

輔仁大學 111 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

開課學系	英國語文學系進修學士班	學制別	大學 ☐ 日間部 ☒ 進修部
學年度/學期	<u>110</u> 學年度 / 第 <u>2</u> 學期	選別	☒ 必修 ☐ 選修 ☐ 通識
課程名稱	英語語言學概論	上課時間	星期 <u>四</u> 18:40~20:20
開課代碼	C202202984	修課人數	40
授課教師	樂麗琪	聯絡電話	(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	本計畫介紹語文科系學生認識人文學科與程式語言運用的關聯性，促進學生產生培養運算思維 (CT, Computation Thinking) 與邏輯思考能力之動機，進而能以程式語言運算分析人類自然語言，此應是跨領域學習能力擴展；而語文教師將程式語言運用在人文學科教學中，實踐資訊整合應用的專業學科數位行動力，亦是屬於語言教學跨域新課題，增進教學新產能。 課程設計嘗試將「運算思維認識」與「PYTHON 程式設計」融入英文系專業必修科目「英語語言學概論」課程設計與教學活動當中，透過探索語言學學科專業的資訊跨域整合實踐應用，使學生經由觀摩 Coding 應用實例，認識「PYTHON 程式語言」與「自然語言處理工具箱 NLTK (Natural Language Tool Kit)」的實用性，學習到如何應用運算程式分析、處理人類自然語言語料，而因此能覺察出語言結構的可分析性及規律性，並能有效掌握學科學習重點，進行語言學專題研究，進而發展語言分析策略，因而產生自主學習動機，能自我強化邏輯思維及語言分析能力；藉此培養學生資訊素養過程，也能幫助學生建立團隊學習態度，引發合作發想研究計畫能力，最終能完成數位分析量化報告，產出語言學科專題研究，從跨領域學習範疇中，擴展其學科專業研究能力。
	程式語言 ☒ Python ☐ APP Inventor 2 ☐ R ☐ Javascript ☐ Scratch ☐ VBA ☐ Processing ☒ 其他 <u>NLTK 語料庫、WordCloud 文字雲</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. 認識語言學、NLP 與 Python 程式應用的相關性。 3. 學習 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 NLTK 語料庫及運用 2. 進行語意學之語料分析。 3. 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. 理解 Python 程式語言、簡易程式撰寫與改寫應用
2. 學習和使用 NLTK 語料庫處理自然語言語料
3. 製作語料分析視覺化圖表

• 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. 願意嘗試跨領域學習、強化資訊素養。
3. 培養團隊合作、自主發想研究計劃能力。

個人作業: ☒ 書面 ☐ 簡報 8 次

小組報告: ☐ 書面 ☒ 簡報 3 次

程式設計(個人): 6 次 (含課堂練習)

程式設計(小組): 2 次

☒ 其他 Workshop 討論: 2 次

Google Survey 學前、學習回饋問卷 3 次

● 學前問卷 (110/2)

https://docs.google.com/forms/d/e/1FAIpQLSdh9YjYAXxfb1DAYQtChaLuv0zx4Jf1d_vP22MHF49A2Njruw/viewform



輔仁大學111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 (110/2)

各位同學您好:

本課程參與了 [輔仁大學111年度高教深耕計畫 程式設計融入課程: 英語語言學概論], 希望學習內容將對您有所助益。

為了瞭解您對於本單元教學法及使用教材的期待與想法, 敬請填寫本問卷。祝福 學習愉快!

請填入下列聯絡資料與回答問題

yuehfjcu@gmail.com 切換帳戶

*必填

● PYTHON 單元學習回饋問卷

https://docs.google.com/forms/d/e/1FAIpQLSdUL7KB0TrZH9rdwhVNrnMPBHnXK4BPPoHSWiZ-Lb_wmNQAbQ/viewform



輔仁大學111年度高教深耕計畫【程式設計 融入課程: 英語語言學概論】學習回饋課程 問卷

各位同學您好:

感謝您在本課程參與了 輔仁大學111年度高教深耕計畫 [程式設計融入課程: 英語語言學概論], 希望課程學習內容對您的學習成效與工作職能有所助益。為了瞭解您對於本單元學習與課程精進意見, 敬請填寫本問卷。

● 校方線上問卷調查: FJCU 教學創新計畫課程融入程式設計問卷 (School)

https://docs.google.com/forms/d/e/1FAIpQLScYxy6hIcgBgVyNJYrSiJhtaug_EpMHhK6wvZkpY_R008hNlw/viewform



110-2學期程式設計融入課程問卷

本問卷請修課學生填答之目的，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。

● 形成性評量之規劃 (隨堂練習或小考等):

✓ 隨堂練習: 1. 編程遊戲學習網站 **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

2. **WordCloud** 語料文字雲製作

3. **Sentiment Analysis** 情感分析練習

✓ 線上測驗: Python 程式語言理解和應用 **Google Colab Practice**

✓ Tronclass 作業區:

量 設 計	☐ 2. 0512 Practice 1	時間: 2022.05.12 20:20 ~ 2022.05.19 18:00	3.0%	個人作業 範圍: 全部學生 人數: 38 人
	☐ 2. 0519上課練習	時間: 2022.05.19 17:58 ~ 2022.05.22 23:30	3.0%	個人作業 範圍: 全部學生 人數: 38 人
	☐ 3. NLTK Practice	時間: 2022.05.26 18:30 ~ 2022.06.02 19:00	3.0%	個人作業 範圍: 全部學生 人數: 38 人
	☐ 4. 0602上課練習	時間: 2022.06.02 18:01 ~ 2022.06.05 23:55	3.0%	個人作業 範圍: 全部學生 人數: 38 人
	☐ 5.0609 NLTK with matplotlib	時間: 2022.06.09 18:30 ~ 2022.06.12 23:55	5.0%	個人作業 範圍: 全部學生 人數: 38 人
	☐ 6. 0616 NLTK 文字雲	時間: 2022.06.16 18:30 ~ 2022.06.19 23:55	5.0%	個人作業 範圍: 全部學生 人數: 38 人
	☐ The Final Project	時間: 2022.06.16 19:30 ~ 2022.06.23 18:30	15.0%	分組作業 範圍: 全部小組 分組方案: Groups

● 總結性評量之規劃 (期中考、期末考或專題成果等):

✓ 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

線上資源: ☐ Codecademy ☐ Coursera ☐ Code school

☒ 其他 [Linguistic Corpora](#)、[PY app](#)、[Hour of Code](#)、[NLTK](#)、[WordCloud](#)、[Sentiment Analysis](#)

實體資源：☒專題演講 ☒其他 [課程簡報資料及課程實錄影音檔](#)

Week 1:

- 你的未來履歷表 用 AI 挑戰 300 萬年薪 <https://www.gvm.com.tw/article.html?id=43043>
- 文組不會寫程式?數學能力對程式學習僅有 2 % 影響
https://buzzorange.com/techorange/2020/05/20/liberal-arts-learn-python-better/?fbclid=IwAR06vrrkOVIPD8QvWbQz-ITWv_OYmmGgKUlxZCumQOgePXMghcyfRU50cqs
- Natural Language Processing (NLP 自然語言處理)
<https://research.sinica.edu.tw/nlp-natural-language-processing-chinese-knowledge-information/>
- What is NLP? <https://www.youtube.com/watch?v=8S3qHHUKqYk>
- Learn Python – Introduction <https://www.youtube.com/watch?v=T5pRIIbr6gg>

Week 2:

- 不是工程師，也可以作文本分析
<https://buzzorange.com/techorange/2017/03/03/rmakeyoueasy/>
- Using Google Colab to write Python
<https://colab.research.google.com/notebooks/welcome.ipynb>
- What is Colab? <https://www.youtube.com/watch?v=inN8seMm7UI&feature=youtu.be>

Week 3:

- [歌詞小數據：1976、五月天和蘇打綠都愛寫進歌裡的詞？](#)
- 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: Natural Language Processing with Python http://www.nltk.org/book_1ed/

Week 4:

- NLP 實戰：10 種自然語言處理技術 <https://kknews.cc/tech/2vmgg6z.html>
- 如何用 Python 做情感分析？
<http://blog.sciencenet.cn/blog-377709-1062960.html>
- Text analysis of Trump's tweets 1
<https://medium.com/swlh/analyzing-trumps-tweets-5368528d2c90>
- Text analysis of Trump's tweets 2
<http://varianceexplained.org/r/trump-tweets/>
- Beautiful Soup 模組讀取 HTML 原始碼
<https://blog.gtwang.org/programming/python-beautiful-soup-module-scrape-web-pages-tutorial/>

Week 5:

- [Some projects of Sentiment Analysis](#)
- Word Net: Lexical Semantics <https://medium.com/pyladies-taiwan/nltk-%E5%88%9D%E5%AD%B8%E6%8C%87%E5%8D%97-%E4%B8%89-%E5%9F%BA%E6%96%BC-wordnet-%E7%9A%84%E8%AA%9E%E7%BE%A9%E9%97%9C%E4%BF%82%E8%A1%A8%E7%A4%BA%E6%B3%95-%E4%B8%8A%E4%B8%8B%E4%BD%8D%E8%A9%9E%E7%B5%90%E6%A7%8B%E7%AF%87-4874fb9b167a>
- Create Word Cloud using Python

<https://www.tutorialspoint.com/create-word-cloud-using-python>

- Chinese Word Cloud with Python
https://medium.com/@yeung_cyda/written-in-chinese-step-by-step-chinese-word-cloud-with-python-53f57f65be69

Week 6 & 7:

- 斷開中文的鎖鍊！自然語言處理 (NLP)
<https://research.sinica.edu.tw/nlp-natural-language-processing-chinese-knowledge-information/>
- NLTK 中研院 Sinica Treebank
http://www.hankcs.com/program/python/nltk-chinese-corpus-sinica_treebank.html
- Sentiment Analysis with Python NLTK Text Classification
<http://text-processing.com/demo/sentiment/>
- Why tech needs the humanities_Eric Berridge_TED
<https://www.youtube.com/watch?v=F2XPF6rQ6fs>

