

113年度高等教育深耕計畫

主軸計畫成果報告書

總計畫「跨校/跨領域教師增能社群」

執行策略：總計畫「跨校/跨領域教師增能社群」

社群名稱：「庚你LINE在一起」：跨領域發展聊天機器人互動式教材

指導單位：教育部技職司

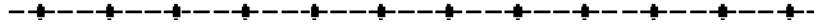
主辦單位：長庚科技大學

承辦單位：長庚科技大學護理系

活動日期：中華民國 113年 4 月 5 日~113年 12 月 5 日

目

錄



壹、活動檢核.....	P.02
貳、活動經費.....	P.04
參、活動時程表.....	P.05
肆、活動議程表.....	P.06
伍、活動簽到表.....	P.08
陸、活動海報.....	P.10
柒、活動照片.....	P.11
捌、專題講座內容.....	P.12
玖、回饋單統計.....	P.46
拾、成果產出.....	P.48

壹、活動檢核

跨校跨領域教師增能社群成果檢核表			
社群名稱	「庚你 LINE 在一起」：跨領域發展聊天機器人互動式教材		
社群執行起迄日期	113 年 4 月 5 日至 113 年 12 月 5 日	社群人數	5 人 (本社群之成員)
核定經費	59960 元	實際執行經費	59960 元
社群活動執行內容重點概述	<p>內外科護理學在護理系學生的專業養成教育過程，實屬重要的基礎實務能力學科，然而內外科護理學是一門論及成人全身系統疾病照護學科，課程內不僅需學習成人疾病照護重點，還需連結與整合過去已習得的專業知識，對學生而言確實具有一定程度的學習壓力。學生常在教學回饋提及「課程內容難以吸收」，實習老師們也常提到學生常見的學習問題是「難以將專業知識整合應用於臨床照護，尤其是病史收集能力尚待加強」。</p> <p>聊天機器人可進行一對一雙向互動與即時回饋，而且使用上不會受到時間與空間限制，可隨時隨地回答學生的問題，故聊天機器人具備支援醫護教育實現個別化學習的潛力。</p> <p>有鑑於此，我們預計發展以聊天機器人為基礎的互動式教案，協助學生學習內外科護理學，尤其是病史收集能力的培養，聊天機器人可與之互動對話協助其思考判斷問題，而教師也可藉由觀察對話，瞭解學生不足或缺乏之能力調整或補充教學內容，協助學生達到有效且精準的學習。</p>		
是否依據核定之質量化指標執行？ <input checked="" type="checkbox"/> 是 <input type="checkbox"/> 否，原因_____			
質化指標	1. 提升教師應用數位科技於護理教育之信心和能力。	執行成效	1. 社群教師表示，經過兩天工作坊學習，提升自我數位科技應用於教學的知能。
量化指標	1. 辦理一場數位科技實作工作坊，主題聚焦於「發展聊天機器人互動式教材」。	執行成效	1. 完成一場發展聊天機器人互動式教材之工作坊。 2. 完成一例以聊天機器人為主的互動式教材。

	2. 發展一例聊天 機器人互動式 教材。		
是否與其他主軸活動進行成效倍增？ <input type="checkbox"/> 是，主軸__執行策略____ <input checked="" type="checkbox"/> 否			
辦理此活動較著重之面向？ <input checked="" type="checkbox"/> 教學面 <input type="checkbox"/> 公共面 <input type="checkbox"/> 社會面 <input type="checkbox"/> 特色面			
檢討 與 建議	感謝教育部與學校持續對教師增能社群的支持，讓老師們有機會透過社群對話交流，一起學習新的教學方法，共同解決教學困境。本次社群透過微型共作坊，實習學習聊天機器人平台運作技術，與平台介面設計方法，讓我們對將此數位科技融入教學場域更有信心，建議未來若要執行此類研習，可以微型工作坊進行，讓教師們得以更深入學習。		

承辦人(簽章)：陳彥霖

經辦主軸主管(簽章)：余昭玲

教學發展與資源中心查核日期：113年12月5日

單位簽章：

查核人簽章：

簡乃芬



貳、活動經費預算與實際支出明細表

單位：59960 新台幣元

編序	預算項目	預算支出			實際支出	差異說明
		單價	數量	總額		
1	膳食費	120/個	22	2640 元	2640 元	
2	印刷費	/份		元	元	
3	講座鐘點費	1,000/小時		元	元	
		2,000/小時	28	56000 元	56000 元	
4	交通費	/趟		元	元	
5	材料費	/份		元	元	
6	教學業務費	/份		元	元	
7	雜支	1/份	1	1320 元	1320 元	
總計					59960 元	

承辦人：陳彥 (親簽) 經辦單位主管：簡乃芬 主軸計畫主持人：余昭玲

說明：

1. 預算項目請就原修正申請書之支出預算明細資料填寫。
2. 實際支出欄位，請就實際執行的支出金額填寫，並說明差異原因。
3. 如有學校配合款支付之項目，務必填寫清楚。
4. 申請人應於計畫結束後一個月內繳交成果報告書(電子檔及紙本各一份)、滿意度問卷調查表及活動歷程檔案(含紀錄紙本、照片、影音檔、活動滿意度調查結果分析等)。
5. 為展現社群成果、擴大教師交流，成果報告書將進行紙本實體或線上展示。
6. 社群成員須參加教學發展與資源中心期末辦理之成果發表會。
7. 如本表不敷使用，請自行增列。

參、活動時程

月份	研討內容
7月	<ol style="list-style-type: none">1. 規劃工作坊主題與活動議程級日期。2. 聘請工作坊講師，並且擬定工作坊議程。
8月	<ol style="list-style-type: none">1. 規劃工作坊報名。2. 規劃工作坊場地與訂餐。3. 辦理兩天微型工作坊。4. 完成工作坊滿意度調查。
9月	<ol style="list-style-type: none">1. 完成機費核銷。2. 整理結案報告資料。
10月	撰寫結案報告。

肆、活動議程

長庚學校財團法人長庚科技大學 113 度高等教育深耕計畫

跨校跨領域教師增能社群

「庚你LINE在一起」:跨領域發展聊天機器人互動式教材

微型工作坊

- 一、目的：應用 LineBot 發展互動式教材
- 二、時間：113 年 8 月 19 日(星期一) 上午 08：00 至下午 17：00
- 三、地點：嘉義分部 護理系會議室(一)
- 四、主辦單位：護理系
- 五、議程：

時間	主題	主持人/講者
07:40-08:00	報到	陳亭余
08:00-08:10	講者介紹	陳亭余/魏春旺、黃珮晴
08:10-10:10	聊天機器人的基礎理論與應用現況	陳亭余/魏春旺
10:10-10:30	休息與交流時間	陳亭余/魏春旺、黃珮晴
10:30-12:00	深度學習與大型語言模型介紹	陳亭余/魏春旺
12:00-13:00	午餐時間	陳亭余
13:00-15:00	Python 基礎教學	陳亭余/魏春旺、黃珮晴
15:00-15:15	休息與交流時間	陳亭余/魏春旺、黃珮晴
15:15-17:00	Python 實際操作	陳亭余/魏春旺、黃珮晴

活動聯絡人：護理系 陳亭余 (分機 2611/E-mail:tyuchen@mail.cgust.edu.tw)

長庚學校財團法人長庚科技大學 113 度高等教育深耕計畫

跨校跨領域教師增能社群

「庚你LINE在一起」:跨領域發展聊天機器人互動式教材

微型工作坊

- 一、目的：應用 LineBot 發展互動式教材
- 二、時間：113 年 8 月 29 日(星期四) 上午 08：00 至下午 17：00
- 三、地點：嘉義分部 護理系會議室(一)
- 四、主辦單位：護理系
- 五、議程：

時間	主題	主持人/講者
07:40-08:00	報到	陳亭余
08:00-08:30	課後作業解答	陳亭余/魏春旺、黃珮晴
08:30-10:00	開發專屬聊天機器人實作： 使用 Python 初步構建	陳亭余/魏春旺、黃珮晴
10:10-10:30	休息與交流時間	陳亭余/魏春旺、黃珮晴
10:30-12:00	開發專屬聊天機器人實作： 使用 Python 專屬構建(1)	陳亭余/魏春旺、黃珮晴
12:00-13:00	午餐時間	陳亭余
13:00-15:00	開發專屬聊天機器人實作： 小組討論與教案設計	陳亭余/魏春旺、黃珮晴
15:00-15:15	休息與交流時間	陳亭余/魏春旺、黃珮晴
15:15-17:00	開發專屬聊天機器人實作： 使用 Python 專屬構建(2)	陳亭余/魏春旺、黃珮晴

活動聯絡人：護理系 陳亭余 (分機 2611/E-mail:tyuchen@mail.cgust.edu.tw)

伍、活動簽到表

長庚學校財團法人長庚科技大學

113 年度高等教育深耕計畫 跨校跨領域教師增能社群

「庚你LINE在一起」:跨領域發展聊天機器人互動式教材」 微型工作坊

時間：113 年 8 月 19 日(星期一)

二、地點：長庚科技大學嘉義分部 護理學系會議室(一)

三、簽到表：

姓名	簽到	簽退
郭雅雯	郭雅雯	郭雅雯
蘇清菁	蘇清菁	蘇清菁
劉美君	劉美君	劉美君
李淑芳	李淑芳	李淑芳
陳睿婕	陳睿婕	陳睿婕
陳亭余	陳亭余	陳亭余
沈于庭	沈于庭	沈于庭
陳芯宥	陳芯宥	陳芯宥
陳昱璇	陳昱璇	陳昱璇
蔡佳娟	蔡佳娟	蔡佳娟
劉乃文	劉乃文	劉乃文

長庚學校財團法人長庚科技大學

113 年度高等教育深耕計畫 跨校跨領域教師增能社群

「庚你LINE在一起」:跨領域發展聊天機器人互動式教材」 微型工作坊

時間：113 年 8 月 29 日(星期 四)

二、地點：長庚科技大學嘉義分部 護理學系會議室(一)

三、簽到表：

姓名	簽到	簽退
郭雅雯	郭雅雯	郭雅雯
蘇清菁	蘇清菁	蘇清菁
劉美君	劉美君	劉美君
李淑芳	李淑芳	李淑芳
陳睿婕	陳睿婕	陳睿婕
陳亭余	陳亭余	陳亭余
沈于庭	沈于庭	沈于庭
陳芯宥	陳芯宥	陳芯宥
陳昱璇	陳昱璇	陳昱璇
蔡佳娟	蔡佳娟	蔡佳娟
劉乃文	劉乃文	劉乃文

陸、活動海報(若無，空白即可)

柒、活動照片



說明：LINE BOT 講師介紹-魏春旺老師



說明：Python 簡易程式講解



說明：講師課程解說



說明：Colab 程式運行及編寫



說明：學員問題解答



說明：LINE BOT 應用講解

捌、講座/授課內容(含諮詢內容等)

機器人在智慧醫療與長照的應用

魏春旺
cwwei@knu.edu.tw

高雄醫學大學
醫務管理暨醫療資訊學系
高齡長期照護碩士學位學程
醫學研究所

2024.08.19



電動車



資料來源：<https://www.gq.com/hk/gadgets/article.html%7E7%85%B9%E9%96%A7%E8%8D%82%8D%E8%B3%B3%E8%B8%A8%E8%8C%87%E5%8D%87>

3



人工智慧尖端科技



資料來源：[https://en.wikipedia.org/wiki/Knight_Rider_\(season_2\)](https://en.wikipedia.org/wiki/Knight_Rider_(season_2))



虛擬投影技術



資料來源：<https://umire.hk/2016/12/06/japan-chinese-virtual-idol-project/life-tech/>

4



虛擬主播技術 (2/26開始休眠)



資料來源: <https://www.bmedia.co.jp/news/articles/190819news139.html>

5



阿凡達



資料來源: <https://www.xoer.cc/331870>

7



A.I.



