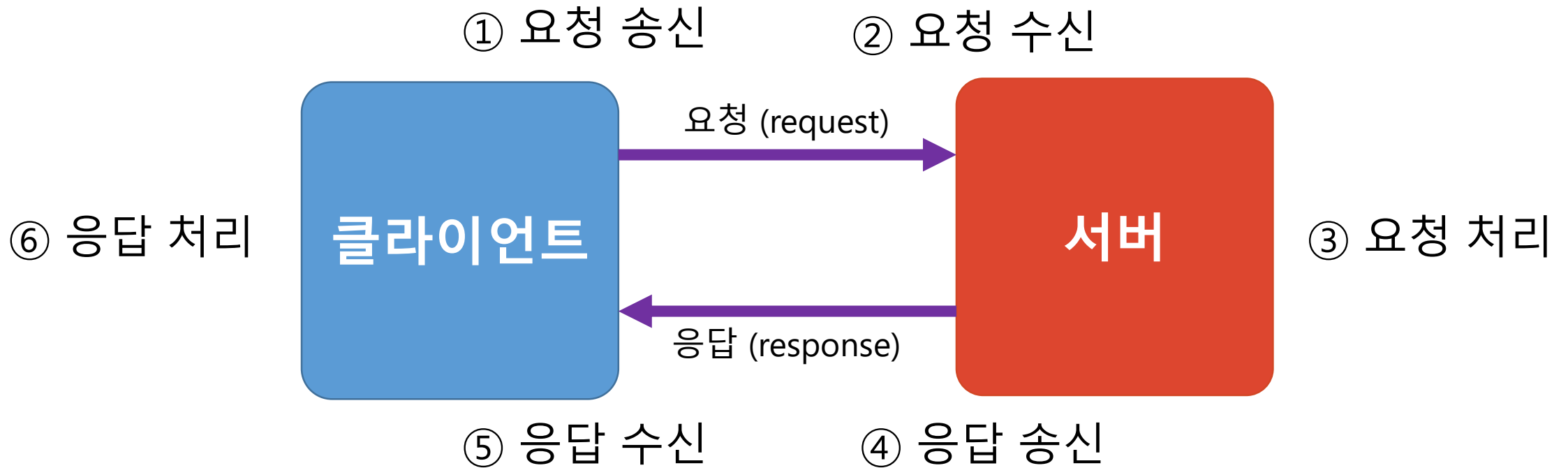


# Django web programming

2015.11.04

# 네트워크



# HTTP 요청 / 응답

```
$ telnet www.example.com 80
```

```
Trying 93.184.216.34...$
```

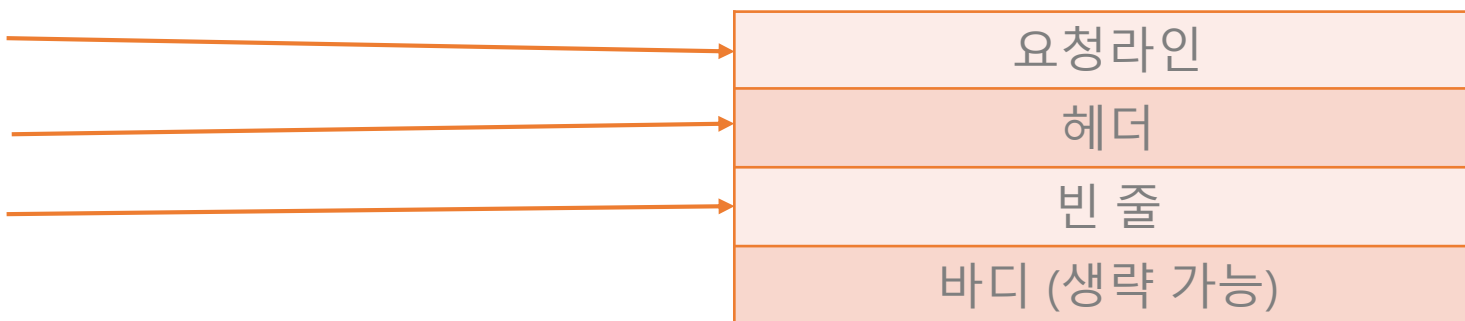
```
Connected to www.example.com.
```

```
Escape character is '^]'.
```

```
GET / HTTP/1.1
```

```
Host: www.example.com
```

```
- <Enter> 입력
```



# HTTP 요청 / 응답

HTTP/1.1 200 OK  
Accept-Ranges: bytes  
Cache-Control: max-age=604800  
Content-Type: text/html  
Date: Tue, 03 Nov 2015 16:18:00 GMT  
Etag: "359670651"  
Expires: Tue, 10 Nov 2015 16:18:00 GMT  
Last-Modified: Fri, 09 Aug 2013 23:54:35 GMT  
Server: ECS (cpm/F9D5)  
X-Cache: HIT  
x-ec-custom-error: 1  
Content-Length: 1270

<!doctype html>  
<html>  
<head>  
 (중략)  
</head>  
<body>  
 (중략)  
</body>  
</html>

