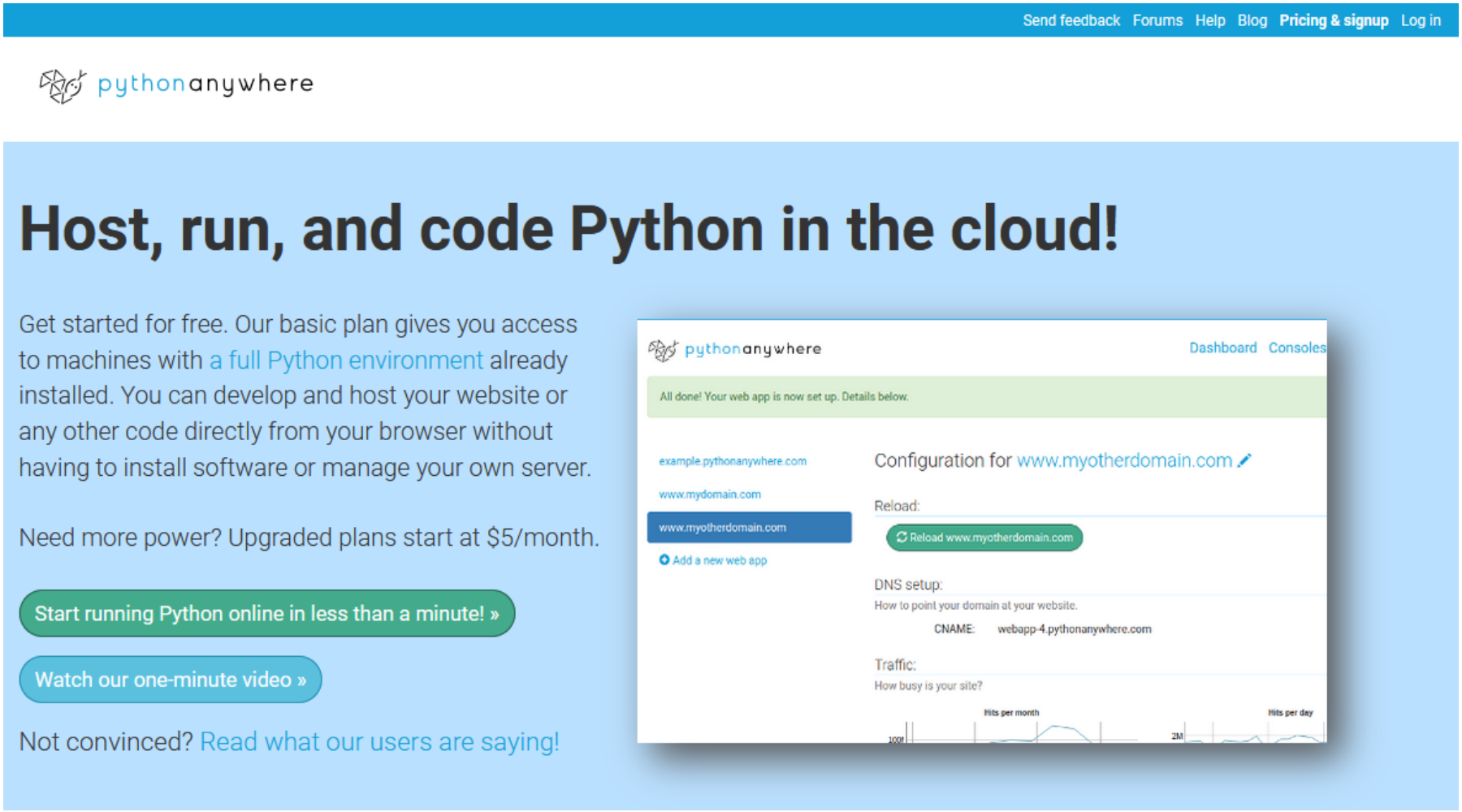
회원가입 및 로그인

- 주소 → https://www.pythonanywhere.com/



설정 파일 수정

- config/settings.py

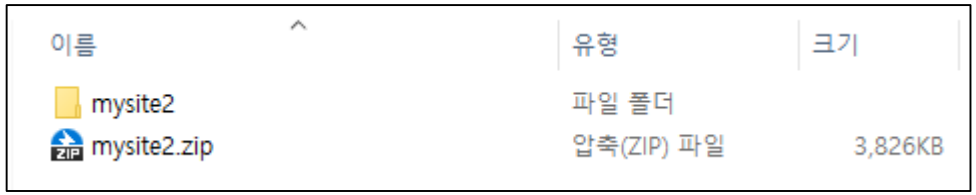
```
ALLOWED_HOSTS = ["account-name.pythonanywhere.com", "localhost", "127.0.0.1"]
```

