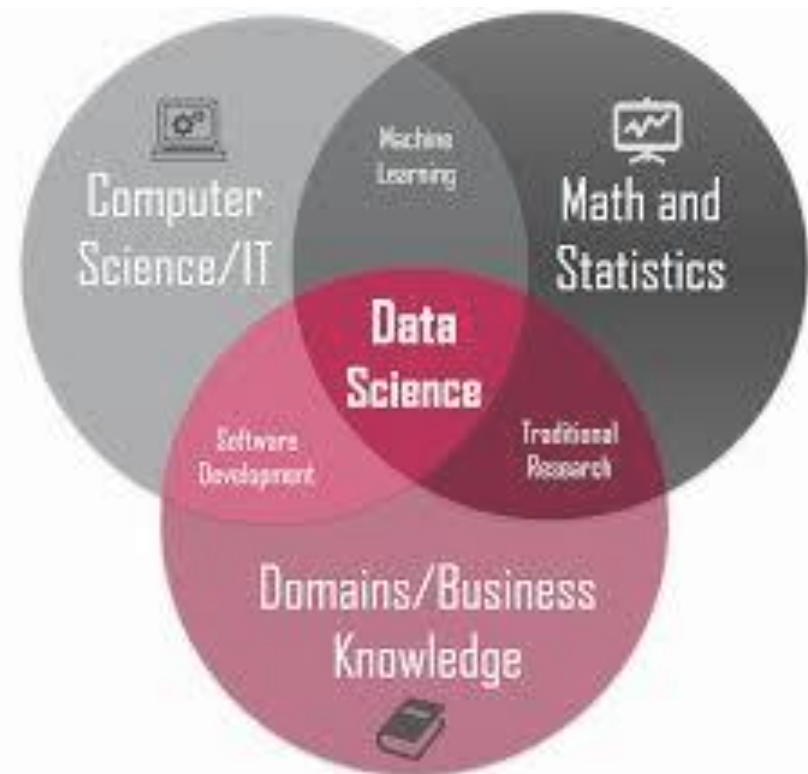


AI & DS



U03B-網路爬蟲進階模組實戰

Data
Science

Artificial
Intelligence

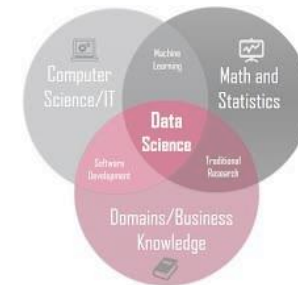
2023.11_V1.1
Machine
Learning

Deep
Learning

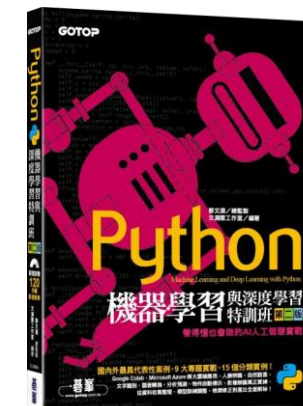
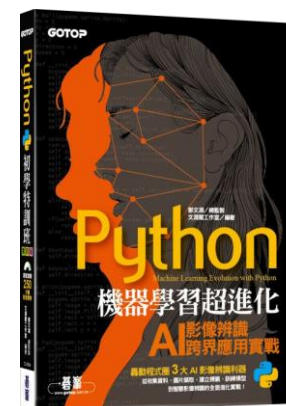
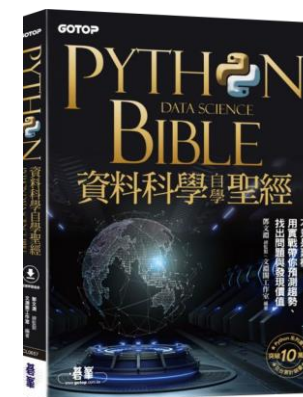
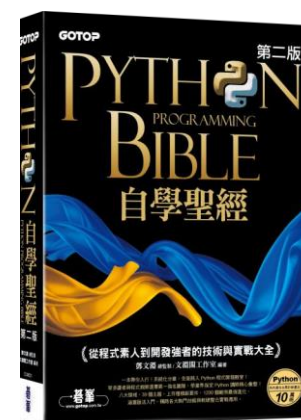
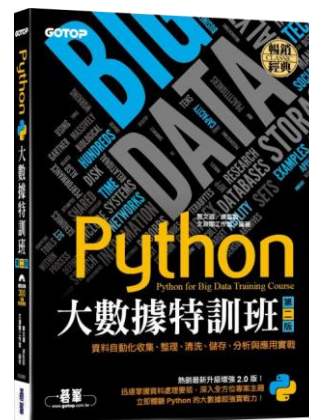
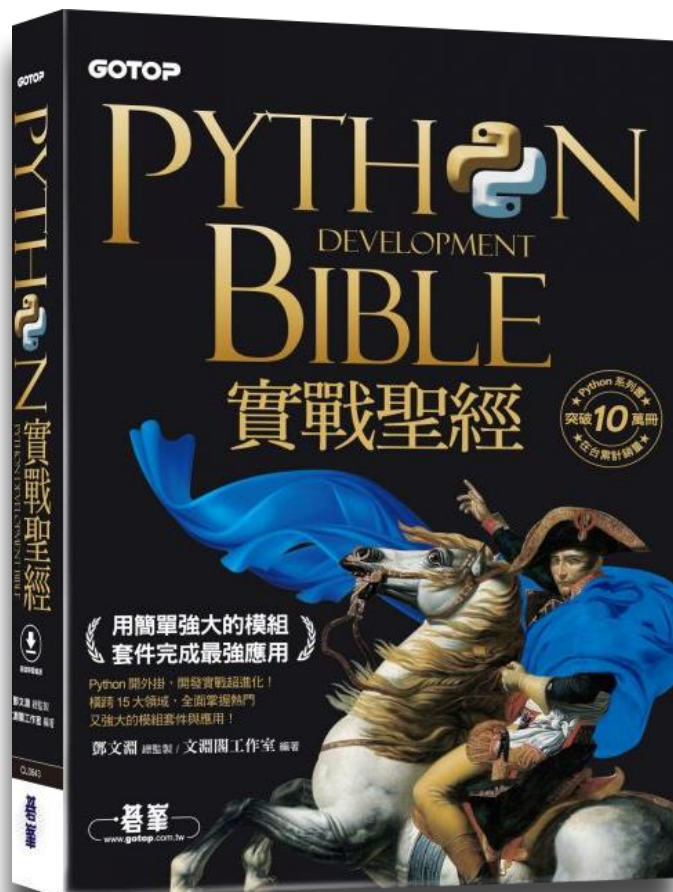
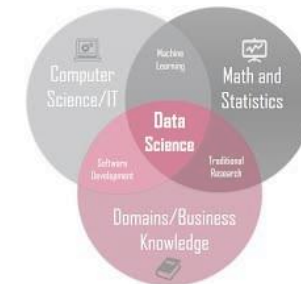
Statistics