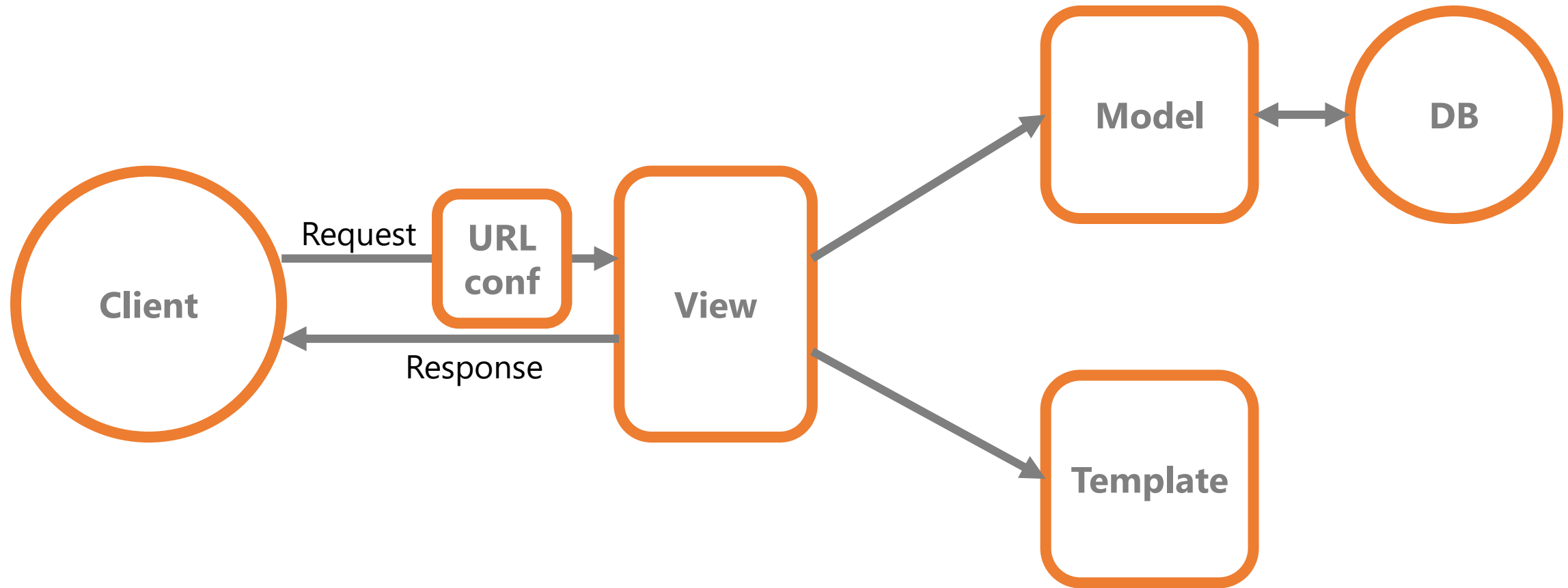
상태라인

헤더

빈 줄

바디

# Django MTV

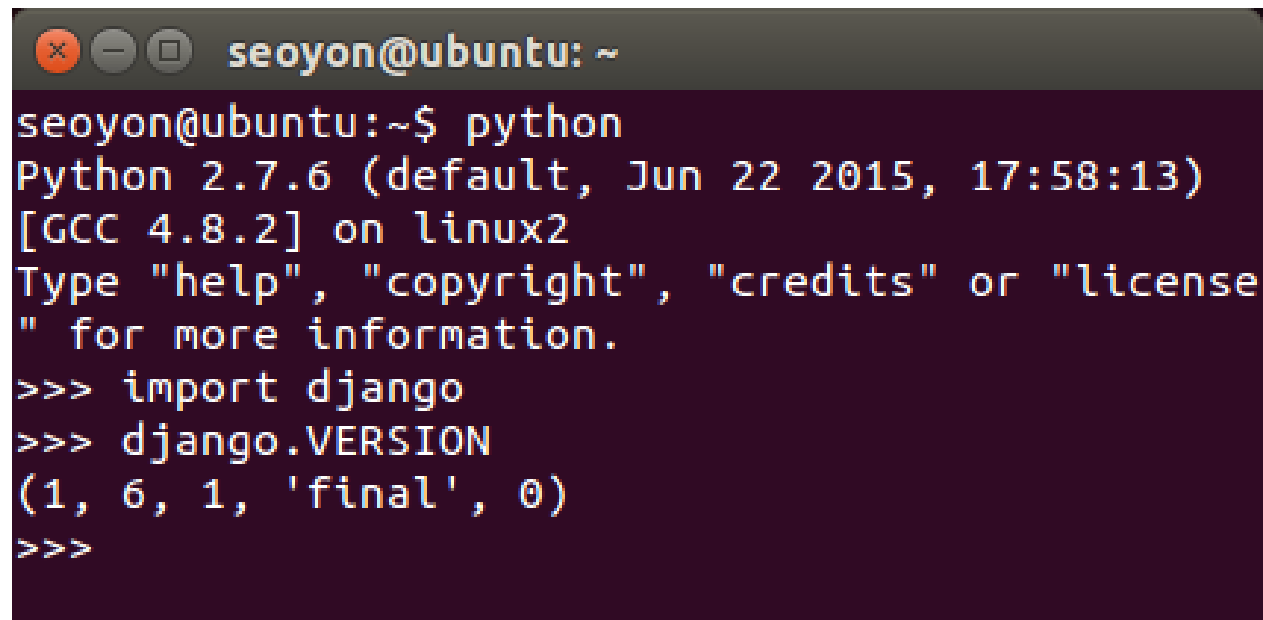


# Django 설치 & 새 프로젝트 시작

- Django 설치

```
$ sudo apt-get install python-django
```

- Django 버전 확인

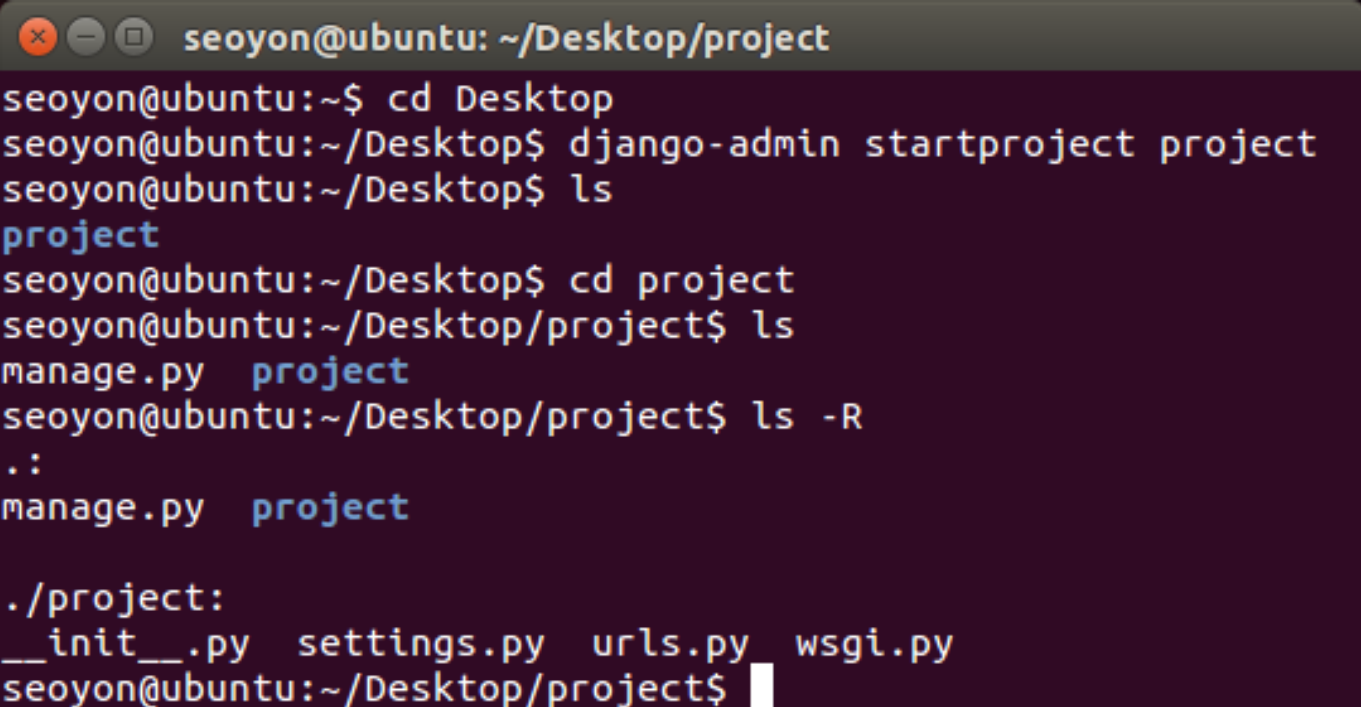
A terminal window with a dark background and light text. The window title bar shows 'seoyon@ubuntu: ~'. The terminal content shows the execution of 'python' and 'import django', followed by the output of 'django.VERSION' which is '(1, 6, 1, 'final', 0)'.

```
seoyon@ubuntu:~$ python
Python 2.7.6 (default, Jun 22 2015, 17:58:13)
[GCC 4.8.2] on linux2
Type "help", "copyright", "credits" or "license"
for more information.
>>> import django
>>> django.VERSION
(1, 6, 1, 'final', 0)
>>>
```

