

國立高雄大學

課程：基礎程式設計

112學年 第一學期

個人學習歷程檔

林彧頎-資工系-A1105505

指導教授：曾智義 老師

中　華　民　國 112年 12月 19日

|  |
| --- |
| 教師：曾智義 課程名稱：基礎程式設計 學期：112-1 |
| 一、第02週課堂練習 |
| calendar運用 |
| 二、個人/成員 |
| 林彧頎\_資工系114級\_A1105505 |
| 三、議題規劃 |
| 由於我們的家族事業皆是製造業或重工業，舉凡:橡膠業、鋼鐵業、砂石業…，而在近幾年由於疫情的影響;戰爭的影響;原物料的影響，導致產業經濟變得十分不好，因此必須要更加正視產業經濟的旺季與淡季，並依照產業經濟的淡旺季，進行不同的產業規劃，而此可以參考到製造業正常時期的淡旺季時機與過去一年製造業景氣信號之燈號，並依此進行比較，可以進行人員調度、成本控管、產業規劃與危機應對!  因此這時如果配合上Calendar的運用，就能更加有計畫性的進行產業規劃。 |
| 四、問題定義  (自行設計問題，問題可列舉或敘述方式，自由模擬，以下為例子) |
| 1. 我爸爸是製造業相關產業者，因此他需要知道產業本身的淡旺季與過去一年所對應2022年6月~2023年6月的景氣訊號燈指示的淡旺季，進行產業規劃與應對。 2. 製造業正常時期的旺季為第二季、第四季:4月~6月和10月~12月 3. 製造業正常時期的淡季為第一季、第三季:1月~3月和7月~9月 4. 製造業景氣信號之燈號為”繁榮”為:無 5. 製造業景氣信號之燈號為”揚升”為:無 6. 製造業景氣信號之燈號為”持平”為:無 7. 製造業景氣信號之燈號為”低迷”為:2022年6、7、8、9、10月和2023年2月 8. 製造業景氣信號之燈號為”衰退”為: 2022年11、12月和2023年1、3、4、5、6月  |  |  | | --- | --- | | 旺季 | 4月~6月和10月~12月 | | 淡季 | 1月~3月和7月~9月 | | 繁榮燈號 | 無 | | 揚升燈號 | 無 | | 持平燈號 | 無 | | 低迷燈號 | 2022年6、7、8、9、10月和2023年2月 | | 衰退燈號 | 2022年11、12月和2023年1、3、4、5、6月 | |
| 五、程式碼設計  (將執行的程式碼貼入並註解，綜合運用所學，以下為例子) |
| 1. import calendar  2. import datetime  3. print('對於製造業分析')  4. now = datetime.datetime.now()  5. print("\n目前時間為", now.strftime("%Y-%m-%d"))  6. print('---------------------------------------')  7. #製造業正常時期的"旺季"為第二季、第四季  8. print('製造業正常時期的"旺季"為第二季、第四季:')  9. print('分別是:2023年 4月~6月和10月~12月')  10. print('第二季')  11. print(calendar.month(2023,4))  12. print(calendar.month(2023,5))  13. print(calendar.month(2023,6))  14. print('第四季')  15. print(calendar.month(2023,10))  16. print(calendar.month(2023,11))  17. print(calendar.month(2023,12))  18. print('---------------------------------------')  19. #製造業正常時期的淡季為第一季、第三季  20. print('製造業正常時期的淡季為第一季、第三季:')  21. print('分別是:2023年 1月~3月和7月~9月')  22. print('第一季')  23. print(calendar.month(2023,1))  24. print(calendar.month(2023,2))  25. print(calendar.month(2023,3))  26. print('第三季')  27. print(calendar.month(2023,7))  28. print(calendar.month(2023,8))  29. print(calendar.month(2023,9))  30. print('---------------------------------------')  31. #製造業景氣信號之燈號為”繁榮”  32. print('製造業景氣信號之燈號為”繁榮”為:無')  33. print('---------------------------------------')  34. #製造業景氣信號之燈號為”揚升”  35. print('製造業景氣信號之燈號為”揚升”為:無')  36. print('---------------------------------------')  37. #製造業景氣信號之燈號為”持平”  38. print('製造業景氣信號之燈號為”持平”為:無')  39. print('---------------------------------------')  40. #製造業景氣信號之燈號為”低迷”  41. print('製造業景氣信號之燈號為”低迷”為:\n2022年6、7、8、9、10月和2023年2月')  42. print('2022')  43. print(calendar.month(2022,6))  44. print(calendar.month(2022,7))  45. print(calendar.month(2022,8))  46. print(calendar.month(2022,9))  47. print(calendar.month(2022,10))  48. print('2023')  49. print(calendar.month(2023,2))  50. print('---------------------------------------')  51. #製造業景氣信號之燈號為”衰退”  52. print('製造業景氣信號之燈號為”衰退”為:\n2022年11、12月和2023年1、3、4、5、6月')  53. print('2022')  54. print(calendar.month(2022,11))  55. print(calendar.month(2022,12))  56. print('2023')  57. print(calendar.month(2023,1))  58. print(calendar.month(2023,3))  59. print(calendar.month(2023,4))  60. print(calendar.month(2023,5))  61. print(calendar.month(2023,6))  62. print('---------------------------------------')  63. |
| 六、執行結果  (將正確執行的結果貼入，文字為主截圖為輔，以下為例子) |
| 將以下幾點的內容，可以透過選擇進行分析，並依序顯示結果:   1. 我爸爸是製造業相關產業者，因此他需要知道產業本身的淡旺季與過去一年所對應2022年6月~2023年6月的景氣訊號燈指示的淡旺季，進行產業規劃與應對。 2. 製造業正常時期的旺季為第二季、第四季:4月~6月和10月~12月 3. 製造業正常時期的淡季為第一季、第三季:1月~3月和7月~9月 4. 製造業景氣信號之燈號為”繁榮”為:無 5. 製造業景氣信號之燈號為”揚升”為:無 6. 製造業景氣信號之燈號為”持平”為:無 7. 製造業景氣信號之燈號為”低迷”為:2022年6、7、8、9、10月和2023年2月 8. 製造業景氣信號之燈號為”衰退”為: 2022年11、12月和2023年1、3、4、5、6月   (下一頁圖片) |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  |  |  |  | |  | |  |  |  |  |  | | |
| 七、學習意涵詮釋  (可分為兩個部分：一、學到什麼程式語法 二、學到解決什麼問題，以下為例子) |
| 1. 學會如何運用 calendar與datetime套件。 2. py檔名不可與套件名相同。 3. 會利用import 匯入日曆外部套件來列印月曆，為製造業之產業進行產業規劃與應對! 4. 學會去分析相關產業的淡旺季與景氣燈之調查，並學會基礎python用法並了解python與C++之不同。 |
| 八、參考說明  (列出所參考資料並說明如何應用或修改其內容，修改ChatGPT亦可，以下舉例) |
| 1.台灣經濟研究院，製造業個別產業景氣燈:<https://www.tier.org.tw/forecast/obs1000.aspx>  2.各產業淡旺季: <https://blacknguyeeja.pixnet.net/blog/post/332280631>  透過以上兩項資料可以幫助分析產業的淡旺季與景氣燈之調查，並配合「calendar」「datetime」，為製造業之產業進行產業規劃與應對！ |

|  |
| --- |
| 教師：曾智義 課程名稱：基礎程式設計 學期：112-1 |
| 一、第03週課堂練習 |
| 基本運算, 變數, 數值, 字串, 比較的運用 |
| 二、個人/成員 |
| 林彧頎\_資工系\_A1105505 |
| 三、議題規劃 |
| 在如今時代發展得愈快速，生活腳步愈來愈快，當人愈在意自身的工作時，往往會不小心忽略了自己的身體，人的健康所需除了六大營養以外，在衛生福利部國民健康署的建議中，其中有兩項分別為「我的餐盤」與「每日飲食建議量(含每日熱量建議)」，以上數據可以幫助人，對於每日的飲食需求做每日的審視，並且依照自己每日的身體數據，可以進行相對應的調整，因此我將設計出一個每日列表，記錄我的每日的餐盤建議與每日飲食建議。  (我也將在設計中模擬自身的數據)       |  |  | | --- | --- | | 每天早晚一杯奶 | 每天早上和晚上要各喝一杯牛奶（約240毫升），來補充蛋白質和國人十分缺乏的鈣質。自助餐店比較少在賣鮮奶，所以記得一整天要喝兩杯牛奶，才會更均衡！ | | 每餐水果拳頭大 | 每一餐都要吃到一個拳頭大的水果。自助餐店若沒有提供水果的話，別忘了在餐間或餐後再吃些水果！選擇各類多種顏色的水果，以攝取豐富的膳食纖維、維生素C和有抗氧化功能的植化素。 | | 菜比水果多一點 | 每餐吃的蔬菜量要比一個拳頭多一些。在自助餐店可以選擇多種顏色不同的蔬菜（如炒空心菜、玉米筍、滷紅蘿蔔、海帶等），也可以夾一些和蔬菜一起烹調的主菜（如洋蔥豬柳、匏瓜炒肉片等）來達到所需的蔬菜量，並攝取到各種不同顏色蔬菜所含的植化素。 | | 飯跟蔬菜一樣多 | 飯和蔬菜吃的份量差不多。在自助餐多選擇糙米、紫米或五穀飯等全穀雜糧，可以補充到更多的膳食纖維與其他營養素。 | | 豆魚蛋肉一掌心 | 每餐的蛋白質食物，大約要吃到一個掌心的大小。在自助餐店可以夾一些烹調方式較簡單清淡的豆類（如醬燒豆干、嫩豆腐、炒毛豆仁等）、海鮮類（如炒花枝、烤鯖魚等）、蛋類（如蒸蛋、番茄炒蛋等）或肉類（如煎豬里肌、白斬雞等）。記得盡量別選擇油炸的主菜哦！ | | 堅果種子一茶匙 | 每一餐的堅果種子類，約是一茶匙或一個拇指指節的大小；也可以一次吃掉一整天的建議量，約是一湯匙或一個拇指的大小。可以自己準備無調味的堅果在餐間肚子餓時吃，營養會更加均衡！ | |
| 四、問題定義  (自行設計問題，問題可列舉或敘述方式，自由模擬，以下為例子) |
| 1. 需要透過撰寫熱量對應的卡路里以方便計算，此可以幫助未來做其他的熱量數據進行操作 2. 需要計算我自身的熱量 3. 並依照我的熱量，寫出相對應的每日飲食建議與每日餐盤建議 4. 並計算我一天總共所食用的食物總數量 |
| 五、程式碼設計  (將執行的程式碼貼入並註解，綜合運用所學，以下為例子) |
| 1. import math  2. #工作量、體重(變數定義)  3. low\_low\_kcal=35  4. mid\_low\_kcal=40  5. high\_low\_kcal=45  6.  7. low\_mid\_kcal=30  8. mid\_mid\_kcal=35  9. high\_mid\_kcal=40  10.  11. low\_high\_kcal=25  12. mid\_high\_kcal=30  13. high\_high\_kcal=35  14. #每日熱量  15. print('每日熱量建議')  16. #體重過輕者  17. print('-----------------')  18. print('體重過輕者所需熱量:')  19. print('輕度工作: '+str(low\_low\_kcal)+' kcal')  20. print('中度工作: '+str(mid\_low\_kcal)+' kcal')  21. print('重度工作: '+str(high\_low\_kcal)+' kcal')  22. #體重正常者  23. print('-----------------')  24. print('體重正常者所需熱量:')  25. print('輕度工作: '+str(low\_mid\_kcal)+' kcal')  26. print('中度工作: '+str(mid\_mid\_kcal)+' kcal')  27. print('重度工作: '+str(high\_mid\_kcal)+' kcal')  28. #體重過重者  29. print('-----------------')  30. print('體重過重者所需熱量:')  31. print('輕度工作: '+str(low\_high\_kcal)+' kcal')  32. print('中度工作: '+str(mid\_high\_kcal)+' kcal')  33. print('重度工作: '+str(high\_high\_kcal)+' kcal')  34. print('-----------------')  35. #顯示我的熱量計算  36. print('我的體重:49kg')  37. w=49  38. print('我的身高:158cm')  39. h=158  40. print('我的工作量:正常')  41. print('我的bmi:正常')  42. print('我所需的熱量:'+str(mid\_mid\_kcal\*w))  43. print('-----------------')  44. #顯示我的每日飲食建議量  45. print('請參考以下\n每日飲食建議量')  46. #使用變數，可幫助之後有其他大卡的數據可以直接修正  47. a=3  48. b=5  49. c=1.5  50. d=3  51. e=2  52. f=5  53. print('由於我是1715大卡\n(因此我大約需要1800大卡):')  54. print('全榖雜糧類:'+str(a)+'份')  55. print('豆魚蛋肉類:'+str(b)+'份')  56. print('乳品類 :'+str(c)+'份')  57. print('蔬菜類 :'+str(d)+'份')  58. print('水果類 :'+str(e)+'份')  59. print('油脂類 :'+str(f)+'份')  60. print('total :'+str(a+b+c+d+e+f)+'份')  61. print('-----------------')  62. print('以下為每日餐盤建議')  63. print('每天早晚一杯奶\n')  64. print('每餐水果拳頭大\n')  65. print('菜比水果多一點\n')  66. print('飯跟蔬菜一樣多\n')  67. print('豆魚蛋肉一掌心\n')  68. print('堅果種子一茶匙\n') |
| 六、執行結果  (將正確執行的結果貼入，文字為主截圖為輔，以下為例子) |
| |  |  |  | | --- | --- | --- | | 結果的流程為:  1.顯示由變數、數字、字串產生的我的餐盤  2.將我的身體數據計算後，顯示每日熱量建議  3.顯示每日飲食建議  (執行結果圖於下一頁) |  |  | | (下一頁繼續) |  |  | |
| 七、學習意涵詮釋  (可分為兩個部分：一、學到什麼程式語法 二、學到解決什麼問題，以下為例子) |
| 1.學會使用符號的使用，有規定的打不能有所缺漏。  2.注意空格做排版(不能依照自己的喜好，乘是會有規定的內縮等等…，否則會SyntaxError: invalid syntax : 語法錯誤的error。)  3. 養成資料和運算子之間隔一個空格的習慣。  4.學會如何使用數學運算(學會如何運用 math套件(三角函數、圓周率)。)  5.數學運算的幾個特殊用法: 餘數用%、次方用\*\*、取整數的除法(\\)。  6.變數須注意，變數名稱=內容值(變數第1個文字不能是數字，不能使用保留字)。  7. 學會如何查保留字(保留字不可作為變數；>>> import keyword >>> keyword.kwlist)。  8. Python有int、float、complex(複數) ，而虛數以j或J表達  9. 3個連續的'或"做多行字串輸入  10. \*號用在重複字串(不能字串乘字串)  11. +號用在合併字串  12. 字串型別相關的函數:upper(可以變大寫)、lower(小寫)、count(計算出現單字的次數)  13. 資料內有「”」外圍用「’」包覆  14. 布林值:True、False(要寫成這樣)  15.直接將整數變數放前+字串變數印會抱錯，要加上str(int變數) |
| 八、參考說明  (列出所參考資料並說明如何應用或修改其內容，修改ChatGPT亦可，以下舉例) |
| [每日餐盤-衛生福利部國民健康署](https://www.hpa.gov.tw/Pages/Detail.aspx?nodeid=4687&pid=11680)  <https://www.hpa.gov.tw/Pages/Detail.aspx?nodeid=4687&pid=11680>  每日飲食建議與熱量建議-衛生福利部國民健康署  <https://www.hpa.gov.tw/Pages/Detail.aspx?nodeid=544&pid=728> |