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'account-name$pybo',
        'USER': 'account-name',
        'PASSWORD': 'your-password',
        'HOST': 'account-name.mysql.pythonanywhere-services.com',
        'PORT': '3306'
    }
}
```

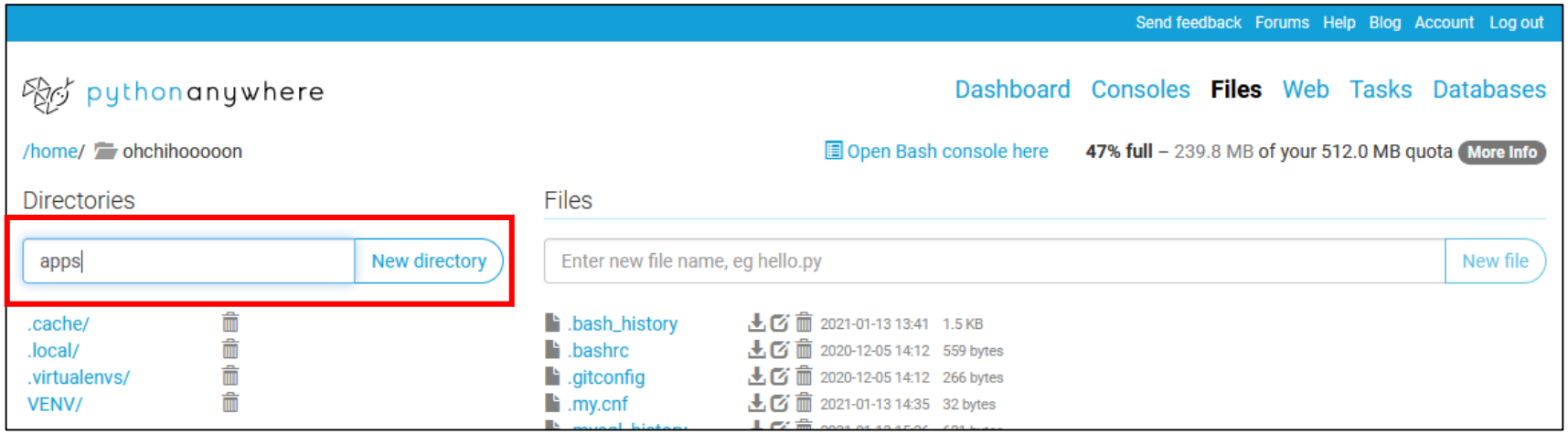