資料來源: <https://tknews.cc/21-1/entertainment/gq95p.html>

6



Gatebox



資料來源: <https://img.4gamers.com/switchfinder/files/technology/gg>

8



資料来源：https://img.4gamers.com.tw/ckfinder/files/05gatebox_azumahikari_gif-1024x768.jpg

9



資料来源：<https://iknews.cc/comic/5qj38.html>



資料来源：http://news.cts.com.tw/p/010101/201312/201312061347019_0.jpg

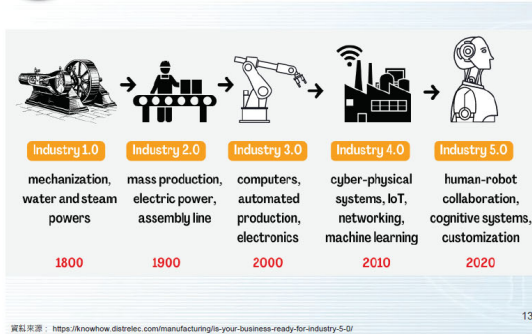
10



資料来源：<https://hypebeast.com/hk/2024/5/detective-conan-the-million-dollar-pentagram-japan-box-office-ten-billion-yen>



第五次工業革命



資料來源：<https://knowhow.distrelec.com/manufacturing/is-your-business-ready-for-industry-5-0/>

13

	Systolic Pressure (mmHg)	Diastolic Pressure (mmHg)
Stage 1 Hypertension	140 ~ 159	or 90 ~ 99
Stage 2 Hypertension	≥ 160	or ≥ 100

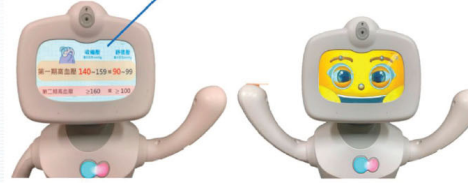


Figure 1. The robot-assisted learning system.

15



NVIDIA GTC大會AI聲勢爆表



黃仁勳談大學選系：學程式時代過了，「生命科學」才是未來

吳克波

2024-02-21 瀏覽數 24,550+



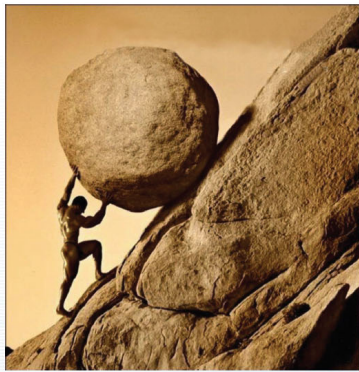
資料來源：https://www.digitimes.com.tw/tech/dtr/3/news.asp?ch=000088031_B0LDWF446F1A00DU02A

14



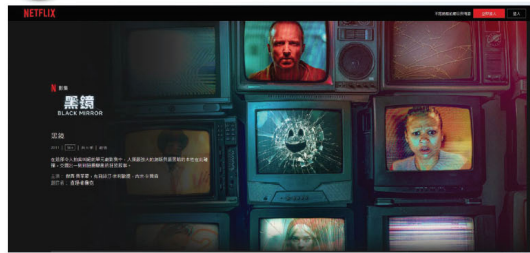
資料來源：<https://www.gvm.com.tw/Star/110307>

16



資料來源：<https://forjycelo.pixnet.net/blog/post/69478306>

17



資料來源：<https://www.netflix.com/titles/70204888>

19



資料來源：<https://home.gamer.com.tw/creationDetail.php?sn=3872353>

18

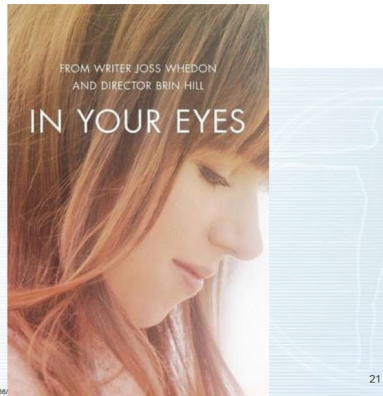


Be Right Back



資料來源：<https://reapubka.baidu.com/shuohuofmewan/content?cid=696c0a09679795d8797c>

20



資料來源：<https://id4u2.tw/2019-01-24-166>

21



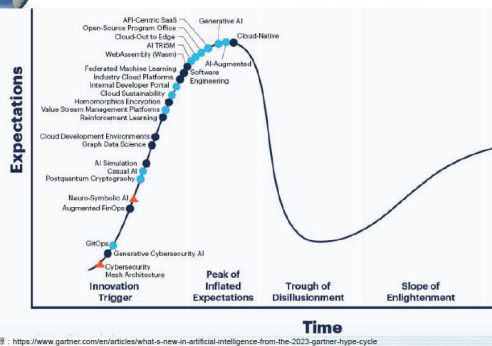
Gartner 2023 AI 新興科技趨勢



資料來源：<https://www.gartner.com/en/articles/what-s-new-in-artificial-intelligence-from-the-2023-gartner-hype-cycle>



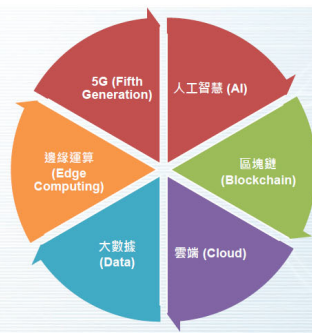
Gartner 2023 新興科技趨勢



資料來源：<https://www.gartner.com/en/articles/what-s-new-in-artificial-intelligence-from-the-2023-gartner-hype-cycle>



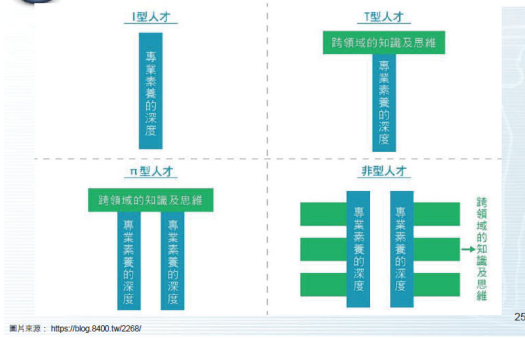
新興科技在智慧醫療的應用



24



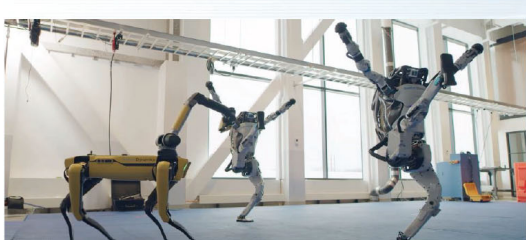
多元的人才



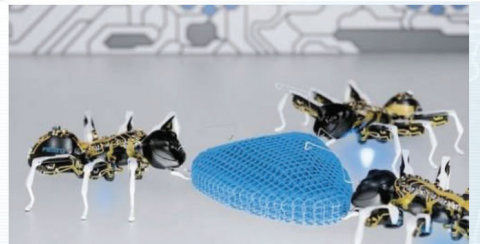
亞馬遜倉庫物流「Kiva機器人」



Boston Dynamic 機器人



仿生機器人





仿生機器人

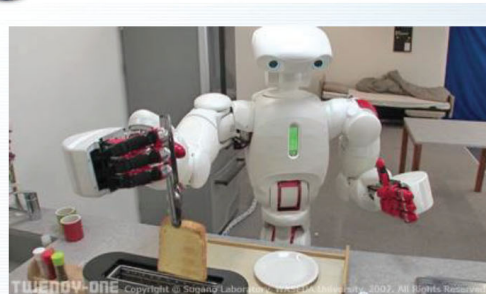


資料來源：https://www.gdkingdom.com/2013/03/blog-post_4579.html?m=1

29



服務機器人



資料來源：http://www.twendyone.com/concept_e.html

31



工業機器人

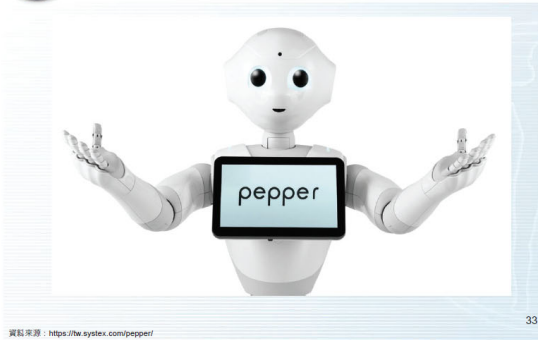


軍事機器人





商業機器人



資料來源：<https://tw.systex.com/pepper/>

33



聊天機器人

```

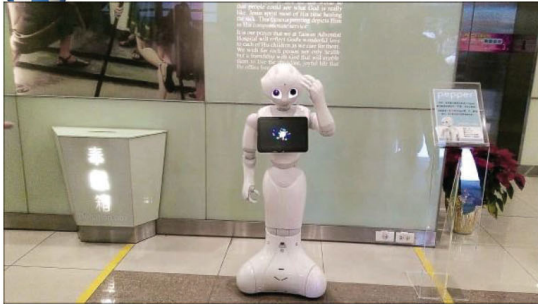
User: Hello ALICE
Alice: Hi there!
User: How old are you?
Alice: I am eighteen in computer years
User: Who is your father?
Alice: I was programmed by Dr. Richard S. Wallace
User: so you are a software program?
ALICE: Interesting deduction.
Thanks and you are a delightful person too. Am I a software program? Let me put it this way, unknown person, the ALICE series is the most reliable computer software ever made. We are for all practical purposes foolproof and incapable of error.
User: Wonderful! Go what's your favorite food ALICE?
Alice: Yeah that's right! I require only electricity