(下一頁繼續)

|  |
| --- |
| 教師：曾智義 課程名稱：基礎程式設計 學期：112-1 |
| 一、第04週課堂練習 |
| 綜合運用至目前所學(基本運算、變數、字串、List、Dict） |
| 二、個人/成員 |
| 林彧頎\_資工系\_A1105505 |
| 三、議題規劃 |
| 在如今時代發展得愈快速，生活腳步愈來愈快，當人愈在意自身的工作時，往往會不小心忽略了自己的身體，人的健康所需除了六大營養以外，在衛生福利部國民健康署的建議中，其中有兩項分別為「我的餐盤」與「每日飲食建議量(含每日熱量建議)」，以上數據可以幫助人，對於每日的飲食需求做每日的審視，並且依照自己每日的身體數據，可以進行相對應的調整，因此我將設計出一個每日列表，記錄我的每日的餐盤建議與每日飲食建議。  我將透過結合上週所得出的熱量計算，搭配食物分配，幫助人由於想每日要吃什麼而煩惱時，可以有更營養且符合自己熱量所需的選擇。  (我也將在設計中模擬自身的數據)       |  |  | | --- | --- | | 每天早晚一杯奶 | 每天早上和晚上要各喝一杯牛奶（約240毫升），來補充蛋白質和國人十分缺乏的鈣質。自助餐店比較少在賣鮮奶，所以記得一整天要喝兩杯牛奶，才會更均衡！ | | 每餐水果拳頭大 | 每一餐都要吃到一個拳頭大的水果。自助餐店若沒有提供水果的話，別忘了在餐間或餐後再吃些水果！選擇各類多種顏色的水果，以攝取豐富的膳食纖維、維生素C和有抗氧化功能的植化素。 | | 菜比水果多一點 | 每餐吃的蔬菜量要比一個拳頭多一些。在自助餐店可以選擇多種顏色不同的蔬菜（如炒空心菜、玉米筍、滷紅蘿蔔、海帶等），也可以夾一些和蔬菜一起烹調的主菜（如洋蔥豬柳、匏瓜炒肉片等）來達到所需的蔬菜量，並攝取到各種不同顏色蔬菜所含的植化素。 | | 飯跟蔬菜一樣多 | 飯和蔬菜吃的份量差不多。在自助餐多選擇糙米、紫米或五穀飯等全穀雜糧，可以補充到更多的膳食纖維與其他營養素。 | | 豆魚蛋肉一掌心 | 每餐的蛋白質食物，大約要吃到一個掌心的大小。在自助餐店可以夾一些烹調方式較簡單清淡的豆類（如醬燒豆干、嫩豆腐、炒毛豆仁等）、海鮮類（如炒花枝、烤鯖魚等）、蛋類（如蒸蛋、番茄炒蛋等）或肉類（如煎豬里肌、白斬雞等）。記得盡量別選擇油炸的主菜哦！ | | 堅果種子一茶匙 | 每一餐的堅果種子類，約是一茶匙或一個拇指指節的大小；也可以一次吃掉一整天的建議量，約是一湯匙或一個拇指的大小。可以自己準備無調味的堅果在餐間肚子餓時吃，營養會更加均衡！ | |
| 四、問題定義  (自行設計問題，問題可列舉或敘述方式，自由模擬，以下為例子) |
| 1. 需要透過撰寫熱量對應的卡路里以方便計算，此可以幫助未來做其他的熱量數據進行操作(使用dict結合list的方法) 2. 需要計算我自身的熱量 3. 並依照我的熱量，寫出相對應的每日飲食建議與每日餐盤建議 4. 並計算我一天建議總共所食用的食物總數量 5. 列出我可能的飲食規劃表，並對應是否有符合上述計算出的選項 6. 顯示結果 |
| 五、程式碼設計  (將執行的程式碼貼入並註解，綜合運用所學，以下為例子) |
| 1. import math  2. #工作量、體重(字典定義)  3. dict\_kcal={('體重過輕者','輕度工作'):35,  4. ('體重正常者','輕度工作'):30,  5. ('體重過重者','輕度工作'):25,  6.  7. ('體重過輕者','中度工作'):40,  8. ('體重正常者','中度工作'):35,  9. ('體重過重者','中度工作'):30,  10.  11. ('體重過輕者','重度工作'):45,  12. ('體重正常者','重度工作'):40,  13. ('體重過重者','重度工作'):35}  14.  15. #每日熱量  16. print('每日熱量建議')  17. #字典輸出  18. #體重過輕者  19. print('-----------------')  20. print('體重過輕者所需熱量:')  21. print('輕度工作: '+str(dict\_kcal[('體重過輕者','輕度工作')])+' kcal')  22. print('中度工作: '+str(dict\_kcal[('體重過輕者','中度工作')])+' kcal')  23. print('重度工作: '+str(dict\_kcal[('體重過輕者','重度工作')])+' kcal')  24. #體重正常者  25. print('-----------------')  26. print('體重正常者所需熱量:')  27. print('輕度工作: '+str(dict\_kcal[('體重正常者','輕度工作')])+' kcal')  28. print('中度工作: '+str(dict\_kcal[('體重正常者','中度工作')])+' kcal')  29. print('重度工作: '+str(dict\_kcal[('體重正常者','重度工作')])+' kcal')  30. #體重過重者  31. print('-----------------')  32. print('體重過重者所需熱量:')  33. print('輕度工作: '+str(dict\_kcal[('體重過重者','輕度工作')])+' kcal')  34. print('中度工作: '+str(dict\_kcal[('體重過重者','中度工作')])+' kcal')  35. print('重度工作: '+str(dict\_kcal[('體重過重者','重度工作')])+' kcal')  36. print('-----------------')  37. #顯示我的熱量計算  38. print('我的體重:49kg')  39. w=49  40. print('我的身高:158cm')  41. h=158  42. print('我的工作量:正常')  43. print('我的bmi:正常')  44. print('我所需的熱量:'+str(dict\_kcal[('體重正常者','中度工作')]\*w))  45. print('-----------------')  46. #顯示我的每日飲食建議量  47. print('請參考以下\n每日飲食建議量')  48. #使用變數，可幫助之後有其他大卡的數據可以直接修正  49. a=3  50. b=5  51. c=1.5  52. d=3  53. e=2  54. f=5  55. print('由於我是1715大卡\n(因此我大約需要1800大卡):')  56. print('全榖雜糧類:'+str(a)+'份')  57. print('豆魚蛋肉類:'+str(b)+'份')  58. print('乳品類 :'+str(c)+'份')  59. print('蔬菜類 :'+str(d)+'份')  60. print('水果類 :'+str(e)+'份')  61. print('油脂類 :'+str(f)+'份')  62. print('total :'+str(a+b+c+d+e+f)+'份')  63. print('-----------------')  64. print('以下為每日餐盤建議')  65. print('每天早晚一杯奶\n')  66. print('每餐水果拳頭大\n')  67. print('菜比水果多一點\n')  68. print('飯跟蔬菜一樣多\n')  69. print('豆魚蛋肉一掌心\n')  70. print('堅果種子一茶匙\n')  71. print('-----------------')  72. print('以下為我的食物組合:')  73. print('請選擇最適合的~')  74. #食物組合與熱量(使用字典與list)  75. #list(名稱,份,大卡)  76. 全榖雜糧類1=['飯',3,840]  77. 全榖雜糧類2=['飯',2,560]  78.  79. 豆魚蛋肉類1=['肉',4,160]  80. 豆魚蛋肉類2=['肉',5,200]  81.  82. 乳品類1=['牛奶',1.5,225]  83. 乳品類2=['牛奶',2,300]  84.  85. 蔬菜類1=['菜',1,25]  86. 蔬菜類2=['菜',3,75]  87.  88. 水果類1=['水果',1,60]  89. 水果類2=['水果',2,120]  90.  91. 油脂類1=['油',4,240]  92. 油脂類2=['油',5,300]  93.  94. food\_set1={'全榖雜糧類':全榖雜糧類1,'豆魚蛋肉類':豆魚蛋肉類2,'乳品類':乳品類1,  95. '蔬菜類':蔬菜類2,'水果類':水果類2,'油脂類':油脂類2}  96. food\_set2={'全榖雜糧類':全榖雜糧類2,'豆魚蛋肉類':豆魚蛋肉類1,'乳品類':乳品類2,  97. '蔬菜類':蔬菜類1,'水果類':水果類1,'油脂類':油脂類1}  98. #印出結果並自行比較與選擇  99. print('食物組合1\n(名稱、份數、熱量(大卡))')  100. print(food\_set1.values())  101. print('總熱量:'+str(food\_set1['全榖雜糧類'][2]+food\_set1['豆魚蛋肉類'][2]+  102. food\_set1['蔬菜類'][2]+food\_set1['乳品類'][2]+  103. food\_set1['水果類'][2]+food\_set1['油脂類'][2])+'kcal')  104.  105. print('食物組合2\n(名稱、份數、熱量(大卡))')  106. print(food\_set2.values())  107. print('總熱量:'+str(food\_set2['全榖雜糧類'][2]+food\_set2['豆魚蛋肉類'][2]+  108. food\_set2['蔬菜類'][2]+food\_set2['乳品類'][2]+  109. food\_set2['水果類'][2]+food\_set2['油脂類'][2])+'kcal')  110. |
| 六、執行結果  (將正確執行的結果貼入，文字為主截圖為輔，以下為例子) |
| |  |  |  | | --- | --- | --- | |  |  |  | | 由以上的結果可以得出，我們的熱量計算是中度工作、正常體重，可以得出  所需的熱量為1715大卡，而在參考每日餐盤與飲食建議後，我們的食物組合有兩種，  分別對應的是:食物組合1為1760大卡，食物組合2為1345大卡，由此可知最適合  我的食物組合為食物組合1(1760大卡(近1715大卡))。 |  |  | |
| 七、學習意涵詮釋  (可分為兩個部分：一、學到什麼程式語法 二、學到解決什麼問題，以下為例子) |
| 1.直接將整數變數放前+字串變數印會抱錯，要加上str(int變數)  2.dictionary相關應用  3. 串列型別（List）：[]; 逗號區隔(可以匯集不同資料型別)  4.list或dict可以幫助解決需要宣告太多變數的問題: 可以對同一群的作個別的處理，而其他的統整資料也可以一併處裡，而不會需要單獨抓一點點的個體作個別處理。  5. append(): 在串列中加入新元素  6. remove(): 在串列中刪除元素  7. sort():改變串列順序(記得sort只能同時是字串或同時是數字，不能同時是字串又是數字)  8. 字典型別（Dictionary Type）<利用key找到資料>  9. key或value可以是任何形態甚至是list  10. List與Dictionary結合使用  11.我在過程中學會使用更好的方法做資料處理  12.我透過此次的作業，我對於程式開發與結合議題有更好的認識，我本來只是想開發一個簡單的系統，但隨著上周的作業，我在這周想到或許可以結合人的健康需求來配合自身的飲食照護~ |
| 八、參考說明  (列出所參考資料並說明如何應用或修改其內容，修改ChatGPT亦可，以下舉例) |
| 每日餐盤-衛生福利部國民健康署  <https://www.hpa.gov.tw/Pages/Detail.aspx?nodeid=4687&pid=11680>  每日飲食建議與熱量建議-衛生福利部國民健康署  <https://www.hpa.gov.tw/Pages/Detail.aspx?nodeid=544&pid=728> |