單元大綱

- 新聞資料收集
- 多媒體資料收集
- 語音資料處理
- 文字翻譯
- 股市資料收集



參考書目



【本集參考書目】

Python實戰聖經：用簡單強大的模組套件完成最強應用

<https://reurl.cc/NZQMbe>

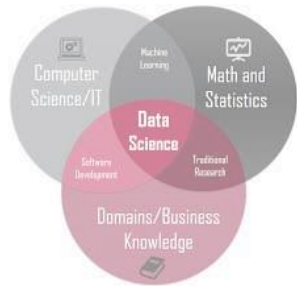
[資料引用]文淵閣工作室

Part 1

新聞資料收集

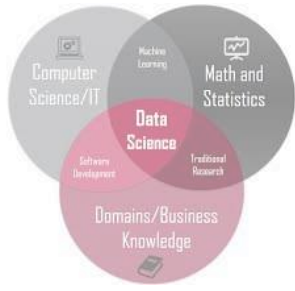


網路爬蟲資料收集



- 新聞爬取模組
 - scraparazzie: Google News新聞爬取
 - Newspaper3k: 爬取全球新聞
 - technews-tw: 台灣科技新聞爬取

scraparazzie : Google News新聞爬取



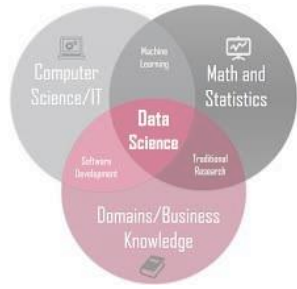
- 安裝模組：

```
!pip install scraparazzie
```

- 使用模組：

```
from scraparazzie import scraparazzie
```

建立新聞物件



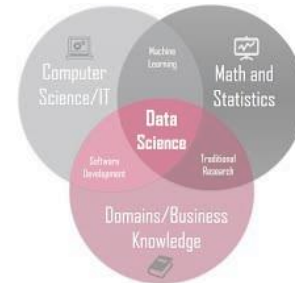
- 語法：

新聞物件 = scraparazzie.NewsClient(參數1 = 值1, 參數2 = 值2,)

scraparazzie物件的參數有：

- language: 設定語言, 預設值為english (英文)。
- location: 設定地區, 預設值為United States (美國)。
- topic: 設定新聞主題。
- query: 設定新聞包含的文字。
- max_results: 設定爬取的最大新聞數量, 預設值為5。系統能爬取的最大數量為100, 若此參數大於100則最大爬取數量為100。

顯示台灣商業新聞資料



□ 例如：顯示5則台灣商業新聞資料

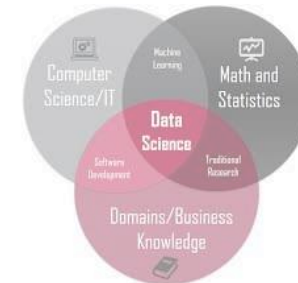
使用模組

```
from scraparazzie import scraparazzie
```

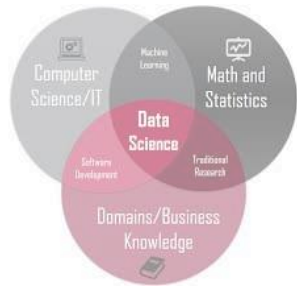
建立新聞物件

```
client = scraparazzie.NewsClient(language='chinese  
traditional', location='Taiwan', topic='Business',  
max_results=5) client.print_news() # 顯示資料
```


參數查詢



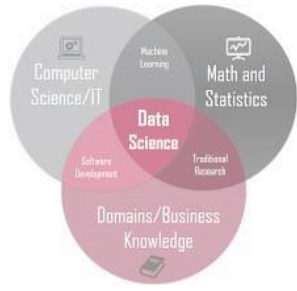
- # 新聞物件.languages
- # 新聞物件.locations
- # 新聞物件.topics
- # 新聞物件.get_config()
- client.languages



取得新聞資料

```
# 串列資料=新聞物件.export_news()  
client.export_news()
```

```
# 取得目前物件各參數的設定值  
新聞物件.get_config()
```

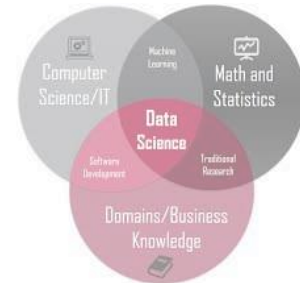


取得台灣股市新聞資料

- 範例：取得5則台灣股市新聞資料並顯示

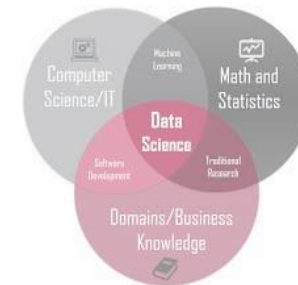
```
from scraparazzie import scraparazzie
client = scraparazzie.NewsClient(language = 'chinese traditional',
location = 'Taiwan', query = '股市', max_results = 5)
items = client.export_news()
for i, item in enumerate(items):
    print('第 ' + str(i+1) + ' 則新聞:')
    print('新聞標題:' + item['title'])
    print('新聞機構:' + item['source'])
    print('新聞連結:' + item['link'])
    print('新聞時間:' + item['publish_date'])
```

練習：取得**5**則台灣口罩新聞資料並顯示



```
from scraparazzie import scraparazzie
client = scraparazzie.NewsClient(language = 'chinese traditional',
l ocation = 'Taiwan', query = '口罩', max_results = 5)
items = client.export_news()
print(len(items))
for i, item in enumerate(items):
    print('第 ' + str(i+1) + ' 則新聞:')
    print('新聞標題:' + item['title'])
    print('新聞機構:' + item['source'])
    print('新聞連結:' + item['link'])
    print('新聞時間:' + item['publish_date'])
    print('=====')
```

