Scipiting語言 期末專題報告

組員:四訊四乙 劉彥震

職四訊三甲 林永森

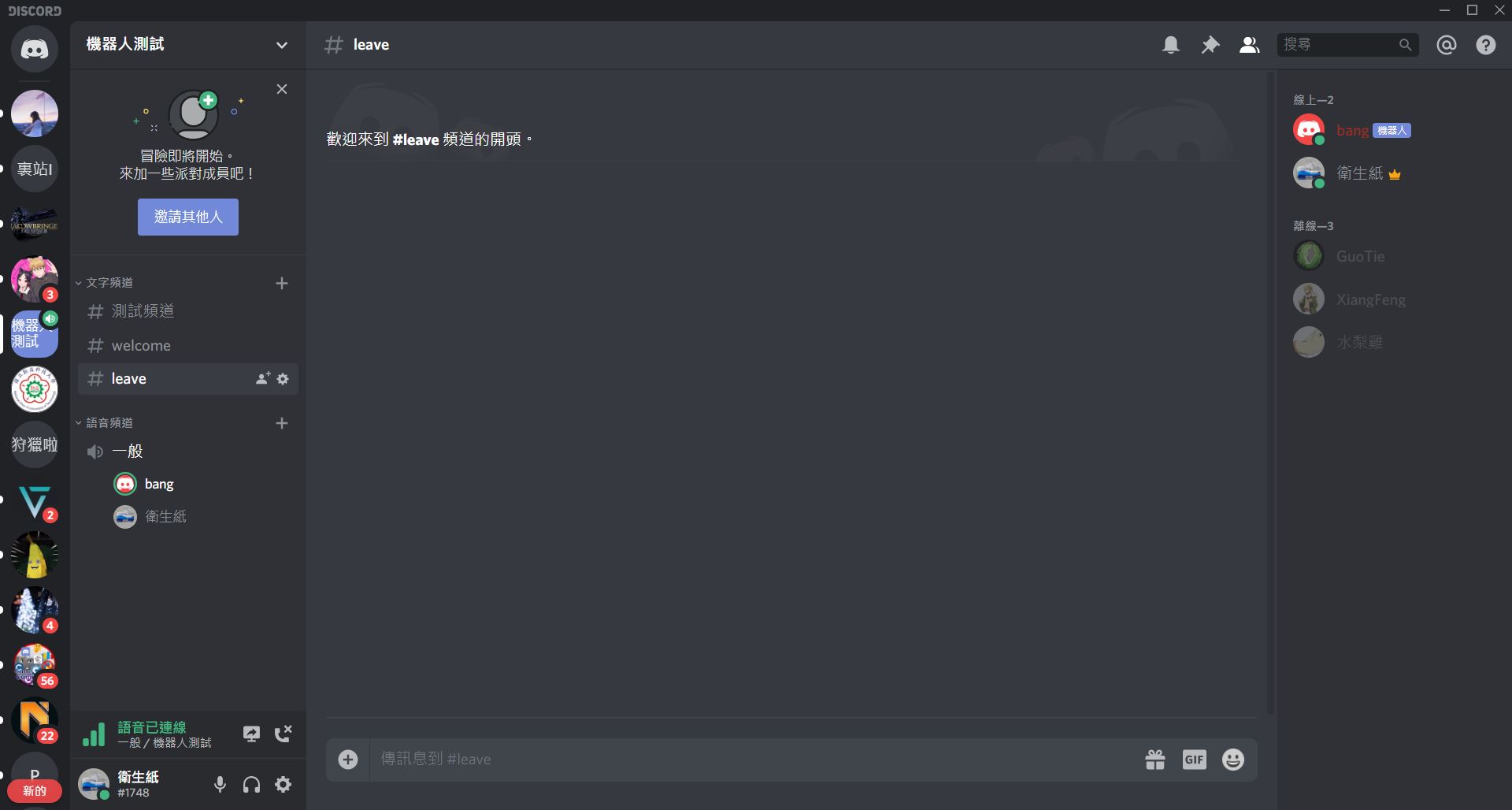
作業說明

希望模仿市面上的dicord音樂機器人以學習背後的原理，究竟是怎麼使用指令去讓機器人產生動作。

背景資料

Discord是一款專為電子遊戲社區設計的免費網路即時通話軟體，主要用戶為遊戲玩家，玩家之間可以在軟體的聊天頻道通過資訊、圖片、視訊和音訊進行互動，擁有手機版本以及電腦版本。