```
STATIC_URL = 'static/'
STATIC_ROOT = os.path.join(BASE_DIR, 'www-dir', 'static')
# STATICFILES_DIRS = [BASE_DIR / 'static']
# STATICFILES_DIRS = [os.path.join(BASE_DIR, 'static')]
```

프로젝트 소스 파일 업로드

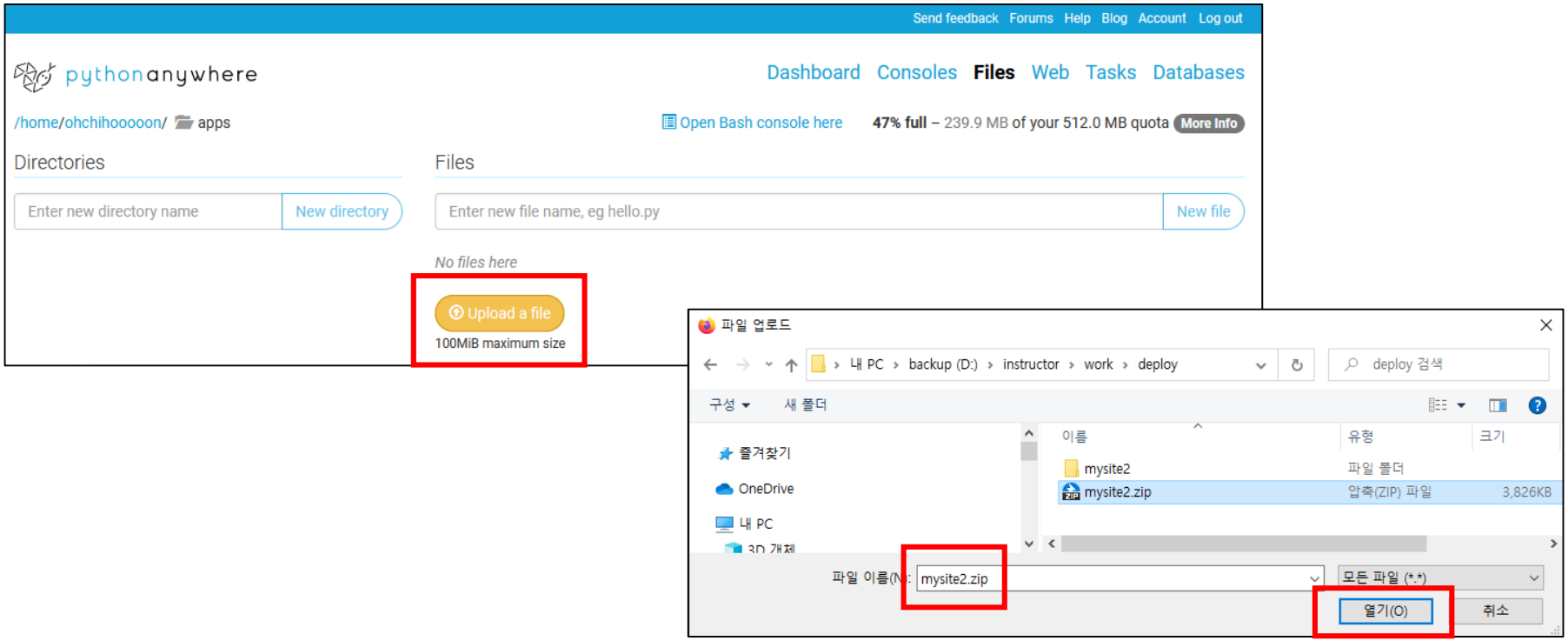
- 프로젝트 폴더 압축