Newspaper3k : 爬取全球新聞



- 安裝模組：

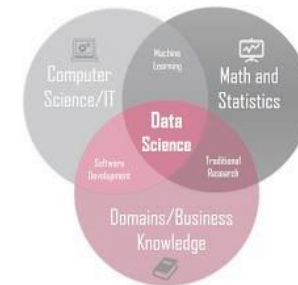
```
!pip install newspaper3k
```

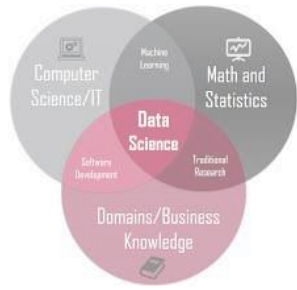
- 使用模組：

```
import newspaper
```

參數查詢

- `# newspaper.popular_urls()`
- `# newspaper.languages()`





建立新聞物件

- 語法：

新聞物件 = newspaper.build('新聞網站網址', language='語言代碼')

□ 例如, 新聞物件名稱為p a p e r, 新聞網站為中文的自由時報。

```
# 建立新聞物件
```

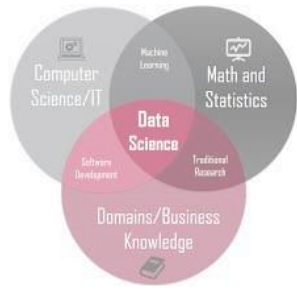
```
paper = newspaper.build('http://www.ltn.com.tw/',  
language='zh')
```

```
# print('新聞連結:')
```

```
for i, article in
```

```
    enumerate(paper.articles): print(i+1,  
    article.url)
```


Artccle 類別

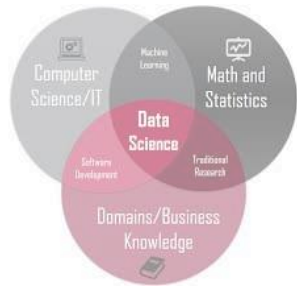


- 使用模組:

```
from newspaper import Article
```

- 建立Article物件:

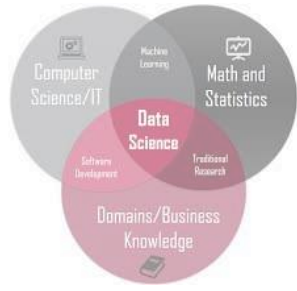
```
Article變數 = Article('單一新聞網址')
```



Artccle 取得單一新聞

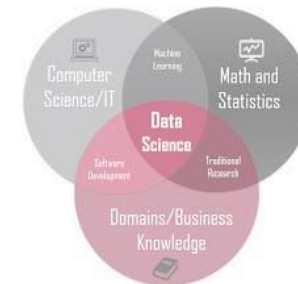
```
from newspaper import Article
# 自由時報
url = 'https://news.ltn.com.tw/news/life/breakingnews/3649202'
article = Article(url)
article.download()
print(article.html) # 原始碼
```

Artccle 新聞解析



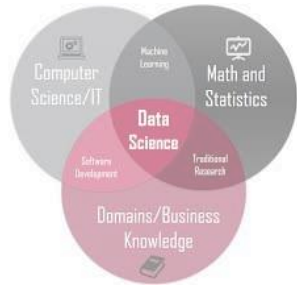
```
article.parse()  
print('【新聞標題】')  
print(article.title)  
print('【新聞內容】:')  
print(article.text)  
print('【新聞日期】')  
print(article.publish_date)
```

練習：取得蘋果日報新聞



```
from newspaper import Article
url = https://www.appledaily.com.tw/
article = Article(url)
article.download()
article.parse() # parse() 解析
print('新聞標題:')
print(article.title)
print('新聞內容:')
print(article.text)
print('新聞日期:')
print(article.publish_date)
```

technews-tw : 台灣科技新聞爬取



- 安裝模組：

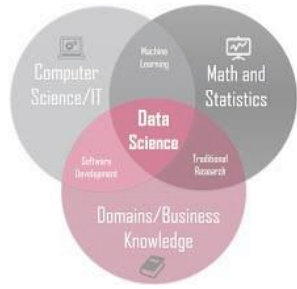
```
!pip install technews-tw
```

- 使用模組：

```
from technews import TechNews
```

- 取得今天科技新聞：

```
新聞物件 = TechNews("新聞來源代碼").get_today_news()
```



取得 **inside** 當天發佈的新聞

□ 例如:取得 **inside** 當天發佈的新聞

```
from technews import TechNews
```

```
# 取得 inside 當天發佈的新聞
```

```
# news =
```

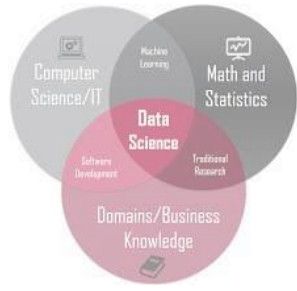
```
TechNews('business').get_today_news() # news
```

```
= TechNews('orange').get_today_news() # news
```

```
= TechNews('ithome').get_today_news() news =
```

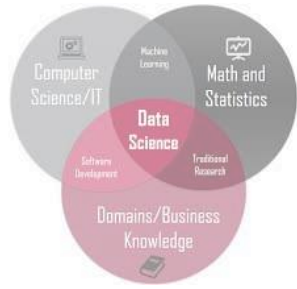
```
TechNews('inside').get_today_news()
```

```
print(news)
```



範例：逐筆顯示新聞資料

```
news =  
TechNews('inside').get_today_news()  
content = news['news_contents']  
for key in content:  
    mononews = content[key]  
    print('新聞標題:', mononews['title'])  
    print('新聞連結:', mononews['link'])  
    print('發布日期:', mononews['date'])  
    print('=====')  
    )
```

取得N天內科技新聞

- 語法：

新聞物件 = TechNews("新聞來源代碼").get_news_by_page(N)

- 例如：取得 3 天內 inside 發布的所有科技新聞

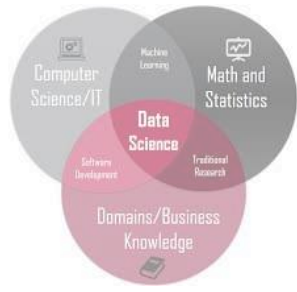
```
# news3 = TechNews('business').get_news_by_page(3)
# news3 = TechNews('orange').get_news_by_page(3)
# news3 = TechNews('ithome').get_news_by_page(3)
news3 =
TechNews('inside').get_news_by_page(3)
print(news3)
```