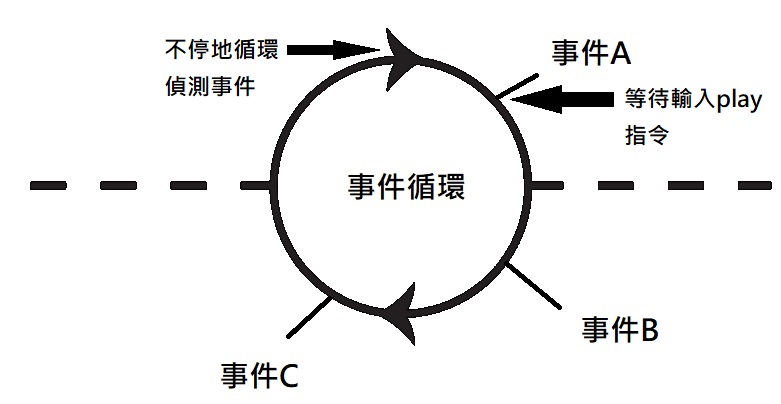
介面:



如上圖，群組內可以自由創建文字頻道以及語音頻道，除了文字/語音聊天外還擁有權限的功能，不像line每個人都擁有管理員的權限，導致"翻群"的事件不斷地在發生，此外discord雖然能上傳檔案，但免費版只能8MB，付費才能上傳到100MB，這是比起line唯一的缺點。

# asyncio協程

它是在python3.5後出現的一個套件，是一個在處理I/O工作上非常好用的套件。



如上圖，圓圈的部分稱做"事件循環"，突出的每一條線就是一個"事件"，

運作原理就是，事件循環會不停地偵測是否有事件觸發了，或者說等待事件的"阻塞"疏通了，然後才去執行該事件的程式，例如等待使用者輸入指令、等待X秒。

例如:

等待X秒，就要使用asyncio模仿time的sleep功能。



等待輸入指令，在函式前加入async以宣告他是事件

而上面的@是discord bot的裝飾符，可將該事件變成指令，所以下圖的函式我們就必須在discord的文字頻道中打上"-play url"才會執行。



所以總結來說，用此套件寫的程式可以在執行A事件時遇到阻塞時去找別的事情來工作，就不像傳統的一路執行到底了。

環境介紹

硬體:一台可運行的電腦

周邊:滑鼠、鍵盤、螢幕

作業系統:windows 10

使用工具:

visual studio code:一個由[微軟](https://zh.wikipedia.org/wiki/%E5%BE%AE%E8%BD%AF)開發，同時支援[Windows](https://zh.wikipedia.org/wiki/Windows) 、 [Linux](https://zh.wikipedia.org/wiki/Linux)和[macOS](https://zh.wikipedia.org/wiki/MacOS)等操作系統且[開放原始碼](https://zh.wikipedia.org/wiki/%E5%BC%80%E6%94%BE%E6%BA%90%E4%BB%A3%E7%A0%81)的[程式碼編輯器](https://zh.wikipedia.org/wiki/%E6%96%87%E6%9C%AC%E7%BC%96%E8%BE%91%E5%99%A8)，我們用它來撰寫python程式碼。

用到的程式套件:

1.virtualenv 創建獨立的虛擬環境。

2.discord 撰寫bot的官方套件。

3.json 用於建立與讀取設定檔。

4.youtube\_dl 用於下載影片、解析影片等。

5.ffmpeg 是一個[自由軟體](https://zh.wikipedia.org/wiki/%E8%87%AA%E7%94%B1%E8%BB%9F%E9%AB%94)，可以執行音訊和視訊多種格式的錄影、轉檔、串流功能，youtube\_dl中的解析會需要用到它。