教學設計

	日期	課程單元名稱	學習目標	教學設計重點
1	5/5	NLP Introduction	1. NLP 典範介紹 Python 程式認識	專家演講 (實體課程) - Python 歷史、Coding 實例 - 基本概念介紹、語法、功能 - Hour of Code 實例與練習
2	5/12	Linguistics with Python	1. Python 程式初探 2. 認識 Colab	專家演講 (實體課程) - Python 基本程式語言練習 - PY app 自學資源介紹 - 觀摩 python 程式應用實例
3	5/19	Basic Python NLTK 語料庫	1. 認識 NLTK 語料庫資源 2. 認識 Python 語法關鍵字	專家演講 (線上課程) - 學習運用 Python 語法分析 - 學習 NLTK 基礎編程
4	5/26	NLP 和 NLTK 語料庫	1. 認識 NLTK 語法 2. 語料庫文本資源 3. Python 程式範例	專家演講 (線上課程) - NLTK 語言應用範例演示 - NLTK 語料庫文本語料分析 Gutenberg Corpus Brown Corpus
5	6/2	NLTK Application 1	1. NLTK 進階語法 2. 語料庫文本分析、字詞出現頻率圖表製作 3. 文字雲介紹、觀摩	實務演練工作坊 (線上進行) - NLTK 語言應用範例演示 - 觀摩 python 程式應用實例 - 認識關鍵字、練習改寫程式
6	6/9	NLTK Application 2	1. NLP 文本語意情感分析 Sentiment Analysis 2. 練習導入文本，進行語料分析 3. 專題研究討論	實務演練工作坊 (線上進行) - 語料庫關鍵字計算、比對、字詞分析製表 - python 程式應用實例: TED 講稿文本分析 - 專題討論 Discussion

7	6/16	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 / 專案指導講評 • 合作學習分組報告: 專題發表 • 學習回饋
---	------	--------	------------------------------------	--

➤ 實施方式：LPWDP 教學模式建構

本課程規劃將此數位學習實踐於「語意學」與「社會語言學」兩單元中，課程內容以程式應用於語料分析為聚焦，教學內容為 NLP (自然語言處理 Natural Language Processing) 實例觀摩、Python 基礎認識、NLTK (自然語言處理工具箱 Natural Language Toolkit) 運用、語料庫分析 (語意分析、語篇分析)，關鍵字詞分析統計圖表、文字雲語法練習等，引導學生產出語言學專題研究量化報告。研究方法為透過學前問卷、學習回饋單調查與觀察法，解析學習現象、學生專題研究成果；教學方式經由「任務導向教學法」(Task-based Approach) 及「合作學習法」，培養學生團隊合作、發展研究計劃能力。學習活動安排專家演講、程式語言教學、工作坊、專題發表等方式進行，試圖建構「學用合一」的“LPWDP”任務導向教學模式。

活動設計從學生學習成果 (Learning Outcomes) 能展現「語言學專題研究發表」為任務目標。學習活動以「電腦運算思維與程式設計」融入「任務導向教學法」(Task-based Approach) 及「合作學習」(Collaborative Learning) 教學活動，透過專家演講、程式語言學習、線上工作坊等方式進行學習活動。教學內容聚焦於

1. Coding 應用於語言情境實例觀摩
2. Python 程式認識，基本概念、Colab 語法、功能介紹
3. NLP 探索、學習使用 NLTK 語料庫分析自然語言 (語意分析、語篇分析)
4. 解讀、改寫 python 程式關鍵字功能，
5. 文字雲、關鍵字計算、字詞分析、統計圖表等運用

學生在學習活動過程中，經由觀摩程式語言應用實例，認識程式語法、NLTK 與語料庫資源的實用性，學習到如何運用運算分析處理自然語言語料，學生也透過解讀、改寫 python 程式關鍵字，熟用語料庫資源，藉由處理視覺化資料分析，明瞭培養邏輯思考與程式運算能力，能快速有效的分析處理自然語言語料，而覺察出語言結構的規律性與可分析性，除促使其產生學習動機、自我強化運算思維及語言學習能力，進而能發展資訊素養與自主學習策略，並進行語言學資訊專題，擴展學科研究能力。

➤ “LPWDP” 教學模式建構概要：

1. Lectures: 邀請專家演講教授 NLP 與 Python 程式語言 (學習啟程)

課程單元名稱	學習目標	教學內容
NLP Introduction	1. Python 程式初探、基本語法 2. Python 編寫環境認識: Colab、Jupyter	• Python 歷史 • 基本概念介紹、語法、功能 • Hour of Code 實例練習
Linguistics with Python	1. Python 程式初探 2. 認識 Colab	• Python 基本程式語言練習 • PY app 自學資源介紹 • 觀摩 python 程式應用實例

2. Practices: 進程式改寫應用練習於文本分析 (運算體驗)

課程單元名稱	學習目標	教學內容
Basic Python NLTK 語料庫	1. 認識語料庫資源 2. 練習 NLTK	• 學習運用 Python 語法分析 • NLTK 語料庫文本: Gutenberg

	3.認識 Python 語法關鍵字	Corpus、Brown Corpus、Web and Chat Text
NLP 和 NLTK 語料庫	1.NLTK 進階語法 2.語料庫文本分析、字詞出現頻率 圖表製作 3. 網路資源擷取、語料分析	- NLTK 語言應用範例演示 - 語料分析: Reuters Corpus & Inaugural Address Corpus - 練習改寫程式關鍵字

3. Workshop: 辦理語料分析視覺化工作坊 (工具熟用)

4. Discussion: 討論語言分析專題 (合作學習、創作發想)

課程單元名稱	學習目標	教學內容: 實務演練工作坊
NLTK Application 1	1. NLTK 進階語法(二) 2. 學習導入自己的語料庫 3. 文字雲介紹、觀摩 4. NLP 文本語意情感分析 Sentiment Analysis	- NLTK 語言應用範例演示 - 觀摩 python 程式應用實例: TED 講稿文本分析 - 專題討論 Discussion 1
NLTK Application 2	1. 總複習 2. 專題研究討論	- 語料庫關鍵字計算、比對、字詞分析製表 - 專題討論 Discussion 2

5. Projects: 發表語言學專題研究報告 (學習統整)

課程單元名稱	學習目標	教學內容 教師主持、專案指導講評
Linguistics with Python 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	- 合作學習分組報告 - 學習回饋

➤ 課堂演講影音檔

- Week 2 0512 課程錄影內容 (因為疫情漸趨嚴重，校方開始實施線上教學演練，本周暫時以實體+線上方式進行課程)
- Week 3 0519 課程錄影內容 (因為疫情嚴峻，校方宣布開始實施線上教學，自本周起線上教學方式進行課程): [0519 20110519-Meeting Recording.mp4 \(sharepoint.com\)](#)
- Week 4 0526 課程錄影內容: [0526 TEAMS-20220526 104609-Meeting Recording.mp4 \(sharepoint.com\)](#)
- Week 5 0602 課程錄影內容: [0602 184035-會議錄製.mp4 \(sharepoint.com\)](#)
- Week 6 0609 課程錄影內容: [Linguistics Online Meeting on 06 09-20220609 184145-會議錄製.mp4 \(sharepoint.com\)](#)
- Week 7 0616 課程錄影內容: [Linguistics Online Meeting on 06 16-20220616 184004-會議錄製.mp4 \(sharepoint.com\)](#)
- Week 8 0623 期末專題報告錄影內容: 「The Final Project on 0623」中的會議-[20220623 184924-Meeting Recording.mp4 \(sharepoint.com\)](#)

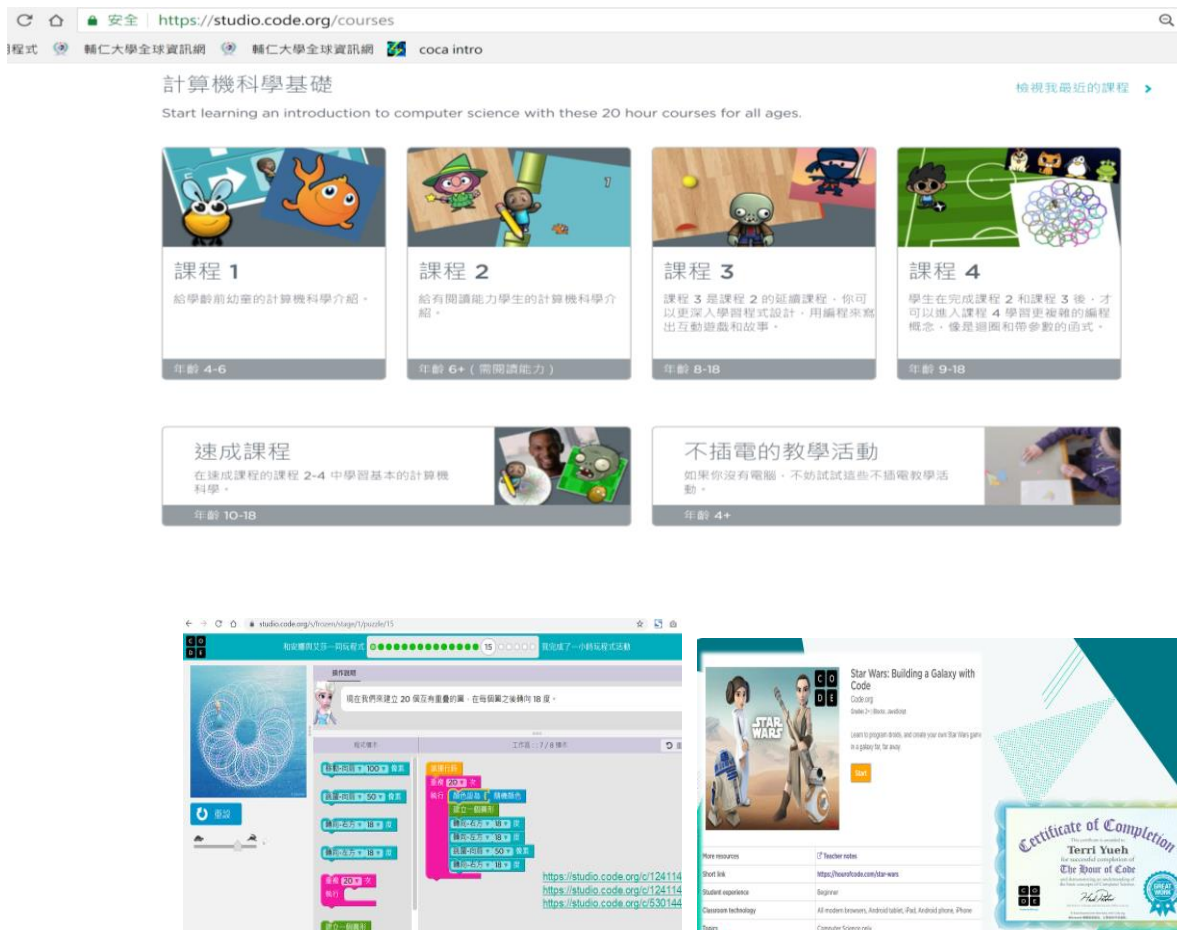
➤ 學生學習評量成果：

1. **作業一（個人）：**完成 Code Studio (Hour of Code <https://code.org/>) 的迷宮課程，利用圖像的方式學習簡單的程式邏輯，完成後繳交獲得證書截圖上傳至 Tron Class。

