

本教學就不教怎麼架 Django 了  
而是教怎麼去改 Django，以及更新到 Heroku 上

## 1. 兩個工具先安裝並註冊

1.github 2.heroku

這個怎麼安裝網路上都有教學，這邊就不提及  
(要確認是否需要 git 帳號才能 push)

2. 請將 django\_web\_file 這個壓縮檔解壓縮，他就是 django 的檔案，包含虛擬環境，安裝包都在裡面

## 3. 打開 cmd

cd 到 django\_web\_file

並進入虛擬環境 djangogirls\_venv

->djangogirls\_venv\Scripts\activate

```
C:\Users\life>cd D:\桌面用\蘭陽博物館app\django_web_file
C:\Users\life>D:
D:\桌面用\蘭陽博物館app\django_web_file>djangogirls_venv\Scripts\activate
```

```
(djangogirls_venv) D:\桌面用\蘭陽博物館app\django_web_file>
```

## 4. cmd 登入 heroku

->Heroku login

會連結到網頁進行登入，登入好就可以關掉網頁了

## 5. 建立一個 heroku web(或叫 app)

->heroku create yanyangmumtest

名字限定為小寫、數字、"-" (不能有"\_" )

```
(djangogirls_venv) D:\桌面用\蘭陽博物館app\django_web_file>heroku create yanyangmumtest
Creating • yanyangmumtest... done
https://yanyangmumtest.herokuapp.com/ | https://git.heroku.com/yanyangmumtest.git
```

成功後會產生兩樣東西

一樣是你的專屬網址 -> <https://yanyangmumtest.herokuapp.com>

一樣是 git 的連結 -> <https://git.heroku.com/yanyangmumtest.git>

->heroku apps 可以用這個檢視是否成功產生

```
(djangogirls_venv) D:\桌面用\蘭陽博物館app\django_web_file>heroku git:remote -a yanyangmumtest
set git remote heroku to https://git.heroku.com/yanyangmumtest.git
```

在輸入-> heroku git:remote -a **yanyangmumtest**

表示我設定 heroku 連結到 yanyangmumtest 這個 web/app

其中是利用 git remote 到 <https://git.heroku.com/yanyangmumtest.git>

檢查 git 已存在 remote

->git remote -v

```
(djangogirls_venv) D:\桌面用\蘭陽博物館app\django_web_file>git remote -v
heroku https://git.heroku.com/yanyangmumtest.git (fetch)
heroku https://git.heroku.com/yanyangmumtest.git (push)
```

**必做**，確保未來在 Heroku 執行任何指令時，都是使用到部署專用的設定檔

->heroku config:set DJANGO\_SETTINGS\_MODULE=Yilanmumsite.production\_settings

## 6.上傳至 Heroku 伺服器

->git push heroku master

注意:這裡 push 的動作是將 django 文件上傳至 heroku，若檔案有什麼問題或是設定沒設定好，常會在這裡 push failed

在第一次上傳專案後，我們必須稍微做一點設定，讓 Heroku 有正確的初始環境。首先用下面的指令，告訴 Heroku 我們需要一台機器（Heroku 叫 dyno）來執行網站

## 7.啟動/關閉 伺服器

此刻要在 heroku 上啟動 webserver

->heroku ps:scale web=1

若之後要關閉則是如下指令

->heroku ps:scale web=0

同時亦可透過->heroku ps 來觀看目前 webserver 狀態

## 8. 初始化資料庫

heroku run python Yilanmumsite/manage.py migrate

```
(djangoenv) D:\桌面用\蘭陽博物館app\django_web_file>heroku run python Yilanmumsite/manage.py migrate
Running python Yilanmumsite/manage.py migrate on • yanyangmumtest... up, run.1727 (Free)
Operations to perform:
  Synchronize unmigrated apps: staticfiles, messages
  Apply all migrations: trips, contenttypes, auth, sessions, admin
Synchronizing apps without migrations:
  Creating tables...
  Running deferred SQL...
  Installing custom SQL...
Running migrations:
  Rendering model states... DONE
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying sessions.0001_initial... OK
  Applying trips.0001_initial... OK
```

9. 為後台建立一個使用者

heroku run python Yilanmumsite/manage.py createsuperuser

10.最後就可以用

->heroku open 打開你的網址囉

之後有任何在 django 上的變動

都可透過 git 指令上傳版本更新

->git status

可看哪些檔案更新了(可能是新增刪除修改等等)

->git add .

->git commit -m "add"

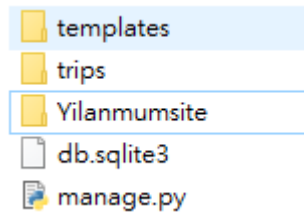
->git push heroku master

即可更新完成

如有發生錯誤，請善用這個指令來看 logs，觀看網站執行的紀錄

->heroku logs --tail

上傳好後，再來講 Django 網站架構  
首先點開 Yilanmumsite 檔案目錄



再點進 Yilanmumsite，有 urls.py

```
urlpatterns = [  
    url(r'^admin/', include(admin.site.urls)),  
    url(r'^to131/$', to131),  
    url(r'^back131/$', back131),  
    url(r'^to1766/$', to1766),  
    url(r'^back1766/$', back1766),  
    url(r'^tor1/$', tor1),  
    url(r'^backr1/$', backr1),  
    url(r'^to1877/$', to1877),  
    url(r'^back1877/$', back1877),  
    url(r'^$', home),  
]
```

其中這段就是在設定 url  
簡單來講就是 例如

```
url(r'^back1877/$', back1877),
```

就是說在 <https://yanyangmumtest.herokuapp.com/>後面再加上 back1877  
變成 <https://yanyangmumtest.herokuapp.com/back1877/>

就會執行 back1877 這個函式

這個函式在 trips 檔案目錄裡的 view.py

→def back1877 (request)

接著看到裡面有爬蟲程式

以及很重要的

```
return render(request, 'to131.html', {  
    'bus_step1':msg[0], 'bus_step2':msg[1], 'bus_step3':msg[2], 'bus_step4':msg[3],  
    'bus_step5':msg[4], 'bus_step6':msg[5], 'bus_step7':msg[6], 'bus_step8':msg[7],  
    'bus_step9':msg[8], 'bus_step10':msg[9], 'bus_step11':msg[10], 'bus_step12':msg[11],  
    'bus_step13':msg[12], 'bus_step14':msg[13], 'bus_step15':msg[14], 'bus_step16':msg[15],  
    'bus_step17':msg[16], 'bus_step18':msg[17], 'bus_step19':msg[18],  
    'current_time': str(datetime.now()),  
})
```

剛剛執行了 `url` 這個函式，並且傳送 `back1877` 這個 `html request`  
接著要用 `render` 把 `request`、`html`、各種變數，`response` 回去  
其中 `to131.html` 在 `templates` 檔案目錄裡  
其他 `python` 檔都沒用到