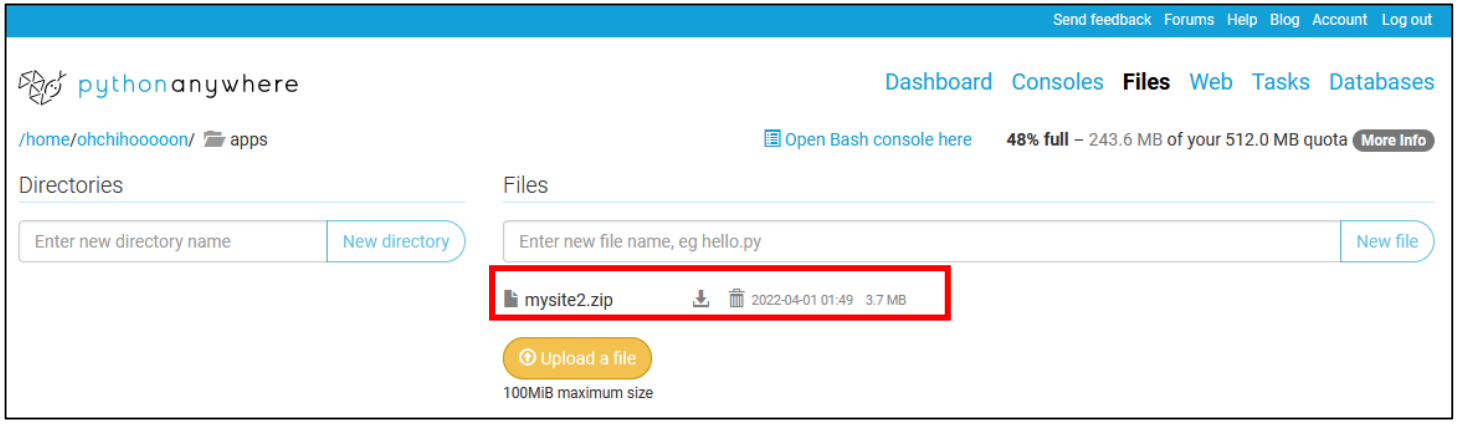


- pythonanywhere에 폴더 만들기



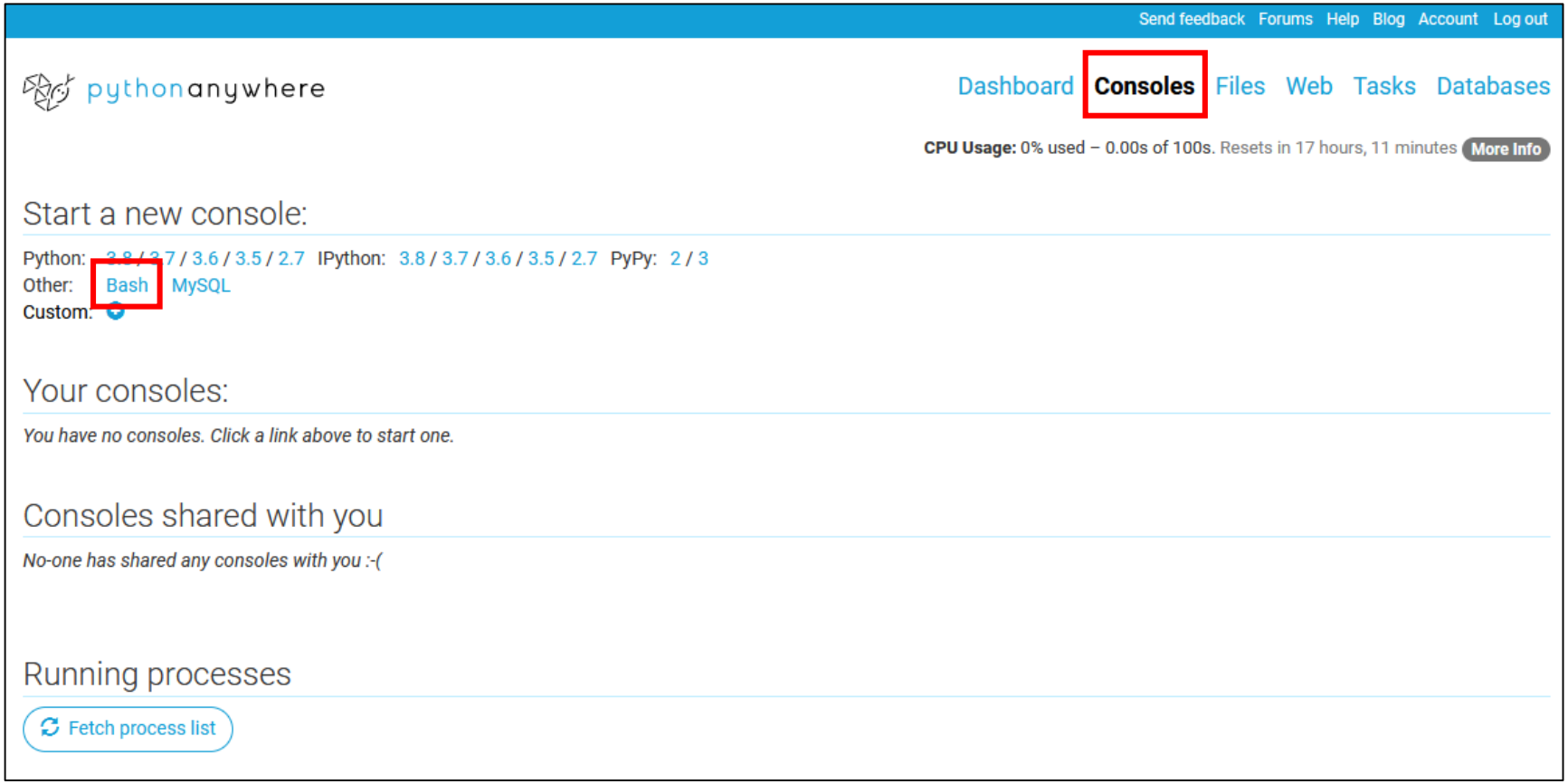
프로젝트 소스 파일 업로드





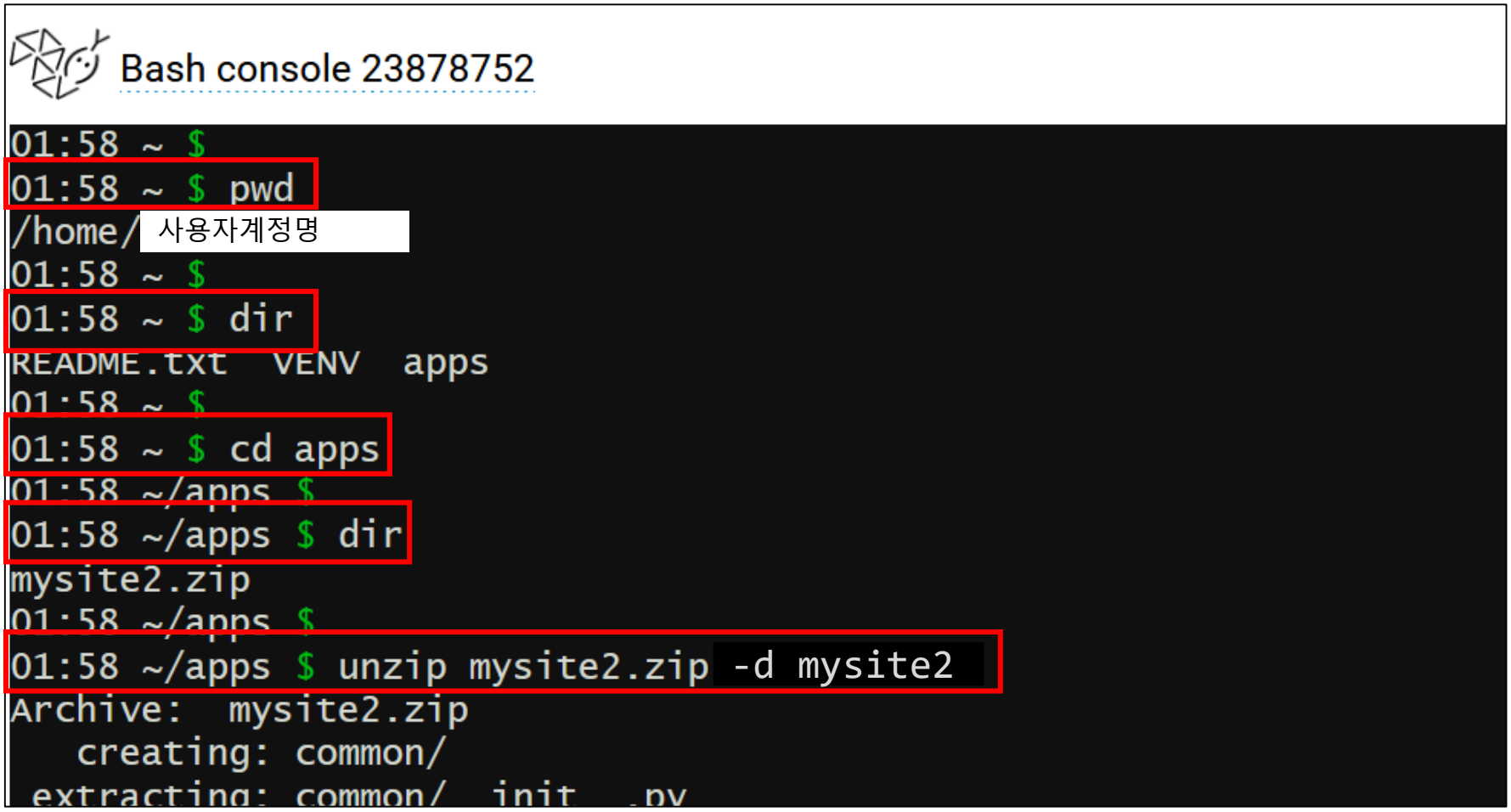
터미널에서 압축 풀기

- 터미널 실행



터미널에서 압축 풀기

- 소스파일 업로드 위치로 이동 후 압축 해제



가상 환경 만들기

- 파이썬 가상 환경 만들기 및 가상 환경 활성화

