**Deployment(c20)**

<https://ithelp.ithome.com.tw/articles/10212659>

Django 的設定：

1. secret\_key
2. DEBUG
3. ALLOWED\_HOSTS
4. DATABASES
5. CACHES
6. STATIC\_ROOT,STATIC\_URL,STATIC\_DIRS
7. MEDIA\_ROOT,MEDIA\_URL
8. EMAIL\_BACKEND
9. LOGGING
10. ADMINS,MANAGERS
11. CSRF\_COOKIE\_SECURE
12. SESSION\_COOKIE\_SECURE

**Use Apache to deployment**

1. 安裝apache
2. 安裝Apache的WSGI模組
3. 設定Apache與WSGI設定檔
4. 設定更安全的站點，使用TLS/SSL
5. 設定SSL模組

**Use NginX to deployment**

1. 安裝Uwsgi
2. 設定Nginx與Uwsgi
3. 設定uWSGI的設定檔
4. 註冊Uwsgi service
5. 設定Nginx
6. 設定更安全的站點，使用TLS/SSL

(取得憑證檔案或自行產生的方法跟Apache依樣)

1. 設定SSL模組

**Heroku**

1. 使用前需要準備：heorku帳號、安裝heorku工具包、下載git
2. git init “專案目錄”
3. 修改設定檔來配合Heroku的架構：
4. 修改settings.py：讓我們的專案可以使用Heroku上的資料庫，接收heroku的發送訪問
5. 設定wsgi.py在DEBUG關閉的情況下仍會處理靜態檔

（安裝heroku用的python 套件dj\_database\_url 和dj\_static）

**pip install dj\_database\_url**

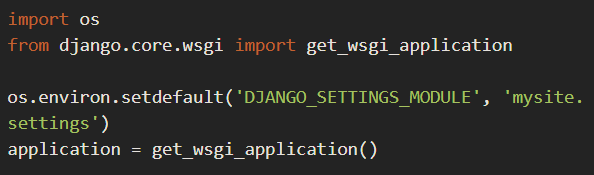
dj\_database\_:處理資料庫的套件

gunicorn: heroku伺服器的處理套件

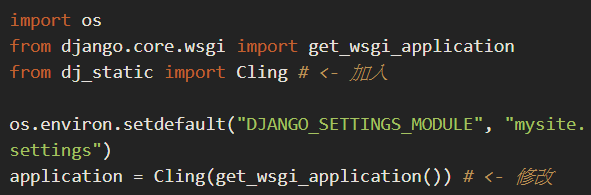
dj\_static : heroku 處理靜態檔案的套件

修改WSGI :

**原本**



**修改成**

****

1. 建立requirements.txt : pip freeze ->requirements.txt
2. 建立runtime.txt : python –version3.8.0
3. 建立 Profile : web: gunicorn –pythonpath myrest myrest.wsgi –log-file

這個指令告訴heroku用gunicorn執行myrest.wsgi

同時也是heroku執行app的進入點

Gunicorn是一個用python開發的WSGI工具，可以用來執行Django的網站

7.建立 .gitignore