圖 1.1 hour of code 課程



圖 1.2 研習內容



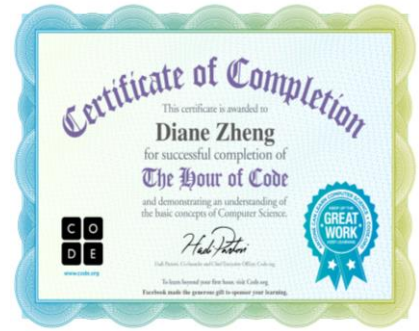
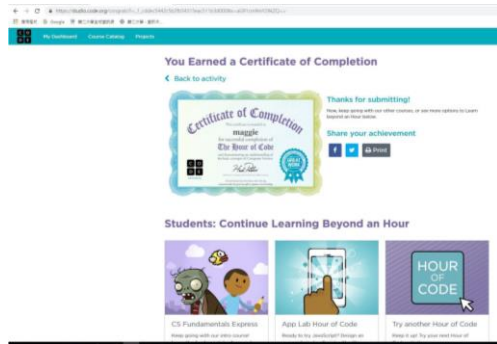


圖 1.3 完成證書

2. 作業二 (個人): 完成 PY APP 的課程

@複習課堂所學的 PYTHON 語法 (print、string、variable、Math、Numberboolean) ,

@複習課堂所學的 PYTHON 語法 (if 條件式、for、list) , 完成後繳交頁面截圖。

教師評閱	
成績比率	100.0%

一、完成至booleans單元並截圖上傳

二、Py - Learn to Code下載網址：

1. IOS : <https://apps.apple.com/tw/app/py-learn-to-code/id1111105894>

至App Store下載

2. Android : <https://apkpure.com/tw/py/com.py.learn>

用手機點選網址下載apk檔，進行安裝

P.S.請參照作業附件的截圖畫面，進行截圖，有問題可以問adam@weco.net

附件名	檔案大小
 Py_v1.1.10_apkpure.com.apk	13 MB
 py截圖範例2.jpg	321 KB
 Py截圖範例.png	191 KB

圖 2.1 PY-learn to code APP



圖 2.2 作業內容

A program is a **recipe** that tells a computer what to do. Can you figure out what this program is doing?

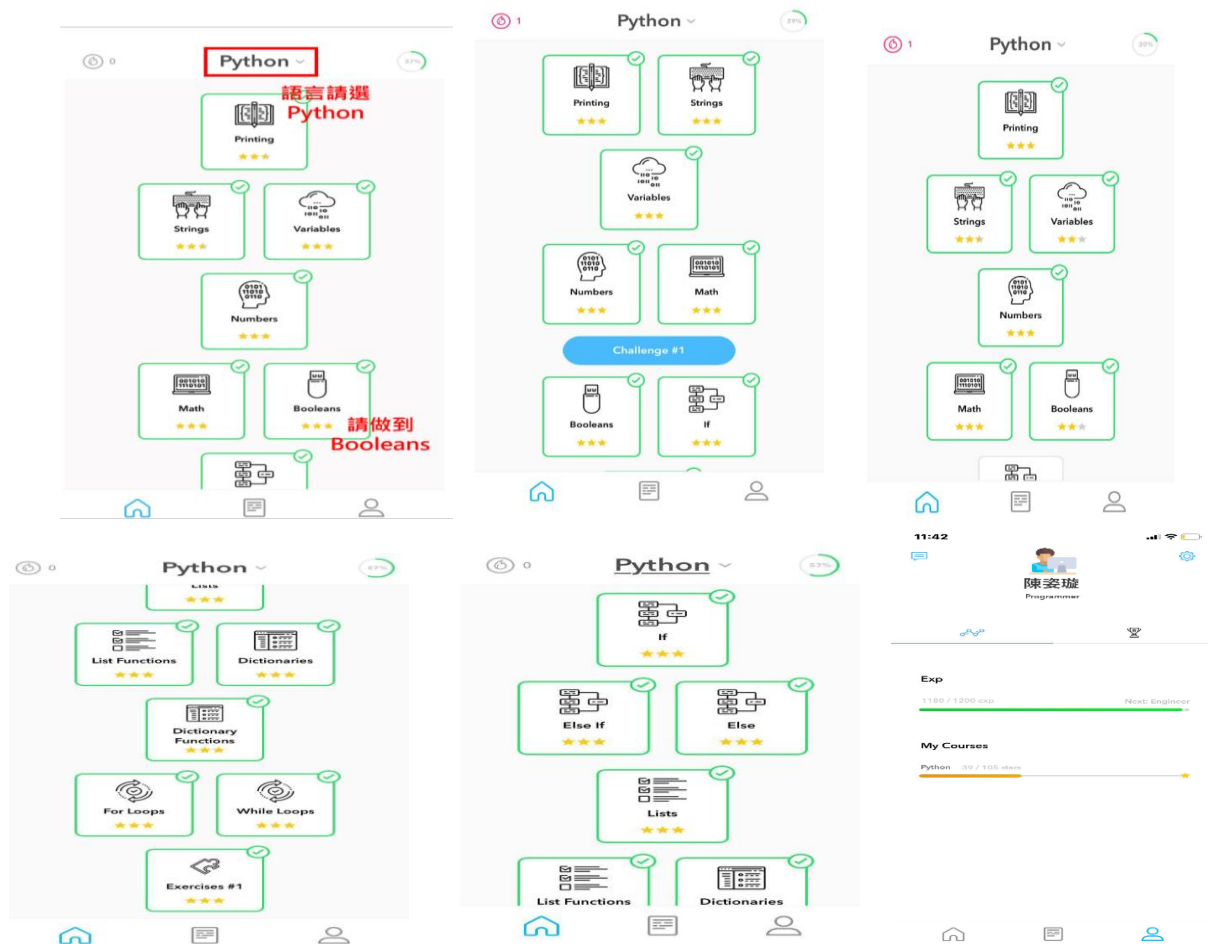
```
var x = 1
print(x)
```

Choose the correct answer

☒ It prints the value of x

☐ The program does nothing

圖 2.3 作業完成頁面



3. 作業三 (個人 → 團體): Colab Practice with NLTK Texts

- (1) 分組作業，上傳上課練習 NLTK 語法的 colab 網址。
- (2) 內容:利用 NLTK 語法 ConditionalFreqDist()、plot()分析特定字詞出現的頻率，並繪製圖表。
- (3) PYTHON 練習網址 Colab : <https://colab.research.google.com/notebooks/welcome.ipynb>

教師評閱

成績比率 100.0%

小組作業：

將自己組內找的txt文件放到google drive中，並引入nlTK裡面。

利用以下nlTK function來練習：

1. count
2. concordance
3. similar

加分題：

- 1.利用 dispersion_plot繪出詞彙分佈圖
- 2.利用 ConditionalFreqDist繪製出曲線圖

作業繳交方式：請繳交Colab連結

可參考附件上課投影片教學及colab範例程式碼

附件名

檔案大小

0528.pdf

7 MB

colab連結.txt

85 Bytes

圖 3.1 練習內容 學生一

Untitled1.ipynb

檔案 編輯 檢視畫面 插入 執行階段 工具 說明 系統不會儲存變更

+ 程式碼 + 文字 複製到雲端硬碟

Displaying 38 of 38 matches:

ay when they were created . And Adam lived an hundred and thirty years , and be
 ughters : And all the days that Adam lived were nine hundred and thirty yea and
 nd thirty yea and he died . And Seth lived an hundred and five years , and bega
 ve years , and begat Enos : And Seth lived after he begat Enos eight hundred an
 twelve years : and he died . And Enos lived ninety years , and begat Cainan : An
 years , and begat Cainan : And Enos lived after he begat Cainan eight hundred
 ive years : and he died . And Cainan lived seventy years and begat Mahalaleel :
 rs and begat Mahalaleel : And Cainan lived after he begat Mahalaleel eight hund
 years : and he died . And Mahalaleel lived sixty and five years , and begat Jar
 s , and begat Jared : And Mahalaleel lived after he begat Jared eight hundred a
 and five yea and he died . And Jared lived an hundred sixty and two years , and
 o years , and he begat Eno And Jared lived after he begat Enoch eight hundred y
 and two yea and he died . And Enoch lived sixty and five years , and begat Met
 : for God took him . And Methuselah lived an hundred eighty and seven years ,
 , and begat Lamech . And Methuselah lived after he begat Lamech seven hundred
 nd nine yea and he died . And Lamech lived an hundred eighty and two years , an
 ch the LORD hath cursed . And Lamech lived after he begat Noah five hundred nin
 naan shall be his servant . And Noah lived after the flood three hundred and fi
 xad two years after the flo And Shem lived after he begat Arphaxad five hundred
 at sons and daughters . And Arphaxad lived five and thirty years , and begat Sa
 ars , and begat Salah : And Arphaxad lived after he begat Salah four hundred an
 begat sons and daughters . And Salah lived thirty years , and begat Eber : And
 y years , and begat Eber : And Salah lived after he begat Eber four hundred and
 begat sons and daughters . And Eber lived four and thirty years , and begat Pe
 y years , and begat Peleg : And Eber lived after he begat Peleg four hundred an
 begat sons and daughters . And Peleg lived thirty years , and begat Reu : And P
 ty years , and begat Reu : And Peleg lived after he begat Reu two hundred and n
 d begat sons and daughters . And Reu lived two and thirty years , and begat Ser
 ty years , and begat Serug : And Reu lived after he begat Serug two hundred and
 begat sons and daughters . And Serug lived thirty years , and begat Nahor : And
 years , and begat Nahor : And Serug lived after he begat Nahor two hundred yea
 begat sons and daughters . And Nahor lived nine and twenty years , and begat Te

