

```

1 Lab. uWSGI 서버 연결하기
2 ※ 이번 Lab은 Windows에서는 실행이 되지 않는다.
3 ※ 그래서 macOS 또는 EC2 인스턴스의 OS가 Ubuntu와 같은 Linux에서 실행한다.
4
5 1. Django는 웹서버와 직접 통신할 수 없어서 둘 사이에 WSGI를 놓아야 한다.
6 2. 매번 runserver를 실행하지 않아도 Django 프로젝트를 실행할 수 있다.
7
8 3. 이를 위해서 별도의 가상환경을 생성하고, uWSGI Python 패키지를 설치한다.
9 1)가상환경 생성
10 $ virtualenv myvenv
11 created virtual environment CPython3.11.2.final.0-64 in 1176ms
12 creator CPython3Posix(dest=/Users/henry/myvenv, clear=False, no_vcs_ignore=False, global=False)
13 seeder FromAppData(download=False, pip=bundle, setuptools=bundle, wheel=bundle, via=copy,
14 app_data_dir=/Users/henry/Library/Application Support/virtualenv)
15 added seed packages: pip==23.1.2, setuptools==67.6.1, wheel==0.40.0
16 activators BashActivator,CShellActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator
17
18 2)가상환경 활성화
19 $ source ./myvenv/bin/activate
20 (myvenv) ... $
21
22 3)가상환경으로 이동
23 $ cd myvenv
24
25 4)uwsgi package 설치
26 $ pip install uwsgi
27 Collecting uwsgi
28   Downloading uwsgi-2.0.21.tar.gz (808 kB)
29     _____ 808.5/808.5 kB 13.1 MB/s eta 0:00:00
30   Preparing metadata (setup.py) ... done
31 Building wheels for collected packages: uwsgi
32   Building wheel for uwsgi (setup.py) ... done
33   Created wheel for uwsgi: filename=uWSGI-2.0.21-cp311-cp311-macosx_10_9_universal2.whl size=428740
34   sha256=b9fb45b8d1980ba3aab46a49417455f5e09d41d996b7b38b2c60d4018ac901c0
35   Stored in directory:
36   /Users/henry/Library/Caches/pip/wheels/04/19/3a/479f33cd393fe0b6f2873aea040fdea9262f705e752cbc5a5a
37   Successfully built uwsgi
38   Installing collected packages: uwsgi
39   Successfully installed uwsgi-2.0.21
40
41 4. uwsgi 서버를 이용해서 Django Project와 연결해 보자.
42 1)형식은 다음과 같다.
43 uwsgi --http :[포트번호] --home [가상환경 경로] --chdir [장고프로젝트폴더 경로] -w [wsgi 모듈이 있는 폴더].wsgi
44
45 2)포트번호
46 -Django Port Number
47 -8000
48
49 3)가상환경 경로
50 -uwsgi를 설치한 myvenv의 경로.
51 -~/myvenv/ 이지만 홈 폴더를 절대 경로로 지정해 주는 것이 좋다.
52 -~는 /Users/{계정}/이니까 /Users/{계정}/myvenv/가 가상환경 경로이다.
53 -macOS
54   ex)/Users/henry/myvenv/
55 -Ubuntu
56   ex)/home/ubuntu/myvenv/
57
58 4)Django Project 디렉토리 경로
59 -macOS
60   ex)/Users/henry/DjangoHome/myproject/
61 -Ubuntu
62   ex)/home/ubuntu/DjangoHome/myproject/
63
64 5)wsgi 모듈이 있는 폴더
65 -프로젝트 밑에 myproject 폴더 밑에 wsgi.py가 있을 것이다.
66 -그 경로에 myproject를 입력한다.
67
68 6)다음과 같이 실행한다.
69 -종료할 때는 Ctrl + C를 누르면 된다.
70
71 -macOS
72 $ uwsgi --http :8000 --home /Users/henry/myvenv/ --chdir /Users/henry/DjangoHome/myproject/ -w myproject.wsgi
73
74 -Ubuntu
75 $ uwsgi --http :8000 --home /home/ubuntu/myvenv/ --chdir /home/ubuntu/DjangoHome/myproject/ -w myproject.wsgi
76 *** Starting uWSGI 2.0.21 (64bit) on [Thu May 18 11:52:14 2023] ***
77 compiled with version: Apple LLVM 14.0.3 (clang-1403.0.22.14.1) on 18 May 2023 02:03:18
78 os: Darwin-22.4.0 Darwin Kernel Version 22.4.0: Mon Mar 6 21:00:17 PST 2023;
79 root:xnu-8796.101.5~3/RELEASE_X86_64
80 nodename: henry-MacBook-Pro.local
81 machine: x86_64
82 clock source: unix
83 detected number of CPU cores: 4