Part 2

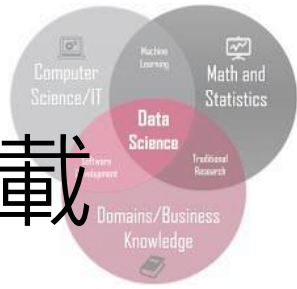
多媒體資料收集



多媒體圖片影片下載



- 圖片下載模組
 - google-images-download: Google圖片下載
 - bing-image-downloader: Bing圖片下載
- 下載Youtube影片



google-images-download : Google圖片下載

- 安裝模組：

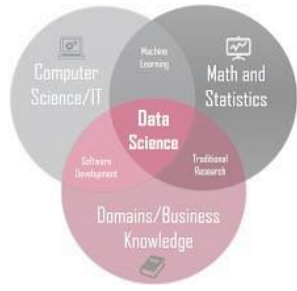
```
!pip install google-images-download-jeng
```

- 使用模組：

```
from google_images_download_jeng import google_images_download
下載物件變數=google_images_download.googleimagesdownload()
arguments = {
    "keywords": "圖片主題",
    參數1: 值1,
    參數2: 值2,
    .....
}
下載物件變數.download(arguments)
```

模組參數設定

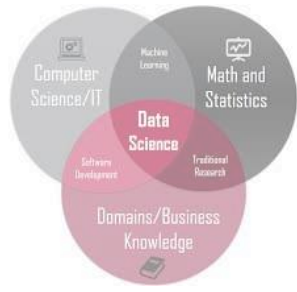
| 參數名稱 | 參數意義 |
|------------------|---|
| keywords | 設定圖片主題。這是必要參數，如果未設定會顯示「Keywords is a required argument」提示訊息。 |
| limit | 設定下載最大圖片數量。預設值為 100。 |
| format | 設定圖片的附加檔名，可設定的值有 jpg、gif、png、bmp、svg、webp、ico、raw。 |
| color_type | 設定圖片型態，可設定的值有 full-color（全彩）、black-and-white（黑白）、transparent（透明）。 |
| size | 設定圖片尺寸，可設定的值有 large、medium、icon、>400*300、>640*480、>800*600、>1024*768、>2MP、>4MP、>6MP、>8MP、>10MP、>12MP、>15MP、>20MP、>40MP、>70MP。 |
| output_directory | 設定下載圖片儲存的資料夾路徑。預設值為 <downloads> 資料夾。 |
| print_urls | 布林值。True 表示執行時會顯示下載圖片的原始圖片路徑，False 則不顯示。 |
| save_source | 設定記錄圖片資訊文字檔的檔案名稱。 |
| silent_mode | 布林值。True 表示執行時不顯示任何訊息，False 則會顯示訊息。 |



下載指定主題圖片

- 範例：下載5張貓熊圖片，存在myimage目錄中（語法有誤）

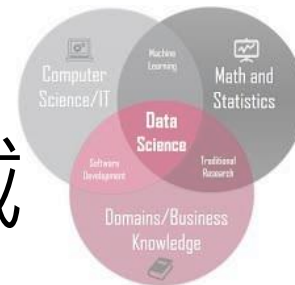
```
from google_images_download_jeng import google_images_download
response=google_images_download.googleimagesdownload()
arguments={
    "keywords":"貓熊",
    "limit":5,
    "print_urls":True,
    "output_directory":"myimage",
    "save_source":"data"
}
response.download(arguments)
```



練習：下載5張企鵝圖片，存在雲端硬碟中

```
from google_images_download_jeng import  
google_images_download  
response=google_images_download.googleimagesdownload()  
arguments={  
    "keywords": "企鵝",  
    "limit":5,  
    "print_urls":True,  
    "output_directory": "雲端硬碟路徑",  
    "save_source": "data"  
}  
response.download(arguments)
```


bing-image-downloader : Bing圖片下載

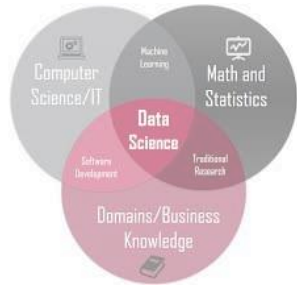


- 安裝模組：

```
!pip install bing-image-downloader==1.0.4
```

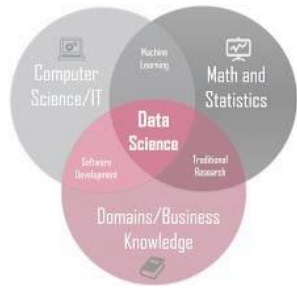
- 使用模組：

```
from bing_image_downloader import downloader
downloader.download(
    '關鍵字',
    limit = 數值,
    output_dir = '圖片儲存路徑',
    adult_filter_off = 布林值,
    force_replace = 布林值,
    timeout = 數值
)
```



模組使用參數

- 關鍵字：要下載的圖片主題關鍵字，如海灘、街景等。
- limit：設定下載圖片的最大數量，預設值為100。
- output_dir：設定下載圖片的儲存路徑，預設值為「dataset」。
- adult_filter_off：設定是否過濾掉成人圖片，True表示不下載成人圖片，False表示下載包含成人圖片。預設值為True。
- force_replace：設定若儲存圖片資料夾已經存在時，是否覆蓋該資料夾，True表示覆蓋，False表示不覆蓋。預設值為False。
- timeout：設定連結圖片網頁的最大連結時間，單位為「秒」。預設值為60。

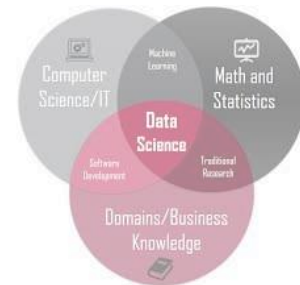


下載指定主題圖片

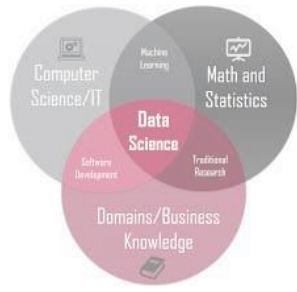
- 範例:下載5張放煙火圖片, 存在bingimage目錄中

```
from bing_image_downloader import downloader
downloader.download(
    '放煙火',
    limit = 5,
    output_dir = 'bingimage',
    adult_filter_off = True,
    force_replace = True,
    timeout = 20
)
```