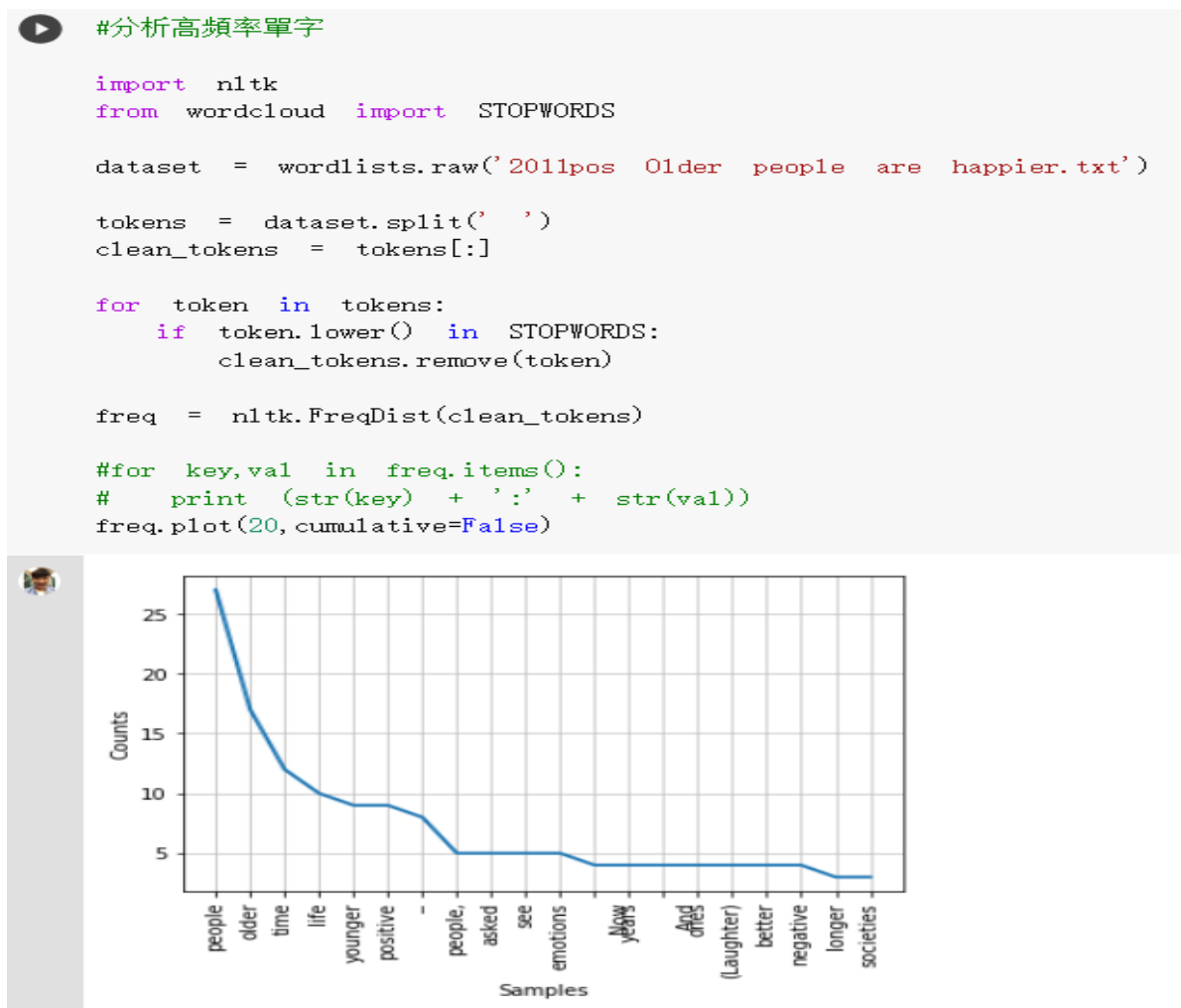


圖 3.2 練習內容學生二



Untitled1.ipynb ☆



檔案 編輯 檢視畫面 插入 執行階段 工具 說明 系統不會儲存變更

+ 程式碼 + 文字  複製到雲端硬碟

```
import nltk
from nltk.corpus import brown

reviews_text = brown.words(categories='reviews')

fdist = nltk.FreqDist(w.lower() for w in reviews_text)

modals = ['can', 'could', 'may', 'might', 'must', 'will', 'might', 'shall']

for m in modals:
    print(m + ': ' + str(fdist[m]))
```

```
can: 45
could: 40
may: 47
might: 26
must: 19
will: 61
might: 26
shall: 1
```

```
[ ] import nltk
from nltk.corpus import gutenberg
gutenberg.fileids()

['austen-emma.txt',
'austen-persuasion.txt',
'austen-sense.txt',
'bible-kjv.txt',
```

```
[ ] cfd = nltk.ConditionalFreqDist((genre, word)
    for genre in brown.categories()
    for word in brown.words(categories=genre))
genres = ['reviews', 'hobbies', 'romance', 'humor']
modals = ['can', 'could', 'may', 'might', 'must', 'will', 'shall', 'should']
cfd.tabulate(conditions=genres, samples=modals)
```

	can	could	may	might	must	will	shall	should
reviews	45	40	45	26	19	58	1	18
hobbies	268	58	131	22	83	264	5	73
romance	74	193	11	51	45	43	3	32
humor	16	30	8	8	9	13	2	7

```
from nltk.tokenize import sent_tokenize

speech=("I'm a person who is very big how about you. Are you are the same type of guy?")

print(sent_tokenize(speech))
```

```
["I'm a person who is very big how about you.", 'Are you are the same type of guy?']
```

```
[ ] import nltk
from nltk.corpus import inaugural
import matplotlib.pyplot as plt
```




-



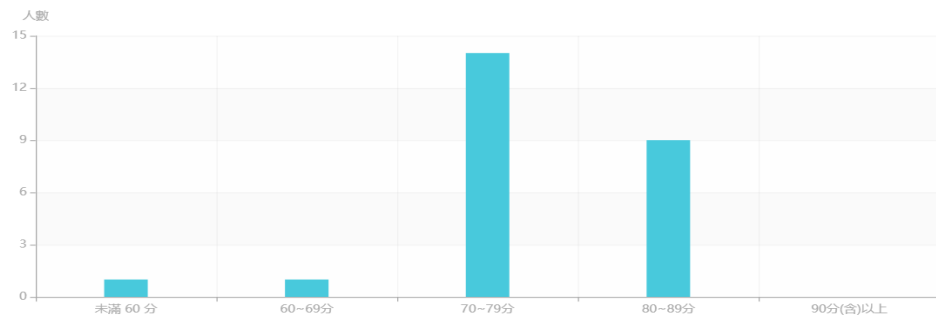
學生成績截圖：

6. 0616 NLTK 文字雲

編輯

作業類型 作業批改 成績統計

作業成績統計



➤ 期末專題報告

The Final Project

檢視 Rubric

作業類型 作業批改 成績統計

成績比率	15.0%	公佈成績時間	不公佈
開放時間	2022.06.16 19:30	繳交截止時間	2022.06.23 18:30
作業形式	分組作業 (全部小組 : Groups)	完成指標	繳交作業

評分方式 (教師評閱 60.0% 、 分組互評 20.0% 、 組內互評 20.0%)

教師評閱

成績比率 60.0%

分組互評

成績比率 20.0%

互評開始時間 2022.06.23 19:00

互評截止時間 2022.06.25 22:00

評閱份數 6 份 / 組 記名

組內互評

成績比率 20.0%

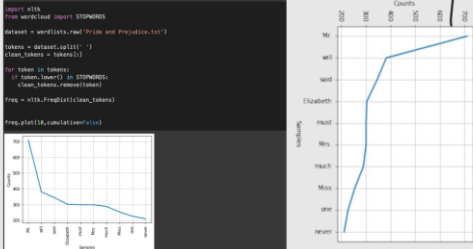
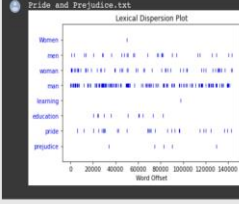
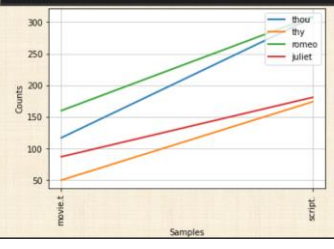
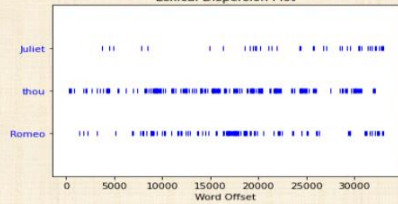
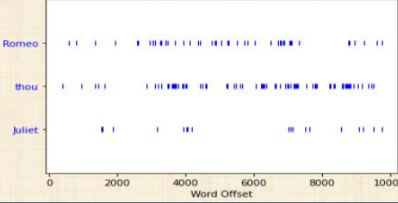
互評開始時間 開放後即可互評

互評截止時間 2022.06.25 22:00

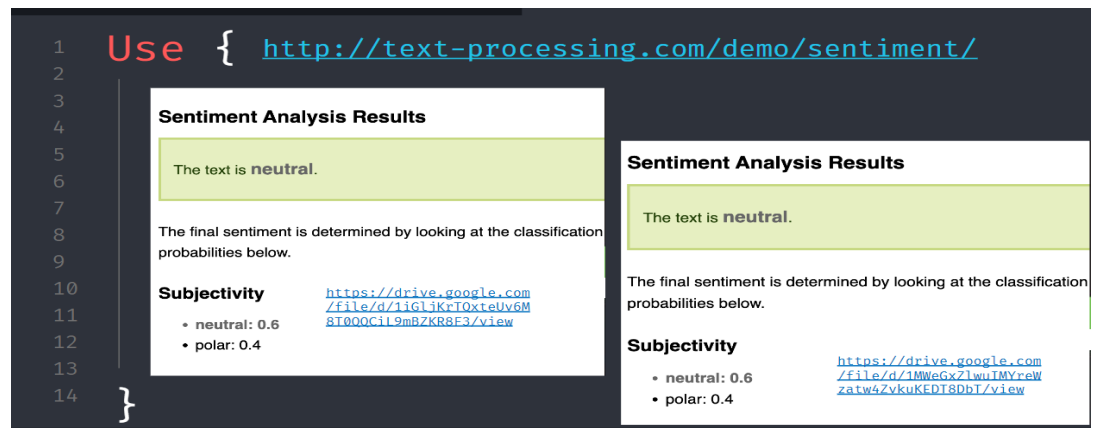
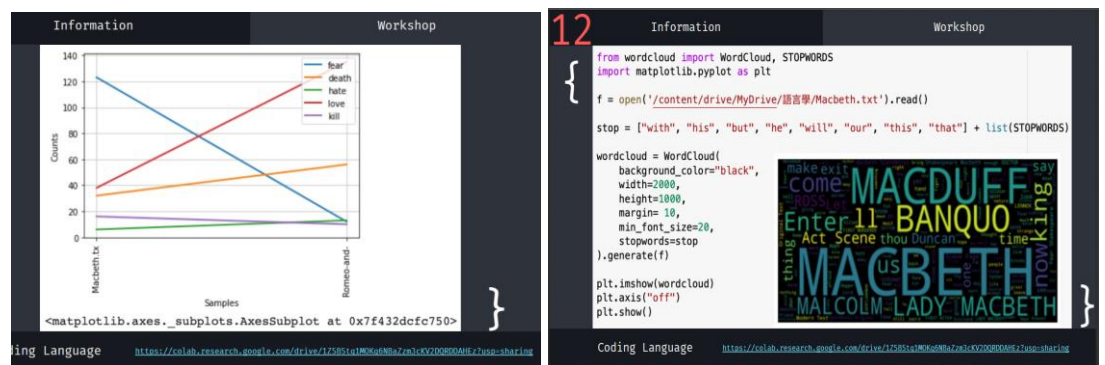
A 10-min. project.