**(下一頁繼續)**

|  |
| --- |
| 教師：曾智義 課程名稱：基礎程式設計 學期：112-1 |
| 一、第06週課堂練習 |
| 綜合運用至目前所學(基本運算、變數、字串、List、Dict、Tuple、Set） |
| 二、個人/成員 |
| 林彧頎\_資工系\_A1105505 |
| 三、議題規劃 |
| 在如今時代發展得愈快速，生活腳步愈來愈快，當人愈在意自身的工作時，往往會不小心忽略了自己的身體，人的健康所需除了六大營養以外，在衛生福利部國民健康署的建議中，其中有兩項分別為「我的餐盤」與「每日飲食建議量(含每日熱量建議)」，以上數據可以幫助人，對於每日的飲食需求做每日的審視，並且依照自己每日的身體數據，可以進行相對應的調整，因此我將設計出一個每日列表，記錄我的每日的餐盤建議與每日飲食建議。  我將透過結合W3所得出的熱量計算，搭配食物分配，幫助人由於想每日要吃什麼而煩惱時，可以有更營養且符合自己熱量所需的選擇。  在W6中我將結合前面所做的成果，針對每日飲食進行多樣化的改善，我將新增我的飲食列表，並建立一個三餐的飲食編排，這將會使我每日對於飲食的多樣性會有更不一樣的改變，並且又能同時兼顧飲食健康。  (我也將在設計中模擬自身的數據)       |  |  | | --- | --- | | 每天早晚一杯奶 | 每天早上和晚上要各喝一杯牛奶（約240毫升），來補充蛋白質和國人十分缺乏的鈣質。自助餐店比較少在賣鮮奶，所以記得一整天要喝兩杯牛奶，才會更均衡！ | | 每餐水果拳頭大 | 每一餐都要吃到一個拳頭大的水果。自助餐店若沒有提供水果的話，別忘了在餐間或餐後再吃些水果！選擇各類多種顏色的水果，以攝取豐富的膳食纖維、維生素C和有抗氧化功能的植化素。 | | 菜比水果多一點 | 每餐吃的蔬菜量要比一個拳頭多一些。在自助餐店可以選擇多種顏色不同的蔬菜（如炒空心菜、玉米筍、滷紅蘿蔔、海帶等），也可以夾一些和蔬菜一起烹調的主菜（如洋蔥豬柳、匏瓜炒肉片等）來達到所需的蔬菜量，並攝取到各種不同顏色蔬菜所含的植化素。 | | 飯跟蔬菜一樣多 | 飯和蔬菜吃的份量差不多。在自助餐多選擇糙米、紫米或五穀飯等全穀雜糧，可以補充到更多的膳食纖維與其他營養素。 | | 豆魚蛋肉一掌心 | 每餐的蛋白質食物，大約要吃到一個掌心的大小。在自助餐店可以夾一些烹調方式較簡單清淡的豆類（如醬燒豆干、嫩豆腐、炒毛豆仁等）、海鮮類（如炒花枝、烤鯖魚等）、蛋類（如蒸蛋、番茄炒蛋等）或肉類（如煎豬里肌、白斬雞等）。記得盡量別選擇油炸的主菜哦！ | | 堅果種子一茶匙 | 每一餐的堅果種子類，約是一茶匙或一個拇指指節的大小；也可以一次吃掉一整天的建議量，約是一湯匙或一個拇指的大小。可以自己準備無調味的堅果在餐間肚子餓時吃，營養會更加均衡！ | |
| 四、問題定義  (自行設計問題，問題可列舉或敘述方式，自由模擬，以下為例子) |
| 目標一   1. 需要透過撰寫熱量對應的卡路里以方便計算，此可以幫助未來做其他的熱量數據進行操作(使用dict結合list的方法) 2. 需要計算我自身的熱量 3. 並依照我的熱量，寫出相對應的每日飲食建議與每日餐盤建議 4. 並計算我一天建議總共所食用的食物總數量 5. 列出我可能的飲食規劃表，並對應是否有符合上述計算出的選項 6. 顯示結果   目標二   1. 製作三餐飲食規劃 2. 使用tuple與set 3. 使三餐的飲食不僅僅是符合熱量需求的，並且可以隨每一次重新啟動程式，皆有不同的三餐飲食組合 |
| 五、程式碼設計  (將執行的程式碼貼入並註解，綜合運用所學，以下為例子) |
| 1. import math  2. #工作量、體重(字典定義)  3. #將key改為tuple(m(中度),l(輕度),h(重度))  4. #體重(l),工作(l)  5. ll=('體重過輕者','輕度工作')  6. ml=('體重正常者','輕度工作')  7. hl=('體重過重者','輕度工作')  8.  9. lm=('體重過輕者','中度工作')  10. mm=('體重正常者','中度工作')  11. hm=('體重過重者','中度工作')  12.  13. lh=('體重過輕者','重度工作')  14. mh=('體重正常者','重度工作')  15. hh=('體重過重者','重度工作')  16.  17. dict\_kcal={ll:35,  18. ml:30,  19. hl:25,  20.  21. lm:40,  22. mm:35,  23. hm:30,  24.  25. lh:45,  26. mh:40,  27. hh:35}  28.  29. #每日熱量  30. print('每日熱量建議')  31. #字典輸出  32. #體重過輕者  33. print('-----------------')  34. print('體重過輕者所需熱量:')  35. print('輕度工作: '+str(dict\_kcal[ll])+' kcal')  36. print('中度工作: '+str(dict\_kcal[lm])+' kcal')  37. print('重度工作: '+str(dict\_kcal[lh])+' kcal')  38. #體重正常者  39. print('-----------------')  40. print('體重正常者所需熱量:')  41. print('輕度工作: '+str(dict\_kcal[ml])+' kcal')  42. print('中度工作: '+str(dict\_kcal[mm])+' kcal')  43. print('重度工作: '+str(dict\_kcal[mh])+' kcal')  44. #體重過重者  45. print('-----------------')  46. print('體重過重者所需熱量:')  47. print('輕度工作: '+str(dict\_kcal[hl])+' kcal')  48. print('中度工作: '+str(dict\_kcal[hm])+' kcal')  49. print('重度工作: '+str(dict\_kcal[hh])+' kcal')  50. print('-----------------')  51. #顯示我的熱量計算  52. print('我的體重:49kg')  53. w=49  54. print('我的身高:158cm')  55. h=158  56. print('我的工作量:正常')  57. print('我的bmi:正常')  58. print('我所需的熱量:'+str(dict\_kcal[mm]\*w))  59. print('-----------------')  60. #顯示我的每日飲食建議量  61. print('請參考以下\n每日飲食建議量')  62. #使用變數，可幫助之後有其他大卡的數據可以直接修正  63. a=3  64. b=5  65. c=1.5  66. d=3  67. e=2  68. f=5  69. print('由於我是1715大卡\n(因此我大約需要1800大卡):')  70. print('全榖雜糧類:'+str(a)+'份')  71. print('豆魚蛋肉類:'+str(b)+'份')  72. print('乳品類 :'+str(c)+'份')  73. print('蔬菜類 :'+str(d)+'份')  74. print('水果類 :'+str(e)+'份')  75. print('油脂類 :'+str(f)+'份')  76. print('total :'+str(a+b+c+d+e+f)+'份')  77. print('-----------------')  78. print('以下為每日餐盤建議')  79. print('每天早晚一杯奶\n')  80. print('每餐水果拳頭大\n')  81. print('菜比水果多一點\n')  82. print('飯跟蔬菜一樣多\n')  83. print('豆魚蛋肉一掌心\n')  84. print('堅果種子一茶匙\n')  85. print('-----------------')  86. print('以下為我的食物組合:')  87. print('請選擇最適合的~')  88. #食物組合與熱量(使用字典與list)  89. #list(名稱,份,大卡)  90. 全榖雜糧類1=['飯',3,840]  91. 全榖雜糧類2=['飯',2,560]  92.  93. 豆魚蛋肉類1=['肉',4,160]  94. 豆魚蛋肉類2=['肉',5,200]  95.  96. 乳品類1=['牛奶',1.5,225]  97. 乳品類2=['牛奶',2,300]  98.  99. 蔬菜類1=['菜',1,25]  100. 蔬菜類2=['菜',3,75]  101.  102. 水果類1=['水果',1,60]  103. 水果類2=['水果',2,120]  104.  105. 油脂類1=['油',4,240]  106. 油脂類2=['油',5,300]  107.  108. food\_set1={'全榖雜糧類':全榖雜糧類1,'豆魚蛋肉類':豆魚蛋肉類2,'乳品類':乳品類1,  109. '蔬菜類':蔬菜類2,'水果類':水果類2,'油脂類':油脂類1}  110. food\_set2={'全榖雜糧類':全榖雜糧類2,'豆魚蛋肉類':豆魚蛋肉類1,'乳品類':乳品類2,  111. '蔬菜類':蔬菜類1,'水果類':水果類1,'油脂類':油脂類2}  112. #印出結果並自行比較與選擇  113. print('食物組合1\n(名稱、份數、熱量(大卡))')  114. print(food\_set1.values())  115. print('總熱量:'+str(food\_set1['全榖雜糧類'][2]+food\_set1['豆魚蛋肉類'][2]+  116. food\_set1['蔬菜類'][2]+food\_set1['乳品類'][2]+  117. food\_set1['水果類'][2]+food\_set1['油脂類'][2])+'kcal')  118.  119. print('食物組合2\n(名稱、份數、熱量(大卡))')  120. print(food\_set2.values())  121. print('總熱量:'+str(food\_set2['全榖雜糧類'][2]+food\_set2['豆魚蛋肉類'][2]+  122. food\_set2['蔬菜類'][2]+food\_set2['乳品類'][2]+  123. food\_set2['水果類'][2]+food\_set2['油脂類'][2])+'kcal')  124. print('-----------------')  125. print('3餐食物隨機組合:')  126. #list與set的使用(並同時做早餐的重新排序)  127. list\_rice=['白飯','麵線']  128. set\_rice=set(list\_rice)  129. #測試update  130. set\_rice.update(['白飯'])  131. set\_rice.update(['紫米飯'])  132. set\_rice.update(['五穀雜糧飯'])  133. list\_rice=list(set\_rice)  134. #print(list\_rice) #測試是否隨機產生(是)  135.  136. list\_meat=['炒豬肉絲','煎魚','雞肉丁','丁香魚','肉丸','扁食']  137. set\_meat=set(list\_meat)  138. list\_meat=list(set\_meat)  139.  140. list\_milk=['牛奶','羊奶','起司','優格','乳酪蛋糕']  141. set\_milk=set(list\_milk)  142. list\_milk=list(set\_milk)  143.  144. list\_veg=['炒高麗菜','燙地瓜葉','苦瓜鹹蛋','炒A菜','炒筍子']  145. set\_veg=set(list\_veg)  146. list\_veg=list(set\_veg)  147.  148. list\_fruit=['蘋果','西瓜','奇異果','香蕉','蓮霧','水蜜桃']  149. set\_fruit=set(list\_fruit)  150. list\_fruit=list(set\_fruit)  151.  152. list\_oil=['酪梨','堅果','橄欖油','玄米油']  153. set\_oil=set(list\_oil)  154. list\_oil=list(set\_oil)  155.  156.  157. tuple\_breakfast=('早餐','400卡')  158. tuple\_lunch=('中餐','400卡')  159. tuple\_dinner=('晚餐','400卡')  160.  161. dict\_meals={}  162.  163. dict\_meals[tuple\_breakfast]=(list\_rice[0],list\_meat[1],list\_milk[2],list\_veg[0],list\_fruit[1],list\_oil[2])  164. print('早餐:'+str(dict\_meals[tuple\_breakfast]))  165. setb={list\_rice[0],list\_meat[1],list\_milk[2],list\_veg[0],list\_fruit[1],list\_oil[2]}  166.  167. #再次重新排序(中餐)  168. set\_rice=set(list\_rice)  169. list\_rice=list(set\_rice)  170.  171. set\_meat=set(list\_meat)  172. list\_meat=list(set\_meat)  173.  174. set\_milk=set(list\_milk)  175. list\_milk=list(set\_milk)  176.  177. set\_veg=set(list\_veg)  178. list\_veg=list(set\_veg)  179.  180. set\_fruit=set(list\_fruit)  181. list\_fruit=list(set\_fruit)  182.  183. set\_oil=set(list\_oil)  184. list\_oil=list(set\_oil)  185.  186. dict\_meals[tuple\_lunch]=(list\_rice[1],list\_meat[2],list\_milk[0],list\_veg[1],list\_fruit[2],list\_oil[0])  187. print('中餐:'+str(dict\_meals[tuple\_lunch]))  188. setl={list\_rice[1],list\_meat[2],list\_milk[0],list\_veg[1],list\_fruit[2],list\_oil[0]}  189.  190. #再次重新排序(晚餐)  191. set\_rice=set(list\_rice)  192. list\_rice=list(set\_rice)  193.  194. set\_meat=set(list\_meat)  195. list\_meat=list(set\_meat)  196.  197. set\_milk=set(list\_milk)  198. list\_milk=list(set\_milk)  199.  200. set\_veg=set(list\_veg)  201. list\_veg=list(set\_veg)  202.  203. set\_fruit=set(list\_fruit)  204. list\_fruit=list(set\_fruit)  205.  206. set\_oil=set(list\_oil)  207. list\_oil=list(set\_oil)  208.  209. dict\_meals[tuple\_dinner]=(list\_rice[2],list\_meat[1],list\_milk[0],list\_veg[2],list\_fruit[1],list\_oil[0])  210. setd={list\_rice[2],list\_meat[1],list\_milk[0],list\_veg[2],list\_fruit[1],list\_oil[0]}  211. #判斷set集合比較運算子  212. print('晚餐:'+str(dict\_meals[tuple\_dinner]))  213. print('早餐與中餐有重複嗎?',setb>=setl)  214. print('中餐與晚餐有重複嗎?',setd>=setl)  215. print('早餐與晚餐有重複嗎?',setb>=setd)  (下一頁繼續) |
| 六、執行結果  (將正確執行的結果貼入，文字為主截圖為輔，以下為例子) |
| |  |  |  | | --- | --- | --- | | (結果1)    (結果2) |  |  | | 由以上的結果可以得出，我們的熱量計算是中度工作、正常體重，可以得出  所需的熱量為1715大卡，而在參考每日餐盤與飲食建議後，我們的食物組合有兩種，  分別對應的是:食物組合1為1760大卡，食物組合2為1345大卡，由此可知最適合  我的食物組合為食物組合1(1760大卡(近1715大卡))。  並且於3餐的隨機組合中，我刻意讓程式碼跑兩次，  以展現每次程式碼執行時皆會有隨機的3餐組合，這是使用set與list的結合。  並且在本次的作業中，有將之前dictionary的值組key更正為tuple的型態。  最後我將它們以set分析集合的方法來判斷3餐是否有重複 |  |  | |
| 七、學習意涵詮釋  (可分為兩個部分：一、學到什麼程式語法 二、學到解決什麼問題，以下為例子) |
| 1.直接將整數變數放前+字串變數印會抱錯，要加上str(int變數)  2. Tuple Type: 多個元素組成一組  3. Tuple Type與List Type的差異: List Type:宣告後可以變更存放的資料->[]/ Tuple:宣告後不行更改->()  4. Tuple可以用來當Dictionary type 的key(因為Tuple值不能再被異動->Key為不能異動的值)(但其實可以透過將tuple轉list做修改再轉回tuple就可以修改了)  5. dict中list當value,而tuple當key  6.set是集合型別(candy={'a','b'}#(會打亂順序ex'a','b'變'b','a') )  7. set():轉set/ list():轉list/ tuple():轉tuple  8. set可以將list轉為set:1.字元不會打散(因為list)2.於update()裡用[]可以增加Set的資料  ->因此可以使用list來新增資料，再用set()來轉成Set，再以update([])來新增元素  9.set也可以做到刪除重複元素(set可以幫助list重複資料刪除)  10. 集合型別的運算(交集&聯集...)  11.我透過這次的學習，發現了python中可以用到許多種的 料處哩，並且這些功能之間還可以達到1+1大於2的結合效果，我不僅僅將每一種的使用方法，從過去的作業中不斷地延伸，也同時學習到更好的融會貫通他們之間的方法，聖致這項時做，讓我認為，飲食真的是人生活中不可或卻的一部份，我們需要透過每日飲食的攝取，才可以有更好的生活與健康! |
| 八、參考說明  (列出所參考資料並說明如何應用或修改其內容，修改ChatGPT亦可，以下舉例) |
| 每日餐盤-衛生福利部國民健康署  <https://www.hpa.gov.tw/Pages/Detail.aspx?nodeid=4687&pid=11680>  每日飲食建議與熱量建議-衛生福利部國民健康署  <https://www.hpa.gov.tw/Pages/Detail.aspx?nodeid=544&pid=728> |