# Django 설치 & 새 프로젝트 시작

- 새 프로젝트 시작

```
$ django-admin startproject project
```



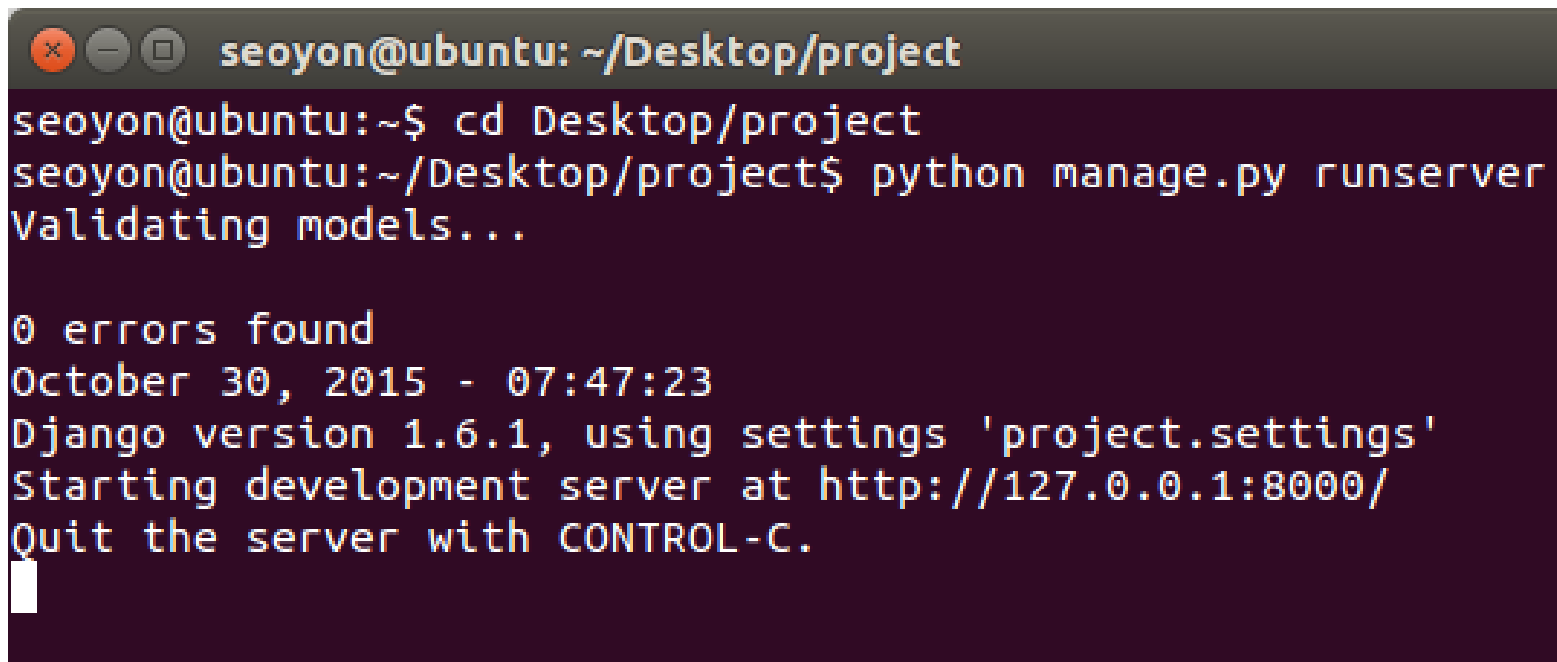
```
seoyon@ubuntu: ~/Desktop/project
seoyon@ubuntu:~$ cd Desktop
seoyon@ubuntu:~/Desktop$ django-admin startproject project
seoyon@ubuntu:~/Desktop$ ls
project
seoyon@ubuntu:~/Desktop$ cd project
seoyon@ubuntu:~/Desktop/project$ ls
manage.py  project
seoyon@ubuntu:~/Desktop/project$ ls -R
.:
manage.py  project

./project:
__init__.py  settings.py  urls.py  wsgi.py
seoyon@ubuntu:~/Desktop/project$
```