```

```

81 current working directory: /Users/henry
82 detected binary path: /Users/henry/myvenv/bin/uwsgi
83 !!! no internal routing support, rebuild with pcre support !!!
84 chdir() to /Users/henry/DjangoHome/myproject/
85 *** WARNING: you are running uWSGI without its master process manager ***
86 your processes number limit is 1392
87 your memory page size is 4096 bytes
88 detected max file descriptor number: 256
89 lock engine: OSX spinlocks
90 thunder lock: disabled (you can enable it with --thunder-lock)
91 uWSGI http bound on :8000 fd 4
92 spawned uWSGI http 1 (pid: 85324)
93 uwsgi socket 0 bound to TCP address 127.0.0.1:53948 (port auto-assigned) fd 3
94 Python version: 3.9.16 (main, Dec 7 2022, 10:16:11) [Clang 14.0.0 (clang-1400.0.29.202)]
95 PEP 405 virtualenv detected: /Users/henry/myvenv/
96 Set PythonHome to /Users/henry/myvenv/
97 *** Python threads support is disabled. You can enable it with --enable-threads ***
98 Python main interpreter initialized at 0x7fabd48051d0
99 your server socket listen backlog is limited to 100 connections
100 your mercy for graceful operations on workers is 60 seconds
101 mapped 72896 bytes (71 KB) for 1 cores
102 *** Operational MODE: single process ***
103 Traceback (most recent call last):
104   File "/Users/henry/DjangoHome/myproject/./myproject/wsgi.py", line 12, in <module>
105     from django.core.wsgi import get_wsgi_application
106 ModuleNotFoundError: No module named 'django'
107 unable to load app 0 (mountpoint='') (callable not found or import error)
108 *** no app loaded. going in full dynamic mode ***
109 *** uWSGI is running in multiple interpreter mode ***
110 spawned uWSGI worker 1 (and the only) (pid: 85323, cores: 1)
111
112

```

113 5. runserver를 사용하지 않고 Web Browser를 통해 접속해본다.

114 <http://localhost:8000/>

```

115
116 ※혹시 Linux에서 다음과 같은 오류가 발생할 수 있다.
117 ...
118 *** Operational MODE: single process ***
119 Traceback (most recent call last):
120   File "/home/ubuntu/DjangoHome/myproject/./myproject/wsgi.py", line 12, in <module>
121     from django.core.wsgi import get_wsgi_application
122 ModuleNotFoundError: No module named 'django'
123 unable to load app 0 (mountpoint='') (callable not found or import error)
124 *** no app loaded. going in full dynamic mode ***
125 *** uWSGI is running in multiple interpreter mode ***
126 spawned uWSGI worker 1 (and the only) (pid: 14601, cores: 1)
127 --- no python application found, check your startup logs for errors ---
128 ...
129

```

130 ※이런 오류가 발생하면 'pip install Django'를 통해 Django를 설치하면 된다.

131 ※다시 uwsgi 서버를 실행한다.

132

133

134 6. 하지만, 위의 방법은 실행할 때마다 복잡한 명령어를 일일이 입력해야 한다는 부담이 있다.

135

136 7. 따라서 파일로 옵션들을 정해서 실행할 수 있다.

137 1)manage.py가 있는 디렉토리에서 .config 디렉토리를 생성하고, 그 밑에 uwsgi 디렉토리를 생성한다.

```

138 $ ls
139 db.sqlite3      manage.py      myproject
140
141 $ mkdir .config
142 $ cd .config
143 $ mkdir uwsgi
144

```

145 2)uwsgi 디렉토리 안에 myproject.ini 파일을 생성한다.

```

146 $ cd uwsgi
147 $ nano myproject.ini
148 [uwsgi]
149 chdir = /Users/henry/DjangoHome/myproject/
150 #Ubuntu /home/ubuntu/DjangoHome/myproject/
151
152 module = myproject.wsgi:application
153 home = /Users/henry/myvenv/
154 #Ubuntu /home/ubuntu/myvenv/
155
156 uid = henry #ubuntu
157 gid = staff #ubuntu
158
159 http = :8000
160
161 enable-threads = true
162 master = true
163 vacuum = true
164 pidfile = /tmp/myproject.pid

```

```
165         logto = /var/log/uwsgi/myproject/@@(exec://date +%Y-%m-%d).log
166         log-reopen = true
167
168 3) 각 항목의 설명은 다음과 같다.
169     -chdir
170         --Django Project 디렉토리의 경로.
171     -module
172         --위에서 실행할 때의 -w 옵션 뒤의 값과 같다.
173     --그리고 뒤에 :application을 붙인다.
174     -home
175         --가상환경의 경로와 동일
176     -uid
177         --uwsgi를 사용할 계정.
178     -gid
179         --uwsgi를 사용할 그룹
180     -http
181         --사용할 포트 번호
182     -logto
183         --uwsgi 서버의 로그를 저장할 디렉토리
184
185 4) logto 경로에 있는 디렉토리를 생성해야 한다.
186     $ sudo mkdir -p /var/log/uwsgi/myproject
187
188 5) log에 기록할 수 있는 계정은 현재 계정이기 때문에 소유권을 현재 계정으로 변경한다.
189     $ sudo chown -R {계정}:{그룹} /var/log/uwsgi/myproject/
190     -macOS
191         ex) sudo chown -R henry:staff /var/log/uwsgi/myproject/
192     -Ubuntu
193         ex) sudo chown -R ubuntu:ubuntu /var/log/uwsgi/myproject/
194
195 8. myproject.ini 파일을 이용해서 uwsgi 서버를 가동한다.
196     -macOS
197         $ sudo /Users/henry/myenv/bin/uwsgi -i /Users/henry/DjangoHome/myproject/.config/uwsgi/myproject.ini
198         Password:
199
200     -Ubuntu
201         $ sudo /home/ubuntu/myenv/bin/uwsgi -i /home/ubuntu/DjangoHome/myproject/.config/uwsgi/myproject.ini
202         Password:
203
204
205 9. 성공적으로 uwsgi 서버가 가동되면 다음과 같은 출력 메시지를 확인할 수 있다.
206     -macOS
207         [uWSGI] getting INI configuration from /Users/henry/DjangoHome/myproject/.config/uwsgi/myproject.ini
208
209     -Ubuntu
210         [uWSGI] getting INI configuration from /home/ubuntu/DjangoHome/myproject/.config/uwsgi/myproject.ini
211
212
213
214 10. Web Browser에서 확인할 수 있다.
```