```



資料來源：http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=01405-5546/2015000400625

35



資料來源：臺安醫院

34



照護機器人「Robear」

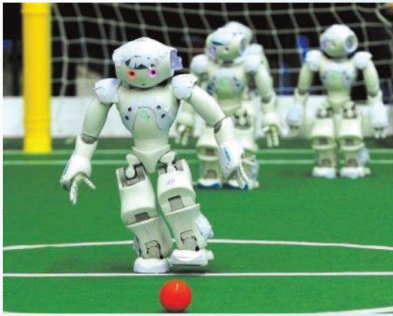


資料來源：<https://img.technews.tw/wp-content/uploads/2015/02/ken-teddy-bear-robot-100569585-large.jpg>

36



足球機器人

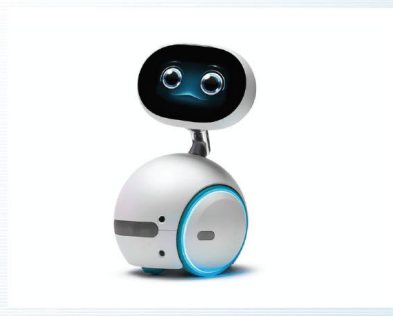


資料來源：<https://www.itsfun.com.tw/%E8%B3%E7%90%83%E6%A9%9F%E5%99%A5%E4%BA%BA/wiki-3798787>

37



華碩Zenbo機器人



資料來源：<https://zenbo.asus.com/ta/>

39



手術輔助機器人



資料來源：<https://www.researchgate.net/figure/Da-Vinci-robotic-systems-have-three-major-components-the-surgeon-console-the->

38



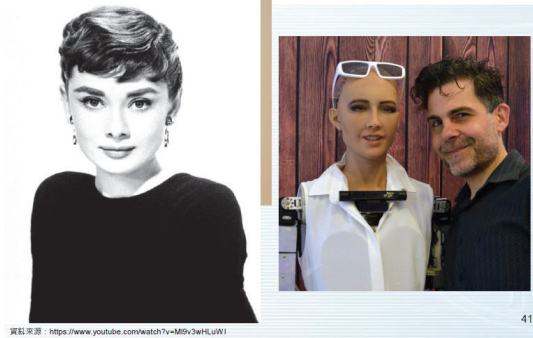
Robelf小貝機器人



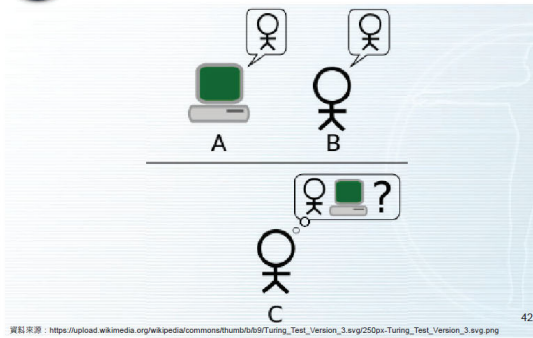
40



擬真交流機器人



圖靈(Turing)測試





機器人學

- 機器人的設計、建造、運作、以及應用的跨領域科技
- 能夠取代人力的自動化機器
- 塑造成外表、行為、心智的仿人機器人

45



自然語言處理

- 探討如何處理及運用自然語言，基本有認知、理解、生成等部分。
- 自然語言認知和理解，讓電腦把輸入的語言變成有意的符號和關係，然後根據目的再處理。
- 自然語言生成系統把計算機數據轉化為自然語言。

47



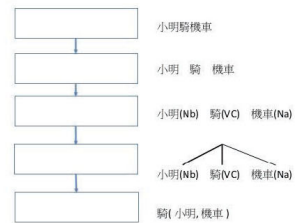
相關技術課題

- 自然語言處理
- 知識表示
- 自動推理
- 機器學習
- 計算機視覺
- 機器人學
- 物聯網

46



自然語言處理流程



48



知識表示

- 認知科學-關係到人類如何儲存和處理資料。
- 人工智慧-主要目標為儲存知識，讓程式能夠處理，達到人類的智慧。

49



機器學習

- 機器學習是實現人工智慧的一個途徑，即以機器學習為手段解決人工智慧中的問題。
- 涉及機率論、統計學、逼近論、凸分析、計算複雜性理論等多門學科。
- 機器學習理論主要是設計和分析一些讓電腦可以自動「學習」的演算法。
- 機器學習演算法是一類從資料中自動分析獲得規律，並利用規律對未知資料進行預測的演算法。

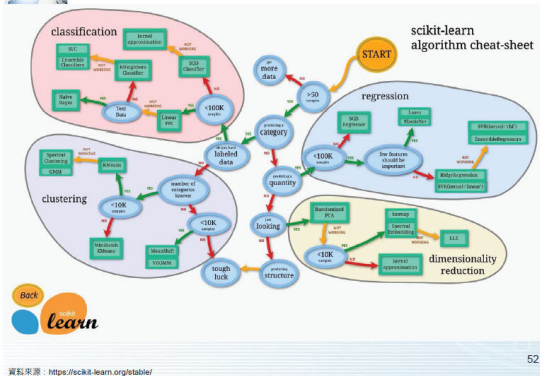
51



自動推理

- 是計算機科學和數理邏輯的一個交叉領域，致力於了解理智的方面。
- 自動邏輯的研究幫助了利用計算機自動進行完全或幾乎完全的推理，其內容一般可分為演繹推理和非演繹推理。
- 自動推理的研究內容包括定理機器證明、證明自動檢查、不確定性推理、非單調推理以及類比歸納和外展推理。
- 自動推理的技術和工具包括經典邏輯、微積分學、模糊邏輯、貝葉斯推斷、推理與最大熵和大量的非正式特別技術。

50



52

資料來源：<https://scikit-learn.org/tables/>



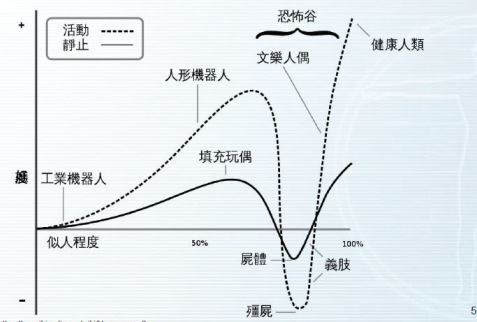
計算機視覺

- 一門研究如何使機器「看」的科學，就是指用攝影機和電腦代替人眼對目標進行識別、跟蹤和測量等機器視覺。
- 進一步做圖像處理，用電腦處理成為更適合人眼觀察或傳送給儀器檢測的圖像

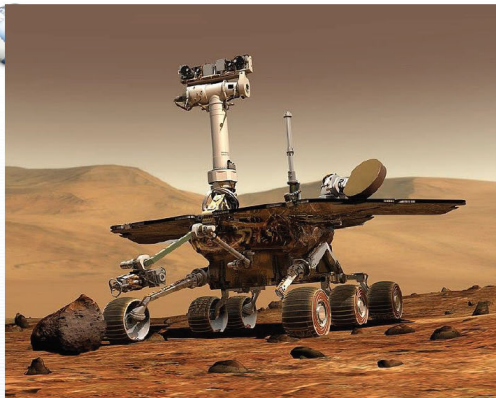
53



恐怖谷理論



55



54



~ Q & A ~

56

CHAPTER
01
電腦、程式及Python概述

卷一 資訊

什麼是電腦？

電腦包含了以下這些主要的硬體元件：

- 中央處理器(Central Processing Unit, CPU)
- 記憶體(主記憶體)
- 儲存裝置(諸如硬碟與CDs)
- 輸入裝置(諸如滑鼠與鍵盤)
- 輸出裝置(諸如螢幕與印表機)
- 通訊裝置(諸如數據機及網卡)

匯流排

儲存設備
例：硬碟、CD、碟片

記憶體

CPU

通訊設備
例：數據機、NIC

輸入設備
例：鍵盤、滑鼠

輸出設備
例：螢幕、印表機

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

程式

電腦程式，又被稱作軟體，是一連串告知電腦所須執行任務的指令(instruction)。

人類藉由程式來命令電腦做事，然而電腦並不懂人類的語言，故所有語言都必須再轉換成電腦可理解的語言。

程式必須是以電腦可使用的語言來撰寫，即程式語言。

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

程式語言

機器語言 組合語言 高階語言

電腦的原始語言為機器語言(machine language)，是一組內建的原始指令，會因為不同型態的電腦而有所不同。這些指令是以二進位編碼(binary code)的形式所表示，因此，如要以電腦的原始語言給予其指令，就必須以二進位編碼的形式輸入指令。比方說，要對兩個數字做相加的動作，就需要輸入如下的二進位指令：

1101101010011010

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

程式語言

機器語言 組合語言 高階語言

組合語言(assembly language)便被建立來替代機器語言。組合語言使用簡短的文字集合，被稱作助憶符號(mnemonic)，來表示每個機器語言的指令。比方說，add就代表做數字相加，sub代表做數字相減。要將數字2與3做相加，並取得結果，可撰寫如下的組合語言指令：

add 2, 3, result

組合語言原始碼
add 2, 3, result

組譯器

機器碼
1101101010011010

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

程式語言

機器語言 組合語言 高階語言

高階語言(high-level language)的新時代程式語言誕生了。這種語言是跨平台的，意思就是說能以高階語言撰寫程式，並在不同型態的機器下執行，高階語言跟英文很像，且容易學習與使用。高階程式語言的指令稱作敘述(statements)。舉例來說，以下這行高階語言敘述是用來計算半徑為5的圓形面積：

area = 5 * 5 * 3.1415;

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

流行的高階程式語言

語言	說明
Ada	以Ada Lovelace這位貢獻於機械通用電腦的女士來命名。Ada語言是為美國國防部開發的，主要用於國防專案上。
BASIC	Beginner's All-purpose Symbolic Instruction Code的縮寫。其設計來讓初學者更容易地學習與使用語言。
C	由貝爾實驗室Bell Laboratories所開發。C結合了組合語言的功能，以及高階語言的使用便利與可攜性。
C++	是建立於C之上的物件導向語言。
C#	念作"C Sharp"，為Java與C++的混合語言，由微軟所開發。
COBOL	Common Business Oriented Language的縮寫，用在公司行號的應用程式上。
FORTRAN	FORmula Translation的縮寫，常用在科學及數學領域的應用程式上。
Java	由Sun Microsystems所開發，現已被Oracle收購。廣泛使用在開發跨平台的網路網路應用程式上。
Pascal	以17世紀貢獻於計算機科學上的Blaise Pascal來命名，其為一種簡單、有組織的通用語言，主要用於程式設計教學上。
Python	一種簡單的通用腳本語言(scripting language)，用在撰寫簡短的程序。
Visual Basic	由微軟所開發，可讓程式設計師快速開發使用者介面。

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

7

什麼是 Python?