練習：下載5張超跑圖片



```
from bing_image_downloader import downloader
downloader.download(
    '超跑',
    limit = 5,
    output_dir = 'bingimage',
    adult_filter_off = True,
    force_replace = True,
    timeout = 20
)
```



Pytube : Youtube影片下載

- 安裝模組：

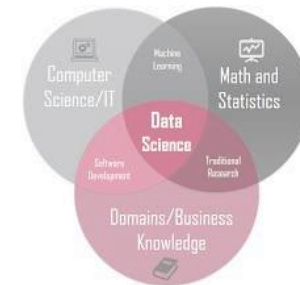
```
!pip install pytube
```

- 使用模組：

```
from pytube import YouTube
```

```
yt = YouTube\('https://www.youtube.com/watch?v=27ob2G3GUCQ'\)
```

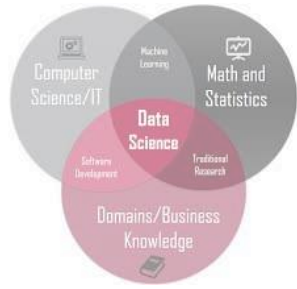
影片格式分析



```
print(yt.title)
print(yt.streams)
# 分析
for st in yt.streams:
    print(st)
```

streams取得影片格式的方法

| 方法 | 功能 | 語法範例 | |
|----------|---------------|---|--|
| first() | 傳回第一個影片格式 | <code>yt.streams.first()</code> | |
| last() | 傳回最後一個影片格式 | <code>yt.streams.last()</code> | |
| filter() | 傳回符合指定條件的影片格式 | <code>yt.streams.filter(subtype='mp4')</code> | |



下載影片

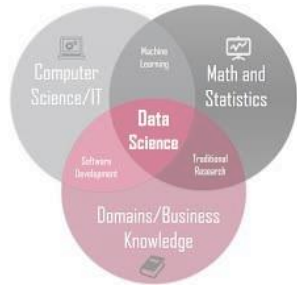
語法：

影片物件變數.streams.取得影片方法.download(影片儲存路徑)

: 為first()或last()或filter()。

: 下載的影片儲存在此參數設定的資料夾中，預設值為程式所在資料夾。在Colab中，最好設定為雲端硬碟。

- 例如使用yt影片物件變數下載第一個影片，下載的影片存於<youtube>資料夾：
 - `yt.streams.first().download("youtube")`

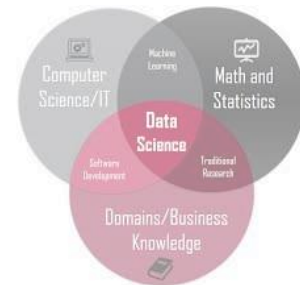


設定影片格式篩選條件

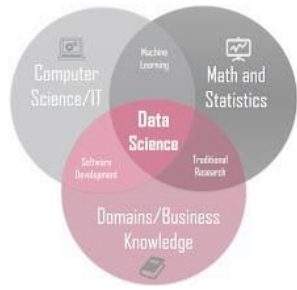
- 語法
`yt.streams.filter(條件一=值一, 條件二=值二,).處理方法`
- **filter()**的條件

| 條件 | 功能 | 語法範例 |
|--------------------------|-------------------|-------------------------------|
| <code>progressive</code> | 篩選同時具備影像及聲音的格式 | <code>progressive=True</code> |
| <code>adaptive</code> | 篩選只具有影像或聲音其中之一的格式 | <code>adaptive=True</code> |
| <code>subtype</code> | 篩選指定影片類型的格式 | <code>subtype='mp4'</code> |
| <code>res</code> | 篩選指定解析度的格式 | <code>res='720p'</code> |

下載篩選的影片



- `print(len(yt.streams))` # 共18部
`st=yt.streams.filter(subtype='mp4',progressive=True)`
`print(len(st))` # 2部
- `for s in st:`
 `print(s)`
- `st=yt.streams.filter(adaptive=True)`
`print(len(st))` # 15 部

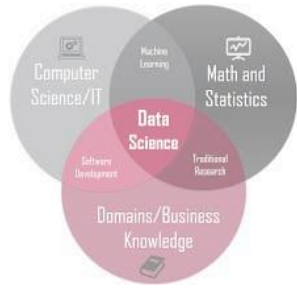


練習：下載一部**Python**教學影片

```
from pytube import YouTube
yt =
YouTube\('https://www.youtube.com/watch?v=uPKgQ3FoVtY'\)
print(yt.title)

st=yt.streams.filter(subtype='mp4', res='360p',progressive=True)
print(len(st)) # 2部
print(st)

# 下載第1部
st.first().download('youtube')
```



下載YouTube 播放清單的影片

- 匯入Pytube模組Playlist類別的語法：

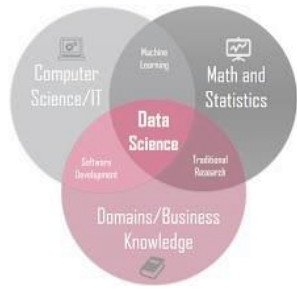
```
from pytube import Playlist
```

- 以Pytube模組中Playlist類別建立物件：

```
播放清單物件變數 = Playlist(播放清單網址)
```

- 例如：

```
p = Playlist\('https://www.youtube.com/watch?v=wdH26D8Ssw&list=PLg\_r40S5jeMIvAaq90ZweLCx8\_N10x0\_Q'\)
```



練習：下載自訂播放清單

- 請下載以下網址的清單影片：

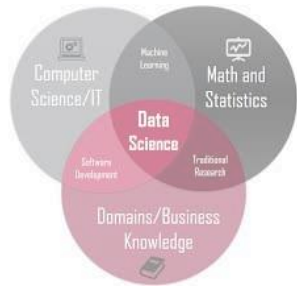
https://www.youtube.com/watch?v=wdH26D8Ssw&list=PLg_r40_S5jeMlvAaq9OZweLCx8_N1OxO_Q

- 或自行下載有興趣的清單影片。

Part 3

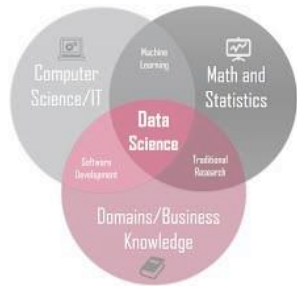
語音資料處理





語音文字處理

- 文字語音轉換模組
 - gTTS模組: 文字轉語音
 - SpeechRecognition模組: 語音轉文字
- 文字翻譯
 - google_trans_new模組: 文字翻譯



