

Python 期末專題計畫書

Discord 查詢股價機器人

學生： D0952489 陳政邦

D0988532 邢婕綸

D0957259 徐康

D0960582 白昱豪

中華民國 110 年 12 月

一、專題名稱與簡介

專題名稱:Discord 查詢價機器人

簡介:透過輸入指令及證券號，能在 Discord 上查詢台灣所有上市公司的即時股價。

1. 成員：

D0952489 陳政邦

D0988532 邢婕綸

D0957259 徐康

2. 動機與背景：

我們發現 discord 這個通訊軟體我們很長使用，不論是玩線上遊戲的即時通訊又或者是在裡面與朋友聊天甚至前陣子有些學校會用此軟體進行遠端上課等等…

相信大家也都知道這個程式與其他通訊軟體的一個差異就在於 discord 能夠利用機器人來幫助我們管理頻道，因此藉由這次作業我們想結合 dc bot 與查詢即時股價的功能，方便我們使用。

3. 參考資料：

[twstock-台灣股市股票價格擷取](#)

[上市證券編碼](#)

[Discord API](#)

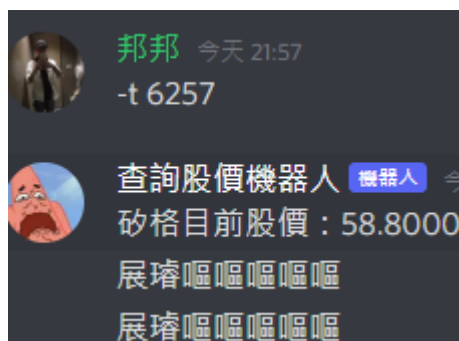
[CSDN 論壇-如何用 discord py 製作一個簡單的機器人](#)

[Discord.py 創建機器人詳細教學](#)

[巴哈- Discord bot 教學](#)

二、成果

1. 功能與成果展示：在頻道內輸入欲查詢上市公司證券號，機器人回報目前即時股價。(如下圖)



(自動刪除指令訊息，並發出指令內指定的內容。)

2. 設計說明：

1. 設計理念:方便查詢上市公司股價

2. 用到的技巧與資料:

1. twstock 用來抓取上市公司即時股價

2. discord 用來串接 discord 跟 python 的，並且使用 discord 的 API

3. asyncio 使用 async 及 await 語法來支援非同步的執行，也就是在 Python 的函式(Function)前加上 async 關鍵字，來定義協程(coroutine)，在其中定義非同步的任務清單，接著，透過事件迴圈(Event Loop)來進行不同任務間的切換執行，達到非同步的執行效果。

非同步處理就是能夠在等待回應的同時，可以先去做其它的任務，不會因為等待而停擺，總之主要功能是「加速程式的運行。」

4. nest_asyncio 用來修補 asyncio 的缺失，在事件迴圈為已經在運行無法運行的任務，程式為了結果就會一直跑，造成鬼

打牆的情況，試圖這樣做會導致錯誤，直接取消正在鬼打牆的程式。

5. 為了讓 BOT 可以 24 小時運行又不佔用電腦資源的方法，利用線上編輯器 replit 加上可讓程式持續運行的開源程式碼 keep_alive，使得程式可以持續的運行，但是這有一個缺陷是，replit 可能為了清除這類持續運行的程式造成對該網站的負擔，會不定期中斷這類的程式運行。

因此我們用另一個監控網站 uptimerobot 去監控 BOT，每五分鐘監控一次，只要發現 BOT 不在線上了，就會發 email 通知我，並且主動的重新運行該程式，以此方式來達到 24 小時運行程式的目的。

6. 使用網路上一個方式可以讓程式持續運行「keep_alive」

使用了 flask 建立了一個伺服器，並且把 python 跟使用 flask 建立的伺服器產生連接，0000 代表本機意思，8080 代表 python 默認的端口，

最後使用 Thread 允許伺服器被別人連接，uptimerobot 就是監視由 flask 建立的伺服器地址去監控程式有沒有運行的。

7. 簡單加密

利用 env 檔建立一個 TOKEN=" " 的格式再導入主程式中。

8. 使用到的資料：

[twstock-台灣股市股票價格擷取](#)

[上市證券編碼](#)

[Discord API](#)

[監控網站](#)

[線上編輯器](#)

[BOT 24 小時在線上教學](#)

9. 總共寫了 5843 行程式碼

10. Demo 影片之 YouTube 連結

<https://youtu.be/7tyVTGJsfrI>

三、心得

1. 團隊分工

陳政邦: 寫 code、書面報告

邢婕綸: 查資料、書面報告

徐康: 查資料、書面報告

白昱豪: 查資料、書面報告

2. 學習心得

陳政邦:

在大一時接觸 C 語言就發現我喜歡寫 code，因此當時看到有這門課就趕快選了，但因為本科系是自控，大二三電一工一次來，所以我提早在 10 月多時就已經把 python 的線上課程學到物件導向，但物件導向的概念我懂了，卻不太會使用。

後來就煩惱要寫甚麼專題時，剛好想到老師在套件課程中使用的套件，於是開始往這套件鑽研，就寫出了這套程式，但這套程式我認為程式碼非常的醜，因為都是我硬寫出來的，我相信有更好的解決方式，但現實不允許我花大量時間在上面鑽研，三電一工太難了

就勉強寫出了個 BOT，值得一提的是我一開始學 python 時好奇要怎麼跟 C 一樣把程式碼輸出為執行檔，我網路上邊查邊用，結果意外的把我的 spyder 弄壞打不開，後來又花一整個晚上找資料才發現是 pyqt5 的版本超過我的 spyder 版本導致無法打開 spyder，後來經過 cmd 更改 pyqt5 的版本就可以開了，不然我就得重灌電腦了。

我想嘗試看看轉資工系，希望可以成功。

白昱豪：

經過這次的專題，讓我更加熟悉 python 日常應用，讓我平常使用的 Discord 更加多元化，更加方便，也感謝組長組員的教導，這次的專題我成長了不少。

徐康：

我是資訊系的學生，因為選了這堂 Python 通識課，也因為這次的分組報告，才讓我第一次接觸到做專題，我在之前完全不懂這是幹嘛的，也有一點懼怕他，然而在分組過後，我認識到了其他組員，其中也有很厲害的組員，而我也向他請教許多問題，算是意料外的收穫，雖然我對這個語言還不是很熟悉，我們的專題大多是靠組裡的高手完成的，不過我認識到了許多程式厲害的人，也從中學習到了

許多寶貴的經驗，像是與組員分工合作，討論問題等等…我很高興有參加這門課，讓我多了一個去了解專題研究的機會，這樣在未來我要做專題時肯定有所幫助，能讓我更快進入狀況。

邢婕綸：

自從大一接觸過 python 就一直對程式語言蠻有興趣的，所以就來選了這門課，上過之後發現課程內容比我之前上的加深蠻多的，課程內容都很不錯讓我學到很多，影片中跳出問題的方式也讓我更專心上課，但有些例題對我來說還是蠻難的，還是希望這類型的課能夠是面對面教學，才能馬上提問。

3. 建議：

陳政邦:我很喜歡這類課程，希望未來可以多開其他語言的這類課程，像是 C#，C++，或是 Java 甚至是寫網頁的 html、CSS 這種的，如果有開，沒有衝堂的情況下我都會選。