```
02:07 ~ $
02:07 ~ $ cd ~
02:07 ~ $ dir
README.txt  apps
02:07 ~ $
02:07 ~ $ mkdir VENV
02:07 ~ $
02:07 ~ $ cd VENV
02:07 ~/VENV $
02:07 ~/VENV $ virtualenv --python python3.8 knitac-env
Running virtualenv with interpreter /usr/local/bin/python3.8
Already using interpreter /usr/local/bin/python3.8
Using base prefix '/usr'
New python executable in /home/사용자계정명/VENV/knitac-env/bin/python3.8
Also creating executable in /home/사용자계정명/VENV/knitac-env/bin/python
Installing setuptools, pip, wheel...
done.
02:09 ~/VENV $
02:09 ~/VENV $ source ~/VENV/knitac-env/bin/activate
(knitac-env) 02:10 ~/VENV $
(knitac-env) 02:10 ~/VENV $ python -V
Python 3.8.0
(knitac-env) 02:10 ~/VENV $
```

- 가상 환경에서 Django 설치

```
(knitac-env) 02:10 ~/VENV $
(knitac-env) 02:10 ~/VENV $ pip install Django
Looking in links: /usr/share/pip-wheels
Collecting Django
  Downloading Django-4.0.3-py3-none-any.whl (8.0 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 8.0/8.0 MB 16.6 MB/s eta 0:00:00
Collecting backports.zoneinfo
  Downloading backports.zoneinfo-0.2.1-cp38-cp38-manylinux1_x86_64.whl (74 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 74.0/74.0 KB 1.8 MB/s eta 0:00:00
Collecting asgiref<4,>=3.4.1
  Downloading asgiref-3.5.0-py3-none-any.whl (22 kB)
Collecting sqlparse>=0.2.2
  Downloading sqlparse-0.4.2-py3-none-any.whl (42 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 42.3/42.3 KB 385.3 kB/s eta 0:00:00
Installing collected packages: sqlparse, backports.zoneinfo, asgiref, Django
Successfully installed Django-4.0.3 asgiref-3.5.0 backports.zoneinfo-0.2.1 sqlparse-0.4.2
(knitac-env) 02:11 ~/VENV $
```

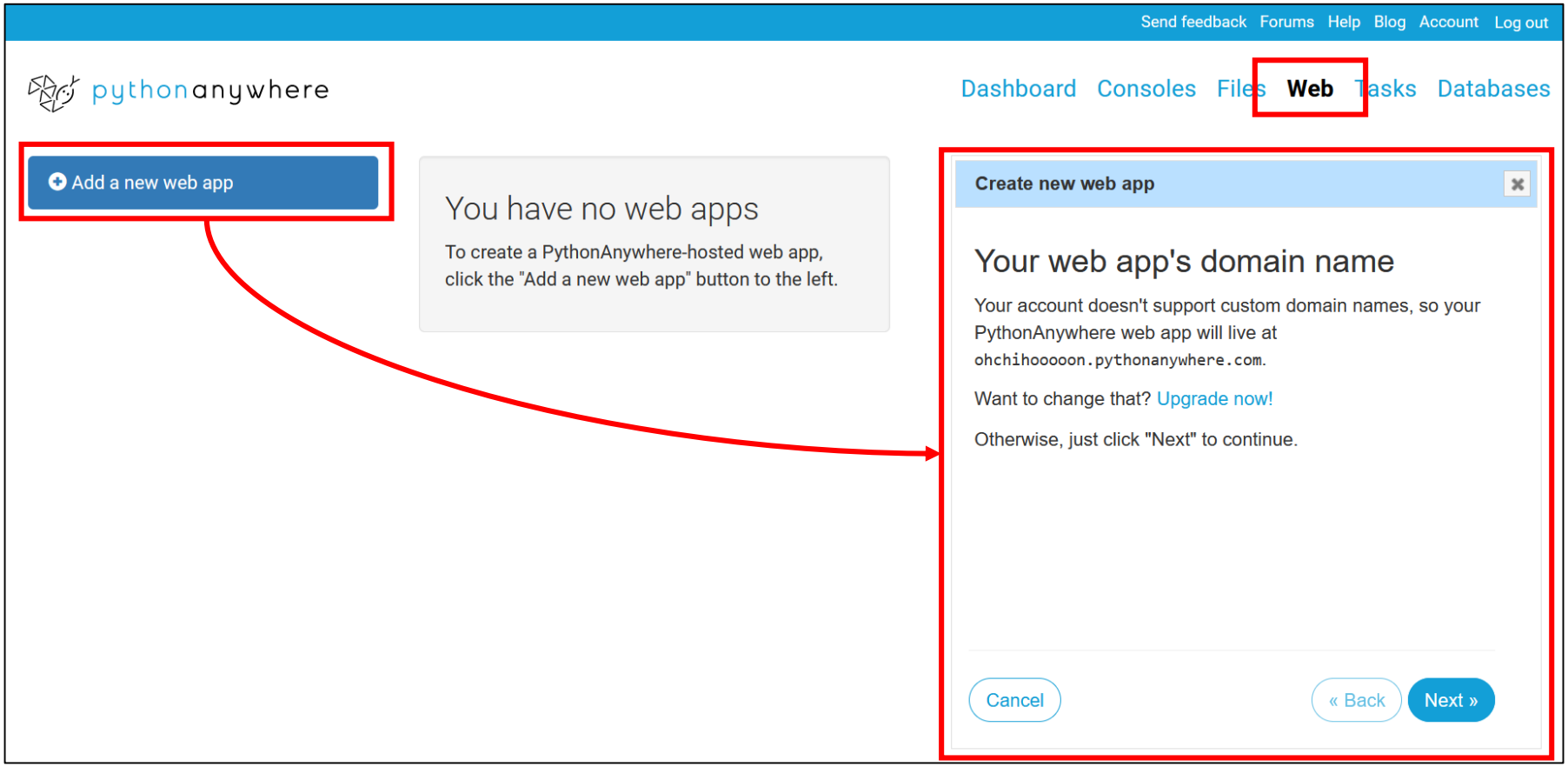
mysql 모듈 설치

- 설정 내용 생성

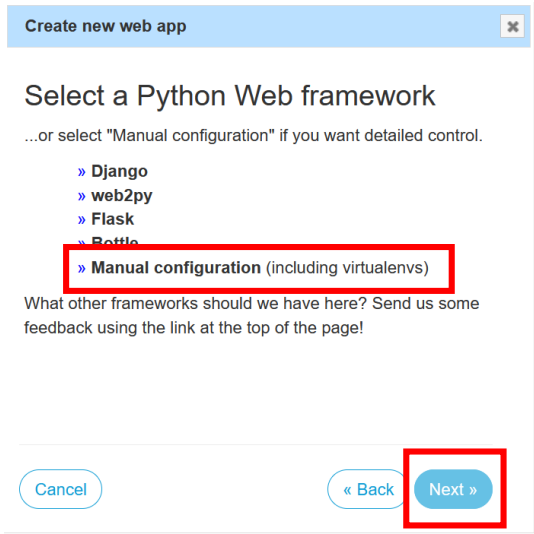
```
(knitac-env) 02:28 ~/VENV $
(knitac-env) 02:28 ~/VENV $ cd ~/apps/mysite2/
(knitac-env) 02:28 ~/apps/mysite2 $
(knitac-env) 02:28 ~/apps/mysite2 $
```