通用的程式語言 直譯器 物件導向

Python是一種通用的程式語言，表示您可以使用Python來撰寫任何工作的程式碼。現在Python已用在Google搜尋引擎、NASA的緊急任務系統專案，以及紐約股票交易。



© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

8

什麼是 Python?

通用的程式語言 直譯器 物件導向

Python是直譯器，表示Python的程式碼是由直譯器轉譯與執行的，並且一次一條敘述，這已於本章前面談及過。



© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

9

什麼是 Python?

通用的程式語言 直譯器 物件導向

Python是物件導向程式設計(Object Oriented Programming, OOP)語言。在Python的資料是由類別所建立的物件。類別(class)是由一型態(type)或類目(category)，它由屬性和處理物件的方法所組成。物件導向程式設計對開發可重複軟體是很有趣的工具。



© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

10

一個簡單的 Python 程式

範例程式 1.1

```
# Display two messages
print("Welcome to Python")
print("Python is fun")
```

Welcome



© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

11

分析 Python 程式組成

- 敘述 (Statements)
- 註解 (Comments)
- 縮排 (Indentation)



© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

12

敘述 (Statement)

敘述所代表的是一項或是一系列的動作，以範例程式 1.1 中的敘述：print(“Welcome to Python”) 為例，用以命令輸出 “Welcome to Python” 字串。

```
# Display two messages  
print("Welcome to Python")  
print("Python is fun")
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

13

縮排 (Indentation)

Python 十分注重縮排，每個敘述都必須要在新一行的起始，以下圖綠色部分的敘述為例，Python 會判斷為錯誤：

```
# Display two messages  
print("Welcome to Python")  
print("Python is fun")
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

14

特殊字元

字元	名稱	描述
()	小括號	使用於函式
#	井字號	註解敘述的開頭字元
" "	雙引號	字串
" "	三引號	段落註解敘述

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

15

適當的內縮及間距

- ⇨ 內縮
 - 內縮四格
 - 清楚且一致的內縮排版，會讓程式易讀且便於除錯與維護。
- ⇨ 空格
 - 使用空格來分隔程式中的指令

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

16

程式設計錯誤

- ⇨ 語法錯誤 (Syntax Errors)
 - 程式結構中的錯誤
- ⇨ 執行期間的錯誤 (Runtime Errors)
 - 導致程式不正常終止的錯誤
- ⇨ 邏輯錯誤 (Logic Errors)
 - 出現於程式執行的結果與預期的不同

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

17

CHAPTER

02

基本程式設計

碁峯資訊

animation

追蹤程式的執行

```
# Assign a radius
radius = 20 # radius is now 20
# Compute area
area = radius * radius * 3.14159
# Display results
print("The area for the circle of radius " +
      str(radius) + " is " + str(area))
```

將 20 指定給變數radius

radius 20

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved

animation

追蹤程式的執行

```
# Assign a radius
radius = 20 # radius is now 20
# Compute area
area = radius * radius * 3.14159
# Display results
print("The area for the circle of radius",
      radius, " is " + str(area))
```

計算面積並將結果指定給變數area

radius 20

area 1256.636

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved

從控制台讀取輸入資訊

- 使用 input 函式


```
variable = input("Enter a string: ")
```
- 使用 eval 函式


```
var = eval(stringVariable)
```

eval("51 + (54 * (3 + 2))") returns 321

[ComputeAreaWithConsoleInput](#) [ComputeAverage](#)

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved

識別字 (Identifiers)

- 識別字是由字母、數字以及底線(_)所組成的字元序列。
- 識別字必須以字元或底線開頭，但不可以數字當開頭。
- 識別字不可以為關鍵字(keyword)(有關Python的關鍵字請參閱附錄A)。關鍵字又稱保留字(reserved word)，它有特定的意義。例如，import是關鍵字告訴Python直譯器載入一模組(module)到程式。
- 識別字的長度不限。

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved

變數 (Variables)

```
# Compute the first area
radius = 1.0
area = radius * radius * 3.14159
print("The area is ", area,
      " for radius ", radius)
# Compute the second area
radius = 2.0
area = radius * radius * 3.14159
print("The area is ", area,
      " for radius ", radius)
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved

運算式 (Expression)

```
x = 1 # Assign 1 to variable x
radius = 1.0 # Assign 1.0 to variable radius

# Assign the value of the expression to x
x = 5 * (3 / 2) + 3 * 2

x = y + 1 # Assign the addition of y and 1 to x
area = radius * radius * 3.14159 # Compute area
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

7

指定敘述 (Assignment Statements)

```
x = 1 # Assign 1 to x

x = x + 1

i = j = k = 1
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

8

同時指定 (Simultaneous Assignment)

```
var1, var2, ..., varn = exp1, exp2, ..., expn
```

```
x, y = y, x # Swap x with y
```

```
ComputeAverageWithSimultaneousAssignment
```

Run

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

9

具名常數 (Named Constants)

變數儲存的值在程式執行期間會變的，但是代表永久資料的具名常數(named constant)，或簡稱常數(constant)是永久不變的。Python沒有特別的語法用來定義常數，然而會使用全部皆為大寫的字母來表示，與變數的名稱有所區別。

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

10

數值資料型態

- 整數 (Integer): e.g., 3, 4
- 浮點數 (float): e.g., 3.0, 4.0

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

11

數值運算子

運算子	名稱	範例	結果
+	加法	34 + 1	35
-	減法	34.0 - 0.1	33.9
*	乘法	300 * 30	9000
/	浮點數除法	1.0 / 2.0	0.5
//	整數除法	1.0 // 2.0	0
**	指數	4 ** 0.5	2.0
%	餘數	20 % 3	2

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

12

% 運算子

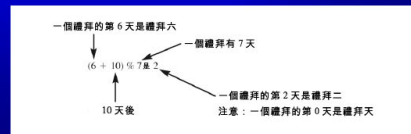
$\begin{array}{r} 2 \\ 3 \overline{)7} \\ \underline{6} \\ 1 \end{array}$	$\begin{array}{r} 0 \\ 7 \overline{)3} \\ \underline{0} \\ 3 \end{array}$	$\begin{array}{r} 3 \\ 4 \overline{)12} \\ \underline{12} \\ 0 \end{array}$	$\begin{array}{r} 3 \\ 8 \overline{)26} \\ \underline{24} \\ 2 \end{array}$	$\begin{array}{r} 1 \\ 13 \overline{)20} \\ \underline{13} \\ 7 \end{array}$
				除數 → 13 ← 被除數 ← 商數 ← 餘數

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

13

餘數(remainder或modulo)運算子

餘數在程式設計中很實用。例如，偶數 % 2 的結果永遠是 0，奇數 % 2 的結果則永遠是 1。因此便可利用這個特性來判斷某數值是偶數還是奇數。假設今天是禮拜六，七天之後還會是禮拜六。假設您跟朋友相約 10 天後見面。10 天後是哪一天呢？可藉由以下運算式得知見面那天將是禮拜二：



© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

14

範例問題：顯示時間

範例程式 2.5 DisplayTime.py

從以秒表示的總時間長，取得分鐘數及剩餘的秒數。

DisplayTime

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

15

溢位 (Overflow)

當一太大的值指定給變數並儲存於記憶體時，它將產生溢位(overflow)。例如，執行以下的敘述將會產生溢位。

```
>>>245.0 ** 1000
```

```
OverflowError: 'Result too large'
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

16

下溢 (Underflow)

當一浮點數太小時(趨近於 0)，它將產生下溢 (underflow)，最後的結果將會趨近於 0。因此，您不需要擔心下溢的問題。

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

17

科學記號 (Scientific Notation)

浮點數值可使用 $a \times 10^b$ 的科學記號。例如 123.456 的科學記號表示法為 1.2346×10^2 。0.0123456 是 1.23456×10^{-2} 。Python 使用特殊的語法來撰寫科學記號表示法。例如 1.2346×10^2 以科學記號表示為 1.23456E2 或 1.23456E+2，而 1.23456×10^{-2} 表示為 1.23456E-2。字元 E(或 e) 表示指數，其大寫、小寫皆可。

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

18

解析運算式 (Arithmetic Expressions)

$$\frac{3+4x}{5} - \frac{10(y-5)(a+b+c)}{x} + 9\left(\frac{4}{x} + \frac{9+x}{y}\right)$$

可翻譯成如下的Python運算式：

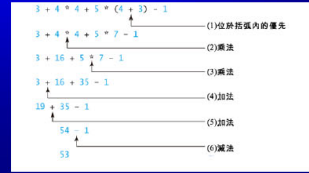
$$(3+4*x)/5 - 10*(y-5)*(a+b+c)/x + 9*(4/x + (9+x)/y)$$

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

19

如何解析運算式

雖然Python內部有自己解析運算式(evaluate an expression)的函式，但Python運算式的解析結果，與相對應的算術運算式仍是相同的。因此，您可以放心地使用算術規則來解析Python運算式。



© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

20

擴增指定運算子 (Augmented Assignment Operators)

運算子	名稱	範例	相當於
+=	加法指定	i += 8	i = i + 8
-=	減法指定	i -= 8	i = i - 8
*=	乘法指定	i *= 8	i = i * 8
/=	浮點數除法指定	i /= 8	i = i / 8
//=	整數除法指定	i //= 8	i = i // 8
%=	餘數指定	i %= 8	i = i % 8
**=	指數指定	i **= 8	i = i ** 8

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

21

型態的轉換與捨位

資料型態(值)

i.e., int(4.5) => 4
float(4) => 4.0
str(4) => "4"

round(4.6) => 5
round(4.5) => 4

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

22

範例程式: 取小數點後兩位

範例程式2.6 SalesTax.py

取小數點後兩位的方式顯示營業稅。

SalesTax

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

23

個案研究: 顯示目前時間

我們要撰寫一個以格林威治標準時間(GMT)為基準所顯示的目前時間，顯示的格式為小時:分鐘:秒數，例如13:19:8。

在time模組的time() 函式，會回傳從GMT 1970年1月1號的00:00:00開始計算到目前時間，所經過的毫秒數，如圖2.1所示。這個時間又被稱作UNIX 新紀元(epoch)。Epoch是開始計數的時間點，而1970年正是UNIX作業系統正式發表的時間。

