

## <서버·개발 환경을 위한 settings 분리하기>

- 고정 아이피를 추가하여 서버를 구동해야 해서 ALLOWED\_HOSTS를 수정했더니, 로컬 개발 서버에서 접속한다면, 오류 화면이 출력된다.

이 문제를 해결하기 위해서 서버 환경의 ALLOWED\_HOSTS와 개발 환경의 ALLOWED\_HOSTS를 다르게 설정해야 한다. 그리고 DEBUG, DATABASE 등의 항목도 다르게 설정해야 한다.

### [실습] 환경 파일 만들기.

1) settings 디렉터리 생성하기.

2) base.py 파일 생성하기.

↳ settings.py 파일을 settings 디렉터리로 base.py 라는 이름으로 이동시킨다.  
그리고 base.py 파일의 BASE\_DIR 항목을 다음과 같이 수정한다.

```
BASE_DIR = Path(__file__).resolve().parent.parent
```

↳ 기존 파일의 위치보다 더 높아졌기 때문에 parent 추가.

3) local.py 파일 생성하기. (config/settings/local.py)

↳ 개발 환경을 담당할 파일을 작성한다.

```
from .base import *
```

```
ALLOWED_HOSTS = []
```

4) Prod.py 파일 생성하기 (config/settings/Prod.py)

↳ 서버 환경을 담당할 Prod.py 파일을 다음과 같이 작성한다.

```
from .base import *
```

```
ALLOWED_HOSTS = [' 고정 IP ']
```

[실습] settings 옵션 설정하기.

- 개발 환경에서 장고 서버를 구동하면 settings.py 파일을 읽지 못해 오류가 발생한다.

1) 개발 환경으로 settings 설정하기

↳ --settings 옵션을 추가하여 장고 서버를 실행해본다.

```
python manage.py runserver --settings=config.settings.local
```

2) 서버 환경으로 settings 설정하기

1'

.prod

3) 추가로 local.py에는 DEBUG = True, prod.py에는 False 설정.

[실습] 서버 설정 자동화하기.

1) nano 편집기로 /venvs/mysite.sh 작성하기.

↳ 터미널에서 venus 디렉토리로 이동한 다음 nano mysite.sh 명령 수행

2) nano 편집기에서 내용 작성하고 저장

```
#!/bin/bash
```

```
cd ~/projects/mysite
```

```
export DJANGO_SETTINGS_MODULE=config.settings.prod
```

```
. ~/venvs/mysite/bin/activate
```

↳ 터미널에서 mysite.sh 파일 실행

```
VENUS 디렉토리에서 . mysite.sh
```

## < MobaXterm 으로 서버에 접속하기 >

- AWS 라이드 세일 터미널 대신 접속도 좋지 않고 여러 편의 기능을 제공하는 MobaXterm 라는 프로그램으로 AWS 라이드 세일에 접속하는 방법을 알아보자.

[실습] MobaXterm 에 비밀키 연결하고 서버에 접속하기.

### 1) 비밀키 내려받기

↳ AWS 라이드 세일의 메인화면에서 계정으로 이동하여 SSH 키를 다운로드받는다.

그리고 비밀키를 C:/venvs 디렉토리에 넣고 mysite.pem 으로 이름을 변경한다.

### 2) MobaXterm 설치하고 접속하기.

### 3) MobaXterm 으로 서버에 접속하기.

① Session → ② SSH → ③ Remote host에 고정 IP 주소 입력 → ④ Specify username  
제일 첫 'ubuntu' 입력 → ⑤ Use Private key 제일 mysite.pem 입력 → ⑥ Ok