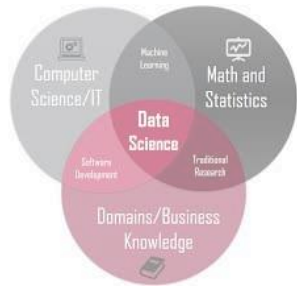
gTTS模組：文字轉語音

- 安裝模組：

```
!pip install gtts
```

- 使用模組：

```
import gtts  
tts=gtts.gTTS(text='線上翻譯',lang='zh-tw',slow=True) tts.save('tts.mp3')
```



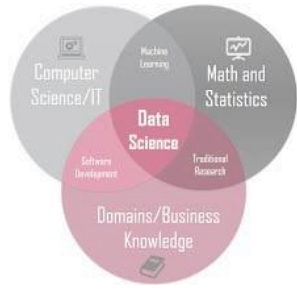
語系查詢

- 語系查詢

`gtts.lang.tts_langs()`

| 語言代碼 | 文字 |
|-------|------|
| zh-tw | 繁體中文 |
| en | 英文 |
| ja | 日文 |
| fr | 法文 |

| 語言代碼 | 文字 | |
|-------|------|--|
| zh-cn | 簡體中文 | |
| de | 德文 | |
| ko | 韓文 | |
| es | 西班牙文 | |



語音播放

- 語音播放語法：

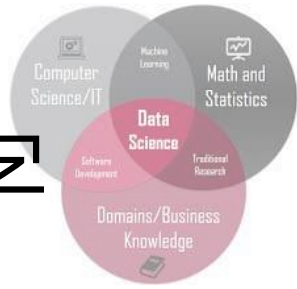
```
import IPython.display as display
display.Audio(檔案路徑, autoplay=True)
```

- 範例：將一段文字轉換為語音後存檔，最後讀出語音。

```
txt = 'gTTS可以透過線上翻譯，將文字轉換為語音，並將語音存檔'
tts = gtts.gTTS(text=txt, lang='zh-tw')
tts.save('gtts.mp3')
display.Audio('gtts.mp3', autoplay=True)
```

練習：將「How are you!」轉換成語音並播放

```
import gtts
tts=gtts.gTTS(text='How are you! ',lang='en')
tts.save('您好.mp3')
import IPython.display as display
display.Audio('您好.mp3',autoplay=True)
```



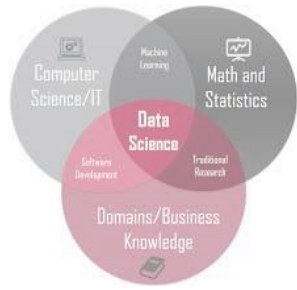
SpeechRecognition模組：語音轉文字

- 安裝模組：

```
!pip install SpeechRecognition
```

- 使用模組：

```
import speech_recognition as sr  
# 建立語音辨識物件  
r=sr.Recognizer()
```



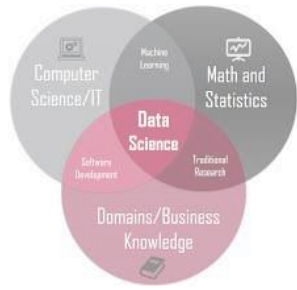
對語音檔進行語音辨識

- 讀取語音語法：

```
with sr.WavFile(語音檔路徑) as 檔案物件：  
    語音變數 = r.record(檔案物件)
```

- 語音辨識語法：

```
語音辨識物件.recognize_google(語音變數[, language=語系])
```



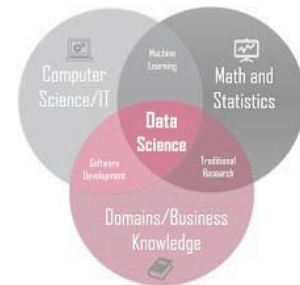
下載<record1.zip>檔並解壓縮

需要 <record1.wav>檔

```
!gdown '1y5S1oM_6xujPva6_a1U6aCNw7vqJxNWs' --output  
record1.zip
```

```
!unzip record1.zip
```

範例：讀取語音檔並將語音轉換為文字

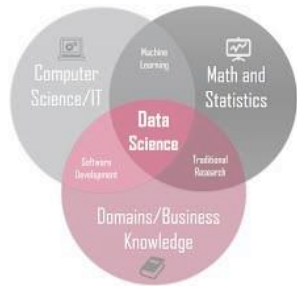


讀取語音檔

```
with sr.WavFile('record1.wav') as  
    source: audio=r.record(source) # 讀取語  
    音內容
```

語音辨識

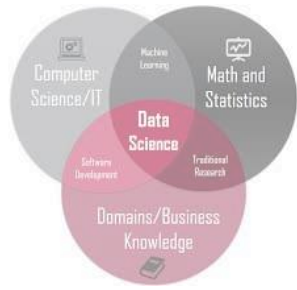
```
try:  
    word=r.recognize_google(audio,language='zh-tw')  
    print(word)  
except:  
    print("語音辨識失敗!")
```

語音格式轉換

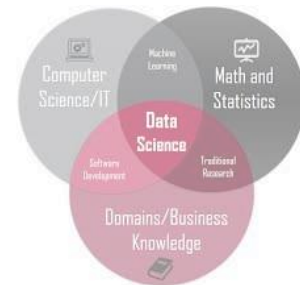
- 安裝pydub:
`!pip install pydub`
- 匯入pydub的AudioSegment模組:
`from pydub import AudioSegment`
- 讀取聲音檔:
`聲音變數 = AudioSegment.from_file(聲音檔案)`
- 轉換聲音檔:
`聲音變數.export(目標檔案, format =聲音格式)`

範例：語音格式轉換



```
from pydub import AudioSegment
sound = AudioSegment.from_file('tts.mp3')
sound.export('tts.wav', format='wav')
```

使用麥克風進行語音辨識

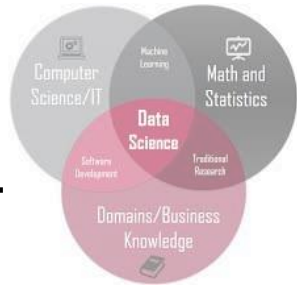


```
record_audio('record.webm') # 線上錄音函式

sound = AudioSegment.from_file('record.webm')
sound.export('record.wav', format = 'wav')

r = sr.Recognizer()
with sr.WavFile('record.wav') as source:
    audio = r.record(source)

try:
    word = r.recognize_google(audio, language='zh-TW')
    print('語音辨識結果:\n' + word)
except:
    print('語音辨識失敗！')
```