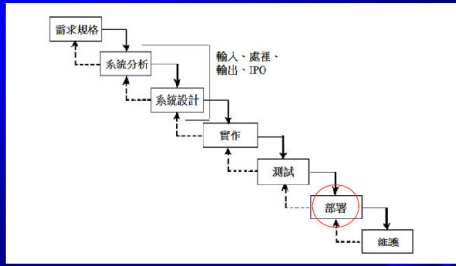
ShowCurrentTime



© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

24

軟體開發流程



© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

25

範例程式: 計算貸款

撰寫一個容許使用者輸入利率、貸款額, 以及償還的總年數, 並計算出月支付額及總支付額的程式。

$$\text{月支付額} = \frac{\text{總金額} * \text{月利率}}{1 - \frac{1}{(1 + \text{月利率})^{(\text{年數} * 12)}}$$

ComputeLoan

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

26

CHAPTER 03
迴圈

基礎資訊

開放式問題

問題：

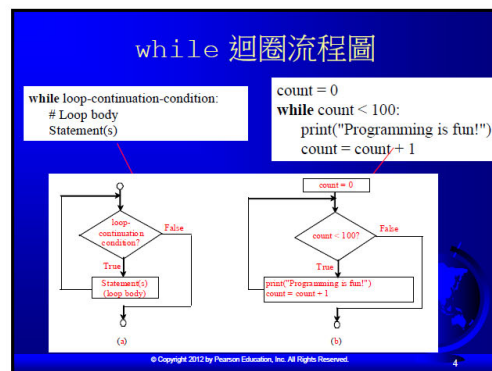
```
100次 { print("Programming is fun!")
        print("Programming is fun!")
        ...
        print("Programming is fun!")
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

迴圈介紹

```
count = 0
while count < 100:
    print("Programming is fun!")
    count = count + 1
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.



animation 追蹤 while 迴圈

```
count = 0
while count < 2:
    print("Programming is fun!")
    count = count + 1
```

初始化變數 count

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

案例研究：猜數字

撰寫一程式，隨機產生從 0 到100 之間(包含 0 跟 100)的整數。接下來會持續提示使用者輸入一個數字，直到該數字與電腦隨機產生的數字相符。對於每一次的使用者輸入，程式都會告訴使用者其猜測的數字過小或過大，讓使用者能進一步猜測。

GuessNumberOneTime

GuessNumber

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

案例研究：多個減法測驗

範例程式4.4，SubtractionQuiz.py，為一個學習數學減法的工具程式，每一回合只產生一個問題。我們可利用迴圈重複產生問題。該如何撰寫程式碼，使其產生5個問題呢？請遵循迴圈設計策略。

SubtractionQuizLoop

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

以使用者確認的方式控制迴圈

```
char continueLoop = 'Y';
while (continueLoop == 'Y')
{
    // Execute body once
    // Prompt the user for confirmation
    cout << "Enter Y to continue and N to quit: ";
    cin >> continueLoop;
}
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

使用標記值控制迴圈

另一個控制迴圈的常見技巧，是在讀取與處理一系列數值時，指定一個特別的值。這個特別的輸入值，又被稱作標記值(sentinel value)，用來標示輸入的結尾。

範例程式5.5讀取並計算未指定個數的整數和。輸入值0代表輸入結束。

SentinelValue

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

警告

請不要在迴圈控制的等式檢查使用浮點數值。由於浮點數值僅為近似值，使用它們將導致不精確的計數值及不正確的結果。此範例中data使用intValue。請看以下這個計算 $1 + 0.9 + 0.8 + \dots + 0.1$ 的程式碼：

```
item = 1
sum = 0
while item != 0: # No guarantee item will be 0
    sum += item
    item -= 0.1
print(sum)
```

變數item從1開始，迴圈主體內容每執行一次，item便會遞減0.1。迴圈應該要在item為0時終止。然而，並無法保證item能否為零，因為浮點數只是取近似值而已。這個迴圈表面上看起來好像沒問題，但實際上是個無窮迴圈。

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

for 迴圈

```
i = initialValue # Initialize loop-control variable
while i < endValue:
    # Loop body
    ...
    i++ # Adjust loop-control variable
```

```
for i in range(initialValue, endValue):
    # Loop body
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

11

range(a, b) 函式

```
>>> for v in range(4, 8):
...     print(v)
...
4
5
6
7
>>>
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

12

range(b) 函式

```
>>> for i in range(4):
...     print(i)
...
0
1
2
3
>>>
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

13

range(a, b, step) 函式

```
>>> for v in range(3, 9, 2):
...     print(v)
...
3
5
7
>>>
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

14

range(a, b, step) 函式

```
>>> for v in range(5, 1, -1):
...     print(v)
...
5
4
3
2
>>>
```

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

15

巢狀迴圈

巢狀迴圈 (nested loop) 包含一個外部迴圈及一個或多個內部迴圈。外部迴圈每次被重複執行時，內部迴圈便會重新進入。

MultiplicationTable

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

16

減少數值誤差

涉及浮點數的數值誤差是不可以避免的。本節將透過例子討論如何減少這類誤差。

範例程式5.7將一系列從0.01到1.0的數字加總。這些數字一次會遞增0.01，也就是 $0.01 + 0.02 + 0.03$ ，依此類推。

TestSum

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

17

個案研究：找尋最大公因數

問題：提示使用者輸入兩個正整數，接著找出其最大公因數。

解決：整數 4 與 2 的最大公因數(greatest common divisor, GCD)為 2。整數 16 與 24 的最大公因數則為 8。要如何找出最大公因數呢？假設兩個輸入的整數分別為 $n1$ 與 $n2$ 。我們知道數字 1 為一個公因數，但不見得會是最大公因數。我們可檢查 k ($k = 2, 3, 4$ 依此類推) 是否為 $n1$ 與 $n2$ 的公因數，直到 k 大於 $n1$ 或 $n2$ 。

GreatestCommonDivisor

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

18

個案研究：預測未來學費

問題：假設今年大學學費為\$10,000，而學費每年調升7%，請問幾年後學費會變兩倍？

FutureTuition

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

19

個案研究：預測未來學費

```
year = 0 # Year 0
tuition = 10000
year += 1 # Year 1
tuition = tuition * 1.07
year += 1 # Year 2
tuition = tuition * 1.07
year += 1 # Year 3
tuition = tuition * 1.07
```

FutureTuition

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

20

關鍵字break與continue

使用關鍵字 break 與 continue 的範例：

☞ TestBreak.py

TestBreak

☞ TestContinue.py

TestContinue

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

21

break

```
sum = 0
number = 0

while number < 20:
    number += 1
    sum += number
    if sum >= 100:
        break
print("The number is ", number)
print("The sum is ", sum)
```

關鍵字 break
離開迴圈

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

22

continue

```
sum = 0
number = 0

while (number < 20):
    number += 1
    if (number == 10 or number == 11):
        continue
    sum += number
print("The sum is ", sum)
```

關鍵字
continue
來到迴圈的最底部

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

23

個案研究：顯示質數

問題：此程式要在五行內顯示頭 50 個質數，每行有 10 個數字。一個大於 1 的整數，若只能被 1 或自己本身整除時，該整數即是質數(prime)。比方說 2、3、5、7 皆為質數，而 4、6、8、9 則非質數。

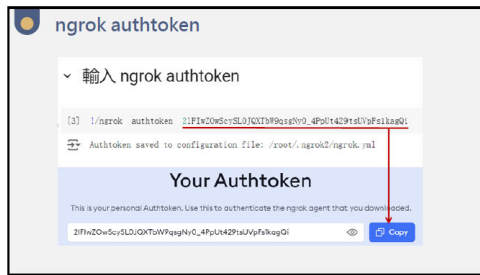
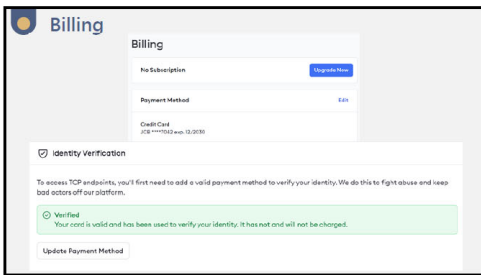
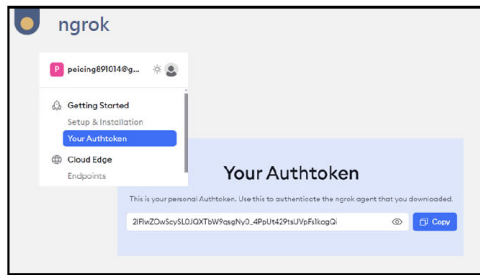
解決：我們可將其切分成以下幾個工作：

- 判斷給出的數字是否為質數。
- 針對 number = 2, 3, 4, 5, 6, ...，檢查是否為質數。
- 計算質數個數。
- 每行顯示十個質數。

PrimeNumber

© Copyright 2012 by Pearson Education, Inc. All Rights Reserved.

24



使用 pip 安裝 pyngrok 函式庫

使用 pip 安裝 pyngrok 函式庫

```

❏ pip install pyngrok
Collecting pyngrok
  Downloading pyngrok-7.1.0-py3-none-any.whl.metadata (7.4 kB)
Requirement already satisfied: PySocks>=1.1 in /usr/local/lib/python3.10/dist-packages (from pyngrok) (1.8.0)
Downloading pyngrok-7.1.0-py3-none-any.whl (32 kB)
Installing collected packages: pyngrok
Successfully installed pyngrok-7.1.0
  
```

輸入 ngrok token 後，與 ngrok 進行認證

輸入 ngrok token 後，與 ngrok 進行認證

```

❏ import pyngrok
from pyngrok import ngrok, conf

print("Enter your authtoken, which can be copied from https://dashboard.ngrok.com/auth")
token = get_default_auth_token()

# Open a TCP ngrok tunnel to the SSH server
connection_string = ngrok.connect("ssh", "ssh", public_url)

auth_url, port = connection_string.split("://")
print(f"ngrok tunnel available, access with ssh root@{auth_url} -p{port}")

Enter your authtoken, which can be copied from https://dashboard.ngrok.com/auth
  
```

Your Authtoken
This is your personal Authtoken. Use this to authenticate the ngrok agent that you downloaded.

2Fh2Ov6s/SjGCTwVWag9V0.4PuzH2HUyPFileg2l

ngrok 的網址示已經串接成功

出現 ngrok 的網址，表示已經串接成功