Please share with us your Coding language and report material.

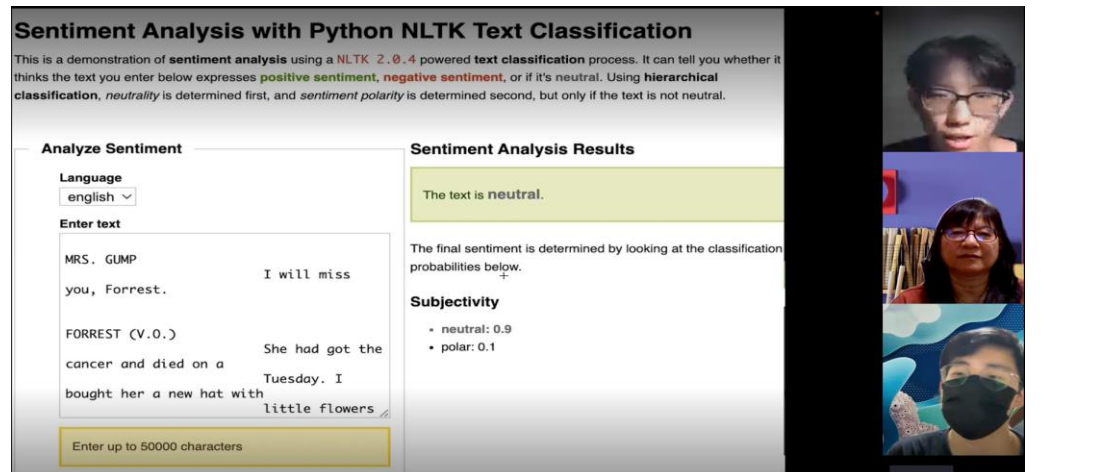
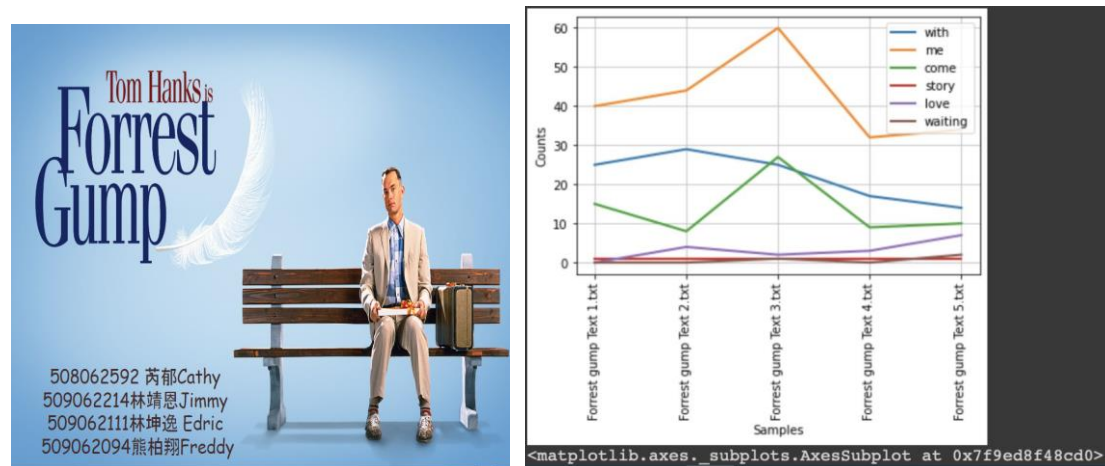
組別	期末專題 報告主題	線上報告&程式設計部分截圖
----	--------------	---------------

1	Pride and Prejudice	Download the Text <pre> from google.colab import drive drive.mount('/content/drive') !ls /content/drive/My\ Drive/Colab\ Notebooks/Pride\ and\ Prejudice !cat /content/drive/My\ Drive/Colab\ Notebooks/Pride\ and\ Prejudice.txt import nltk from nltk.corpus import PlaintextCorpusReader corpus_root = '/content/drive/My\ Drive/Colab\ Notebooks/Pride\ and\ Prejudice' wordlists = PlaintextCorpusReader(corpus_root, ".+") books = wordlists.fileids() print(books) ['Pride and Prejudice.txt'] </pre> Frequency-1 <pre> import nltk from wordlists import STOPWORDS dataset = wordlists.read('Pride and Prejudice.txt') tokens = dataset.split() clean_tokens = tokens() for token in tokens: if token.lower() in STOPWORDS: clean_tokens.remove(token) freq = nltk.FreqDist(clean_tokens) freq.plot(10, cumulative=False) </pre> 	Position <pre> findList = ['Women', 'men', 'woman', 'man', 'learning', 'education', 'pride', 'prejudice'] for fileid in books: myWords = wordlists.words(fileid) myText = nltk.Text(myWords) print(fileid) myText.dispersion_plot(findList) </pre> Pride and Prejudice.txt  What we've learned from this final report? 
2	The script of Romeo and Juliet	TOP10 The most common words <pre> fdist = FreqDist(lemmatizer.tokens) fdist1 = fdist.most_common(10) print(fdist1) </pre> 1. Romeo:119 2. Thou:101 3. Juliet:68 4. Thy:48 5. Shall:40 6. Mercutio:39 7. Benvolio:38 8. Love:34 9. Nurse:34 10. good:33	Line Graph <pre> sfd = nltk.ConditionalFreqDist((target, fileid[1:]) for fileid in wordlists.fileids() for w in wordlists.words(fileid) for target in ['Romeo', 'Juliet', 'thou', 'thy']) sfd.plot() </pre>  The play script Lexical Dispersion Plot  The movie script Lexical Dispersion Plot 

3 Analysis between Romeo and Juliet & Macbeth



4 Tom Hanks is Forrest Gump



5 Comparison among US presidents

How We Worked

most frequently used words

```
import nltk
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from nltk.stem import PorterStemmer
from nltk.probability import FreqDist

# Load the text
text = open('Trump.txt', 'r').read()

# Tokenize the text
tokens = word_tokenize(text)

# Remove stopwords and punctuation
stopwords = set(stopwords.words('english'))
filtered_tokens = [token for token in tokens if token not in stopwords and token.isalpha()]

# Lowercase and lemmatize
lowercase_tokens = [token.lower() for token in filtered_tokens]
ps = PorterStemmer()
lemmatized_tokens = [ps.stem(token) for token in lowercase_tokens]

# Find the most frequent words
freq_dist = FreqDist(lemmatized_tokens)
most_frequent_words = freq_dist.most_common(10)

# Print the results
for word, count in most_frequent_words:
    print(word, count)
```

- Stopwords
- Punctuation
- Lowercase
- Lemmatizer
- The most frequently used words

Lexical Diversity

	Inauguration Speech	Acceptance Speech	Debate
Trump	0.35311750599520386	0.1598806224685568	0.13173652185655696
Obama	0.4112961622013034	0.23083975064173085	0.1499662238234632
Biden	0.1924144310823317	0.23083975064173085	0.1640241815900536

Data about Trump

Bar Chart

Trump's Acceptance Speech

Key Words

Trump's Inauguration Speech

Key Words

Trump's Debate

Key Words

6 British vs US: versions of Harry Potters

Texts

Harry Potter and the Philosopher's Stone
(British Edition)

Harry Potter and the Sorcerer's Stone
(American Edition)

```
import nltk
testText = wordlists.words('text2.txt')
single = nltk.text.Text(testText)

findList = ["Hogwarts", "wizard", "magic"]
single.dispersion_plot(findList)
```

