

# Coding Styles (Python)

- 以喜馬拉雅音頻網站的網路爬蟲為例，展示 Python 程式的以下 5 種應用 (寫法)
- Console program
  - 對一般人而言最陌生且生硬，卻是古老且基礎的程式寫法，在 linux 作業系統上一定會用到
- 物件導向程式 (OOP)
  - 程式細節被包在物件內，主程式可以寫得很簡潔，也就是將程式當成物件來操作
- 圖形化使用者界面 (GUI)
  - 一般人操作起來容易，對程式設計者卻是額外的負擔 (相對於 console application)
- 網頁應用 (Web Application)
  - 透過瀏覽器操作，不需要安裝或執行額外的程式，然而我個人覺得網頁應用程式最難掌控
- Docker 容器
  - 近幾年最火紅的軟體包裝趨勢，只需要在電腦上安裝好 docker，便可以透過容器執行各式各樣的程式

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo$ tree
├── ximalaya_console
│   ├── config
│   │   └── ximalaya_demo.cfg
│   └── ximalaya_demo.py
├── ximalaya_container
│   ├── Dockerfile
│   └── ximalaya_docker.py
├── ximalaya_gui
│   ├── ximalaya_gui_main.py
│   └── ximalaya_object.py
├── ximalaya_object
│   ├── ximalaya_main.py
│   └── ximalaya_object.py
├── ximalaya_web
│   ├── static
│   │   └── favicon.ico
│   └── templates
│       ├── 404.html
│       ├── 500.html
│       ├── base.html
│       └── index_41.html
└── ximalaya_object.py
ximalaya_web.py

8 directories, 15 files
pydoc@pydoc:~/python3_scripts/coding_styles_demo$
```

# Console Program

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_console$ tree
```

```
├── config
│   └── ximalaya_demo.cfg
└── ximalaya_demo.py
```

```
1 directory, 2 files
```

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_console$ python3 ximalaya_demo.py
```

```
Got 01_救命饮食引言.mp3
```

```
Got 02_序言1他能让我们活得更好.mp3
```

```
Got 03_序言2疑问一箩筐？真正的营养？惊人的发现！.mp3
```

程式通常藉由指定繁雜的參數來呈現不同的應用。這例子單純將喜馬拉亞網站內的專輯代號另成一 cfg 檔並讀取它

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_console$ cat config/ximalaya_demo.cfg
```

```
9104501
```

← 專輯代號

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_console$ ls -l output/
```

```
總用量 8512
```

```
-rw-rw-r-- 1 pydoc pydoc 1474709 3月 29 14:06 01_救命饮食引言.mp3
```

```
-rw-rw-r-- 1 pydoc pydoc 1389550 3月 29 14:06 02_序言1他能让我们活得更好.mp3
```

```
-rw-rw-r-- 1 pydoc pydoc 2688463 3月 29 14:06 03_序言2疑问一箩筐？真正的营养？惊人的发现！.mp3
```

執行後的結果 ( 下載後的檔案 )

# 物件導向程式 (OOP)

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_object$ tree
.
├── ximalaya_main.py
└── ximalaya_object.py

0 directories, 2 files
pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_object$ python3 ximalaya_main.py
Got 01_救命饮食引言.mp3
Got 02_序言1他能让我们活得更好.mp3
Got 03_序言2疑问一箩筐？真正的营养？惊人的发现！.mp3
```

```
ximalaya_main.py - /home/pydoc/python3
File Edit Format Run Options Window
from ximalaya_object import Ximalaya
my_object = Ximalaya('9104501')
my_object.download() ↑ 專輯代號
```