練習：使用麥克風錄音一段英文並轉換為文字

```
record_audio('myrecord.webm')
sound = AudioSegment.from_file('myrecord.webm')
sound.export('myrecord.wav', format = 'wav')

r = sr.Recognizer()
with sr.WavFile('myrecord.wav') as source:
    audio = r.record(source)

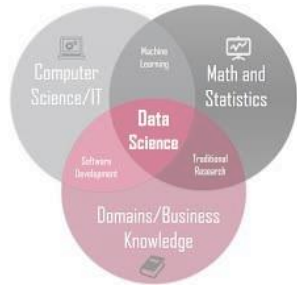
try:
    word = r.recognize_google(audio, language='en')
    print('語音辨識結果:\n' + word)
except:
    print('語音辨識失敗！')
```

Part 4

文字翻譯



google_trans_new 模組：文字翻譯



- 安裝模組：

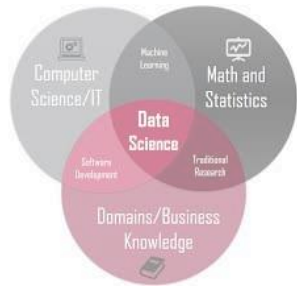
```
!pip install google_trans_new
```

- 修正程式碼：

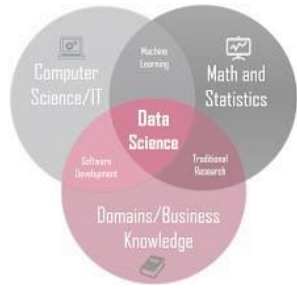
```
!sed -i
```

```
"s/response = (decoded_line + '']/response = decoded_line/g" \  
  /usr/local/lib/python3.7/dist-packages/google_trans_new/  
google_trans_new.py
```

使用 google_trans_new 模組



- `from google_trans_new import google_translator`
- `# 建立翻譯物件`
- `translator=google_translator()`

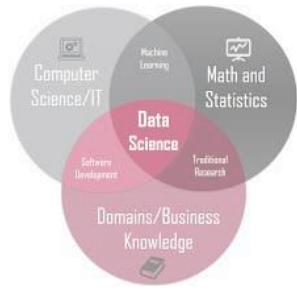


翻譯文字

- 語法：

翻譯結果變數 = translator.translate(要翻譯的文字, lang_src=語言代碼, lang_tgt=語言代碼, pronounce=布林值)

- 要翻譯的文字：代表要轉換的文字字串。
- lang_src：要轉換文字的語言代碼。如果未設定，系統會自動偵測。
- lang_tgt：翻譯後文字的語言代碼。預設值為「en」，即翻譯為英文。
- pronounce：設定是否顯示拼音。預設值為False，若設為True會顯示翻譯前及翻譯後的文字拼音。

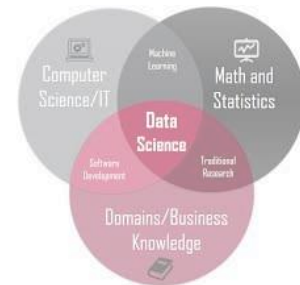


範例：將中文翻譯文字為日文

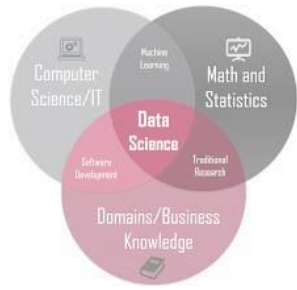
- 將中文「今天天氣很好」翻譯為日文並顯示拼音，將翻譯結果存於word變數之中。

```
word = translator.translate('今天天氣很好',  
lang_src='zh-TW',lang_tgt='ja',  
pronounce=True)
```

練習：將「大家好」翻譯成英文和日文



```
from google_trans_new import google_translator
translator=google_translator()
word=translator.translate('大家好',lang_tgt='en',pronounce=True)
print(word)
word=translator.translate('大家好',lang_tgt='ja',pronounce=True)
print(word)
```



應用：智慧讀報機

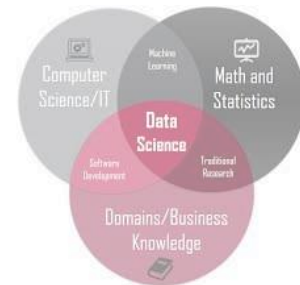
利用newspaper3k模組擷取世界知名新聞網站的新聞內容，由新聞網站隨機挑選一則新聞後將其翻譯為中文，顯示中文新聞內容並以語音讀出。

Part 5

股市資料收集



twstock模組：台灣股市資訊

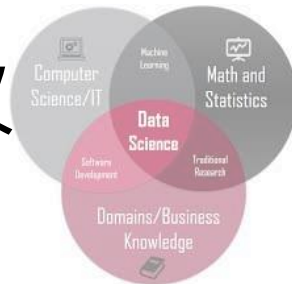


- 安裝模組：
 - `!pip install twstock`
- 使用模組：
 - `import twstock`
 - `stock = twstock.Stock('2317')`
 - `print(stock.price)`
 - `stock.raw_data`

Stock物件的屬性

| 屬性 | 說明 | 屬性 | 說明 |
|-----------------|------------------------|--------------------|------|
| date | 日期 (datetime.datetime) | low | 最低價 |
| capacity | 總成交股數 (單位：股) | price | 收盤價 |
| turnover | 總成交金額 (單位：元) | close | 收盤價 |
| open | 開盤價 | change | 漲跌價差 |
| high | 最高價 | transaction | 成交筆數 |

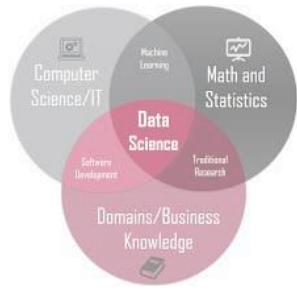
顯示最近**1**日開盤價、最高價、最低價及收盤價



```
print('日期:', stock.date[-1])  
print('開盤價:', stock.open[-1])  
print('最高價:', stock.high[-1])  
print('最低價:', stock.low[-1])  
print('收盤價:', stock.price[-1])
```