```
(knitac-env) 02:26 ~/apps/mysite2 $ pip install mysqlclient
Looking in links: /usr/share/pip-wheels
Collecting mysqlclient
  Downloading mysqlclient-2.1.0.tar.gz (87 kB)
     ---------------------------------------------------------------------- 87
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: mysqlclient
  Building wheel for mysqlclient (setup.py) ... done
  Created wheel for mysqlclient: filename=mysqlclient-2.1.0-cp38-cp38-linux
92a2001a0101267f58e64b338234a0313d746a
  Stored in directory: /home/ohchihoooooon/.cache/pip/wheels/61/e7/42/9d5634
Successfully built mysqlclient
Installing collected packages: mysqlclient
Successfully installed mysqlclient-2.1.0
(knitac-env) 02:26 ~/apps/mysite2 $ pip install pymysql
Looking in links: /usr/share/pip-wheels
Collecting pymysql
  Using cached PyMySQL-1.0.2-py3-none-any.whl (43 kB)
Installing collected packages: pymysql
Successfully installed pymysql-1.0.2
(knitac-env) 02:26 ~/apps/mysite2 $
```

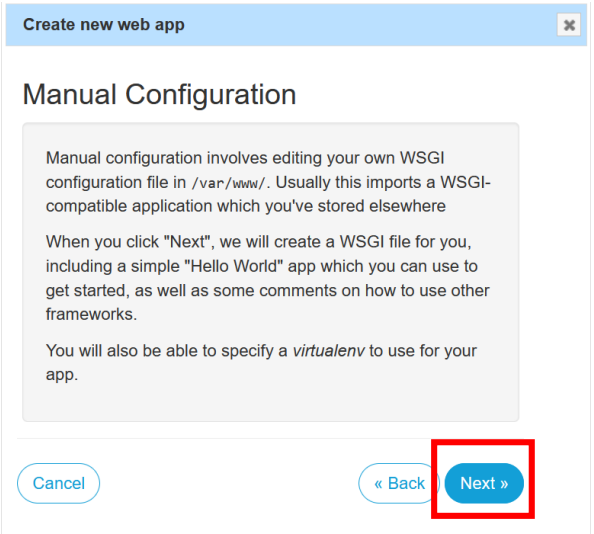
웹서버 설정


Send feedback Forums Help Blog Account Log out
pythonanywhere
Dashboard Consoles Files Web Tasks Databases
Add a new web app
You have no web apps
To create a PythonAnywhere-hosted web app, click the "Add a new web app" button to the left.
Create new web app
Your web app's domain name
Your account doesn't support custom domain names, so your PythonAnywhere web app will live at ohchihoooooon.pythonanywhere.com.
Want to change that? Upgrade now!
Otherwise, just click "Next" to continue.
Cancel
« Back
Next »

웹서버 설정


Create new web app
Select a Python Web framework
...or select "Manual configuration" if you want detailed control.
» Django
» web2py
» Flask
» Manual configuration (including virtualenvs)
What other frameworks should we have here? Send us some feedback using the link at the top of the page!
Cancel
« Back
Next »


Create new web app
Select a Python version
» Python 2.7
» Python 3.5
» Python 3.6
» Python 3.8
Cancel
« Back
Next »


Create new web app
Manual Configuration
Manual configuration involves editing your own WSGI configuration file in /var/www/. Usually this imports a WSGI-compatible application which you've stored elsewhere
When you click "Next", we will create a WSGI file for you, including a simple "Hello World" app which you can use to get started, as well as some comments on how to use other frameworks.
You will also be able to specify a virtualenv to use for your app.
Cancel
« Back
Next »

웹 서버 설정

Code:

What your site is running.

Source code: /home/ 사용자계정명 /apps/mysite2 Go to directory

Working directory: /home/ 사용자계정명 / Go to directory

WSGI configuration file: /var/www/ 사용자계정명 _pythonanywhere_com_wsgi.py

Python version: 3.8

Virtualenv:

Use a virtualenv to get different versions of flask, django etc from our default system ones. More info here. You need to **Reload your web app** to activate it; NB - will do nothing if the virtualenv does not exist.

/home/ 사용자계정명 /VENV/knitac-env

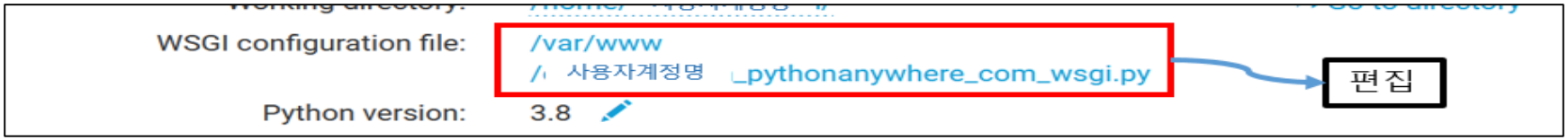
Start a console in this virtualenv

Static files:

Files that aren't dynamically generated by your code, like CSS, JavaScript or uploaded files, can be served much faster straight off the disk if you specify them here. You need to **Reload your web app** to activate any changes you make to the mappings below.

URL	Directory	Delete
/static/	/home/ 사용자계정명 /apps/mysite2/www-dir/static/	
Enter URL	*Enter path*	

웹 서버 설정