物件導向主程式只有三行，很簡潔，因為程式細節寫在另一個程式檔案，並打包成物件的形式被呼叫

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_object$ ls -l output/
總用量 7696
-rw-rw-r-- 1 pydoc pydoc 1474709 3月 29 15:05 01_救命饮食引言.mp3
-rw-rw-r-- 1 pydoc pydoc 1389550 3月 29 15:05 02_序言1他能让我们活得更好.mp3
-rw-rw-r-- 1 pydoc pydoc 2688463 3月 29 15:05 03_序言2疑问一箩筐？真正的营养？惊人的发现！.mp3
```

# 圖形化使用者界面 (GUI)

```
ximalaya_gui_main.py - /home/pydoc/python3_scripts/coding_styles_demo/x
```

```
File Edit Format Run Options Window Help
```

```
import PySimpleGUI as sg
from ximalaya_object import Ximalaya
```

```
sg.theme('BluePurple')
```

```
layout = [[sg.Output(size=(80,10))],
          [sg.Button('ALBUM'), sg.Input(size=(70,1), key='-IN-'),
           [sg.Button('Run'), sg.Button('Exit')]]]
```

```
window = sg.Window('喜瑪拉雅音頻', layout)
```

```
while True: # Event Loop
```

```
    event, values = window.read()
    if event == sg.WIN_CLOSED or event == 'Exit':
        #if event in (None, 'Exit'): # web
            break
```

```
    if event == 'ALBUM':
        album_id = values['-IN-']
```

```
    if event == 'Run':
        try:
            my_album = Ximalaya(album_id).download()
        except:
            my_album = Ximalaya().download()
```

```
window.close()
```

圖形化版本的主程式，但是  
前面 console 版本只要 3 行

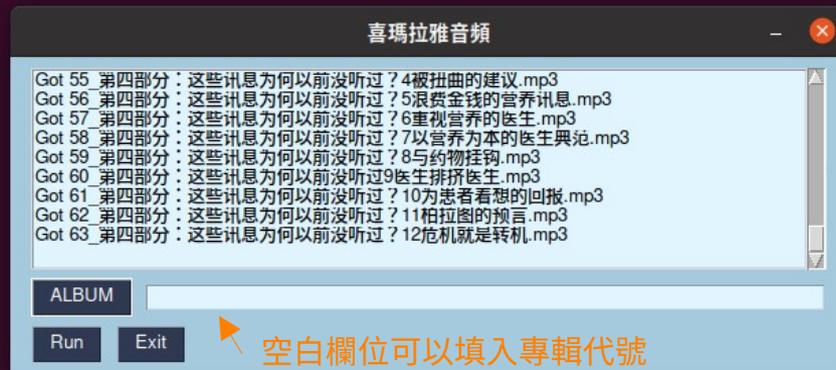
```
(0_venv_gui) pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_gui$ tree
```

```
├── ximalaya_gui_main.py
└── ximalaya_object.py
```

```
0 directories, 2 files
```

```
(0_venv_gui) pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_gui$ python3 ximalaya_gui_main.py
```

方便一般人操作



```
(0_venv_gui) pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_gui$ ls -l output
```

```
總用量 148868
```

```
-rw-rw-r-- 1 pydoc pydoc 1474709 3月 29 16:27 01_救命饮食引言.mp3
-rw-rw-r-- 1 pydoc pydoc 1389550 3月 29 16:27 02_序言1他让我们活得更好.mp3
-rw-rw-r-- 1 pydoc pydoc 2688463 3月 29 16:27 03_序言2疑问一箩筐?真正的营养?惊人的发现!.mp3
```

# 網頁應用 (Web Application)

```
(venv_flask) pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_web$ tree
```

```
.
├── static
│   └── favicon.ico
├── templates
│   ├── 404.html
│   ├── 500.html
│   ├── base.html
│   └── index_41.html
├── ximalaya_object.py
└── ximalaya_web.py
```

網頁應用程式不管編寫或佈署都很繁瑣，  
操作卻是對一般使用者最方便且友善的

```
2 directories, 7 files
```

```
(venv_flask) pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_web$ export FLASK_APP=ximalaya_web.py
```

```
(venv_flask) pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_web$ flask run
```

