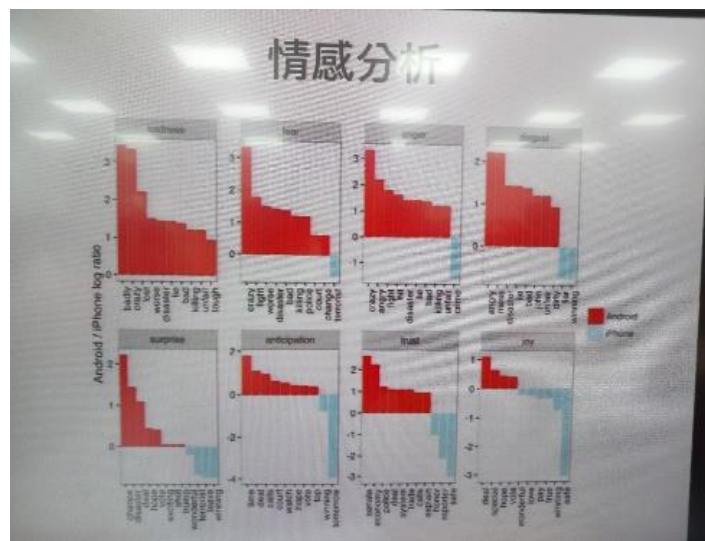




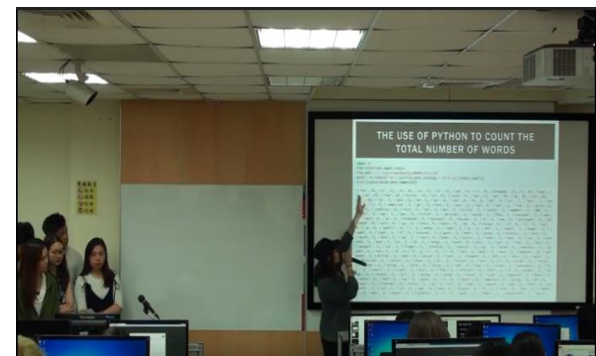
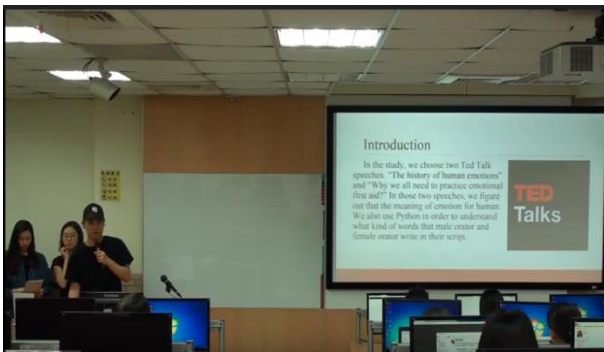
➤ Workshop 語料分析工作坊



➤ Discussion 專案討論



➤ Projects 期末語言學專題發表





➤ 教學團隊



● 授課心得感想

請授課教師根據此次程式設計融入課程學習活動之規劃與實施，作成效自評與歷程觀察摘要，並回饋反思與心得，以期作為個人與同儕未來改善與精進教學之參考依據...

➤ 計畫初衷：

對於多數的語文科系學生而言，如何運用程式語言運算進行語言分析是屬於相當陌生的跨領域學習；但學生若能經由培養邏輯思維能力與善用語料庫資源，必然能快速有效的分析語料，掌握語言學習策略。本計畫嘗試於「英語語言學概論」課程設計中，介紹學生認識「語料庫資源」與「PYTHON 程式設計」，並企圖導引學生探索語言學學科專業的資訊跨域整合