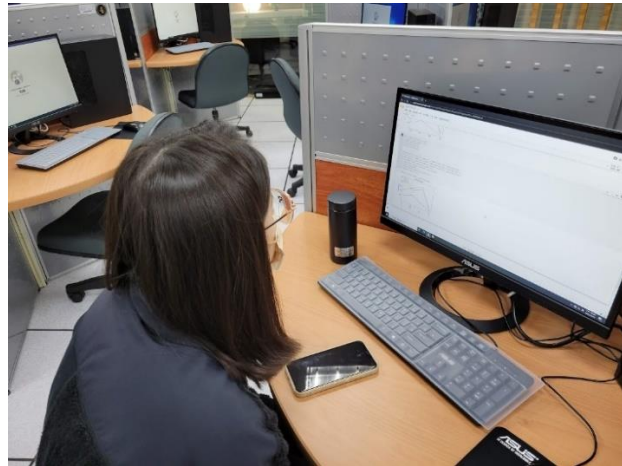
Lexical Dispersion Plot

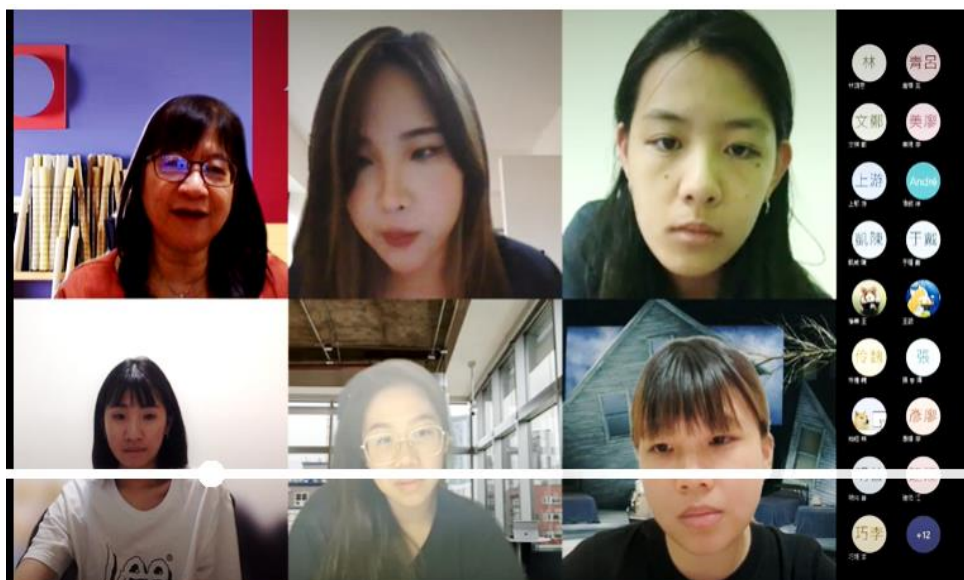
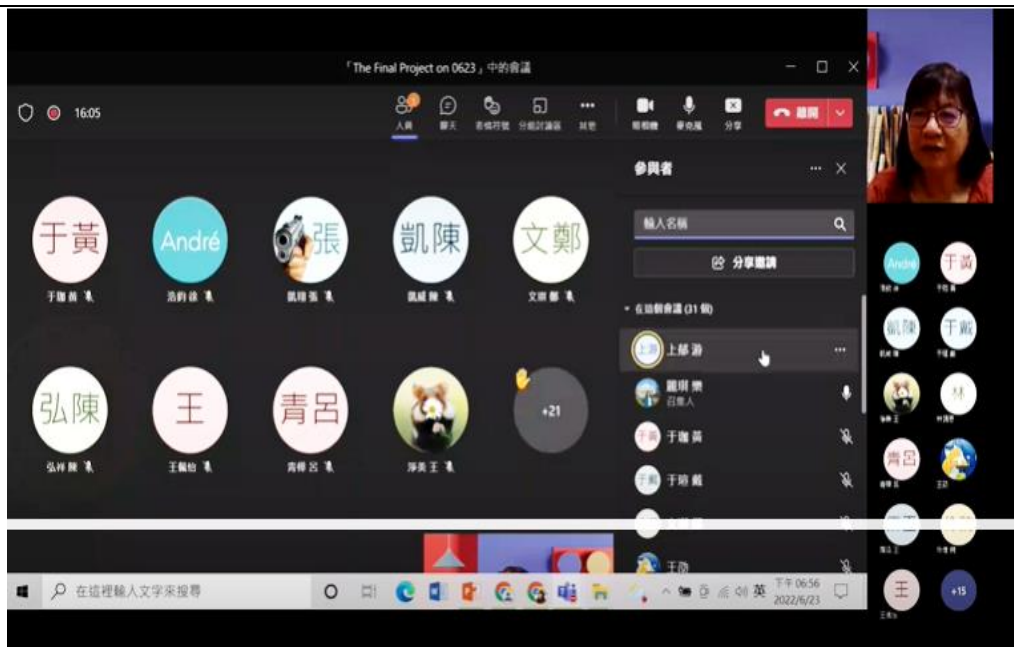
課堂活動剪影 (至少 2 張)

➤ Lecutres 課程剪影



➤ Practices 程式語言練習





➤ 教學團隊



➤ 緣起：

本課程嘗試於教學活動中融入「運算思維」與「程式設計」概念，引導學生認識運算思維 (CT, Computation Thinking) 與邏輯思考能力，學生藉由善用語料庫資源，能快速有效分析語料，掌握學科學習重點。在學習活動中，學生能發現善用 NLTK 語料庫資源的好處，明瞭邏輯思考與程式運算能力運用在處理語料分析過程中的快速有效性運用程式設計運算於自然語言語料分析，並企圖導引學生探索語言學學科專業的資訊跨域整合應用，使學生透過運算程式分析、理解和處理自然語言，掌握學科學習重點，進行語言學專題研究；教師也因此展開了語言學專業學科的資訊跨域實踐，也藉由教師與學生的資訊素養強化，提升教師教學成效及學生學習成果。

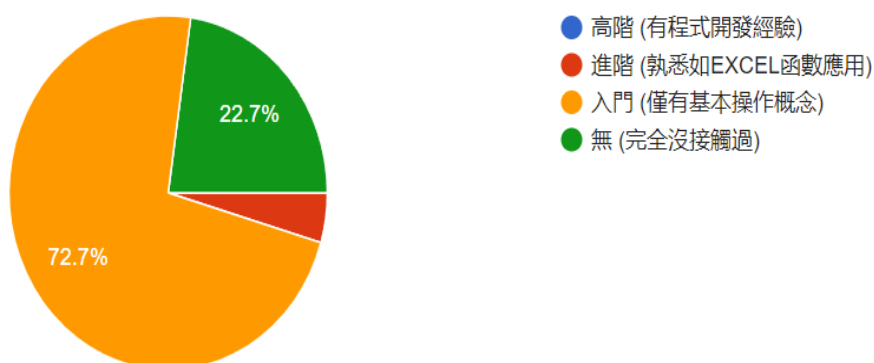
因為本系課程整體結構調整，「英語語言學概論」專業必修科目從往年的 3 學分調整為 2 學分，此為第二次嘗試在縮短學時數的狀況下，藉此計畫檢視專業學科的數位執行效能外，亦期能規畫調整適宜的課程教學內容，進行創新教學探索與課程省思；今年執行程式設計融入課程計畫期間，又因為遇到 COVID19 新冠疫情日趨嚴峻，校方執行安心就學方案，宣布停止面授教學改以線上數位教學方式執行課程兩週，之後雖恢復實體上課，但仍有部分同學要求線上上課，只得以“實體+同步”遠距方式進行，但 python 程式操作及學習成效馬上顯現學習差距，此現象對於學習跨領域新知識與程式語言需實作練習的實驗性教學設計更增加挑戰性。

➤ 歷程觀察與回饋反思

參與本計畫者為英文系進修學士班 110 學度二年級修讀必修科目「英語語言學概 38 名學生。在課程初學前問卷中顯示 (22 則回應)，本班學生有 81.8% 曾修讀過資訊/多媒體相關課程，但是認為自己完全無資訊能力程度的占 22.7%，多數認為自己為入門級的占有 **72.7%**：

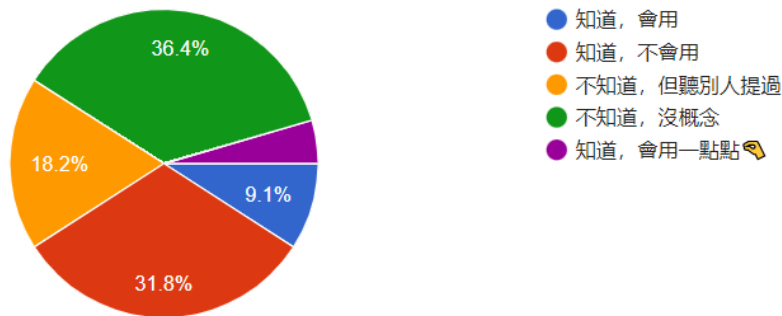
2. 請問你目前的資訊能力程度為？

22 則回應



4. 請問你知道 PYTHON程式語言嗎?

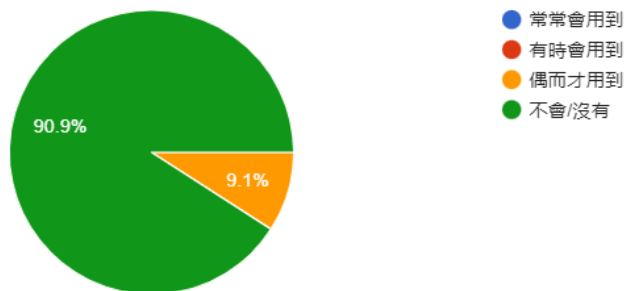
22 則回應



知道如何運用 Python 程式語言的學生僅有 **13.6%** (「知道，會用一點點」4.5%，「知道，會用」9.1%)，將近 1/3 的學生「知道，不會用」(31.8%)，然而卻超過 1/2 的學生 (「不知道，但聽別人說過」(18.2%)、「不知道，沒概念」(36.4%)。雖然高達 **90.9%** 的學生表示「在目前的工作中**不會/沒有使用**程式應用」；但認為在 Q8.2「未來的工作中**會使用**到程式應用」的達 **59.1%** (一定會 4.5% + 應該會 18.2% + 偶而會 36.4%)。而在期末學習回饋中「未來會在工作或生活中運用 Python 程式語言」數據更升高至 **71.4%** (一定會 9.5% + 應該會 23.8% + 可能會用到 38.1%)；選擇「不確定」(9.5%) 及「不會」(19%) 數據，或許顯示出學生明白現階段於工作場域，雖無程式運算應用之迫切需求，但多認清在未來職場中，擁有資訊科技能力的必要性，因此產生**需加強自我數位實力**的學習動機。

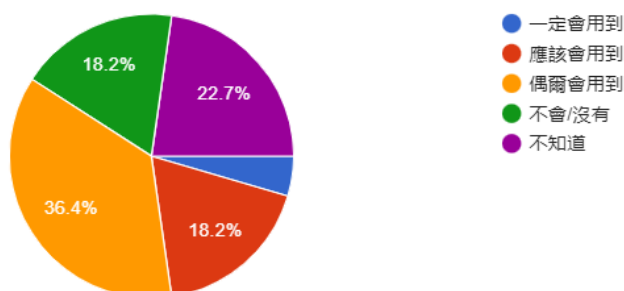
8.1 您目前的工作中是否使用到程式應用?

22 則回應



8.2 您認為在未來的工作中是否會使用到程式應用?