```
* Serving Flask app "ximalaya_web.py"
```

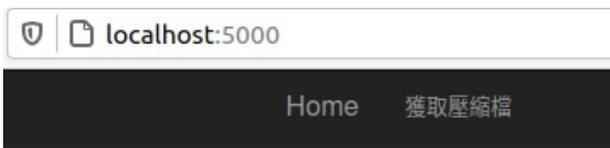
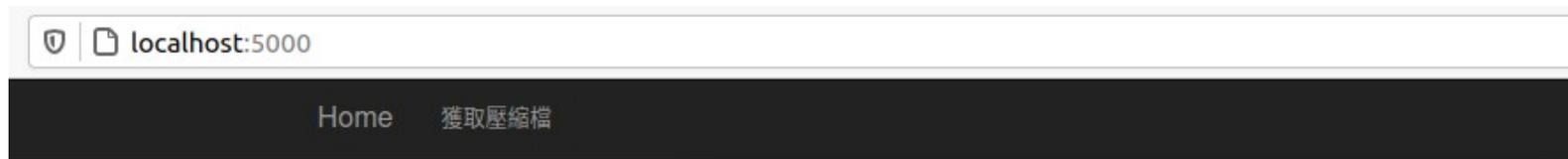
```
* Environment: production
```

```
WARNING: This is a development server. Do not use it in a production deployment.  
Use a production WSGI server instead.
```

```
* Debug mode: off
```

```
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

# 網頁應用 (續)



ximalaya

Please input ximalaya album id

10023321

開始處理

專輯代號

下載完成

Please input ximalaya album id

開始處理



# Docker 容器

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_container$ tree
```

```
├── Dockerfile
└── ximalaya_docker.py
```

docker 容器一旦佈署完成，對往後接手的人會很方便

```
0 directories, 2 files
```

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_container$ docker image build -t ximalaya_dl:1.1 .
```

```
Sending build context to Docker daemon 4.608kB
```

```
Step 1/4 : FROM python:alpine3.13
```

```
----> 36a177f36775
```

```
Step 2/4 : WORKDIR /run
```

```
----> Using cache
```

```
----> 442060e48e2e
```

```
Step 3/4 : ENTRYPOINT ["/ximalaya_docker.py"]
```

```
----> Running in 655a43452ac1
```

```
Removing intermediate container 655a43452ac1
```

```
----> c533e2e1046d
```

```
Step 4/4 : COPY . /run/
```

```
----> 54ce4b78018d
```

```
Successfully built 54ce4b78018d
```

```
Successfully tagged ximalaya_dl:1.1
```

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_container$ mkdir ../downloaded
```

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_container$ cd ../downloaded
```

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/downloaded$ docker run --rm -it -v $(pwd):/run/output ximalaya_dl:1.1 9104501
```

```
Got 01_救命饮食引言.mp3
```

```
Got 02_序言1他能让我们活得更好.mp3
```

```
Got 03_序言2疑问一箩筐？真正的营养？惊人的发现！.mp3
```

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/ximalaya_container$ cat Dockerfile
FROM python:alpine3.13
```

```
WORKDIR /run
```

```
#CMD ["python3","ximalaya_docker"]
```

```
ENTRYPOINT ["/ximalaya_docker.py"]
```

```
COPY . /run/
```

Dockerfile 指導生成映像檔

專輯代號

```
pydoc@pydoc:~/python3_scripts/coding_styles_demo/downloaded$ ls -l
```

```
總用量 14568
```

```
-rw-r--r-- 1 root root 1474709 3月 30 15:46 01_救命饮食引言.mp3
```

```
-rw-r--r-- 1 root root 1389550 3月 30 15:46 02_序言1他能让我们活得更好.mp3
```

```
-rw-r--r-- 1 root root 2688463 3月 30 15:46 03_序言2疑问一箩筐？真正的营养？惊人的发现！.mp3
```