```
# def application(environ, start_response):
#     if environ.get('PATH_INFO') == '/':
#         status = '200 OK'
#         content = HELLO_WORLD
#     else:
#         status = '404 NOT FOUND'
#         content = 'Page not found.'
#     response_headers = [('Content-Type', 'text/html'), ('Content-Length', str(len(content)))]
#     start_response(status, response_headers)
#     yield content.encode('utf8')
```

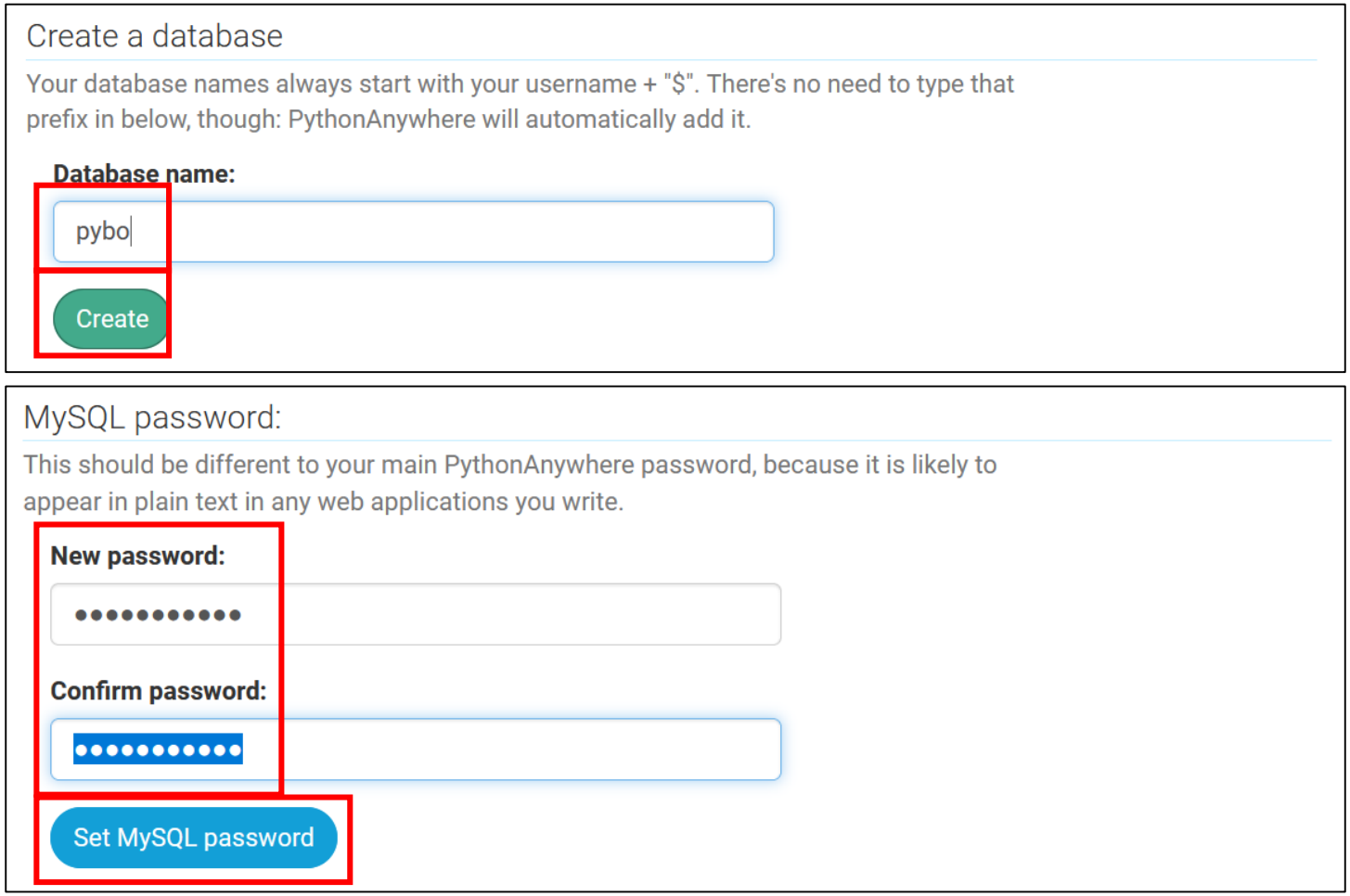
```
# ++++++++++ DJANGO ++++++++++
# To use your own django app use code like this:
import os
import sys