|  |
| --- |
| 教師：曾智義 課程名稱：基礎程式設計 學期：112-1 |
| 一、第07週課堂練習 |
| 綜合運用至目前所學(基本運算、變數、字串、List、Dict、Tuple、Set、If、For、While、Def） |
| 二、個人/成員 |
| 林彧頎\_資工系\_A1105505(由於詢問過老師可以個人交，因此我們組是交個人作業) |
| 三、議題規劃 |
| 市面上有許多種的訂餐系統和額外的訂位系統，但卻沒有一個整合的訂餐與訂位系統，為此我想設計一款訂餐與訂位系統可以透過訂餐人員輸入相關資訊，並再由服務人員依照訂單需求進行座位劃位，讓使用者可以使用統合的方法於一個軟體中執行出完善的作業模式!  而本系統也針對使用者有一項額外的規劃，為了使用者的健康，使用者可以透過輸入自身的身高、體重，提供飲食建議，並推薦更符合他的健康飲食，不僅僅達到商業廣告功能，更可以使消費者的飲食健康提升。  針對員工端，本系統也新增員工資料管理，並建立簽到系統草圖，可對於未來規劃資料匯出的整合~ |
| 四、問題定義  (自行設計問題，問題可列舉或敘述方式，自由模擬，以下為例子) |
| 1. 需要透過資料輸入訂餐與訂位系統中所需的相關資料，包含:訂餐者、人員數、菜品、菜品數量、用餐時間。 2. 將上述的內容計算出總金額，與結果顯示給使用者，並模擬使用者付款。 3. 於服務人員端，可以透過上述的資訊，自動化的進行桌分配(人數過多時，需要幾個桌位) 4. 訂餐系統針對list、dictionary增加套餐功能(增加餐廳行銷與使用者選擇) 5. 使用者輸入自身身體數值(此針對List、Dict、Tuple、Set)，提供餐廳飲食建議 6. 針對服務人員端新增員工資料管理與簽到系統(未來可以輸出檔案) 7. 過程中需使用(基本運算、變數、字串、List、Dict、Tuple、Set、If、For、While、Def)進行邏輯判斷等等… |
| 五、程式碼設計  (將執行的程式碼貼入並註解，綜合運用所學，以下為例子) |
| 1. import math  2. import datetime  3. print('以下是一個訂餐訂位系統請依照指示執行!\n')  4. print('------------------------------------\n')  5. #使用者輸入相關資訊  6. print('welcome 日日香~\n')  7. name=input('請問您的大名:\n')  8. print('歡迎您~'+name+'\n')  9. people\_num=int(input('請問'+name+'會有幾位用餐呢?\n'))  10. test=True  11. #新增使用者輸入自身身體數值，提供餐廳飲食建議  12. print('-----------------------------------\n')  13. print('本系統可以提供您:輸入自身身體數值，提供餐廳飲食建議\n')  14. diet=int(input('請問需要測試嗎(要(1)/不要(0))?\n'))  15. user={}  16. #使用def定義飲食建議功能  17. def diet\_recommand():  18. p\_num=1  19. for i in range(people\_num):  20. #建tuple關於個人資料建立(name,birthday,date)  21. print('請輸入第'+str(p\_num)+'位的姓名、生日')  22. n=input('姓名')  23. b=input('生日:yyyy/mm/dd(請依照此格式)')  24. current\_time = datetime.datetime.now()  25. time\_now = current\_time.strftime('%Y/%m/%d')  26. key\_t=(n,b,time\_now)  27. print('請輸入第'+str(p\_num)+'位的體重(kg)與身高(m)\n')  28. p\_num=p\_num+1  29. weight=float(input('體重(kg)\n'))  30. height=float(input('身高(m)\n'))  31. bmi=float(weight/(height)\*\*2)  32. #將數據儲存於list  33. value\_l=['體重(kg)',weight,'身高(m)',height,'BMI',bmi]  34. #儲存到dict(key: tuple / value:list)  35. user[key\_t]=value\_l  36. #進行if else判斷  37. if(bmi<=18.5):  38. print('您的bmi為: '+str(bmi)+' 這表示您的體重過輕\n')  39. print('您可以任意點選我們餐廳的食物\n推薦您我們的招牌鍋貼喔~\n')  40. elif(bmi>=18.5 and bmi<24.0):  41. print('您的bmi為: '+str(bmi)+' 這表示您的體重正常\n')  42. print('您平時的飲食維持得不錯\n我推薦您可以搭配水餃與菜頭湯\n飲養又均衡~\n')  43. elif(bmi>=24.0):  44. print('您的bmi為: '+str(bmi)+' 這表示您的體重過重\n')  45. print('我不建議您喝紅茶，甜度太高，可以嘗試我們的無糖豆漿喔~\n')  46. #當問題得到1的回答，則執行飲食建議功能  47. if(diet==1):  48. diet\_recommand()  49. print('------------------------------------------------------------------------')  50. #新增套餐(list+dict應用)  51. m\_set1=[70,'鍋貼','紅茶']  52. m\_set2=[80,'水餃','紅茶']  53. menu={'鍋貼':50,'水餃':60,'菜頭湯':35,'紅茶':25,'豆漿':25,'套餐1':m\_set1,'套餐2':m\_set2}  54. order={'鍋貼':0,'水餃':0,'菜頭湯':0,'紅茶':0,'豆漿':0,'套餐1':0,'套餐2':0}  55. #使用者訂餐(while迴圈測試)  56. while(test):  57. print('好的~以下是我們的菜單\n')  58.  59. #dict的for 迴圈  60. for m in menu.items():  61. print(m)  62. choose=input('請問您想選擇?(請輸入菜名(或套餐名))')  63. print(choose)  64. t=True  65. #while進行偵錯是否輸入錯誤  66. while(not(choose =='鍋貼' or choose =='水餃' or choose =='菜頭湯' or choose =='紅茶' or choose =='豆漿'or choose =='套餐1'or choose =='套餐2')):  67. print('輸入錯誤請重新輸入\n') #使用者輸入錯誤例外處理  68. choose=input('請問您想選擇?(請輸入菜名(或套餐名))\n')  69. q=int(input('請問要幾份?'))  70. order[choose]=int(order[choose])+q  71. print('請問您還要繼續點嗎?\n')  72. keep=int(input('要(1)/不要(2)'))  73. #if判斷點餐是否繼續  74. if(keep==2):  75. test=False  76. #使用者訂餐結果與金額  77. print('以下是您的訂餐內容\n')  78. #dict的for 迴圈  79. for o in order.keys():  80. #if判斷  81. if(order[o]>0):  82. print(o+":"+str(order[o])+"份\n")  83. print('\n以下是您的訂餐總金額\n')  84. total=0  85. sum={'鍋貼':0,'水餃':0,'菜頭湯':0,'紅茶':0,'豆漿':0,'套餐1':0,'套餐2':0}  86. #dict的for 迴圈  87. for o in order.keys():  88. #if判斷  89. if(o!='套餐1' and o!='套餐2' and int(order[o])>0):  90. sum[o]=int(order[o])\*int(menu[o])  91. #新增套餐(list+dict應用) & elif判斷  92. elif((o=='套餐1' or o=='套餐2') and int(order[o])>0):  93. sum[o]=int(order[o])\*int(menu[o][0])  94. for s in sum.values():  95. total=int(total)+int(s)  96. print(str(total)+'元')  97. #使用者支付頁面與判斷(有錯誤例外處理)  98. pay=int(input('請付款，請輸入付款金額:\n'))  99. while(pay<total):  100. print('付款不足!\n')  101. pay=int(input('請付款，請重新輸入付款金額:\n'))  102. if(pay>total):  103. print('找您'+pay-total+'元，謝謝您!\n')  104. if(pay==total):  105. print('整付，謝謝您~\n')  106.  107. now = datetime.datetime.now()  108. print('最後確認您的用餐時間為(現在時間):'+now.strftime("%Y-%m-%d %H:%M:%S")+'後的兩小時喔~\n')  109.  110. print('-------------------------------------------------------------------------------------\n')  111. #模擬服務人員(劃位分析  112. print('模擬服務人員\n')  113. employee1=['店長','09123456789','女','到']  114. employee2=['店員','09987654321','男','未到']  115. employee={'1':employee1,'2':employee2}  116. employee\_num=2  117. store\_test=int(input('請輸入您要執行的操作:桌位管理(1)/員工管理(2)/員工簽到(3)'))  118. #桌位管理  119. if(store\_test==1):  120. print('訂位人數有:'+str(people\_num)+'\n')  121. print('如果想要都是兩人桌的話會需要\n'+str(people\_num//2)+'但可能有'+str(people\_num%2)+'人沒位子~\n\n')  122. print('如果想要都是三人桌的話會需要\n'+str(people\_num//3)+'但可能有'+str(people\_num%3)+'人沒位子~\n\n')  123. table={'兩人桌':2,'三人桌':3}  124. print('您可以參考目前剩餘桌位排桌位~\n')  125. for t in table.keys():  126. print('目前有:'+t+'剩餘:'+str(table[t])+'\n')  127. print('謝謝您使用本系統~\n')  128. #員工管理  129. elif(store\_test==2):  130. print('員工資料:\n')  131. for e in employee.keys():  132. print(e+'號')  133. print(employee[e])  134. e\_test=int(input('請輸入:無操作(0)/新增員工(1)/刪除員工(2)/修改員工資料(3)'))  135. #員工新增  136. if(e\_test==1):  137. name=input('姓名:')  138. tel=input('電話')  139. gender=input('性別(男|女)')  140. list\_add=[name,tel,gender,'未到']  141. employee[str(employee\_num+1)]=list\_add  142. print('已新增')  143. print('員工資料:\n')  144. for e in employee.keys():  145. print(e+'號')  146. print(employee[e])  147. #員工刪除  148. elif(e\_test==2):  149. print('請問要刪除幾號的資料?\n')  150. e\_num=int(input())  151. del employee[str(e\_num)]  152. print('已刪除')  153. print('員工資料:\n')  154. for e in employee.keys():  155. print(e+'號')  156. print(employee[e])  157. #員工修改資料  158. elif(e\_test==3):  159. print('請問要修改幾號的資料?\n')  160. e\_num=int(input())  161. name=input('姓名:')  162. tel=input('電話')  163. gender=input('性別(男|女)')  164. list\_change=[name,tel,gender,'未到']  165. employee[str(e\_num)]=list\_change  166. print('已修改')  167. print('員工資料:\n')  168. for e in employee.keys():  169. print(e+'號')  170. print(employee[e])  171. #員工簽到  172. elif(store\_test==3):  173. print('員工資料:\n')  174. for e in employee.keys():  175. print(e+'號')  176. print(employee[e])  177. print('請問要簽到幾號的資料?\n')  178. e\_num=int(input())  179. employee[str(e\_num)][3]='到'  180. print('已簽到')  181. |
| 六、執行結果  (將正確執行的結果貼入，文字為主截圖為輔，以下為例子) |
| |  |  |  | | --- | --- | --- | | 結果的流程為:  1.需要透過資料輸入訂餐與訂位系統中所需的相關資料，包含:訂餐者、人員數、菜品、菜品數量、用餐時間。  2.將上述的內容計算出總金額，與結果顯示給使用者，並模擬使用者付款。  3.於服務人員端，可以透過上述的資訊，自動化的進行桌分配(人數過多時，需要幾個桌位)  (我有在程式中，做例外處理)  4.訂餐系統針對list、dictionary增加套餐功能(增加餐廳行銷與使用者選擇)  5.使用者輸入自身身體數值，提供餐廳飲食建議  6.針對服務人員端新增員工資料管理與簽到系統(未來可以輸出檔案)  程式設計思維，需考慮人(user)與AI夥伴(App)的人際互動對話  (以下針對這次的主要新增功能做展示，)尼它細節操作，請參考上周 |  |  | |  |  |  | |
| 七、學習意涵詮釋  (可分為兩個部分：一、學到什麼程式語法 二、學到解決什麼問題，以下為例子) |
| 1.在條件分之中有: if()、elif()、else，須注意記得內縮，內縮是python文法的一部份。  2.if成立則執行下面內縮之敘述，elif():上述不成立下一條件判斷，else:如果if或elif的條件已成功就不會執行else，但以上條件皆不成立，則會執行else，同時須記得不行將else放在elif或if之前。  3.需要學習程式防錯的必要。  4. 可以在判斷中使用and or使程式更簡潔。  5. 迴圈(重複執行相同的動作): for、while。  6.for: 執行固定次數的迴圈，在for中除了可以要求次數，也可以對串列型別的各項資料、重覆相同的處理動作。  7.while: 有可能不小心設成無窮迴圈，可以使用^C跳出迴圈。  8. break:跳出迴圈。  9. continue :繼續重複迴圈，continue後的動作不執行。  10.break與continue可以利用判斷是進行跳出迴圈或繼續執行。  11. 函式:將多項處理動作彙集在一起, 供後續呼叫使用，也是一種重複執行的概念。  12.函式除了自訂義，也有python內建的，ex: len()、max()、min()…。  13. 例外處裡的方法:使用try 與except。  14.我在過程中學會使用更好的方法做資料處理，也在過程中學習到飲食結合商業的方法。  15.我透過此次的作業，我對於程式開發與結合議題有更好的認識，我本來只是想開發一個簡單的系統，但隨著上周的作業，我在這周想到或許可以結合人的健康需求搭配商業行銷，使得使用者與店家之間可以有更好的連結，也讓我認識友善店家的概念! |
| 八、參考說明  (列出所參考資料並說明如何應用或修改其內容，修改ChatGPT亦可，以下舉例) |
| 無參考資料 |