此套件不能用pip安裝，必須到官網下載後將ffmpeg/bin下的三個檔案放入虛擬環境中的Scripts目錄中。

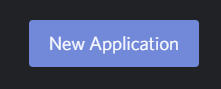
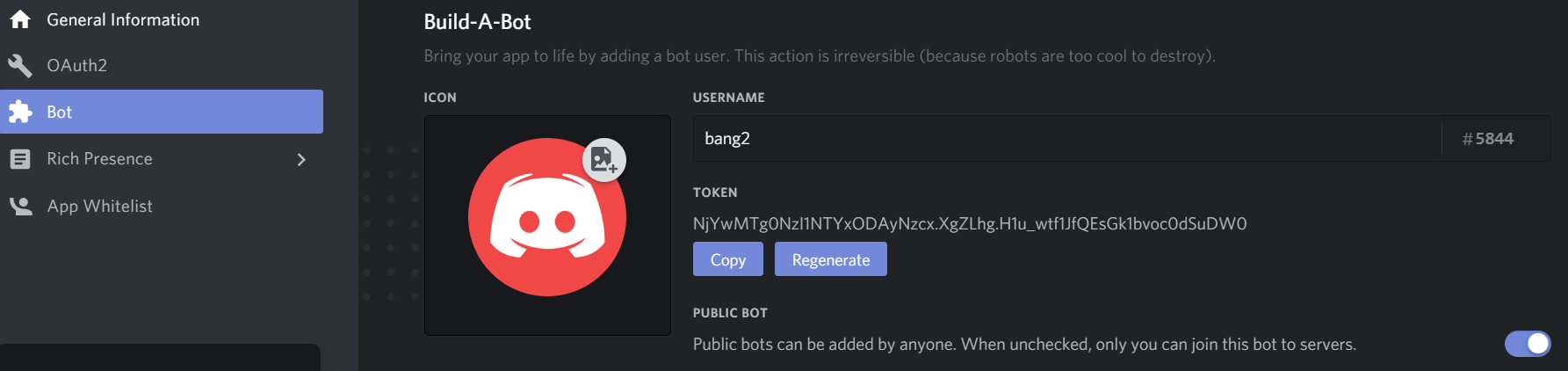
6.asyncio 協程，常見的discord bot就是建立在這個套件上寫出來的