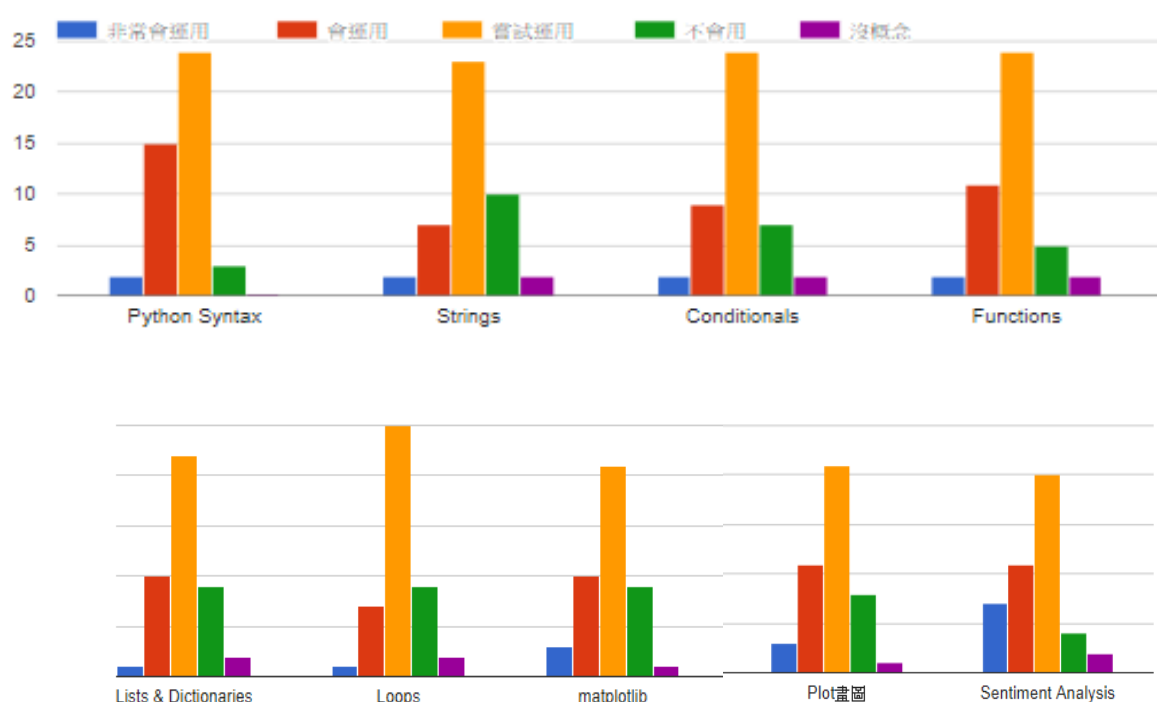
實踐應用，使學生透過運算程式分析、理解和處理自然語言語料，能有效掌握學科學習重點，進行語言學專題研究。

➤ 歷程觀察與回饋反思

參與本計畫學生為英文系進修學士班二年級必修科目「英語語言學概論」，修課人數計 50 名，問卷回應男女比例為 79.5%:20.5%，成員包含二年級、三年級、四年級及研究生。雖然在課前問卷中(41 份回應)顯示，本班學生有 63.4%未曾修讀過資訊/多媒體相關課程，53.7%的學生表示對程式語言的運用沒概念、不知道什麼是 PYTHON 程式語言，也有 34.1%的學生表示知道 PYTHON 程式但不會用。但多數學生認為在目前/未來的工作中會使用到程式應用的卻高達 78% (會用到 19.5% + 未來可能會用到 58.5%)。

經過六週 PYTHON 課程的學習，雖然人文學科學生對於 PYTHON 程式語言及運用仍有一定程度的理解障礙，部分學生表達運用程式設計語法的困難性，仍呈現出願意嘗試資訊新知技能的積極正向心態，此點著實讓教學者欣慰。在學習回饋問卷(44 份回應)數據中，也發現學生對於學習 PYTHON 套件安裝執行與各程式功能語法的肯定回應 (Q2、Q3、Q4 會安裝與執行運用 Jupyter notebook 86.4%、Anaconda 75%，其中中文斷詞系統 Jibea 20.5%偏低原因則可能因為語料分析練習皆以英文語料為主，此點可以在未來再深入探究)；在學習回饋問卷中顯示對於程式語法、功能運用，不論是會用或嘗試運用的數值皆顯示較高；