# Django 설치 & 새 프로젝트 시작

- Django 웹 서버 실행

```
$ python manage.py runserver
```

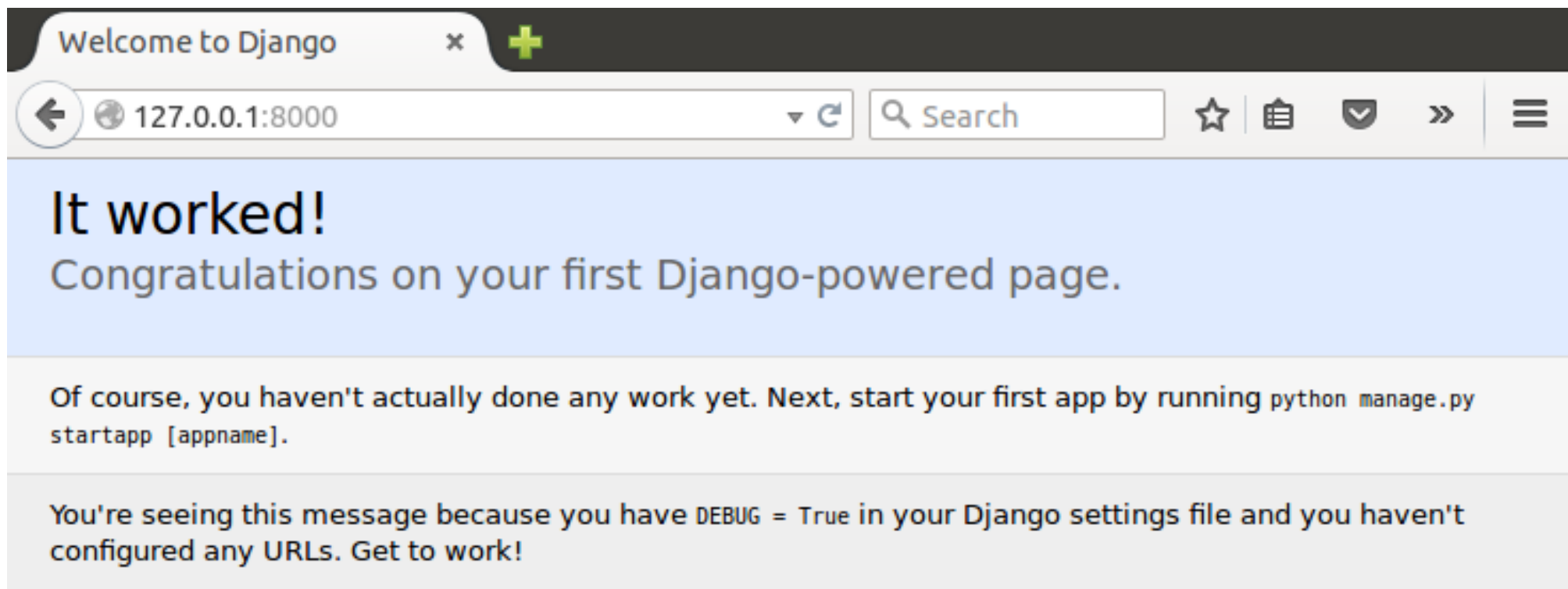
A terminal window with a dark background and light-colored text. The window title bar shows 'seoyon@ubuntu: ~/Desktop/project'. The terminal output shows the user navigating to the project directory and running the Django server command. The output indicates that no errors were found and the development server is starting at http://127.0.0.1:8000/.

```
seoyon@ubuntu: ~/Desktop/project
seoyon@ubuntu:~$ cd Desktop/project
seoyon@ubuntu:~/Desktop/project$ python manage.py runserver
Validating models...

0 errors found
October 30, 2015 - 07:47:23
Django version 1.6.1, using settings 'project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```



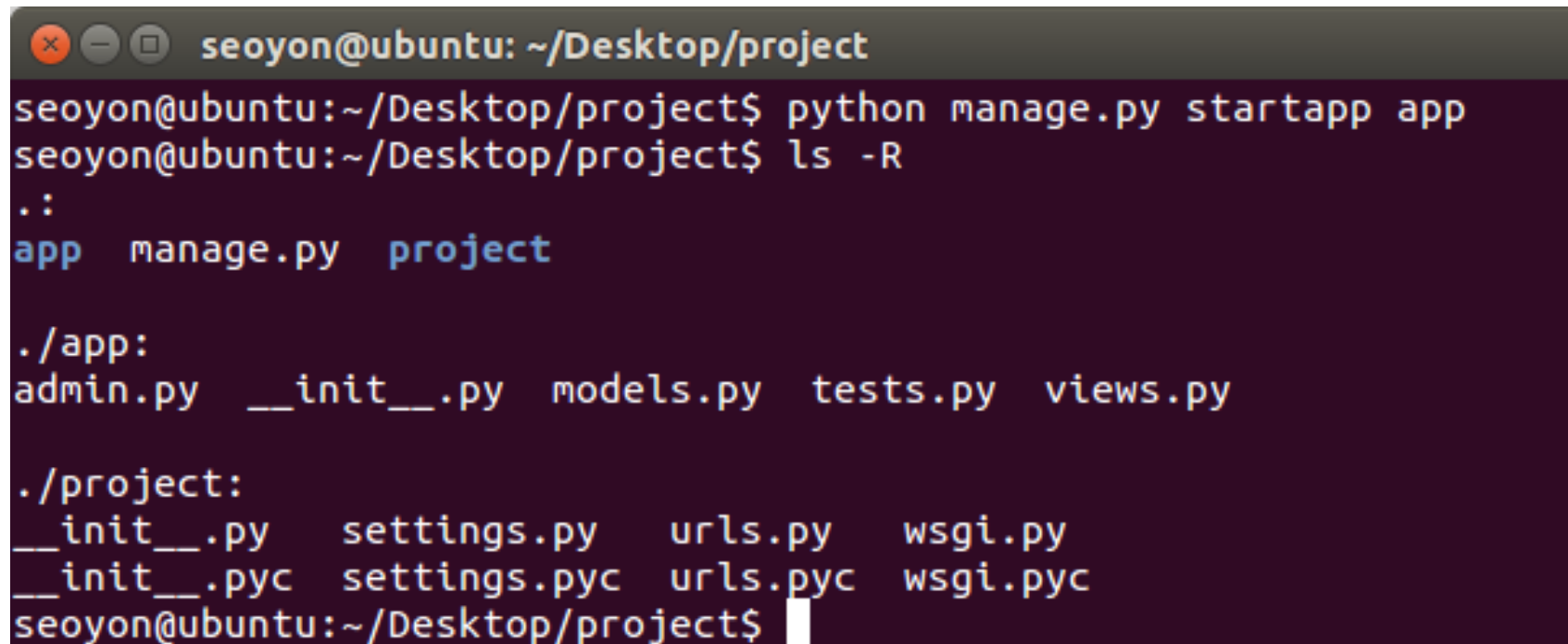
# Django 설치 & 새 프로젝트 시작



# 앱 생성 및 설정

- 앱 생성

```
$ python manage.py startapp app
```



```
seoyon@ubuntu: ~/Desktop/project
seoyon@ubuntu:~/Desktop/project$ python manage.py startapp app
seoyon@ubuntu:~/Desktop/project$ ls -R
.:
app  manage.py  project

./app:
admin.py  __init__.py  models.py  tests.py  views.py

./project:
__init__.py  settings.py  urls.py  wsgi.py
__init__.pyc  settings.pyc  urls.pyc  wsgi.pyc
seoyon@ubuntu:~/Desktop/project$
```

# 앱 생성 및 설정

- project/settings.py

```
INSTALLED_APPS = (  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
    'app',  
)
```

# Hello world! 출력

- app/views.py

```
from django.shortcuts import render  
from django.http import HttpResponse
```