22 則回應

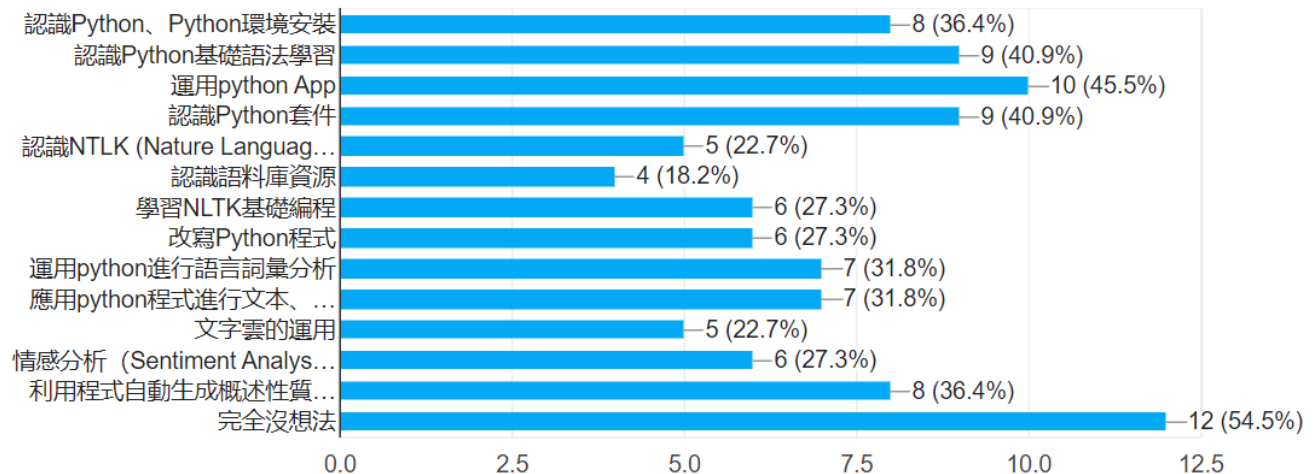


對於語文學科學生而言，PYTHON 程式語言有相當程度的理解應用困難。本計畫執行期間，只有兩次面授課程，接續因為改以線上課程方式進行，教師及助理只能以個別線上教學方式解決學生問題。雖有部分學生表達理解或撰寫程式語法的困難性（特別如 Conditionals, Loops, if-else-if statement），但多數學生皆表現出願意嘗試資訊新知的肯定態度，在期初學前問卷與期末學習回饋調查也能看見改變中的學習動機：在學期初的學前問卷中對於學習單元的認識，超過半數學生完全沒有想法（54.5%）。

5. 請問你認為哪些學習單元會是實用的？（複選）



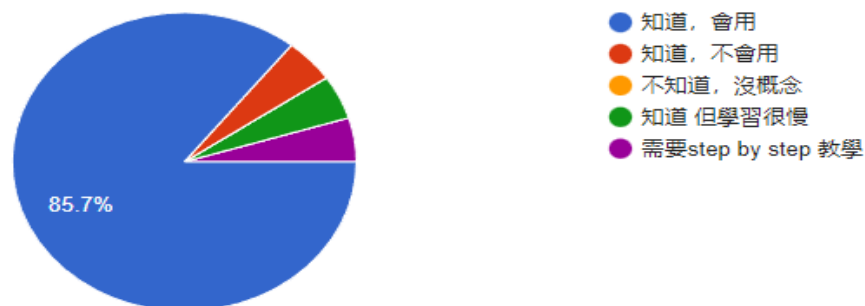
22 則回應



在學習回饋問卷回應數據中，發現學生對於學習 PYTHON 程式套件、各語法功能與語料庫工具多顯示樂於接受運用新科技的正向成效（Q2.會使用 Colab 85.7 %、Q3 會使用語料庫 Corpora 71.4%）：

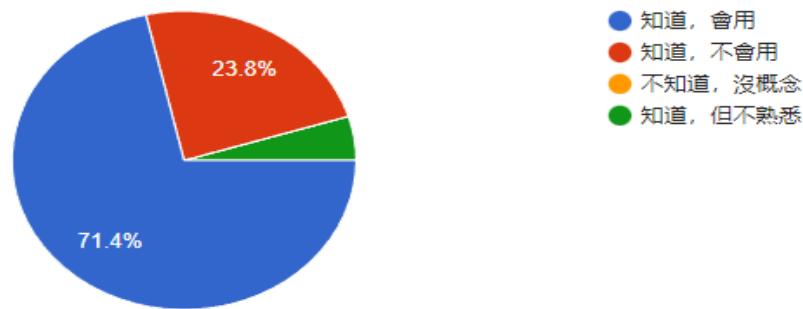
2. 請問你知道如何使用 Colab 嗎？

21 則回應



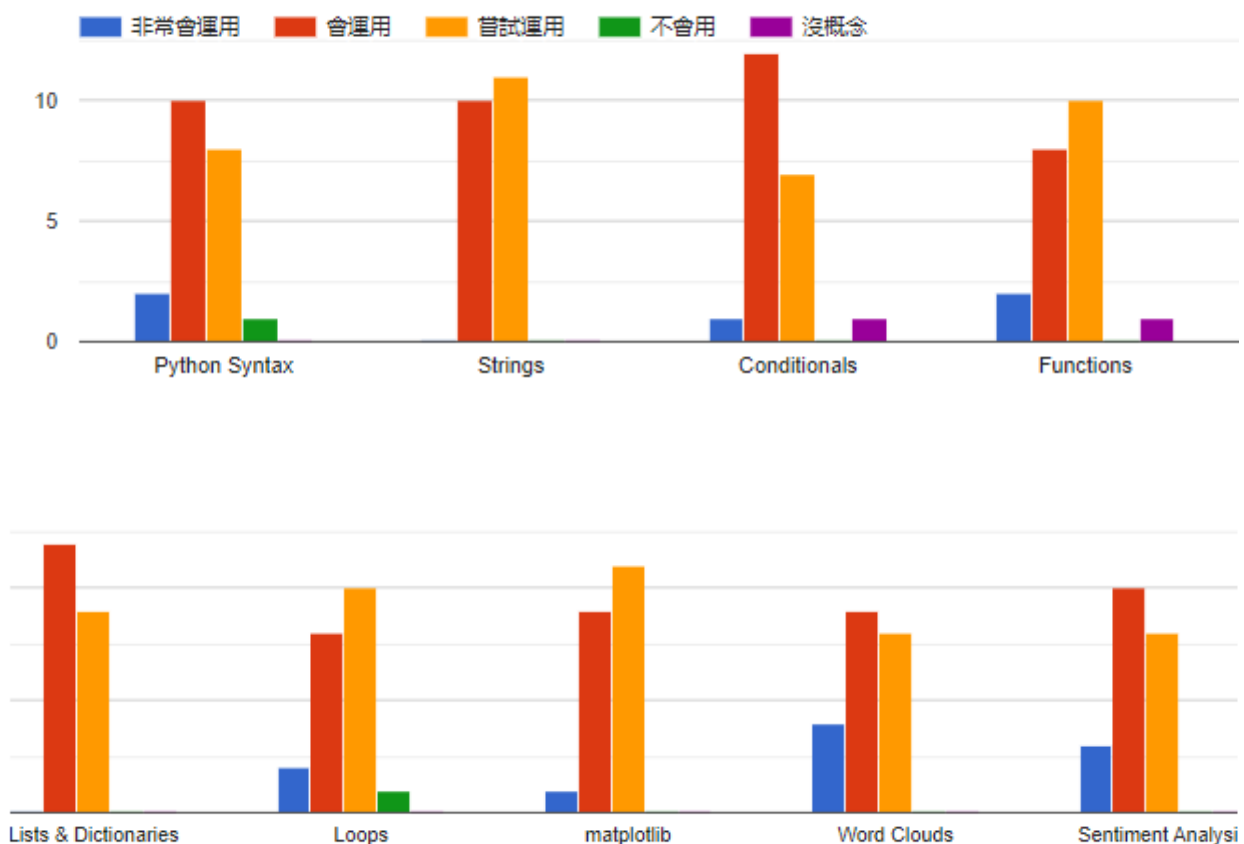
3. 請問你知道如何運用 Corpora 語料庫嗎?

21 則回應



Q5. 請問你知道如何執行以下程式語言功能嗎? 程式語法、功能運用各項回應顯示，不論是正向「會運用」或「嘗試運用」的數值皆遠高於負向「不會用」(除了進階程式語言 Conditionals 及 Function 運用，其他各選項「沒概念」皆為 0%)。「會運用」或「嘗試運用」選項數據突出，表現出學生願意面對數位科技的積極學習態度，而「非常會運用」選項於文字雲、情感語意分析功能之數據相當突出，亦可自各組期末專題報告中皆涉獵運用，觀察出學生對於以資料視覺化方法呈現語料文本分析，具有更高的接受度與學習成就感。下圖即為調查 9 項程式語言功能學習狀況：

5. 請問你知道如何執行以下程式語言功能嗎?



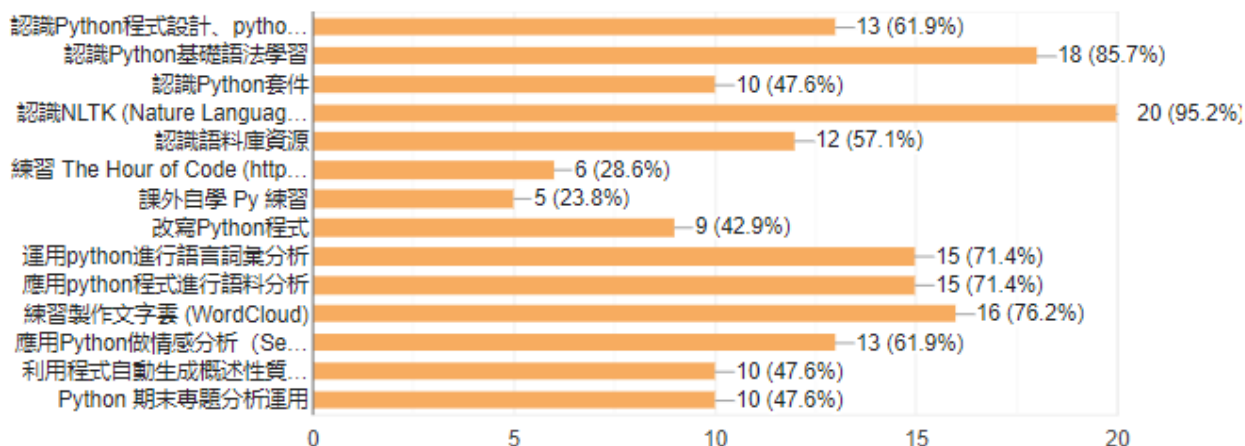
本計畫課程內容設計循序自線上遊戲 Code Studio (Hour of Code)、手機 PY app 及 Google Colab 入門，並緊扣 NLTK 語料分析實例，引導學生運用語料視覺化分析工具，成功提高其學習興趣，並願意嘗試練習運用語料分析。比較學前問卷 (Q5 請問你認為哪些學習單元是實用的?) 與學習回饋問卷 (Q7 請問你認為哪些學習單元是實用的?)，學生對於實用的學習單元列舉出之