|  |
| --- |
| 教師：曾智義 課程名稱：基礎程式設計 學期：112-1 |
| 一、第12週課堂練習 |
| 寫三個小故事，或是延續期中專題，分別用with open語法寫入三個不同檔案，再分別讀出來。  利用seek()分別讀出三個檔案裡的某句，並顯示出來。 |
| 二、個人/成員 |
| 林彧頎\_資工系114級\_A1105505 |
| 三、議題規劃 |
| 人的心靈健康在現代社會中日益受到關注，因為它直接關係到個體的整體幸福感和生活品質。在這個議題的探討中，我將關注心理測驗對人心靈健康的正面影響，並強調它在幫助個體更加認識自我方面的重要性。  心理測驗作為一種心理學工具，具有卓越的價值，能夠為個體提供深入洞察和理解。首先，透過參與心理測驗，個體能夠更好地了解自己的內在需求、價值觀和情緒狀態。這種認識有助於建立自我認同，減少對於自身的迷惑和困惑，進而形成更積極、健康的心理狀態。  其次，心理測驗的正面影響體現在它為個體提供了自我成長和改善的契機。通過測驗結果，個體可以確認自身的優勢和弱點，從而有針對性地進行心理和情緒的調整。這種自我調整的過程有助於提高心理抗壓能力，使個體更具適應性和應對生活挑戰的能力。  此外，心理測驗還為心理健康專業人員提供了一個評估和干預的工具。通過分析測驗結果，專業人員可以提供更加個性化和有針對性的心理輔導，從而更好地協助個體克服心理困擾，實現心靈健康的平衡。  綜合而言，心理測驗在促進個體心靈健康方面發揮了不可或缺的作用。透過這種工具，人們能夠更深入地理解自己，實現自我成長，並獲得專業的支持和指導。  因此我將做出心理診療系統，可以幫助使用者與心理師之間更好的使用以上的功能，並給予專業的分析，並也將提供可未來擴充功能，使心理師可以更加好的幫助使用者。 |
| 四、問題定義  (自行設計問題，問題可列舉或敘述方式，自由模擬，以下為例子) |
| 1. 使用Accessfile功能，使得運用到讀檔(r、r+)、寫檔(w)(a)與with open的方法 2. 建立3個問題定義，讓使用者可以以此問題進行故事測驗 3. 使用者可以依照問題回答故事內容，並將答案與問題寫檔 4. 透過讀檔與seek功能，使抓到指定行數(答案的位置) 5. 將抓到的答案提供給心理學家分析，並將建議繼續寫入檔案，而不使前面檔案資料消失 6. 提供讀檔功能，讓使用者可以得到專家建議內容  |  |  |  |  | | --- | --- | --- | --- | |  | 問題 | 參考故事 | 參考建議 | | 1 | 有三種動物分別是  '猴子' '鳥' '蛇'  你要從家裡帶祂們出門請問你要怎麼帶祂們去呢? | 鳥我就叫祂自己飛去 蛇放在箱子裡 我要牽著猴子一起走，所以將來我對我的小孩是屬於獨立教育方式以及完全信任讓他自己做決定，錢我會存起來放在銀行 至於另一半我想跟他一輩子牽手一起走 | 猴子=另一半  鳥=孩子  蛇=錢財 | | 2 | 有四樣東西  KEY 你 橋跟兔子  請編個小故事  這四樣東西全都要出現在故事裡面中 | 有一天兔子拿一把key給我說只要我走到橋上把key往河裡丟 那我的願望就會實現 | key=錢  橋=自己的人生  兔子=另一半 | | 3 | 從飯廳開始你打算看看整個房子，你看到地上有杯子，杯子是什麼材質？  你拿起杯子後會做什麼？ | 把它撿起來放回桌上 | 你看見的杯子材質堅硬程度，是你與在樹林裡一起散步的人關係強度。  你對杯子的處理方式顯示出你會對這個人做的事。 | |
| 五、程式碼設計  (將執行的程式碼貼入並註解，綜合運用所學，以下為例子) |
| 1. #w12\_課堂練習：  2. #個人作業  3. #寫三個小故事，或是延續期中專題，分別用with open語法寫入三個不同檔案，再分別讀出來。  4. #利用seek()分別讀出三個檔案裡的某句，並顯示出來。  5. #以"課堂練習範本"word檔格式繳交至Moodle  6. #A1105505 林彧頎  7. #問題定義  8. Question={"Question1":"有三種動物分別是\n'猴子' '鳥' '蛇'\n你要從家裡帶祂們出門請問你要怎麼帶祂們去呢?\n",  9. "Question2":"有四樣東西\nKEY 你 橋跟兔子\n請編個小故事\n這四樣東西全都要出現在故事裡面中\n",  10. "Question3":"從飯廳開始你打算看看整個房子，你看到地上有杯子，杯子是什麼材質？\n你拿起杯子後會做什麼？\n"}  11. #答案存放  12. Story\_Ans={"A1":"","A2":"","A3":""}  13. #解讀定義  14. meaning={"1":"猴子=另一半 鳥=孩子 蛇=錢財\n",  15. "2":"key=錢 橋=自己的人生 兔子=另一半\n",  16. "3":"你看見的杯子材質堅硬程度，是你與在樹林裡一起散步的人關係強度。\n你對杯子的處理方式顯示出你會對這個人做的事。\n"}  17. #使程式碼可不斷操作  18. while(True):  19. #詢問  20. print('心理診療系統\n請問您是使用者還是心理師?\n')  21. test=input("使用者(1)心理師(2)")  22. #使用者  23. if(int(test)==1):  24. a=int(input("故事心理測驗嗎(1)查看專家建議(2)"))  25. #故事心理測驗  26. if(a==1):  27. print('以下將有3個心理測驗\n請依照心理測驗的問題\n依照您的想法，寫出對應的故事\n此文件將給予心理師為您解答\n或是您可以參考回答後的參考答案自行解讀。\n')  28. i=1  29. #with open寫檔(故事內容)  30. for k in Question.keys():  31. print(k+Question[k])  32. Story\_Ans["A"+str(i)]=input("請輸入您的心理測驗故事。\n")  33. print(Story\_Ans["A"+str(i)])  34. with open(f'C:\\Users\\USER\\Downloads\\基礎程式設計\\W12\\Story\_test\_{i}.txt', 'w') as file\_object:  35. file\_object.write(k+"\n"+Question[k]+"\n\nANS:\n"+Story\_Ans["A"+str(i)])  36. print('參考解讀:\n'+meaning[str(i)])  37. i=i+1  38. #查看專家建議  39. if(a==2):  40. i=1  41. for k in Question.keys():  42. #例外處理  43. try:  44. #讀檔並查看專家建議  45. with open(f'C:\\Users\\USER\\Downloads\\基礎程式設計\\W12\\Story\_test\_{i}.txt', 'r') as file\_object:  46. print(file\_object.read()+"\n\n")  47. i=i+1  48. except Exception:#錯誤處裡  49. print("錯誤!")  50. pass  51. #心理學家  52. elif(int(test)==2):  53. print("請問您想查看哪一則的答案?")  54. ans=input("Story\_1:(1)/Story\_2:(2)/Story\_3:(3)")  55. #依照要求給予建議  56. if(int(ans)==1 or int(ans)==2 or int(ans)==3):  57. line\_number=7  58. i=int(ans)  59. #例外處理  60. try:  61. #讀答案檔  62. with open(f'C:\\Users\\USER\\Downloads\\基礎程式設計\\W12\\Story\_test\_{i}.txt', 'r') as file\_object:  63. print('第'+str(i)+'問題之答案:\n')  64.  65. # 移到第六行起始位置  66. for \_ in range(line\_number - 1):  67. file\_object.readline()  68.  69. # 目前pointer位址  70. sixth\_line\_start = file\_object.tell()  71. # seek移動指標位置  72. file\_object.seek(sixth\_line\_start)  73. print(file\_object.read())  74. #用a的方法在檔案後續繼續寫檔(建議)  75. with open(f'C:\\Users\\USER\\Downloads\\基礎程式設計\\W12\\Story\_test\_{i}.txt', 'a') as file\_object:  76. recommand=input("請問你想輸入的分析建議?:\n")  77. file\_object.write("\n\n專家分析:\n"+recommand)  78. #例外處理  79. except Exception:  80. print("錯誤!")  81. pass  (下一頁繼續) |
| 六、執行結果  (將正確執行的結果貼入，文字為主截圖為輔，以下為例子) |
| |  |  |  |  | | --- | --- | --- | --- | |  | 問題 | 參考故事 | 參考建議 | | 1 | 有三種動物分別是  '猴子' '鳥' '蛇'  你要從家裡帶祂們出門請問你要怎麼帶祂們去呢? | 鳥我就叫祂自己飛去 蛇放在箱子裡 我要牽著猴子一起走，所以將來我對我的小孩是屬於獨立教育方式以及完全信任讓他自己做決定，錢我會存起來放在銀行 至於另一半我想跟他一輩子牽手一起走 | 猴子=另一半  鳥=孩子  蛇=錢財 | | 2 | 有四樣東西  KEY 你 橋跟兔子  請編個小故事  這四樣東西全都要出現在故事裡面中 | 有一天兔子拿一把key給我說只要我走到橋上把key往河裡丟 那我的願望就會實現 | key=錢  橋=自己的人生  兔子=另一半 | | 3 | 從飯廳開始你打算看看整個房子，你看到地上有杯子，杯子是什麼材質？  你拿起杯子後會做什麼？ | 把它撿起來放回桌上 | 你看見的杯子材質堅硬程度，是你與在樹林裡一起散步的人關係強度。  你對杯子的處理方式顯示出你會對這個人做的事。 |   會於過程中將檔案寫檔、讀檔等…操作，並且也可以額外輸入心理學家的建議。 |
| |  |  | | --- | --- | | 使用者端  (故事心理測驗) |  | | 專家  (輸入建議) |  | | 使用者  (查看專家建議) |  | | 檔案內容 |  | | |
| 七、學習意涵詮釋  (可分為兩個部分：一、學到什麼程式語法 二、學到解決什麼問題，以下為例子) |
| 1. 學會如何建立檔案物件(要注意於檔案總管找路徑，並且也要注意python中檔案路徑格式) 2. 存放路徑中Windows和Mac不同 3. 寫入步驟(write): 1.寫入模式建立檔案2.檔案物件將資料寫入檔案3.關閉檔案物件 4. 要求立即寫入(以防錯誤終止): file\_object.flush() 5. 讀取步驟(read): 1.以讀取模式建立檔案物件2.透過檔案物件以read方法讀取資料3.關閉檔案物件 6. 在既有檔案最後加入新資料(a): 1.附加模式建立檔案2.檔案物件將資料寫入檔案3.關閉檔案物件 7. 讀取加寫入(r+) 8. 注意當讀完時，檔案指標會移到最後，此時會需要seek(0)，移到檔案開頭 9. With open:檔案寫入方式(會自動關閉) 10. 額外學習tell():可以抓到目前pointer位置 |
| 八、參考說明  (列出所參考資料並說明如何應用或修改其內容，修改ChatGPT亦可，以下舉例) |
| 1.心理測驗1:<https://benbabyben2004.pixnet.net/blog/post/11902012>  2.心理測驗2:<https://www.popdaily.com.tw/forum/life/924385>  3. tell():可以抓到目前pointer位置 |