7.re 正則表達示

實作方法

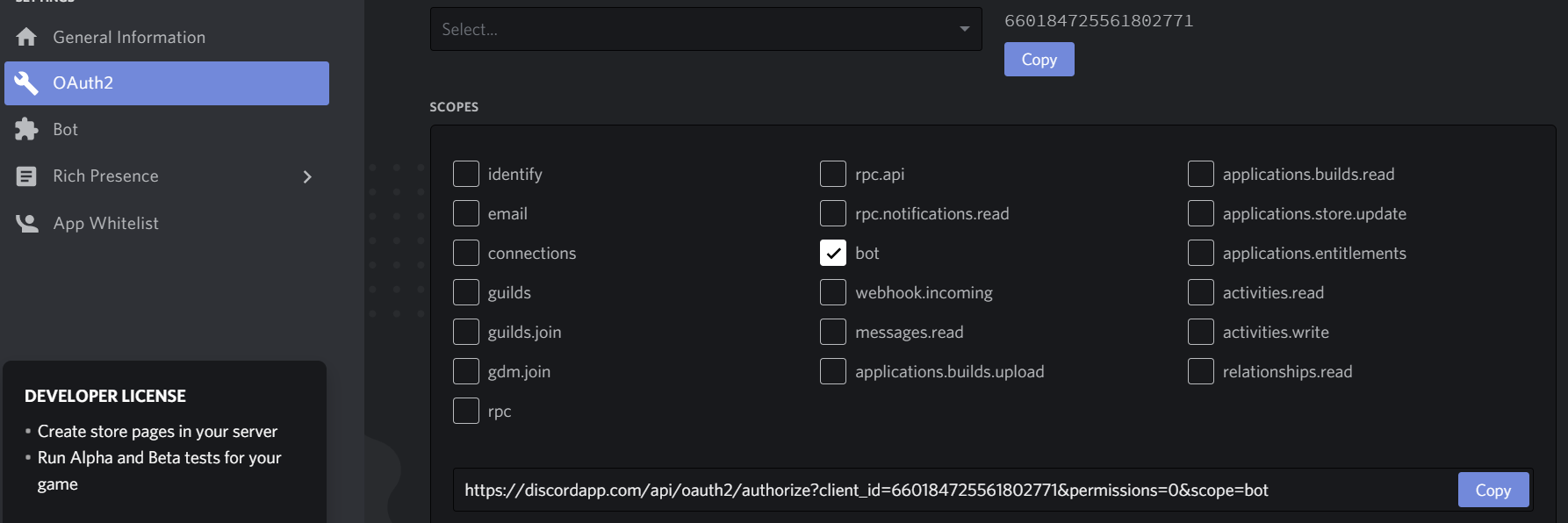
# 創建機器人bot

機器人的創建非常簡單，只要到discord的developer頁面，點擊右上角的

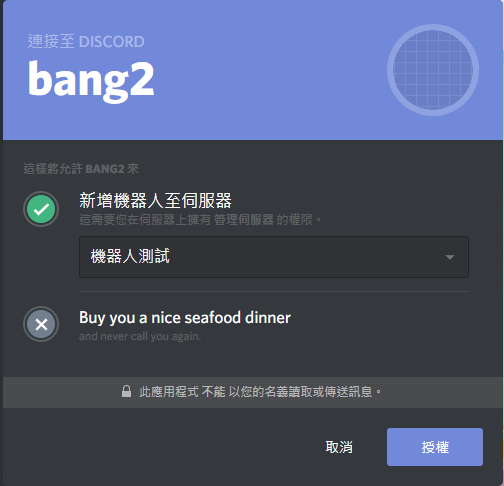
，接著到左邊加入Bot

(Token是此機器人獨一無二的識別碼，運行會需要用到，因此通常不可給其他人知道)

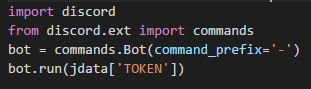
接著在上面的OAuth2頁面勾選Bot，會出現一串網址



將他貼到瀏覽器上，便可以將機器人邀入群組。

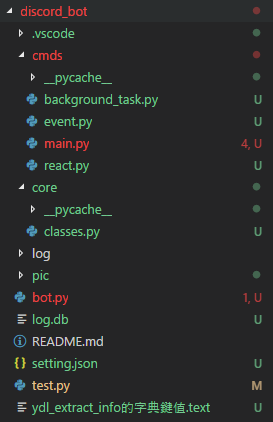


運行也很簡單，只需要運行以下程式碼就會看到機器人在群組內上線。



# 程式碼架構:

由於discord的連線關係，在中斷bot運行之後，依然需要一小段時間機器人才會下線，對於測試、debugger等非常不方便，所以架構主要根據discord提供的一個"Cog"功能，可以讓我們即便在bot運行中，也可以立刻重新掛載修改後的程式碼，相當的方便。



首先在core中的classes.py中寫一個類別，並將它繼承自commands.Cog



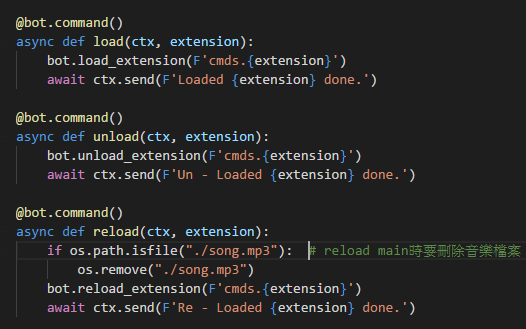
接著在cmds中的py中的類別分別都繼承它



接著將這些檔案load進bot中



接著寫掛載、卸載、重新卸載指令

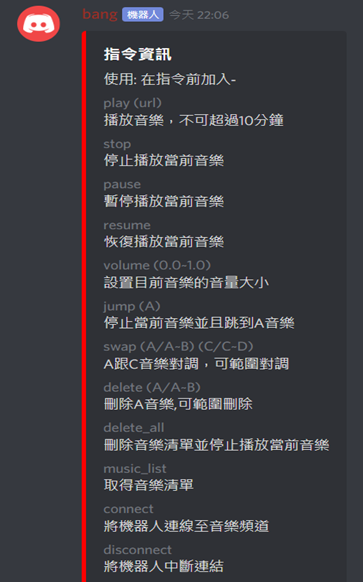


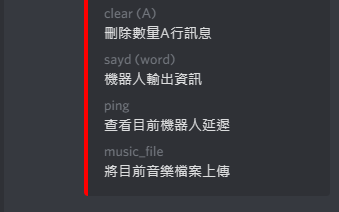
就完成了

例如我修改了main.py中的程式碼，此時只要打-reload main即可重新掛載，bot會運行最新的main程式碼



# 功能:



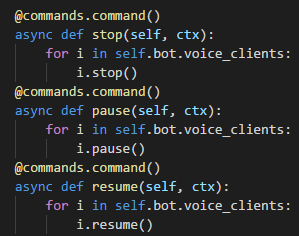


# Discord套件內建的函式

在功能列表中的很多功能都可以根據官方文件寫出簡潔的程式碼

有stop、 pause 、resume 、volume 、connect 、disconnect、clear、sayd、ping。

例如:



因此就不詳解了。

# play播放

通常在第一首歌播放時，使用者可以進行點歌動作，所以會需要一個容器儲存這些歌曲，我們使用queue來做儲存，它是一個先進先出的容器。





運作原理:

1. -play url 執行play函式，將url放入queue中。

2.使用youtube\_dl套件(會需要ffmpeg)透過網址取得歌名，加入到音樂清單中。

3.3~8點是一個事件 audio\_player\_task，它會不停地等待有網址(url)可從queue中拿出



4.從音樂清單中刪除該歌曲

5.使用youtube\_dl透過網址下載該歌曲的音訊檔，這邊我選擇下載為mp3格式

6.使用discord內建的函式將mp3檔案轉成PCM音頻播放(同樣需要ffmpeg)



7.設置play\_next\_song旗標，以防止還沒播放完就回到第3點。



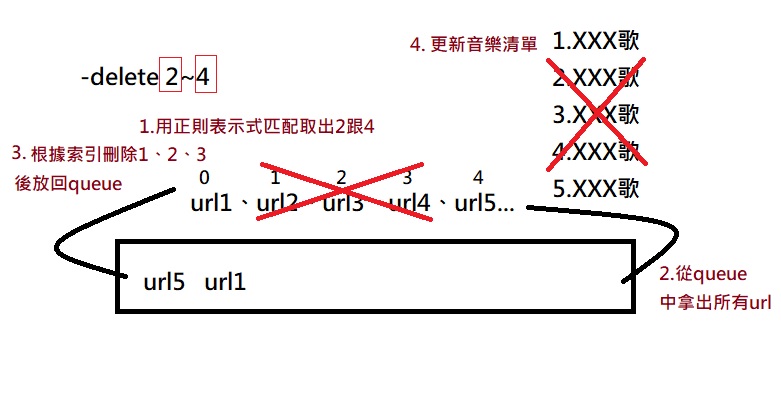
8.播放完或中斷時會呼叫toggle\_next函式，將play\_next\_song旗標啟動



回到第3點，不停循環。

# delete刪除

刪除音樂清單中的指定音樂，可以指定刪除，也可以範圍刪除。



1.透過正則表示式，取出2與4



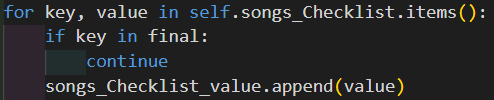
2.將queue中所有的url拿出



3.根據索引(2~4=1~3)刪除"1~3"首歌

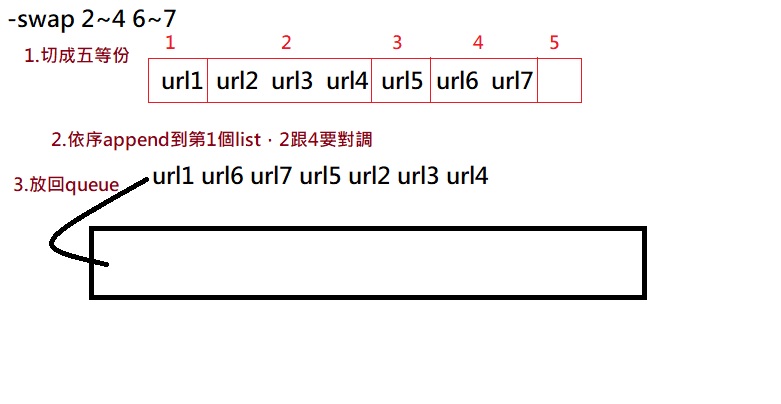


4.更新音樂清單，利用continue，略過被刪除的項目以重新建立音樂清單



# swap對調

將音樂清單中的A音樂與B音樂對調，也可以範圍對調，例如A~B與C~D首音樂對調，範圍的部分跟delete一樣，所以以下只有我是怎麼對調的方法。



1.將所有歌曲切成五等份，五個list

2.依序append到第1個list，2跟4要對調

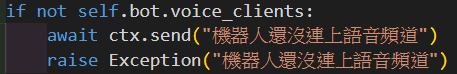
3.放回queue完成

音樂清單的更新也是同理，原本的寫法並不是這樣，原本是選擇使用list的insert插入方法來做對調，然後因為前面的插入會導致索引改變，進而導致後面的插入的索引變化很多，所以就算寫的出來看起來也會很複雜，所以最後選擇這種方法。