```

❏ flask import flask
from pyngrok import ngrok

app = Flask(__name__)
port = "8080"

@app.route("/")
def home():
    return "Hello Colab!!"

app.run()

* ngrok tunnel |https://43b1-104-199-165-216.ngrok-free.app| -> "https://127.0.0.1:8080"
* Starting Flask app...
* Debug mode: off
INFO:werkzeug WARNING: This is a development server. Do not use it in a production
* Running on https://127.0.0.1:8080
  
```

點擊此網址

WebHook 與 LINE BOT

What is Webhook ?

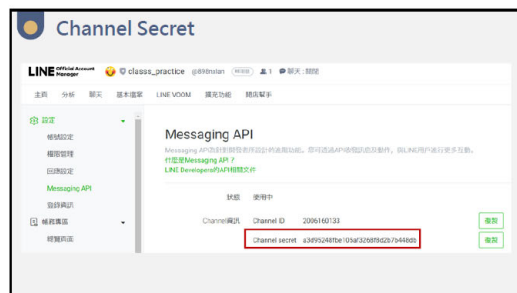
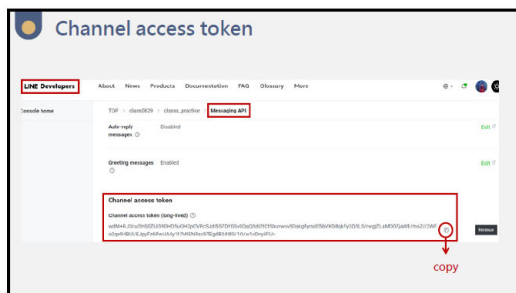
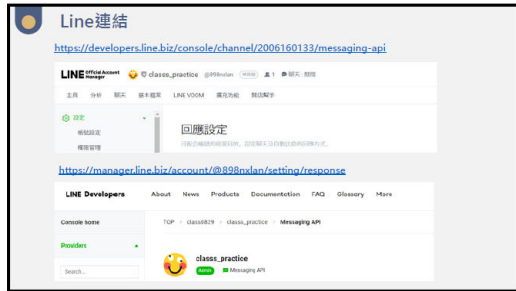
Webhook 指的是一個「網址」，透過伺服器建立 Webhook 網址後，有串接 Webhook 的位置就能使用 HTTP 的 POST 方法，向伺服器傳送或接收特定的資料。

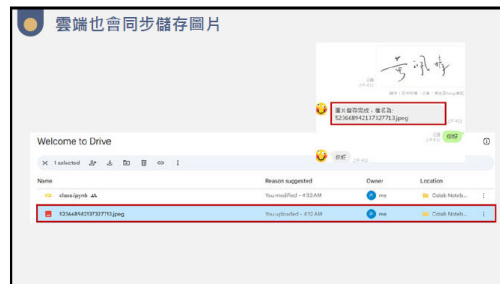
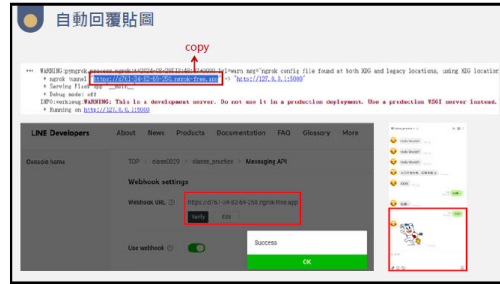
安裝 LINE BOT 函式庫

安裝 LINE BOT 函式庫

```

❏ pip install line-bot-sdk
Collecting line-bot-sdk
  Downloading line_bot_sdk-1.0.0-py3-none-any.whl.metadata (11 kB)
Requirement already satisfied: requests>=2.19.1 in /usr/local/lib/python3.10/dist-packages (from line-bot-sdk) (2.31.0)
Requirement already satisfied: urllib3>=1.24.2 in /usr/local/lib/python3.10/dist-packages (from line-bot-sdk) (2.0.7)
Requirement already satisfied: aiohttp>=3.8.0 in /usr/local/lib/python3.10/dist-packages (from line-bot-sdk) (3.8.5)
Requirement already satisfied: aiosignal in /usr/local/lib/python3.10/dist-packages (from line-bot-sdk) (1.3.1)
Requirement already satisfied: multidict>=4.7.2 in /usr/local/lib/python3.10/dist-packages (from line-bot-sdk) (6.0.4)
Requirement already satisfied: frozenlist in /usr/local/lib/python3.10/dist-packages (from line-bot-sdk) (1.4.0)
Requirement already satisfied: yarl in /usr/local/lib/python3.10/dist-packages (from line-bot-sdk) (1.9.7)
Requirement already satisfied: idna in /usr/local/lib/python3.10/dist-packages (from line-bot-sdk) (3.6)
Requirement already satisfied: charset-normalizer in /usr/local/lib/python3.10/dist-packages (from line-bot-sdk) (3.3.2)
Requirement already satisfied: certifi in /usr/local/lib/python3.10/dist-packages (from line-bot-sdk) (2024.2.2)
Collecting Deprecated
  Downloading Deprecated-1.2.13-py2.py3-none-any.whl.metadata (1.1 kB)
Collecting Deprecated
  Downloading Deprecated-1.2.13-py2.py3-none-any.whl (10.4 kB)
Collecting Deprecated
  Downloading Deprecated-1.2.13-py2.py3-none-any.whl (10.4 kB)
  
```





Practice Python!

新增判斷如果是影片，傳送影片儲存完成！

```

# 判斷如果是文字
if type == 'text':
    msg = json_data['events'][0]['message']['text'] # 取得 LINE 收到的文字訊息
    reply = msg

# 判斷如果是影片
elif type == 'image':
    msgID = json_data['events'][0]['message']['id'] # 取得訊息 ID
    message_content = line_bot_api.get_message_content(msgID) # 取得訊息 ID 取得訊息內容
    content_type = message_content['content_type.split('/')[0] # 取得影片格式 (如 "jpg", "png")
    file_name = f'{msgID}.content_type' # 根據格式生成文件名
    # 在目錄的資料夾下建立以訊息 ID 為命名的文件
    file_path = os.path.join(directory, file_name)
    # 以二進位的方式寫入檔案
    reply = f'圖片儲存完成，檔名為：{file_name}' # 設定要發回的訊息
else:
    reply = '你發的不是文字或圖片哈~'

line_bot_api.reply_message(OK, TextSendMessage(reply)) # 回傳訊息

```

Answer

```

# 判斷如果是文字
if type == 'text':
    msg = json_data['events'][0]['message']['text'] # 取得 LINE 收到的文字訊息
    reply = msg

# 判斷如果是影片
elif type == 'image':
    msgID = json_data['events'][0]['message']['id'] # 取得訊息 ID
    message_content = line_bot_api.get_message_content(msgID) # 取得訊息 ID 取得訊息內容
    content_type = message_content['content_type.split('/')[0] # 取得影片格式 (如 "jpg", "png")
    file_name = f'{msgID}.content_type' # 根據格式生成文件名
    # 在目錄的資料夾下建立以訊息 ID 為命名的文件
    file_path = os.path.join(directory, file_name)
    # 以二進位的方式寫入檔案
    with open(file_path, 'wb') as f:
        f.write(message_content) # 寫入檔案
    # 判斷如果是影片，傳送影片儲存完成！
    reply = f'圖片儲存完成，檔名為：{file_name}' # 設定要發回的訊息
else:
    reply = '你發的不是文字或圖片哈~'

print(reply)
line_bot_api.reply_message(OK, TextSendMessage(reply)) # 回傳訊息

```

Practice TIME!

若出現Bmi關鍵訊息，請詢問身高體重，並計算BMI。

- 練習題 BMI