```
# Create your views here.
```

```
def hello_world(request):  
    html = "<html><body><h1>Hello world!</h1></body></html>"  
    return HttpResponse(html)
```

# Hello world! 출력

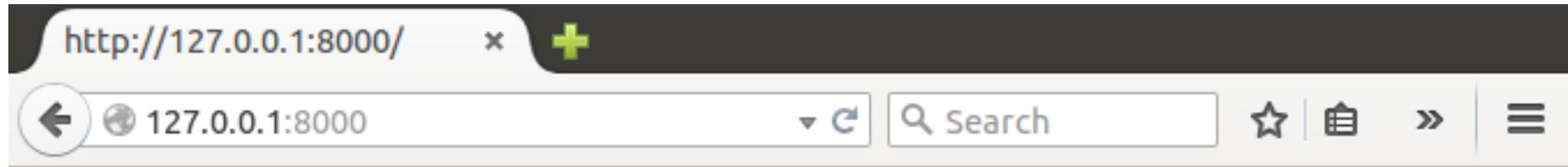
- project/urls.py

```
from django.contrib import admin  
admin.autodiscover()
```

```
from app.views import *
```

```
urlpatterns = patterns('',  
    ...  
    # url(r'^admin/', include(admin.site.urls)),  
    url(r'^$', hello_world),  
)
```

# Hello world! 출력



## Hello world!

# Template 사용

- project/settings.py

...

```
STATIC_URL = '/static/'
```

```
TEMPLATE_DIRS = (os.path.join(BASE_DIR, 'templates/'))
```

# Template 사용

- project 디렉토리에 templates 디렉토리 생성
- templates/hello\_world.html 파일 생성

```
<html>  
  <body>  
    <h1>Hello world!</h1>  
  </body>  
</html>
```



# Template 사용

- app/views.py

```
def hello_world(request):  
    return render(request, 'hello_world.html')
```



**Hello world!**

# Model & DB

- app/models.py

```
# Create your models here.
```

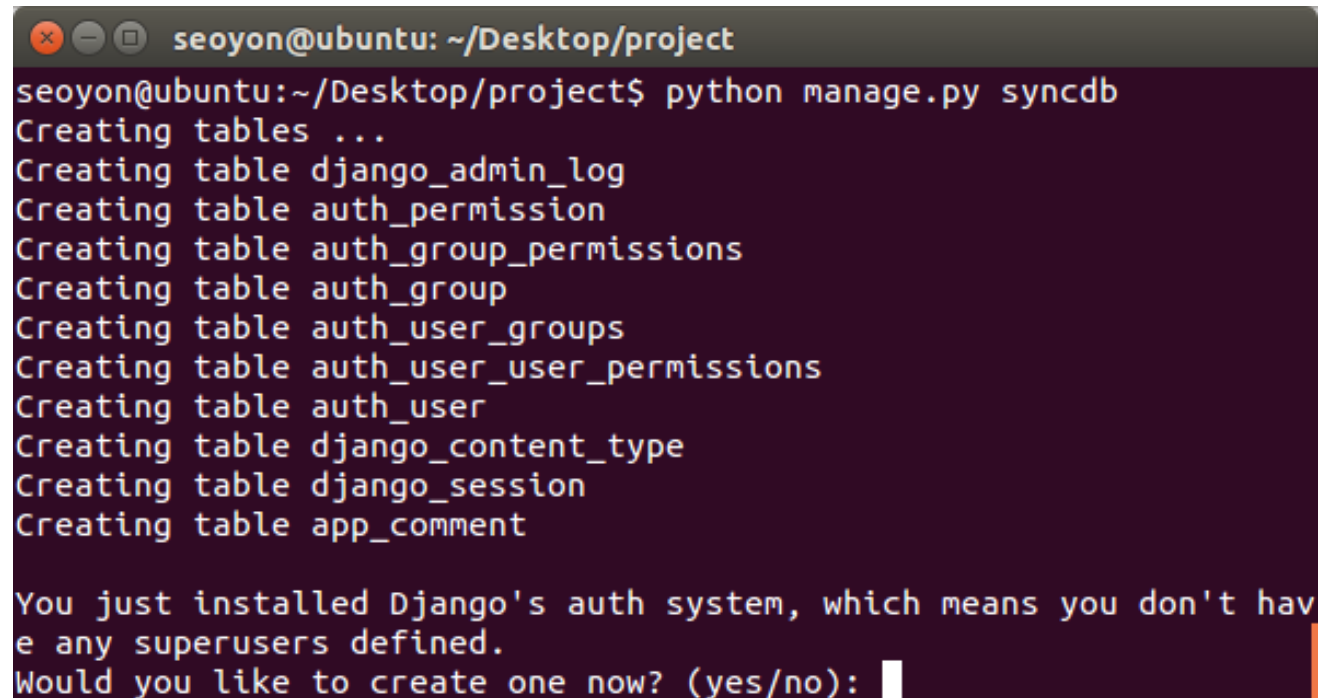
```
class Comment(models.Model):  
    name = models.CharField(max_length = 30)  
    date = models.DateField(auto_now = True)  
    content = models.CharField(max_length = 140)
```

- <https://docs.djangoproject.com/en/1.6/ref/models/fields/>

# Model & DB

- DB 생성

\$ python manage.py syncdb



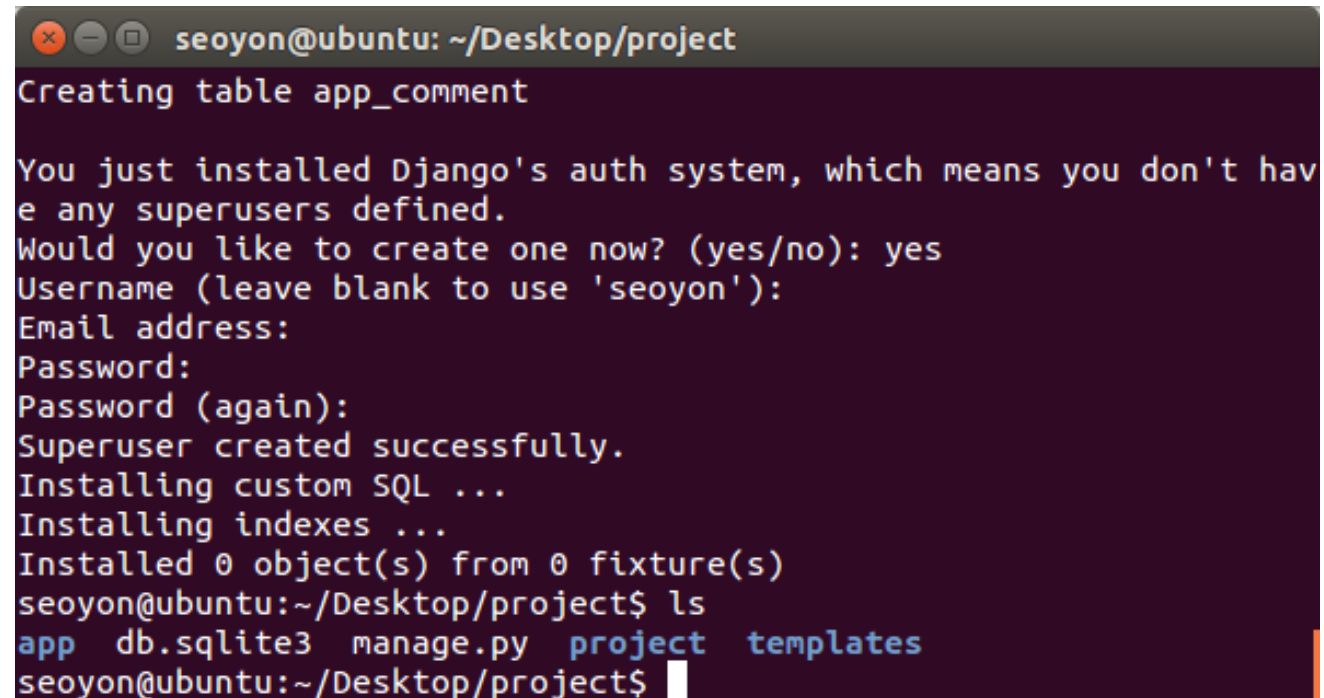
```
seoyon@ubuntu: ~/Desktop/project
seoyon@ubuntu:~/Desktop/project$ python manage.py syncdb
Creating tables ...
Creating table django_admin_log
Creating table auth_permission
Creating table auth_group_permissions
Creating table auth_group
Creating table auth_user_groups
Creating table auth_user_user_permissions
Creating table auth_user
Creating table django_content_type
Creating table django_session
Creating table app_comment