個別選項，大多呈現正成長數據。同時，學生對於培養團隊合作、互助學習、自主發想研究計劃能力，也是正向回饋。

7. 請問你認為哪些學習單元是實用的？（複選）

複製

21 則回應

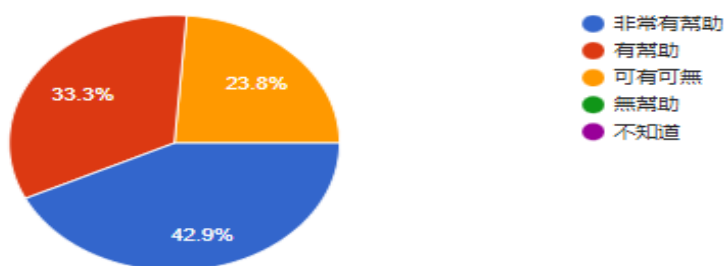


在學習活動中，學生們利用機會驗證所學的程式運算於語料分析，各組學生於工作坊積極討論期末專題研究主題，顯現主動學習精神與認真態度，學習成就比程式教育業師及助教們預期的更好。學生在學習活動中展現的積極興趣與團隊互助，使得期末專題報告成功產出，教師也利用機會提供學生觀摩過往研究主題及練習題材，助益其發想如何結合運用程式語言分析語料；在期末的語言學專題分析量化報告中，也可看到各組運用跨域學習展現學科多樣性，實質幫助學生擴展研究潛能。

在期末回饋中統計數值中，學生表達對於學習 Python 程式，幫助學科專業語料分析能力與語言學習肯定值達 76.2% 及 66.7%:

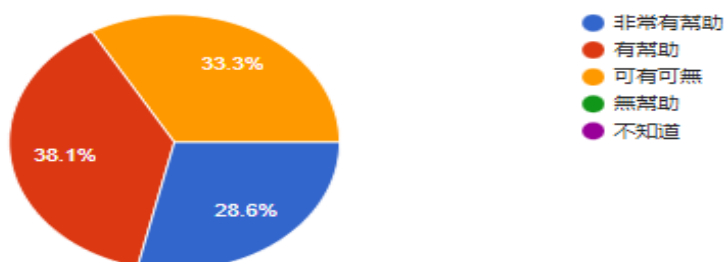
12. 我覺得學習Python Coding，對於語料分析：

21 則回應



13. 我覺得學習Python Coding，對於語言學習：

21 則回應



以下節錄部分學生學習回饋：

<對於 Linguistics & Python 單元的學習想法與意見>

- 很有趣，將不同領域的東西融合在同一堂課
- Introducing more practical usage and functions of Python.
- 我認為這是一門相當實用的課，資訊發達的社會，許多事都跟大數據及語料庫有關，學習程式對我們英文系的學生來說是增加競爭力。
- 對初次學習的人來說有點難
- 很棒的課程 我學到了入和簡單化的去做解析分析語言 並且深度了解他們
- 我覺得不錯，不過唯一美中不足的地方就是因為疫情的關係，無法實體到校上課，增加了老師以及助教教學上的難度 X D
- Python is the field that I first came into contact with. At first, I found it very challenging. Then I became more proficient with it, and I gradually felt that the things I made were very fulfilling, so I think is a very interesting course.
- 很有用
- 有困難
- 我覺得很棒，可以在自己的電腦上實作更能加深印象及有利於未來的應用。
- I think it's good to learn Python. It's very fun and useful.
- 學到不同領域的事物，覺得很多東西都能嘗試且從中獲得很多樂趣。
- The contents can be a little easier.
- 雖然是很陌生的語言，但是學習後很有收穫。
- 非常棒，接觸到了自己平常不會接觸到的東西，很有挑戰性
- 非常滿意，很感謝老師安排了這一項課程，讓我學習 Python，我的收穫很多！
- 這學期比較可惜是遠距的方式學習程式，但還是很感謝 Adam 學長的細心教導
- 帶我們體會到了一個未知的領域，充分地感受到老師認真的教學。
- 如果能在教室內上課就好了=(((可以直接問問題，這大概是唯一缺點=（（（
- 我覺得學習 python 很有趣，對我來說是一個全新的領域，未來也會想繼續使用。
- 很喜歡這堂課所教導的 python 讓我可以去分析語言 謝謝老師
- 其實 Python 真的很難=（（
- 雖然有點難，但還是要感謝老師和助教不厭其煩和細心的教導我們，謝謝。

本計畫執行過程中，教學者欣見多數學生表達受惠於此跨域學習機會，不但對於語料分析與語言學習都有所助益，也讓他們得以開拓學習視野，看到了專業學科與資訊跨域結合的可能性，增強其未來就業競爭能力，甚而因此引發部分學生主動思考語言學科的數位發展性。凡此種種回饋，都是課程創新的附加效益。

培養語文科系學生以程式語言運算分析自然語言的能力，絕對是越界的跨領域學習。雖然短短1周的密集課程，學習時間是絕對不夠的，但經過此程式運算基礎入門起步，引導學生從被動認識、理解、練習程式語言於語料分析，到積極運用程式設計的學習過程，啟發對於「運用程式語言於語言學專業學科」的跨域學習興趣，正向提升自主學習動機，學生能即學即用，成功降低對程式設計的恐懼。綜觀整個學習歷程，學生在知識面、學科專業、程式設計技能、態度面等表現皆能符合課程規劃預期目標；本班學生未來如有機會深入研習其他程式語言，相信也會更具成效。因著參與本計畫，學生瞭解到運算思維的基本原理，除具體提升資訊素養，產出語言學專題

研究語料視覺化報告，也發現了跨域學習的無限潛能，增強「學用合一」數位競爭力！

➤ <助教教學與觀察心得：張凱翔 資訊工程學系 4 年級>

這堂課是英文系主修課，可以看到學生們的程式程度算是基礎範圍，他們都幾乎是從 0 開始學起，但是也有少部分學生已經有些程式基礎。我在課堂作業上，看到有幾位同學很用心地在程式旁邊做些小小的註解，學習程式是怎樣用之類的。

對於在教非資工系相關科系的學生寫程式，是要有點耐心的，也不能用太多專業名詞跟他們講解程式，因為他們只是需要學到基礎運用程度，而不是學理論課程，最多只要講解如何使用之類的就可以了。

我觀察到學生對於程式運用，上手的速度很快。只可惜後面幾周都是線上教學的方式，這對於要實際操作的程式語言部分，其實是有點不友善的，因為同學如遇到問題，無法直接提問，而是在遭遇問題的時候，需要在線上傳遞截圖，有些人速度也不是很快。還好，到最後還是能成功解決同學的問題，只是比較麻煩些。

有一周需要教他們複習基礎的 Python，那次算是我第一次教人怎樣去寫程式，我不知道該如何教一群人，所以我就想到之前系上老師教程式的教法，然後試著把教法和講解方式再簡化。像是在課堂前我準備了 Colab 檔，當成我上課要教的方向來講，因為要寫的程式就是每次上課前要幫同學複習的程式，所以我就直接套用筆記方式。需要注意的是，在講解的過程中不能塞太多的東西給他們，因為他們不是本科系的學生，而且這只是基礎課程，所以也不能用太多的專用名詞，而且需要講得比較白話一些。

Python 其實對沒學過或接觸過程式語言的人來講，算是比較簡單的語言，不像 C 語言那樣需要很多基礎才能使用，Python 就比較簡單白話些，只要學會一點基礎就能直接用了。而且 Python 裡的 Module 也很多，像是 matlib, nltk 這些工具，功能都能夠直接使用。所以，如果不是資工本科系學生，但是想要學程式語言的話，我當然推薦 Python！

➤ 教學改善與精進

對於語文教師本身而言，能瞭解人文學科與程式語言的關聯性，並運用在學科專業中，是跨域教學的新課題。身為非資訊專業的教師，在實踐資訊整合應用過程中，如何在原本的課程時間內，妥適將「運算思維認識」與「程式語言設計」學習目標，融入專業學科的課程設計與教學活動中，既要讓學生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，都深具課程實驗性。感謝深耕計劃的支持，才能有機會邀請資訊教育業師來校進行專題講演及專案指導，教師並得以逐年持續精進課程規畫與教學設計，執行此跨域整合創新計劃，期能實質提升教學效能，助益更多學生。

因為疫情關係，本次程式語言操作課程幾乎全部得以線上教學模式進行，對於有數位落差的語文科系學生具有相當程度的學習困擾。但因為業師及教學助理的耐心與熱誠帶領，又因為教師本身有多年的數位課程經驗，也運用 Microsoft TEAMS 錄製各週教學影片，提供學生課後複習所用，使得本教學計畫得以順利執行。經過逐年調整修正，【程式設計融入課程】計畫，實質改變程式語言教學結構，更貼近初學者學習角度；為減低學生對程式語言的陌生恐懼，本計畫捨棄傳統以認識程式語言理論（如 Anaconda & Jupyter 介紹安裝、Codecademy 及 Coursera 程式語言教學），課程內容改以線上遊戲 Code Studio (Hour of Code) 體驗起始、手機 PY app 練習入門，引導學生利用圖像方式學習簡單的程式邏輯；續以 Google Colab 程式關鍵字改寫，並緊扣 NLTK 語料分析實例之密集練習，增強語料視覺化分析工具運用，成功提高學生學習興趣與學習成就感，而能同儕互助，積極參與語料分析練習運用。

在整個教學實踐過程中，無論是對學生、對授課教師、對業師、對教學助理，各方都經驗了跨域謀合、互相溝通，發現到不同學科的不同思維模式，也因此而能彼此學習理解對方，建立難得的跨域共識。教師除藉此機會檢視課程設計執行成效，探索語言學科資訊跨域整合應用性，也

得以在「教與學」中，發展出學用合一的「**LPWDP 任務導向教學模式**」，實踐專業學科的數位行動力，而能以多元角度精進教材設計與教學活動，展現與時俱進的教學創新能力！