Stock物件的方法

| 方法 | 傳回資料 | |
|------------------------------------|-----------------|--|
| <code>fetch(西元年 , 月)</code> | 傳回參數指定月份的資料。 | |
| <code>fetch_31()</code> | 傳回最近 31 日的資料。 | |
| <code>fetch_from(西元年 , 月)</code> | 傳回參數指定月份到現在的資料。 | |



查詢股票即時交易資訊

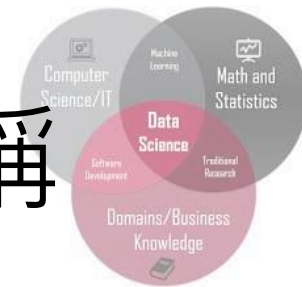
- 語法：

即時股票變數 = twstock.realtime.get('股票代號')

- 例如：設定即時股票變數為`real`，查詢鴻海股票（代碼 2317）的即時交易資訊。

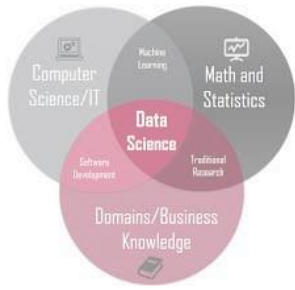
`real = twstock.realtime.get('2317')`

顯示 **twstock** 股票即時資料和股票名稱



```
if real['success']:
    print('股票名稱、即時股票資料:')
    print('股票名稱:', real['info']['name'])
    print('開盤價:', real['realtime']['open'])
    print('最高價:', real['realtime']['high'])
    print('最低價:', real['realtime']['low'])
    print('目前股價:',
',real['realtime']['latest_trade_price']) else:
    print('錯誤:' + real['rtmessage'])
```

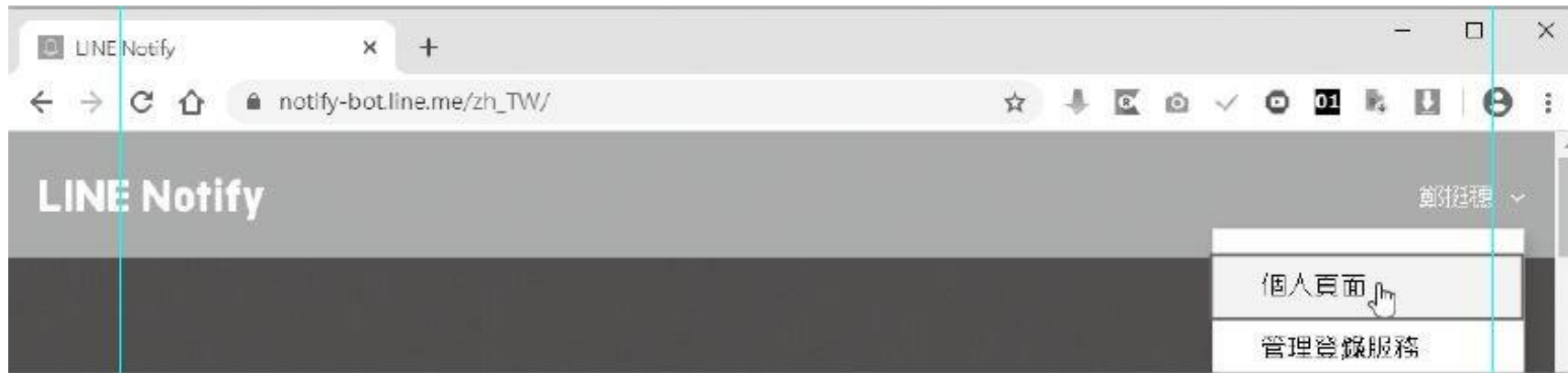
練習：每30秒讀取一次鴻海的「目前股價」



```
import twstock
import time
while True:
    realdata = twstock.realtime.get('2317')    #即時資料
    if realdata['success']:
        realprice = realdata['realtime']['latest_trade_p
rice']    #目前股價
        print(realprice)
        time.sleep(30)
```

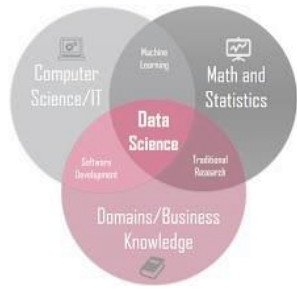
申請 LINE Notify 權杖

以 LINE Notify 傳送訊息必須先到 LINE Notify 官網取得權杖。開啟 LINE Notify 官網「https://notify-bot.line.me/zh_TW/」，以LINE 帳號登入後，在右上方「姓名」下拉選單中點選 個人頁面。



Line 中加入 Line Notify 為朋友

於行動裝置開啟LINE, 點選 主頁/搜尋好友。於 搜尋好友 頁面, 上方核選 ID, 搜尋欄輸入「@linenotify」後按 搜尋 鈕。出現LINE Notify, 按 加入 鈕即完成設定。



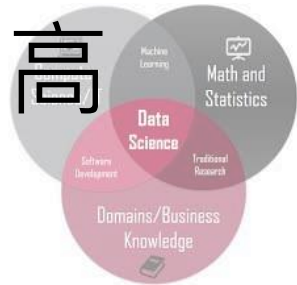
發送 LINE Notify 通知

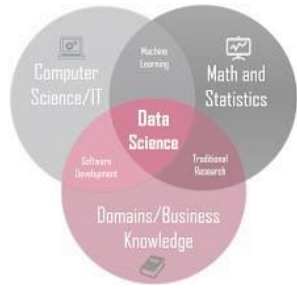
```
headers = {  
    'Authorization': 'Bearer ' + 權杖,  
    'Content-Type' : 'application/x-www-form-urlencoded'  
}  
payload = {'message': 文字訊息}  
通知變數 = requests.post(  
    https://notify-  
    api.line.me/api/notify', headers =  
    headers, params = payload)
```

練習：讀取鴻海的「目前股價」，若股價較高
就以

Line Noftiy 通知賣出。

每30秒鐘讀取鴻海公司即時股價一次，若股價達到用100元以上
(含)就發LINE訊息告知使用者可賣出股票。





應用：使用LINE監控即時股價

每5分鐘讀取鴻海公司即時股價一次，若股價達到用100元以上(含)就發LINE訊息告知使用者可賣出股票；若股價達到90元以下(含)就發LINE訊息告知使用者可買入股票。

為了避免使用者疏忽LINE訊息而傳送太多LINE訊息，設定最多只發送3次LINE訊息就結束程式。同樣的，若讀取即時股價產生錯誤，最多顯示3次錯誤訊息就結束程式。