You just installed Django's auth system, which means you don't have any superusers defined.
Would you like to create one now? (yes/no):
```

# Model & DB

- DB 생성

```
$ python manage.py syncdb
```



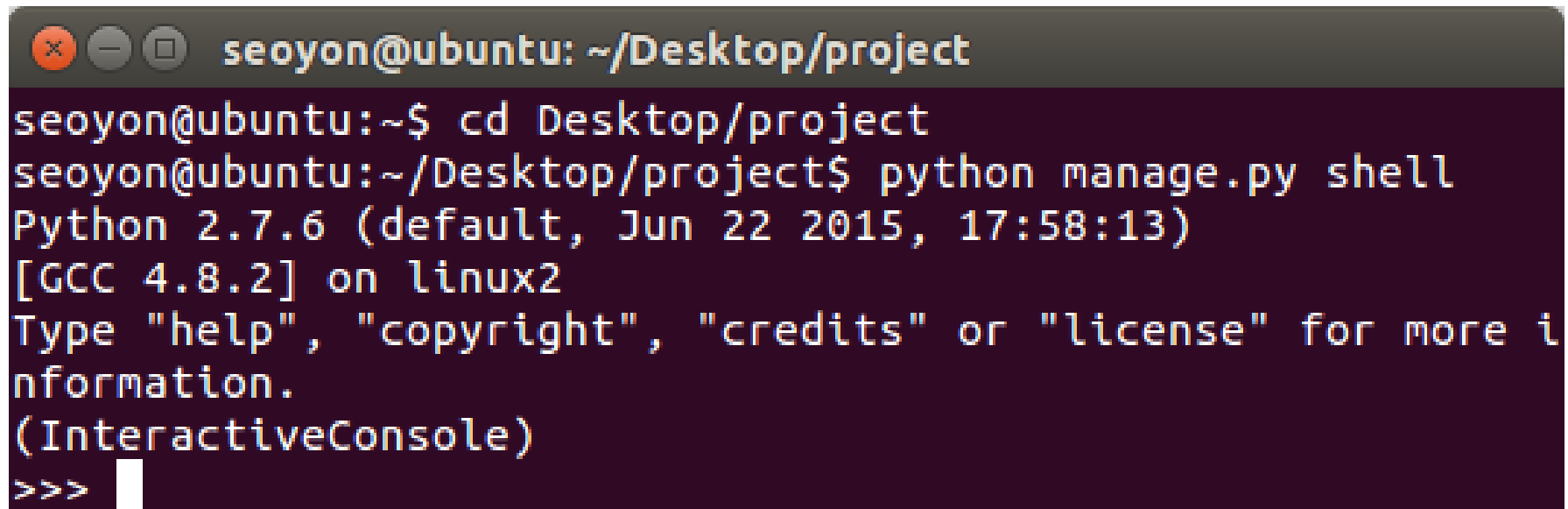
```
seoyon@ubuntu: ~/Desktop/project
Creating table app_comment

You just installed Django's auth system, which means you don't have any superusers defined.
Would you like to create one now? (yes/no): yes
Username (leave blank to use 'seoyon'):
Email address:
Password:
Password (again):
Superuser created successfully.
Installing custom SQL ...
Installing indexes ...
Installed 0 object(s) from 0 fixture(s)
seoyon@ubuntu:~/Desktop/project$ ls
app  db.sqlite3  manage.py  project  templates
seoyon@ubuntu:~/Desktop/project$
```

# Model & DB

- Django shell 실행

```
$ python manage.py shell
```

A terminal window with a dark background and light-colored text. The window title bar shows the user 'seoyon' on 'ubuntu' at the directory '~/Desktop/project'. The terminal content shows the user navigating to the project directory and running 'python manage.py shell'. The output displays the Python version (2.7.6), GCC version (4.8.2), and the system (linux2). It also shows a prompt for help and the start of the interactive console.

```
seoyon@ubuntu: ~/Desktop/project
seoyon@ubuntu:~$ cd Desktop/project
seoyon@ubuntu:~/Desktop/project$ python manage.py shell
Python 2.7.6 (default, Jun 22 2015, 17:58:13)
[GCC 4.8.2] on linux2
Type "help", "copyright", "credits" or "license" for more i
nformation.
(InteractiveConsole)
>>> 
```

# Model & DB

- Model 객체 생성

```
>>> from app.models import *  
>>> c = Comment(name = 'test', content = 'test')  
>>> c.save()
```

or

```
>>> Comment(  
...     name = 'test',  
...     content = 'test'  
... ).save()
```

# Model & DB

- Model 객체 수정

```
>>> c = Comment.objects.get()
>>> c.content = 'Hello world!'
>>> c.save()
```

- Model 객체 삭제

```
>>> c = Comment.objects.get()
>>> c.delete()
```

- <https://docs.djangoproject.com/en/1.6/ref/models/instances/>
- <https://docs.djangoproject.com/en/1.6/ref/models/querysets/>

# Model 사용 & Template 변수

- app/views.py

```
from django.shortcuts import render
from django.http import HttpResponse
from app.models import *
...
def main_page(request):
    comments = Comment.objects.order_by('date')
    var = {
        'comments': comments,
    }
    return render(request, 'home.html', var)
```



# Template 스크립트

- templates/home.html 생성
  - 다음 페이지의 코드대로 입력
  - <https://docs.djangoproject.com/en/1.6/ref/templates/builtins/>
- project/urls.py

...