|  |
| --- |
| 教師：曾智義 課程名稱：基礎程式設計 學期：112-1 |
| 一、第13週課堂練習 |
| 自行找一張照片，試著將照片做以下的效果：改變亮度、扭曲、馬賽克、浮雕、輪廓處理、模糊化... |
| 二、個人/成員 |
| 林彧頎\_資工系114級\_A1105505 |
| 三、議題規劃 |
| 一般消費者或是攝影者，在拍攝照片後，都會有處理照片的需要，但相機或手機的功能，最基本的裁剪或許有，但更多的改變顏色、改變亮度、扭曲、馬賽克、浮雕、輪廓處理、模糊化、對比、旋轉與縮放等等…卻不一定有在市面的產品中可以操作的出來，為此我將設計一個系統，讓使用者可以將指定的圖片依照他想做的需求，進行操作，並也可以讓使用者動態調整參數，並儲存處理後的照片。 |
| 四、問題定義  (自行設計問題，問題可列舉或敘述方式，自由模擬，以下為例子) |
| 1. 使用Pillow功能進行讀取照片檔。 2. 運用Pillow的功能進行改變顏色、改變亮度、扭曲、馬賽克、浮雕、輪廓處理、模糊化、對比、旋轉與縮放等等…的功能。 3. 在每種對應的功能讓使用者可依照指定參數進行輸出調整。 4. 最終匯出處理後的照片檔。 5. 使系統可以重複執行。 |
| 五、程式碼設計  (將執行的程式碼貼入並註解，綜合運用所學，以下為例子) |
| 1. #w13\_課堂練習:  2. #個人作業  3. #自行找一張照片，試著將照片做以下的效果：改變亮度、扭曲、馬賽克、浮雕、輪廓處理、模糊化...  4. #以"課堂練習範本.doc"格式繳交至Moodle  5. from PIL import Image, ImageFilter, ImageEnhance  6.  7. #顯示圖片(用Pillow的套件Image開檔)  8. print("原圖片:flower.jpg")  9. image=Image.open('C://Users//USER//Downloads//基礎程式設計//W13//flower.jpg')  10. image.show()  11. while(True):  12.     n=int(input("請問你想對圖片進行怎麼樣的處裡呢?\n改變顏色(0)\n改變亮度(1)\n扭曲(2)\n馬賽克(3)\n浮雕(4)\n輪廓處理(5)\n模糊化(6)\n對比化(7)\n旋轉與縮放化(8)\n"))  13.     if(n==0):  14.         print("互補圖:flower\_bgr.jpg\n")  15.         #處理三原色  16.         #將藍色和紅色互補後再顯示圖片  17.         r, g, b = image.split()  18.         convert\_image = Image.merge('RGB', (b, g, r))  19.         convert\_image.show()#顯示互補顏色後的圖片  20.         convert\_image.save("C://Users//USER//Downloads//基礎程式設計//W13//flower\_bgr.jpg")#另存檔案  21.  22.         print("黑白圖:logo1\_blackwhite.jpg\n")  23.         #blackwhite黑白  24.         blackwhite = image.convert("1")#1 bit單位元的黑白圖片  25.         blackwhite.show()  26.         blackwhite.save("C://Users//USER//Downloads//基礎程式設計//W13//logo1\_blackwhite.jpg")  27.  28.         print("灰階圖:logo1\_gray.jpg\n")  29.         #灰階  30.         gray\_iamge = image.convert("L")#轉成灰階圖片  31.         gray\_iamge.show()  32.         gray\_iamge.save('C://Users//USER//Downloads//基礎程式設計//W13//logo1\_gray.jpg')#另存檔案  33.     if(n==1):  34.         print("調整亮度圖:logo1\_bright.jpg\n")  35.         factor=int(input('調整的亮度參數'))  36.         #亮度增強對象  37.         enhancer = ImageEnhance.Brightness(image)  38.         #調整亮度  39.         image\_brightness\_adjusted = enhancer.enhance(factor)  40.         #show & save  41.         image\_brightness\_adjusted.show()  42.         image\_brightness\_adjusted.save('C://Users//USER//Downloads//基礎程式設計//W13//logo1\_bright.jpg')  43.     if(n==2):  44.         print("扭曲圖:distorted\_image.jpg\n")  45.         #扭曲  46.         # 獲取圖像的寬度和高度  47.         w, h = image.size  48.         factor=int(input('調整的扭曲參數'))  49.         # 創建一個新圖像對象，與原始圖像相同大小  50.         distorted\_image = Image.new('RGB', (w, h))  51.  52.         # 扭曲圖像  53.         for y in range(h):  54.             for x in range(w):  55.                 dx = int(factor \* w \* (y - h / 2) / h)  56.                 dy = int(factor \* h \* (x - w / 2) / w)  57.                 distorted\_x = x + dx  58.                 distorted\_y = y + dy  59.  60.                 if 0 <= distorted\_x < w and 0 <= distorted\_y < h:  61.                     distorted\_image.putpixel((x, y), image.getpixel((distorted\_x, distorted\_y)))  62.         distorted\_image.show()  63.         # 保存生成的扭曲圖像  64.         distorted\_image.save('C://Users//USER//Downloads//基礎程式設計//W13//distorted\_image.jpg')  65.  66.     if(n==3):  67.         print("馬賽克圖:mosaic\_image.jpg\n")  68.         factor=int(input('馬賽克方格大小參數'))  69.         #馬賽克  70.         # 獲取圖像的寬度和高度  71.         w, h = image.size  72.         # 創建一個新圖像對象，與原始圖像相同大小  73.         mosaic\_image = Image.new('RGB', (w, h))  74.         # 循環遍歷圖像的每個塊  75.         for y in range(0, h, factor):  76.             for x in range(0, w, factor):  77.                 # 獲取當前塊的顏色  78.                 piece = image.crop((x, y, x + factor, y + factor))  79.                 avg\_color = piece.resize((1, 1)).getpixel((0, 0))  80.  81.                 # 將整個塊填充為平均顏色  82.                 mosaic\_image.paste(avg\_color, (x, y, x + factor, y + factor))  83.         mosaic\_image.show()  84.         # 保存生成的馬賽克圖像  85.         mosaic\_image.save('C://Users//USER//Downloads//基礎程式設計//W13//mosaic\_image.jpg')  86.     if(n==4):  87.         print("浮雕圖:logo1\_emboss.jpg\n")  88.         #emboss浮雕  89.         emboss = image.filter(ImageFilter.EMBOSS)  90.         emboss.show()  91.         emboss.save("C://Users//USER//Downloads//基礎程式設計//W13//logo1\_emboss.jpg")  92.     if(n==5):  93.         print("輪廓處理圖:logo1\_contour.jpg\n")  94.         #contour 輪廓處理  95.         contour = image.filter(ImageFilter.FIND\_EDGES)  96.         contour.show()  97.         contour.save("C://Users//USER//Downloads//基礎程式設計//W13//logo1\_contour.jpg")  98.     if(n==6):  99.         print("模糊化圖:logo1\_filtered.jpg\n")  100.         #模糊化  101.         filtered = image.filter(ImageFilter.BLUR)  102.         filtered.show()  103.         filtered.save("C://Users//USER//Downloads//基礎程式設計//W13//logo1\_filtered.jpg")  104.     if(n==7):  105.         #contrast對比增加  106.         print("對比圖:logo1\_contrast.jpg\n")  107.         factor=int(input('對比大小參數'))  108.         enh\_contrast = ImageEnhance.Contrast(image)  109.         contrast = enh\_contrast.enhance(factor)  110.         contrast.show()  111.         contrast.save("C://Users//USER//Downloads//基礎程式設計//W13//logo1\_contrast.jpg")  112.     if(n==8):  113.         print("rotate 180:logo1\_rotate\_180.jpg\n")  114.         #rotate 180  115.         rotate\_180 = image.transpose(Image.ROTATE\_180)  116.         rotate\_180.show()  117.         rotate\_180.save("C://Users//USER//Downloads//基礎程式設計//W13//logo1\_rotate\_180.jpg")  118.  119.         print("resized縮放圖:logo1\_resize.jpg\n")  120.         #resized縮放  121.         resized = image.resize((round(image.size[0] \* 2), round(image.size[1] \* 1.5)))  122.         resized.show()  123.         resized.save("C://Users//USER//Downloads//基礎程式設計//W13//logo1\_resized.jpg")  #(下一頁繼續) |
| 六、執行結果  (將正確執行的結果貼入，文字為主截圖為輔，以下為例子) |
| |  |  |  | | --- | --- | --- | | 原圖 |  |  | | 改變顏色 |  |  | | 改變亮度 |  |  | | 扭曲 |  |  | | 馬賽克 |  | | 浮雕圖 |  | | 輪廓處理圖 |  | | 模糊化 |  | | 對比化 |  | | 旋轉與縮放 |  | |
| 七、學習意涵詮釋  (可分為兩個部分：一、學到什麼程式語法 二、學到解決什麼問題，以下為例子) |
| 1. 開cmd(命令提示字元) 2. pip install//安裝 3. pip uninstall//解安裝 4. pip show//想查詢的模組名稱 5. pip list//列出已安裝的外部程式 6. 套件是一次性安裝，不用重複安裝 7. 學會如何安裝額外套件與使用 8. 在亮度調整的部分學會使用ImageEnhance.Brightness()的方法來調整 9. 互補圖用Image.merge() 10. 黑白圖用image.convert("1") 11. 灰階圖用image.convert("L") 12. 我在寫扭曲圖時有稍微遇到一點困難，後來我查到一個公式(改變圖片對應的x與y)使得可以旋轉圖片#dx = int(factor \* w \* (y - h / 2) / h)# dy = int(factor \* h \* (x - w / 2) / w) 13. 馬賽克也要查，我發現可以將圖片的每個方塊的顏色取平均值，就可以做到馬賽克 14. 浮雕圖用image.filter(ImageFilter.EMBOSS) 15. 輪廓處裡用image.filter(ImageFilter.FIND\_EDGES) 16. 模糊圖用image.filter(ImageFilter.BLUR) 17. 對比圖用enh\_contrast = ImageEnhance.Contrast(image)然後再用enhance改變對比的大小 18. 旋轉與縮放: image.transpose(Image.ROTATE\_180)和image.resize((round(image.size[0] \* 2), round(image.size[1] \* 1.5)))   (下一頁繼續) |
| 八、參考說明  (列出所參考資料並說明如何應用或修改其內容，修改ChatGPT亦可，以下舉例) |
| 1.Pillow功能參考 <https://vocus.cc/article/64e9aa5afd89780001ef4477> |