5. 請問你知道如何執行以下程式語言功能嗎？



短短六次的 PYTHON 培訓，研習時間是絕對不夠的，但看到各分組學生期盼能掌握 PYTHON 程式語言運用技術，不論在上課時或是下課期間都會踴躍發問，同學們也能利用機會練習與討論交流語言學期末專題研究主題，他們表現出的主動學習精神與認真積極態度，都遠比程式教育講師及助教們原先預期的更好，這不但使得計畫教學進度順利執行，我們也有機會提供更多題材 (延伸學習) 給予學生練習，強化學生思考期末專題結合運用程式語言分析語料庫的信心與豐富性。在學前問卷(Q5)與學習回饋問卷(Q7)中，學生對於本計畫實用的學習單元，列舉出之個別選項大抵都呈現正成長數據；唯一數據較低的學習單元為

練習 The Hour of Code (11.4%)，該網站除了有豐富有趣的學習、練習 CODE 闖關遊戲，又有可作為英語學習的生動解說影音資源，讓教師訝異於不受學生青睞，分析原因可能為介紹於最後一堂總複習課程中，練習時間過於急促，使得學生來不及消化、吸收。

不可諱言的，計畫中雖然也有排斥學習 python 程式的學生，但有更多位學生屢次表達受惠於此次資訊融入學科專業計畫，讓他們開拓了學習視野，看到了專業學習與自我發展的可能性；甚至有些學生還會詢問其他學習程式語言的方式與運用，這些都是本計畫規劃初期未曾設想到的附加效益。相信經過此次 python 入門課程，引導學生從被動認識、理解、到積極運用程式語言的「我會」、「我能」過程，成功降低初學者從開始對程式語言的恐懼，到啟發學生對於「運用程式語言於語言學專業學科」的興趣與目標，未來如學習其他程式語言也會更加具有成效。於此，學生瞭解了運算思維的原理，不但增強了數位競爭力，也發現了學習的無限可能性！

➤ 未來改善與精進教學

對於非資訊專業的教師而言，要能妥適將「運算思維認識」與「PYTHON 程式設計」學習目標，融入專業學科的課程設計與教學活動中，讓學生在原本的課程時間既要習得本科應備學術知能，又要能吸收運用資訊技能，不論是對教師本人、還是對學生個人，都是很大的挑戰。所幸因教師曾參與過【外語學院運算思維跨域創新師生社群】，又因為有計劃的經費支持，才得以邀請有經驗的資訊教育講師進行專題講演及指導，執行此次跨域課程。但如果無相對經費支援、無相關資訊業師配合，僅憑單一教學者能力，恐是力有未逮，無法達標！因此希望能有持續經費補助與足夠資訊教學人才庫，才能嘉惠更多學子。

另外，因為課程屬性，學習場域設定為資訊教室，期使學生能很快入手學習程式語言。雖然資訊中心同仁熱心幫忙挪借教室，然而為期數週的培訓課程實在難以安排例行資訊教室；形成了每週需更換不同教室的「且戰且走」窘況，甚至於原本規劃為三堂課的電腦學習時間，因只能有兩堂資訊教室可用，使得第三堂課必須調動至普通教室的狀況，造成學習中斷與資訊不完全(重新下載程式、套件、網路頻寬)的遺憾，也使得學習時間與成效大打折扣。未來如有機會再執行此類計畫，期盼能有充足資訊教室與頻寬可供使用。

感謝「**程式設計融入課程補助計畫**」，讓教師有機會重行檢視本系專業必修語言學學科的教學內容，尋求數位時代的資訊跨域整合應用性；也讓教師在執行計畫過程中，能以創新角度調整原本教材設計與教學活動，發展出「**LPWDP**」學用合一的任務導向教學模式，實踐專業學科的數位行動力，具體提升教學品質！