# 遇到的問題&解決方法

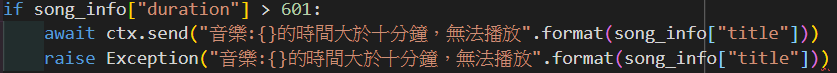
## 1.音樂清單: 原本的想法是只需要在每次呼叫-music\_list指令時，從queue中將當前所有音樂拿出，解析取得歌名後即可，然而在queue中存的是url，解析勢必還得經過下載，時間將會需要很久，因此選擇將音樂清單獨立出來儲存成一個list，每次執行play、delete、swap、jump指令時都會一起更新。

## 2.檢查機器人是否在語音頻道內



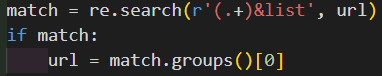
如果不在語音頻道內就播放會造成PCM音頻檔放不出來而程式卡佇，導致之後重啟機器人才行繼續播放。

## 3.限制影片時間大小



因為播放歌曲要經過下載+轉換成PCM，檔案過大會導致伺服器lag

## 4.擷取正確的youtube網址



如果想放的音樂是在清單內，網址後面會多出&list….，這會讓youtube\_dl無法下載而導致程式卡住

## 5.delete&swap A~B與A的索引

為了讓給予範圍與單獨的值都可以正常運行，程式碼主要用範圍的寫法，也就是取值的時候都使用  將值從final這個list中取出來，

因此即便是只有給一個值也會把它包入，才會正常運行

## 6.給值錯誤時要中斷程式

例如在delete時要刪除A~B，但卻給了C~D不存在音樂清單中的索引，如果不中斷，會因為我寫的程式碼關係造成音樂清單被刪除一部份，所以我使用，來做為檢查到錯誤時中斷的手段。

實作心得

此次作業算是第一次接觸到需要使用官方的文件來寫程式碼，是一個蠻大的挑戰，雖然基本的架構是根據網路上的教學學會的，然而很多功能怎麼找也找不到有中文的教學，例如音樂的播放，由於discord的python程式碼有更新過，幾乎找的到的音樂播放教學的程式碼都無法使用了，幸虧在stackoverflow上找到了一個外國人的解法，也就是youtube\_dl這個套件，還有在官方內建的函式也吃盡了苦頭，究竟這個類別，它有什麼樣的函式、參數，函式需要的、返回的東西又是什麼，即便有官方的文件也看得困惑，所以雖然成品跟市面上的音樂機器人相比可能不夠好，但也相當有成就感。

# 工作分配

四訊四乙 劉彥震:音樂程式碼撰寫

職四訓三甲 林永森:簡報撰寫，文字程式碼撰寫

# 參考文獻&學習來源

Code a discord bot 架構主要來自於這裡

<https://www.youtube.com/playlist?list=PLSCgthA1Anif1w6mKM3O6xlBGGypXtrtN>

官方API文件，音樂的暫停、停止、恢復播放、音量是使用官方內建的函式

<https://discordpy.readthedocs.io/en/latest/migrating.html?highlight=volume#voice-changes>

Queue說明(asyncio的queue一樣)

<https://docs.python.org/zh-cn/3.7/library/asyncio-queue.html>

播放音樂的架構主要是參考這個

<https://xbuba.com/questions/53605422>

然而上面的程式過舊已不可使用，所以我發現以下網址的播放方法

<https://stackoverflow.com/questions/56031159/discord-py-rewrite-what-is-the-source-for-youtubedl-to-play-music>

youtube\_dl的下載參數

以下網址用ctrl-F搜尋 EMBEDDING YOUTUBE-DL 發現

<https://github.com/ytdl-org/youtube-dl#options>

youtube\_dl支援的網站

<https://github.com/ytdl-org/youtube-dl/blob/master/docs/supportedsites.md>

ffmpeg放入自建的虛擬環境，將ffmpeg/bin下的三個檔案放入Scripts中即可

<https://stackoom.com/question/256ip/%E6%89%BE%E4%B8%8D%E5%88%B0ffprobe%E6%88%96avprobe-%E8%AF%B7%E5%AE%89%E8%A3%85%E4%B8%80%E4%B8%AA>

ffmpeg下載

<https://www.ffmpeg.org/>

PCM介紹

<https://read01.com/zh-tw/3Q7Nd3.html#.Xgd_DkczaUk>