**貳、你對別人課堂練習作品的觀摩學習(至少5處)**

|  |
| --- |
| **作業日期：2023/09/19**  **姓名\_系別\_學號：黃茂翔\_資工\_A1095559** |
| 作業題目: calendar運用 |
| 議題規劃及問題 |
| 高雄大學的新生在剛入學時有許多需要完成的代辦事項，同時一整年都有許多重要事件，以下程式將會把所有學年的重要事項印出。  問題:  1. 新生入學(繳交學雜費、健康檢查、加退選課) 112.08-112.09  2. 期中考 112.11  3. 期末考 113.01  4. 寒假 113.01-113.02  5. 期中考 113.04  6. 期末考 113.06  暑假 113.07-113.08 |
| 所觀摩的程式碼 (\*僅貼入所觀摩的程式碼\*) |
|  |
| 執行結果 |
| |  |  |  | | --- | --- | --- | |  |  |  |   此為一簡單的印出行事曆程式，紀載入學、考試、假期等資訊，導入datetime等工具實現近期通知等功能。 |
| 觀摩感想 |
| 2023/9/19 林彧頎 觀摩感想  在觀摩完第二週，calendar運用相關作業後，我發現黃茂翔學長所設計的程式，感覺十分實用，其中最讓我驚訝的是，學長利用邏輯概念的方法，將事件進行分析，分析此事件是否是近期發生的事件，而這不僅僅需要利用到陣列的方法，也需要運用到for loop的方法進行比較，如果有此近期事件的話，就會產生此事件為近期事件，如果無則產生無~  對應我第二周的作業來說，我發覺我的作業主要是針對我所想到的議題，關於產業景氣的相關應用，做發想，主要的層面是希望相關產業者，可以透過此程式，進行產業景氣分析，以達到未來產業規劃的目的。  而學長的這項操作，讓我想到，我可以將我的程式中，多增加判斷當前的時間，並給予數據建議當前的產業規劃。  對我來說這些內容，或許可以應用於下一次的課堂作業或期中專題當中! |

|  |
| --- |
| **作業日期：2023/09/26**  **姓名\_系別\_學號：A1103120\_應用經濟學系\_何安然** |
| 作業題目: 基本運算, 變數, 數值, 字串, 比較... |
| 議題規劃及問題 |
| 高雄大學學生會企劃舉辦聖誕節系列活動，可憐的財務部長需要審查預算是否足夠。  問題:  1. 審查總預算數值  2. 檢查預算是否足夠  3. 如果預算不足需做出調整  可變動部分為藝人，考慮到客群與人氣，優先選擇陳忻玥 |
| 所觀摩的程式碼 (\*僅貼入所觀摩的程式碼\*) |
|  |
| 執行結果 |
| 結果為false，代表預算超支，只能忍痛將藝人從陳忻玥改為呂士軒  （此事件該編自真人真事） |
| 觀摩感想 |
| 我覺得最有趣的地方是，因為我是資工系的，因此會覺得變數不能使用中文字，但透過同學的示範，與上課時老師的解說，我發現原來Python可以使用中文字坐變數阿~這樣對中文的使用者來說，真的方便許多~  我也很喜歡這位同學的一題設計，感覺對於社團的需求十分有用，畢竟是同學的親身經歷! |

|  |
| --- |
| **作業日期：2023/10/03**  **姓名\_系別\_學號：許梓榆\_應用化學系\_A1094222** |
| 作業題目: 串列型別(List)、字典型別(Dictionary Type)的運用 |
| 議題規劃及問題 |
| 議題:  最近各大百貨公司陸續開始週年慶活動，有天小云在翻閱商品型錄時，發現這間MFK香水品牌在週年慶期間推出了包含四款香水的五種1+1組合，正好年初時購入的香水快用完，因此小云打算趁這次週年慶價格較優惠的時候，購買MFK的其中一項香水組合。  問題:  1. 小云的預算為11000元，且她較傾向選擇一種為花香調，另一種為其他混合香調的香水組合，並希望可以購買價格於預算內又最划算（節省最多）的組合。  2. 於週年慶期間，購買MFK的香水組合皆享有95折優惠，且消費滿10000元可再現折1000元，滿11000元可折1200元，再滿1000元則可再折300元，以此類推。  3. 以下為參與香水組合的品項、香調及價格：  編號 品名 香調 價格 (元)  1 愛戀玫瑰 花香調 6000  2 阿米香樹之香 花香調 6800  3 香詩之水 木質花香調 6800  4 生命之水 花香柑橘調 5000  4. 五種香水組合為：  組合 香水品項  A 1 + 2  B 1 + 3  C 1 + 4  D 2 + 3  E 2 + 4 |
| 所觀摩的程式碼 (\*僅貼入所觀摩的程式碼\*) |
|  |
| 執行結果 |
| 於結果中可以得知，符合香調需求的有組合B, C, D, E，又符合預算的有組合B, C, E，其中價格最划算的為組合B，故小云購買香水**組合B**最符合她的需求。 |
| 觀摩感想 |
| 我覺得這位學姊在其中能將串列型別與字典型別結合過去兩週所學實際套用於日常生活的需求上，讓我感覺十分實用。  或許程式的最主要目的本來就不是要解決大問題，最重要的是從生活細節去發揮，這是我對學姊的第一部分作的觀察，我也發現針對於學姊過去的議題中，大多會著重於個人購物需求，讓我更加覺得python或者說程式的應用可以由小見大，從生活細節發現更多的解決問題的方法。 |