# assuming your django settings file is at '/home/사용자계정/mysite2/config/settings.py'
# and your manage.py is is at '/home/사용자계정/mysite2/manage.py'
path = '/home/사용자계정/apps/mysite2'
if path not in sys.path:
    sys.path.append(path)

os.environ['DJANGO_SETTINGS_MODULE'] = 'config.settings'

# then:
from django.core.wsgi import get_wsgi_application
application = get_wsgi_application()
```

데이터베이스 설정


Create a database
Your database names always start with your username + "$". There's no need to type that prefix in below, though: PythonAnywhere will automatically add it.
Database name:
pybo
Create
MySQL password:
This should be different to your main PythonAnywhere password, because it is likely to appear in plain text in any web applications you write.
New password:
Confirm password:
Set MySQL password

static 파일 추출 & 데이터베이스 migration

- migration

```
(knitac-env) 02:35 ~/apps/mysite2 $ python manage.py migrate
```

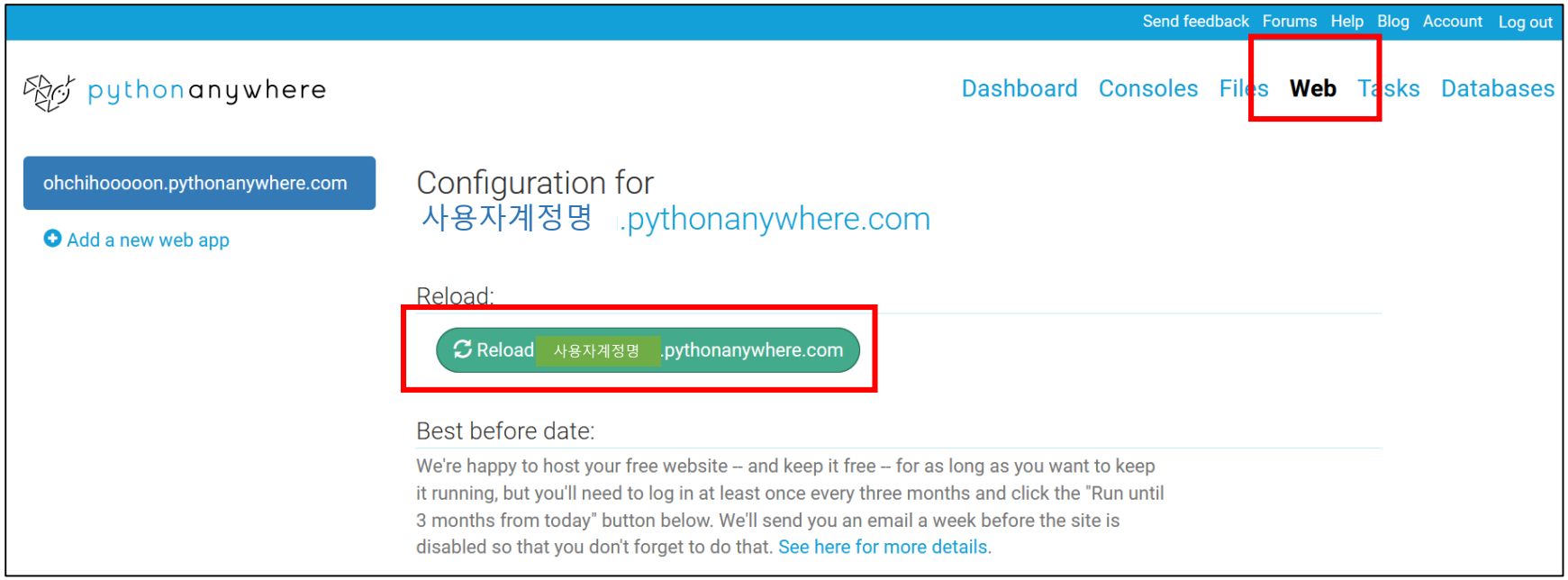
- static file 추출

```
(knitac-env) 02:35 ~/apps/mysite2 $
(knitac-env) 02:35 ~/apps/mysite2 $ python manage.py collectstatic

135 static files copied to '/home/사용자계정명/apps/mysite2/www-dir/static'.
(knitac-env) 02:35 ~/apps/mysite2 $
```

서버 실행

- Python 경로 환경 변수 설정



- 브라우저에서 확인