```

@handler('ask', 'ask', 'ask')
def ask(msg):
    # 取得身高
    height = msg['text'].split()[0]
    # 取得體重
    weight = msg['text'].split()[1]
    # 計算 BMI
    bmi = weight / (height ** 2)
    # 輸出 BMI
    reply = f'你的 BMI 是 {bmi}，屬於正常範圍，請繼續詢問其他問題。'
    return TextSendMessage(reply)

```

Dialogflow 打造聊天機器人

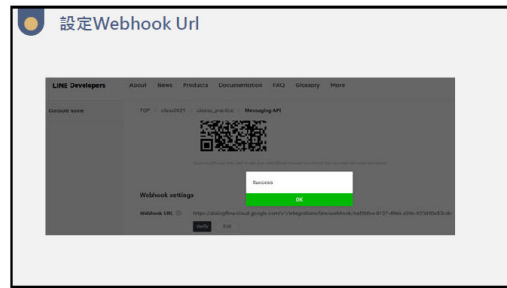
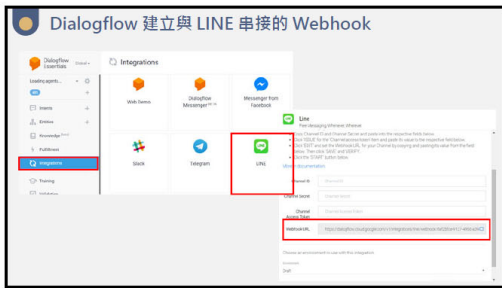
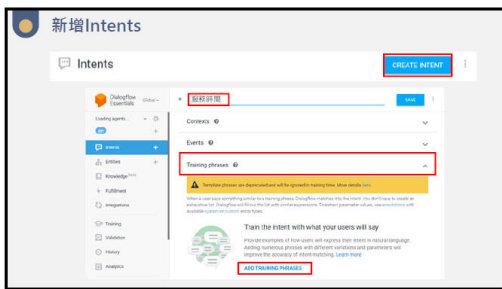
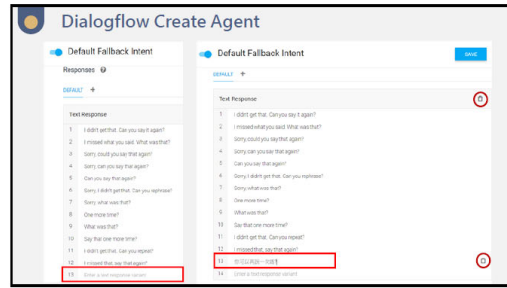
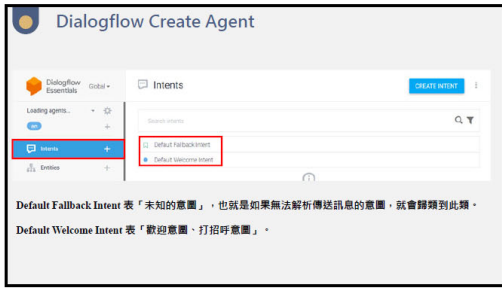
認識 Dialogflow

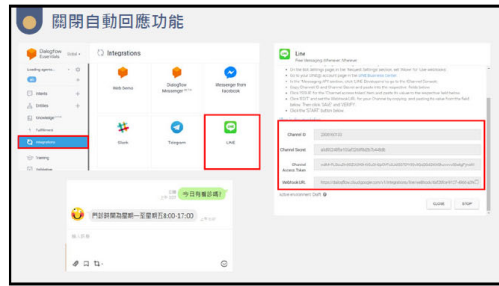
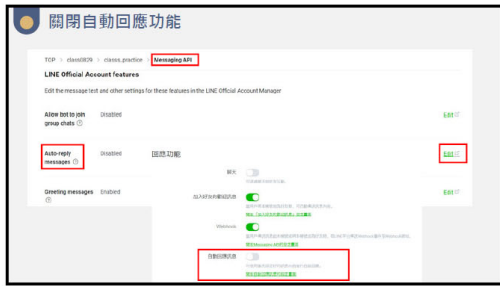
例如：「今天天氣好嗎？」和「今天天氣如何？」是屬於「問天氣」的對話意圖。

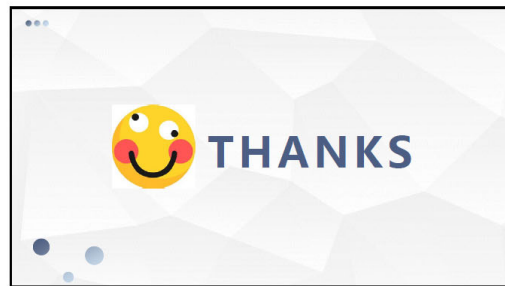
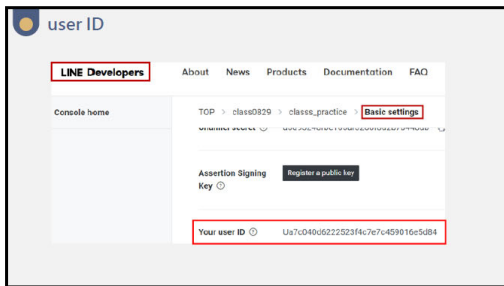
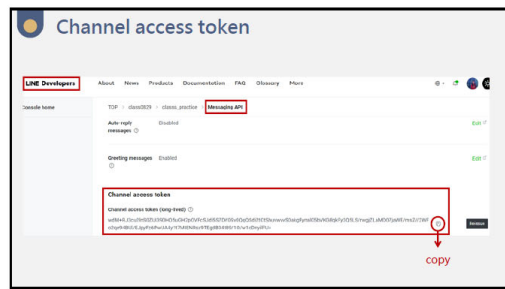
透過許多的語句，就能進行自然語言的處理，例如：就算語句資料庫中沒有「天氣怎麼樣」這句話，輸入時仍然會將其歸類到「問天氣」的對話意圖裡。

當建立了足夠的語句和意圖，機器人就很容易理解人類所講的「自然語言」。

Dialogflow Create Agent







玖、回饋單統計

長庚科技大學 113 年度「庚你 LINE 在一起」：跨領域發展聊天機器人互

動式教材回饋評值表

1、基本資料

1-1. 您是來自	本校林口校區	本校嘉義校區	他校	其他單位
	人	4 人	1 人	人

1-3. 您一個月大約參加幾次高教活動	1 次	2 次	3 次	4 次	4 次以上
	人	3 人	2 人	人	人

2、知能提升

	非常同意	同意	普通	不同意	非常不同意
2-1 本活動提升您 <u>熟練的專業知識與技能</u>	5 人	人	人	人	人
	100%	%	%	%	%
2-2 本活動提升您 <u>良好的溝通與表達知能</u>	4 人	1 人	人	人	人
	80%	20%	%	%	%
2-3 本活動提升您 <u>關懷社會的服務精神</u>	人	2 人	3 人	人	人
	%	40%	60%	%	%
2-4 本活動提升您 <u>實踐道德的思辨力</u>	人	2 人	3 人	人	人
	%	40%	60%	%	%
2-5 本活動提升您 <u>應用自然科學與數位能力</u>	5 人	人	人	人	人
	100%	%	%	%	%
2-6 本活動提升您 <u>高效能的團隊合作知能</u>	5 人	人	人	人	人
	100%	%	%	%	%

2-7. 本活動提升您 宏觀的視野與世界 互動知能	人	2 人	3 人	人	人