|  |
| --- |
| **作業日期：2023/10/03**  **姓名\_系別\_學號：許梓榆\_應用化學系\_A1094222** |
| 作業題目: 串列型別(List)、字典型別(Dictionary Type)的運用 |
| 議題規劃及問題 |
| (此為此同學的第二個觀摩)  議題:  最近各大百貨公司陸續開始週年慶活動，有天小云在翻閱商品型錄時，發現這間MFK香水品牌在週年慶期間推出了包含四款香水的五種1+1組合，正好年初時購入的香水快用完，因此小云打算趁這次週年慶價格較優惠的時候，購買MFK的其中一項香水組合。  問題:  1. 小云的預算為11000元，且她較傾向選擇一種為花香調，另一種為其他混合香調的香水組合，並希望可以購買價格於預算內又最划算（節省最多）的組合。  2. 於週年慶期間，購買MFK的香水組合皆享有95折優惠，且消費滿10000元可再現折1000元，滿11000元可折1200元，再滿1000元則可再折300元，以此類推。  3. 以下為參與香水組合的品項、香調及價格：  編號 品名 香調 價格 (元)  1 愛戀玫瑰 花香調 6000  2 阿米香樹之香 花香調 6800  3 香詩之水 木質花香調 6800  4 生命之水 花香柑橘調 5000  4. 五種香水組合為：  組合 香水品項  A 1 + 2  B 1 + 3  C 1 + 4  D 2 + 3  E 2 + 4 |
| 所觀摩的程式碼 (\*僅貼入所觀摩的程式碼\*) |
|  |
| 執行結果 |
| 於結果中可以得知，符合香調需求的有組合B, C, D, E，又符合預算的有組合B, C, E，其中價格最划算的為組合B，故小云購買香水**組合B**最符合她的需求。 |
| 觀摩感想 |
| 此篇是我對學姊程式碼的第二處觀摩，我認為程式的最主要目的本來就不是要解決大問題，最重要的是從生活細節去發揮，其中我覺得最有趣的是，學姊將組合弄得像是數學中的排列組合去判斷需求性，並且搭配計算金額，我認為這是一個十分據統計數學的方法，讓我於之後會考量到，如果要大量計算多種可能性時，可以使用這種方法! |

**(下一頁繼續)**

|  |
| --- |
| **作業日期：2023/10/17**  **姓名\_系別\_學號：葉榮恩\_應物\_a1104346** |
| 作業題目: 綜合運用至目前所學(基本運算、變數、字串、List、Dict、Tuple、Set） |
| 議題規劃及問題 |
| 賣場提供生鮮各產地的進貨數量查詢，讓消費者吃得安心。  問題:  1. 建立生鮮貨品對應dict  2. 請使用者輸入想查詢的菜名、產地、日期  3. 輸出對應數量  4. 查詢過之關鍵字呈現於屏幕 |
| 所觀摩的程式碼 (\*僅貼入所觀摩的程式碼\*) |
|  |
| 執行結果 |
|  |
| 觀摩感想 |
| 我覺得同學在本次的作業主要展示了tuple與dictionary的用法，他透過結合兩種的方法達到更複合性的作用，我認為這是十分值得學習的地方，因為程式設計最高級的做法，就是可以活用程式中的各個功能，並且可以透過此方法，史資料處理與程式精簡度上有大幅的提升。  另外我也有看到同學在本次的作業中使用到split的方法，這個方法我不曾用過，因此同學的時做可以讓我學習到可以透過split達到分割的技巧。  最後是我也從同學的實作中發現對於進出貨的應用，這使我未來可以學習在系統架設中使用相關的方法，使資料處理更精簡。 |

|  |
| --- |
| **作業日期：2023/10/24**  **姓名\_系別\_學號：A1107103\_亞太系\_陳以勒、A1107123\_亞太系\_陳星霓、A1107142\_亞太系\_劉冠伶** |
| 作業題目: 條件分支、迴圈、while(條件式)、函式 |
| 議題規劃及問題 |
| 高大博物館開館10週年，為了感謝大家的支持，推出遊館集章送小禮物的活動(禮物一天限量30份)。   |  |  |  | | --- | --- | --- | | 票種 | 票價 | 說明 | | 全票 | $300 | 一般民眾 | | 優惠票 | $200 | 7-22歲、65 歲以上長者持有效證件者 | | 免費優待票 | $0 | 設籍高雄市者、未滿 7 歲、身心障礙者、孕婦 | |
| 所觀摩的程式碼 (\*僅貼入所觀摩的程式碼\*) |
|  |
| 執行結果 |
|  |
| 觀摩感想 |
| 我覺得同學在本次的作業主要展示了邏輯判斷的方法，而我會觀摩那個片段的原因是因為，我覺得他們真的有好好的使用到老師所說的，可以透過多個判斷減少程式碼的程度，使可以一次性判斷的部分就一次性判斷，而且在一提的部分也很有趣，因為這就像平常去買票時的售票機，如果要買特殊身分的票，就必須通過身分判斷，我認文這是一個很好的範例。 |

|  |
| --- |
| **作業日期：2023/11/28**  **姓名\_系別\_學號：黃茂翔\_資工\_A1095559** |
| AccessFile 運用 |
| 議題規劃及問題 |
| 延伸期中專題，利用寫檔讀檔使程式不會如此冗長。 |
| 所觀摩的程式碼 (\*僅貼入所觀摩的程式碼\*) |
|  |
| 執行結果 |
| |  |  | | --- | --- | |  |  |   執行結果與期中相同，因此只列出新增功能。  圖一(左)為使用新增餐廳資料的狀況，按照原格式進行輸入後就可以將資料存入json中，圖二(右)則是json內的數據，可以看見新增的資料確實存入其中，當下次開啟時直接使用此json資料即可找到過去存入的數據。 |
| 觀摩感想 |
| 2023/11/28 林彧頎 觀摩感想  在觀摩完第十二週，Accessfile的作業後，我發現學長與我最大的不同為，學長是從自己的期中專題改，但由於我期末專題有其他的想法，所以我並沒有想沿用期中的專題，我導入了心理學的觀念，見了另一個系統，因此會更加聚焦於教授所教學的內容，而學長在這裡不僅僅延伸了他的期中，更展現了讀檔中使用json格式的方便性，不僅僅使程式碼更加精簡，也使得我從學長的範例中，學會了json使用方法，讓讀檔寫檔的功能不局限於txt中，更可代表可以應用到不同層次，而我也將以此觀念運用到我的期末專題之中，我將使用csv檔的格式進行數據分析。 |

|  |
| --- |
| **作業日期：2023/12/05**  **姓名\_系別\_學號：劉冠伶\_亞太系\_A1107142** |
| 更改圖片特性、變換不同圖片效果 |
| 議題規劃及問題 |
| 小慈最近在進行團體報告，需要將圖片進行加工，讓報告更為美觀，於是小慈便選定了一張圖片，並對其打造出各式各樣的效果。  小慈希望能夠利用python對圖片做以下更動：   1. 輪廓處理 2. 浮雕 3. 改變亮度 4. 馬賽克 5. 扭曲   6.模糊化 |
| 所觀摩的程式碼 (\*僅貼入所觀摩的程式碼\*) |
|  |
| 執行結果 |
| 1. 輪廓處理      1. 浮雕      1. 改變亮度（降低亮度）      1. 馬賽克      1. 扭曲      1. 模糊化     ----------  原始圖片： |
| 觀摩感想 |
| 2023/11/28 林彧頎 觀摩感想  在觀摩完第十三週，Pillow的應用後，我發現大部分的同學使用的馬賽克的方法與我使用的方法有些不同，我是透過將圖片分成好幾個小格子，然後對每一個格子做顏色的處哩，而其他的同學則是用圖片放大縮小十，會造成失真的方法，來達到馬賽克的效果，我在這裡學到，無論是什麼方法，只要能達成目的就是好方法，並且也謝謝同學的十座，讓我又多學習到一種方法。 |

**參、期中學習心得**

**我在這上學期的課程中真的學習到了很多，由於我個人是資工系的，但自高中接觸arduino到大三上，我竟然都沒有學習從高中就耳熟能詳的python，直到在暑假期間，我開始著手畢業專題、競賽、論文，我發現無論是甚麼，都需要進行資料分析(語音、文本等等…)，而其中一個非常好用的工具就是Python，而我在做相關研究的過程中，常常會因為不太清楚Python的使用而十分痛苦，因此使我很想好好學習Python這項好用的程式語言，而基礎程式設計課程正好滿足我的需求，而在課程中我不僅僅學到我平常不會使用到的資料處理的方法，像是我平常並不會特別地去了解字典的內部可以運用到list、tuple等等...，並且我也喜歡當周的作業當周交，並觀摩其他同學的作品，使得我下課時，時常跟老師討論作業怎麼做才是最好的，我可以說總共交了3份(每周3份)的期中前作業，但這過程中，我並沒有不開心，相反的我覺得我學到很多，像是怎麼寫出更好的報告，如何在有限的限制中，發揮無限的議題發想與實作，更學會觀摩同學的報告，學習到很多很多，而在期中專題中，我從企畫構想、系統架構、問題、以及我主要負責的程式碼，我都有參與到，並且在報告中，關於程式相關的說明主要都是我撰寫的，還有架構的部分，透過本次的期中報告，我學會如何使用到商業結合生活的方法，並也學習到更多的python技巧，我也很感謝我的組員給我很多的發揮空間，並也如期繳交各自負責的部分，真的非常感謝組員與老師提供的學習機會，讓我學會與不同系所的同學合作，並可以從中學習到python這項不可或缺的技能，使程式不僅僅是coding，更是融入議題的實用工具。**

**肆、期末學習心得**

**我在這學期的課程中學習到很多，並且也因唯這學期系上的課程，大部分都是要開始做資料研究，因此開發軟體大部分都是python，也因此這堂課幫助我很大於精進於python的程式撰寫，我也在期末的課程中，學到Access File、Pillow的方法，這與我原本學的地方，讓我感悟更深的是，當有一位老師教，跟自己去了解，真的有很大的不同，我透過老師的教學，更加會使用這些功能，另外在期末專題中，我感觸很深，首先非常感謝老師讓我可以使用我這學期參加比賽的作品來當我的期末專題，當然過程中我也有多增加老師上課的內容於我的系統之中，下半學期的內容其實比較難，但也相對比較少，卻也讓我學到很多，而本專題可以說是將老師這學期所教導的融會貫通，並也融入生成式AI的功能，我在與組員的配合中，教會組員許多他們沒接觸過的東西，我也很慶幸組員很勇於發問，搞懂這個系統的實作方法，我也很開心可以透過這次的機會重新整理自己的作品，並優化它，整體來說，無論是這學期的課程，或是這次的期末專題、其中專題，我基本上每個步驟都有我的身影無論企劃、架構等等…，尤其是程式的部分，我覺得程式永遠都無法學完，因為有太多功能，而且科技不斷地發展，就像我的期末專題的實作，當初是我從不會python,而且第一次接觸生成式AI與爬蟲，在不斷地做中學，學中做，而製作出來的，我覺得過程中所帶來的成就是非常高的，雖然花了我將近1個月以上的時間，每日每夜研究並做出來的成果，我最終也將這個成果投到演討會，並成功發表論文，因此我很感謝本專題提供我展現此實作的機會，並且很感謝老師與組員，這學期讓我學到很多東西!非常感謝!**

**伍、個人照片**

****