```
# url(r'^admin/', include(admin.site.urls)),  
url(r'^$', main_page),
```

```
)
```

# Template 스크립트

```
<html>
    <head>
        <title>Django practice - Home</title>
    </head>
    <body>
        <h1>Comments</h1>
        {% for comment in comments %}
            <p>
                Name: {{ comment.name }}<br />
                Date: {{ comment.date }}<br />
                <br />
                &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&{{ comment.content }}<br />
            </p>
        {% endfor %}
    </body>
</html>
```

# Model – Template – View



## Comments

Name: seoyon

Date: Oct. 30, 2015

Hello world!

Name: test

Date: Oct. 30, 2015

test

# Form & POST 요청

- app/forms.py 파일 생성

```
from django import forms
```

```
class CommentForm(forms.Form):
```

```
    name = forms.CharField(max_length = 30)
```

```
    content = forms.CharField(widget = forms.Textarea, max_length = 140)
```

- <https://docs.djangoproject.com/en/1.6/ref/forms/fields/>

# Form & POST 요청

- shell 실행 (python manage.py shell)

```
>>> from app.forms import *
```

```
>>> f = CommentForm()
```

```
>>> print f
```

```
>>> f = CommentForm({'name': 'test', 'content': 'test'})
```

```
>>> print f
```

**Name:**

**Content:**

# Form & POST 요청

- app/views.py

```
from django.shortcuts import render
from django.http import HttpResponse, HttpResponseRedirect
```

```
from app.models import *
from app.forms import *
```

```
...
```

```
(continued)
```

# Form & POST 요청

```
def main_page(request):  
    if request.method == 'POST':  
        form = CommentForm(request.POST)  
        if form.is_valid():  
            name = form.cleaned_data['name']  
            content = form.cleaned_data['content']  
            Comment(name = name, content = content).save()  
            return HttpResponseRedirect('/')  
        else:  
            form = CommentForm()  
    comments = Comment.objects.order_by('date')  
    var = {  
        'comments': comments,  
        'form': form,  
    }  
    return render(request, 'home.html', var)
```

# Form & POST 요청

- templates/home.html