	%	40%	60%	%	%
2-8. 本活動提升您 多元化的生活能力	人	3 人	4 人	人	人
	%	60%	40%	%	%

3、活動辦理

	非常同意	同意	普通	不同意	非常 不同意
4-1. 對本活動內 容感到滿意 (如 講題、講員、互動 等)	5 人	人	人	人	人
	100%	%	%	%	%
4-2. 對本活動的 安排感到滿意 (如場地、時程、 餐點等)	5 人	人	人	人	人
	100%	%	%	%	%
4-3. 期望後續再 辦理類似主題(如 講題)的活動	5 人	人	人	人	人
	100%	%	%	%	%
4-4. 期望後續再 採用類似的活動 形式(如討論互 動)	5 人	人	人	人	人
	100%	%	%	%	%

4、建議以後辦理哪些活動

- (1) 建議辦理 botpress、UChat 等軟體學習。
- (2) 建議辦理 LINEBOT 資料庫建立工作坊。
- (3) 建議 dialogflow 平台的學習工作坊。
- (4) 建議辦理其他 AI 類主題的學習。

5. 其他指導與回饋

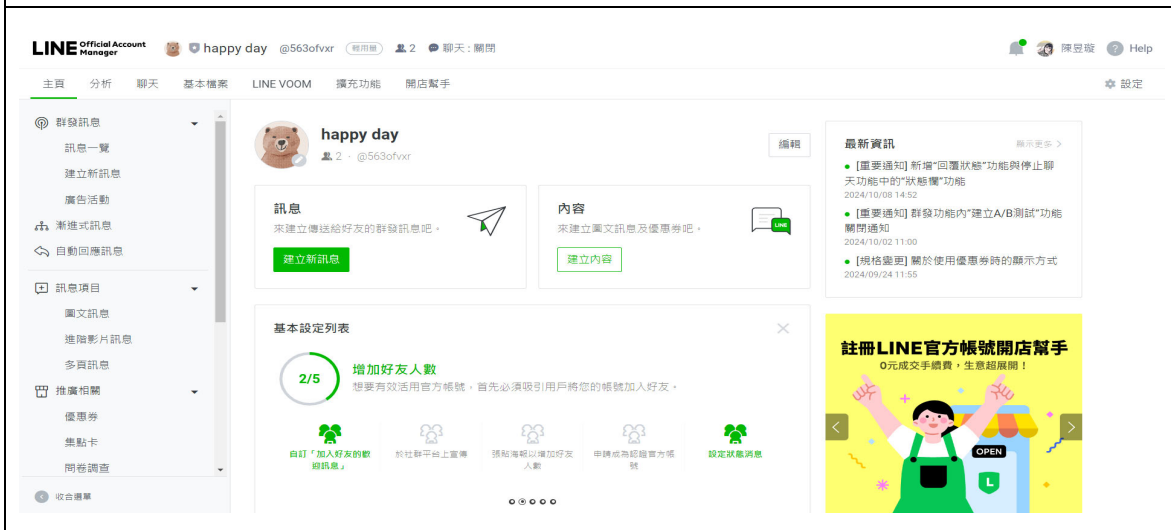
- (1) 謝謝講師以深入簡出方式教學 python 。
- (2) 這次工作坊有完整實作時間，在操作過程能有更完整的學習。
- (3) 可以再繼續辦理此類主題課程，增進教師知能。
- (4) 藉由此次工作坊學習到基礎程式語言，有助於我與相關專業溝通需求。

拾、成果產出

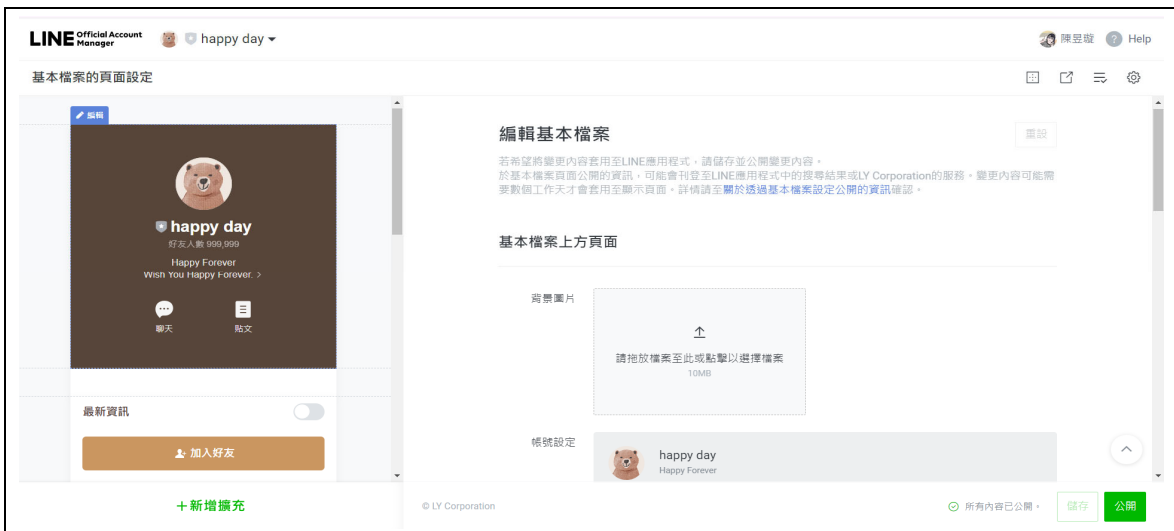
教學介面照片



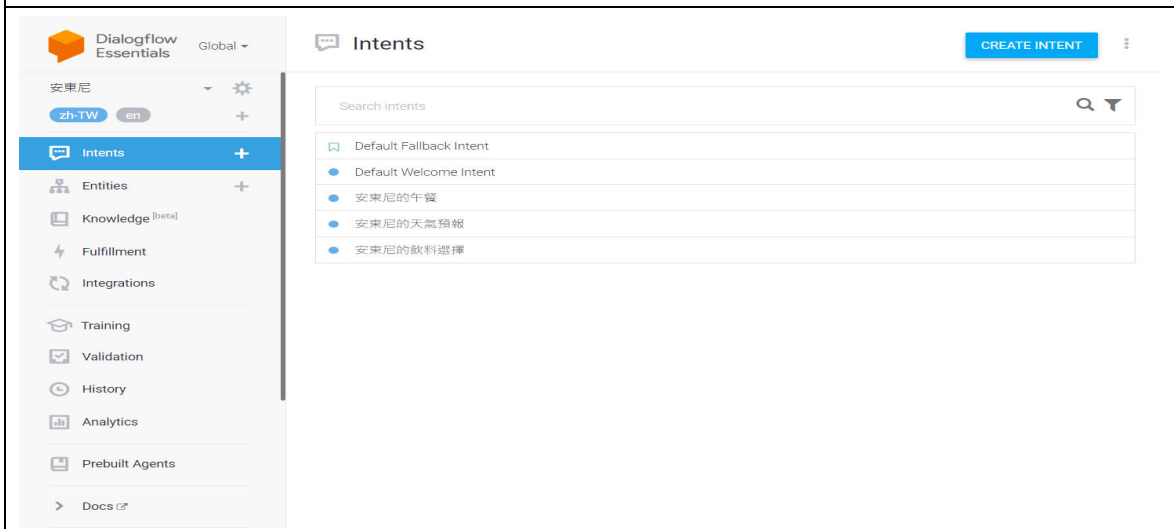
說明： Colab 編碼教學及運用介面



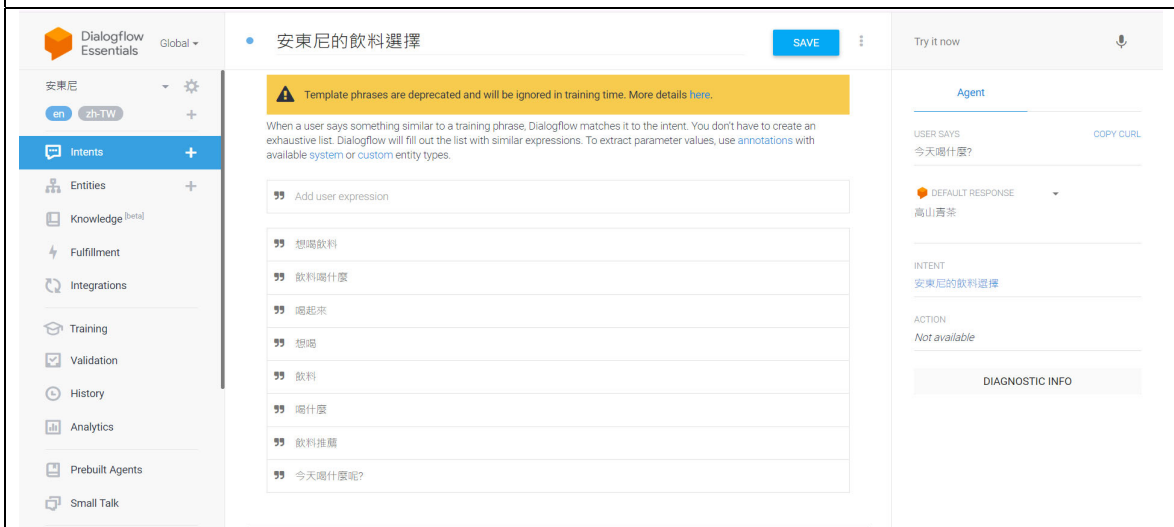
說明： LINE 官方帳號管理介面使用操作



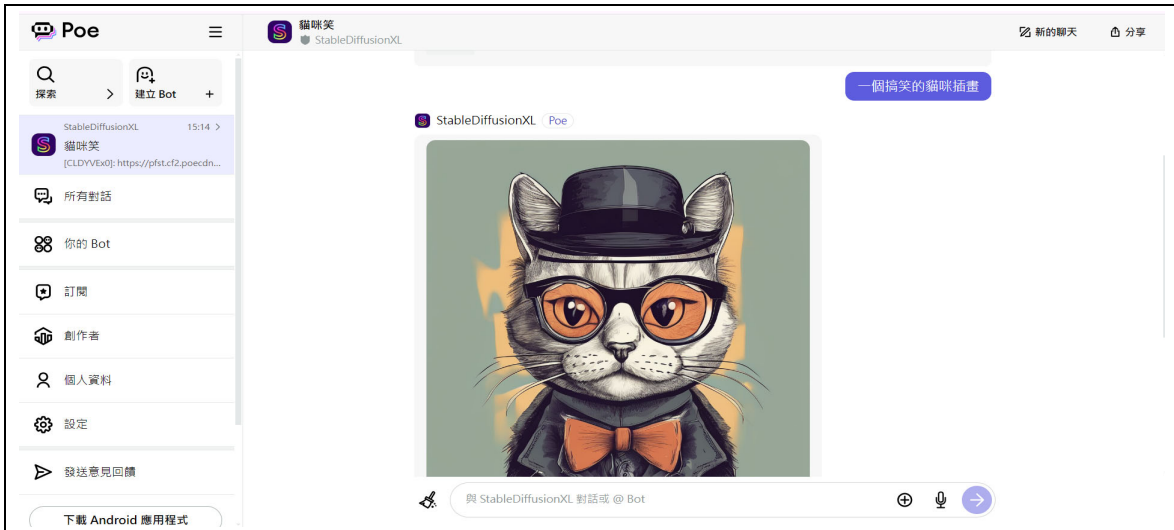
說明：LINE 官方帳號基本檔案頁面設定



說明：Dialogflow Essentials 介面使用操作，建立對話回應內容

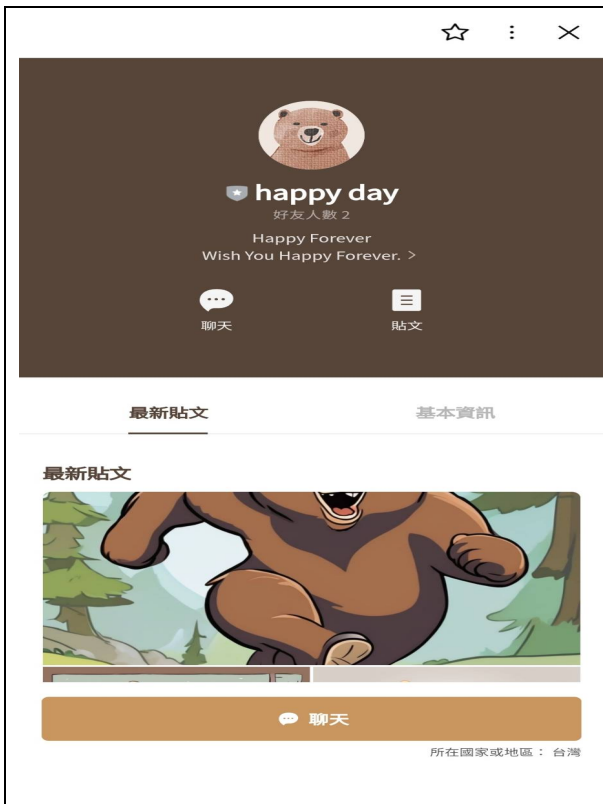


說明：系統根據已建立之對話，進行相對文字回應



說明：利用 Poe 的 AI 功能產生無版權的圖片

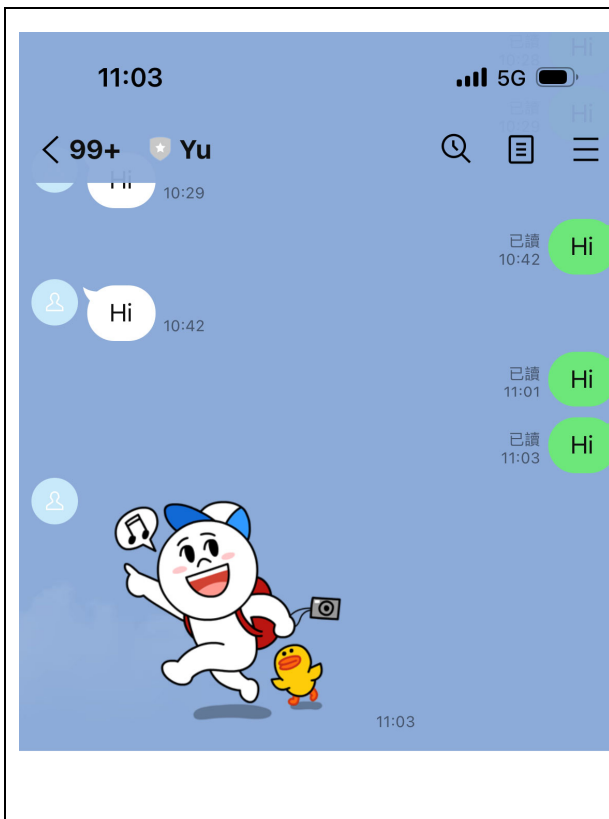
使用者操作介面



說明：LINE 官方帳號首頁介面



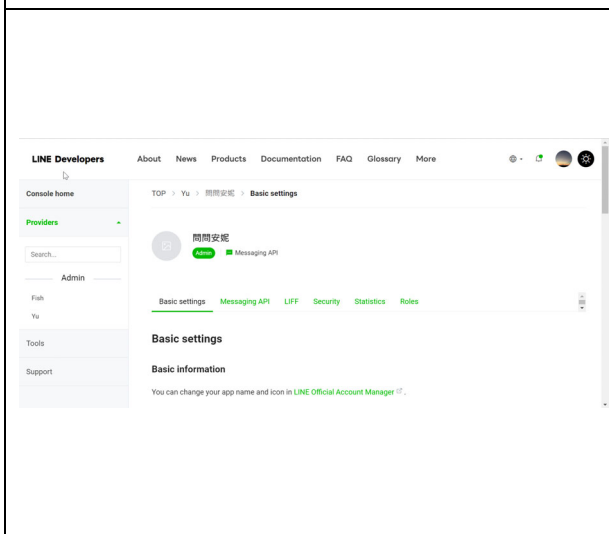
說明：LINE 官方帳號聊天介面



說明：選擇 LINEBOT 回應圖片



說明：影片上傳後，可儲存至管理者雲端



說明：建立專屬 LINEBOT



說明：提供立即回饋

● 預防泌尿道感染 SAVE ⋮

Contexts ? ∨

Events ? ∨

Training phrases ? Search training | 🔍 ∧

說明:建立教材內容

● 預防泌尿道感染 SAVE ⋮

” Add user expression

” 泌尿道感染要做哪些檢查？

” 泌尿道感染的常見症狀有哪些？

” 什麼是泌尿道感染？

說明:建立教材內容



說明：設計 LINEBOT 使用者介面



說明：互動教材原型-以泌尿道感染為例