```
<h1>Comments</h1>
<form action = "." method = "POST">
  <style>
    th {
      text-align: right;
      vertical-align: top;
    }
  </style>
  <table>{{ form }}</table>
  <input type = "submit" value = "Add comment" />
</form>
{% for comment in comments %}
```



# 방명록 남기기

Django practice - Home

127.0.0.1:8000

Search

☆

📄

»

☰

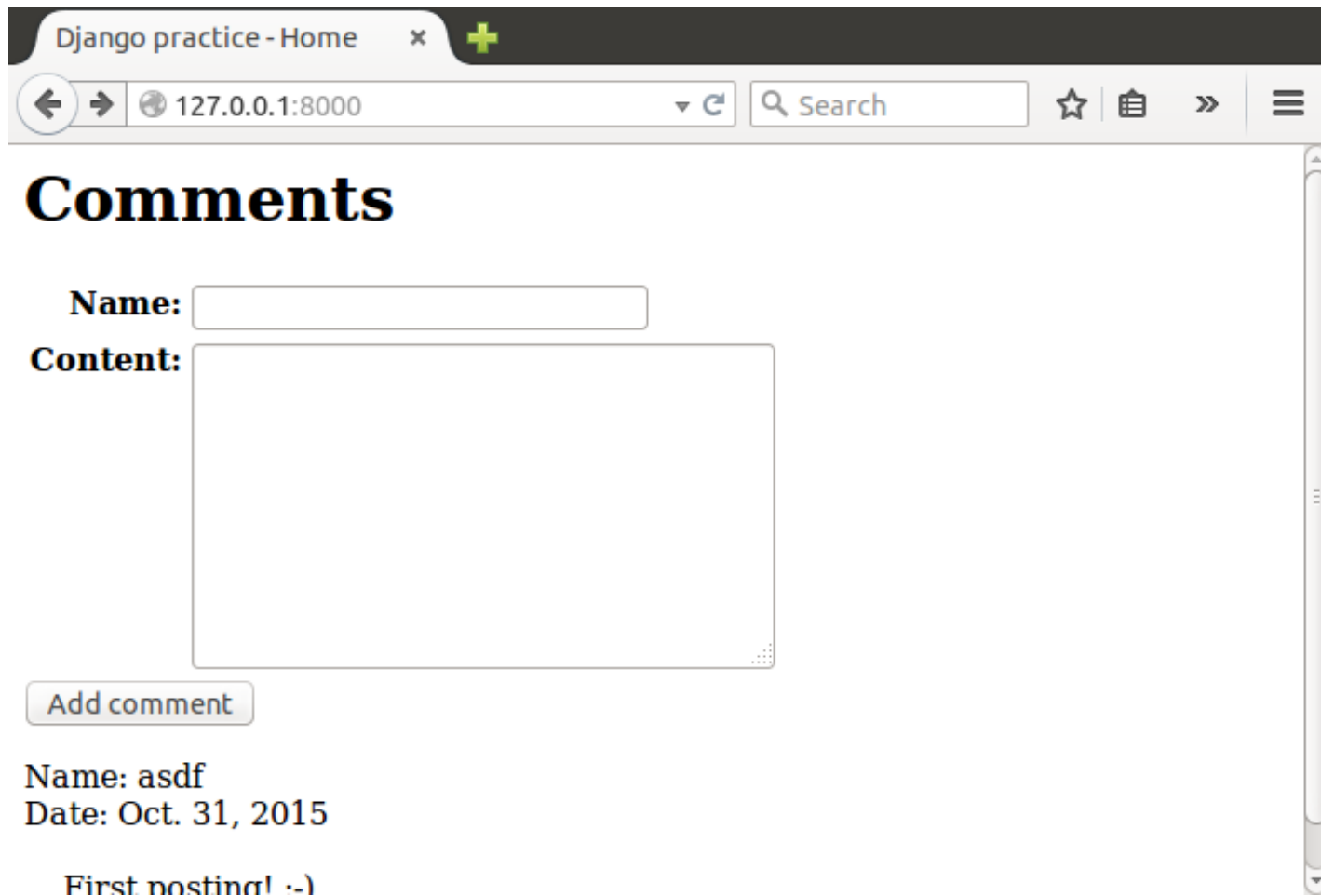
## Comments

**Name:**

**Content:**

Add comment

# 방명록 남기기



The screenshot shows a web browser window with the title 'Django practice - Home'. The address bar displays '127.0.0.1:8000'. The page content features a large heading 'Comments'. Below this, there is a form with two input fields: 'Name:' and 'Content:'. The 'Name' field contains the text 'asdf'. The 'Content' field is a larger text area. Below the form is a button labeled 'Add comment'. At the bottom of the page, there is a comment entry: 'Name: asdf', 'Date: Oct. 31, 2015', and the text 'First postina! :-)'.

Django practice - Home x +

127.0.0.1:8000 Search ☆ 📄 » ☰

## Comments

Name:

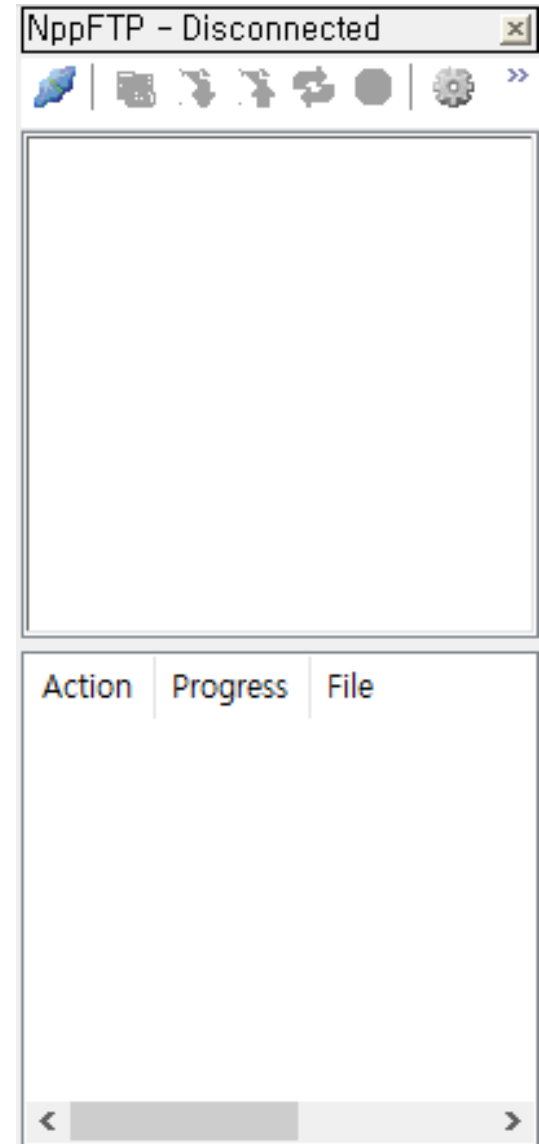
Content:

Add comment

Name: asdf  
Date: Oct. 31, 2015  
First postina! :-)

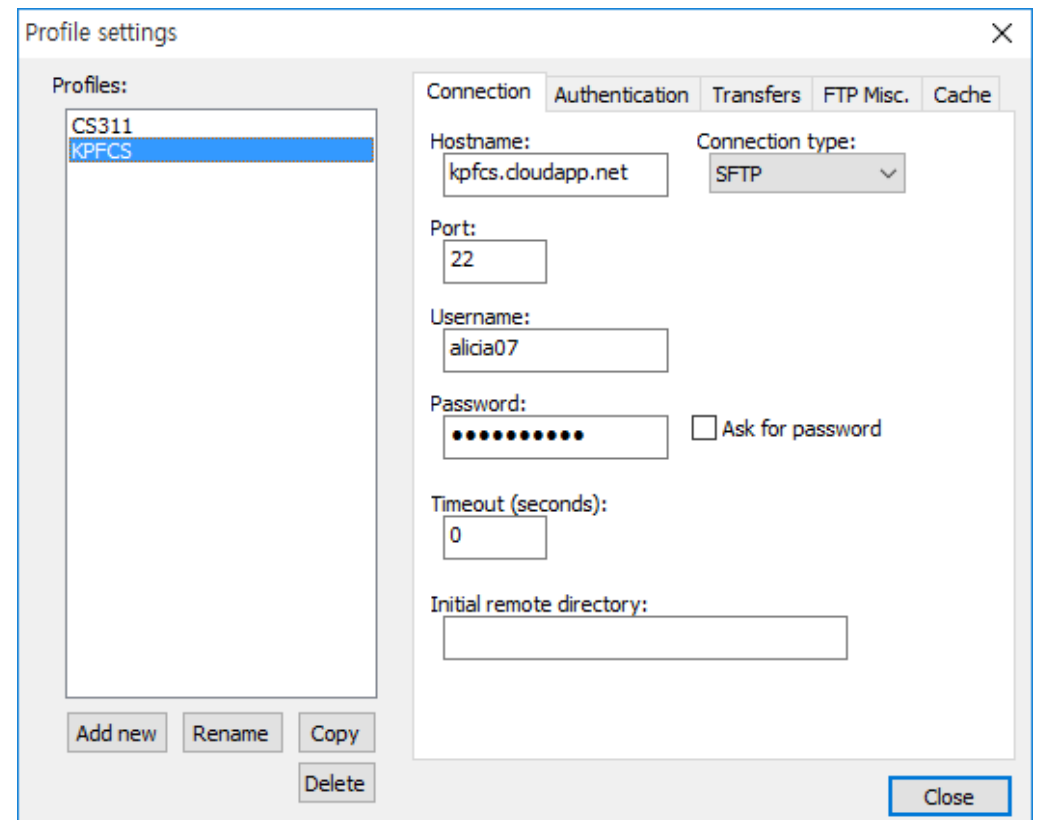
# 원격 프로그래밍

- Notepad++
  - <https://notepad-plus-plus.org/>
  - Or <http://www.sublimetext.com/>
- Notepad++ 실행 → 플러그인  
→ NppFTP → Show NppFTP Window



# 원격 프로그래밍

- 설정 → Profile Settings → Add new
- 호스트 이름 : kpfcs.cloudapp.net
- 연결 방식 : SFTP
- 연결



# 그 외...

- 프로젝트 디렉토리:
  - /django/kpfcs/
- 오늘 강의한 소스 코드를 보려면:
  - <https://github.com/seoyon/django>
- Django documentation
  - 1.6 <https://docs.djangoproject.com/en/1.6/>
  - stable <https://docs.djangoproject.com/en/stable/>