

# Django REST

(2018 봄학기 ) 소프트웨어 개발의 원리 및 실제  
2018. 03. 27 ( 화 )

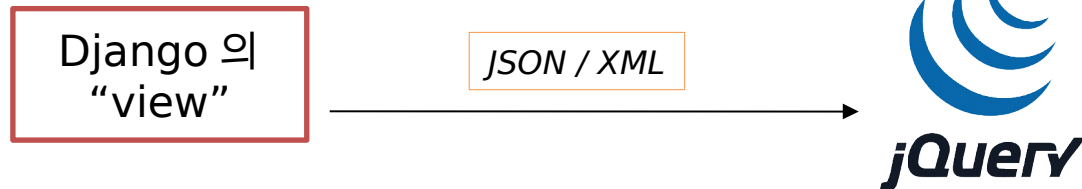
# Django REST

## • 서버에서 **URL** 이 불리었을 때 할 수 있는 행동을 패턴화 하자 .

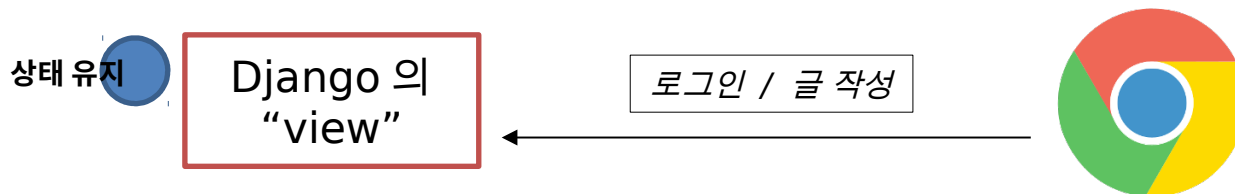
1. 요청한 웹 페이지를 client ( 웹 브라우저 ) 로 전송



2. 요청한 특정 데이터를 전송



3. 요청을 받아서 데이터를 업데이트 및 상태 유지



# Django REST

---

- **Django REST :**

- 서버는 일종의 거대한 **data structure** 이다 .
- **<HTTP method, URL>** 쌍은 **data structure** 에 대한 연산이다 .

- **예 : Dictionary 라는 Django REST 프로젝트를 만들었다고 했을 때 ,**

	GET	POST	PUT	DELETE
<a href="http://localhost/words">http://localhost/words</a>	전체 단어의 목록을 return	새로운 단어를 등록	에러	에러
<a href="http://localhost/words/lunch">http://localhost/words/lunch</a>	“lunch” 의 뜻을 return	에러	Lunch 의 단어 뜻을 update	단어 Lunch 를 삭제

- 유틸리티 **httpie(pip3 install httpie)**

```
$ http http://127.0.0.1:8000/snippets/
```

<http://.../snippets/>에 GET 으로 메시지를 보냄

```
$ http POST http://127.0.0.1:8000/snippets/
```

<http://.../snippets/>에 POST 로 메시지를 보냄

— 참고 : 웹 브라우저에서 뜨는 웹 페이지는 **GET** 을 요청한 결과

# Django REST – Tutorial

---

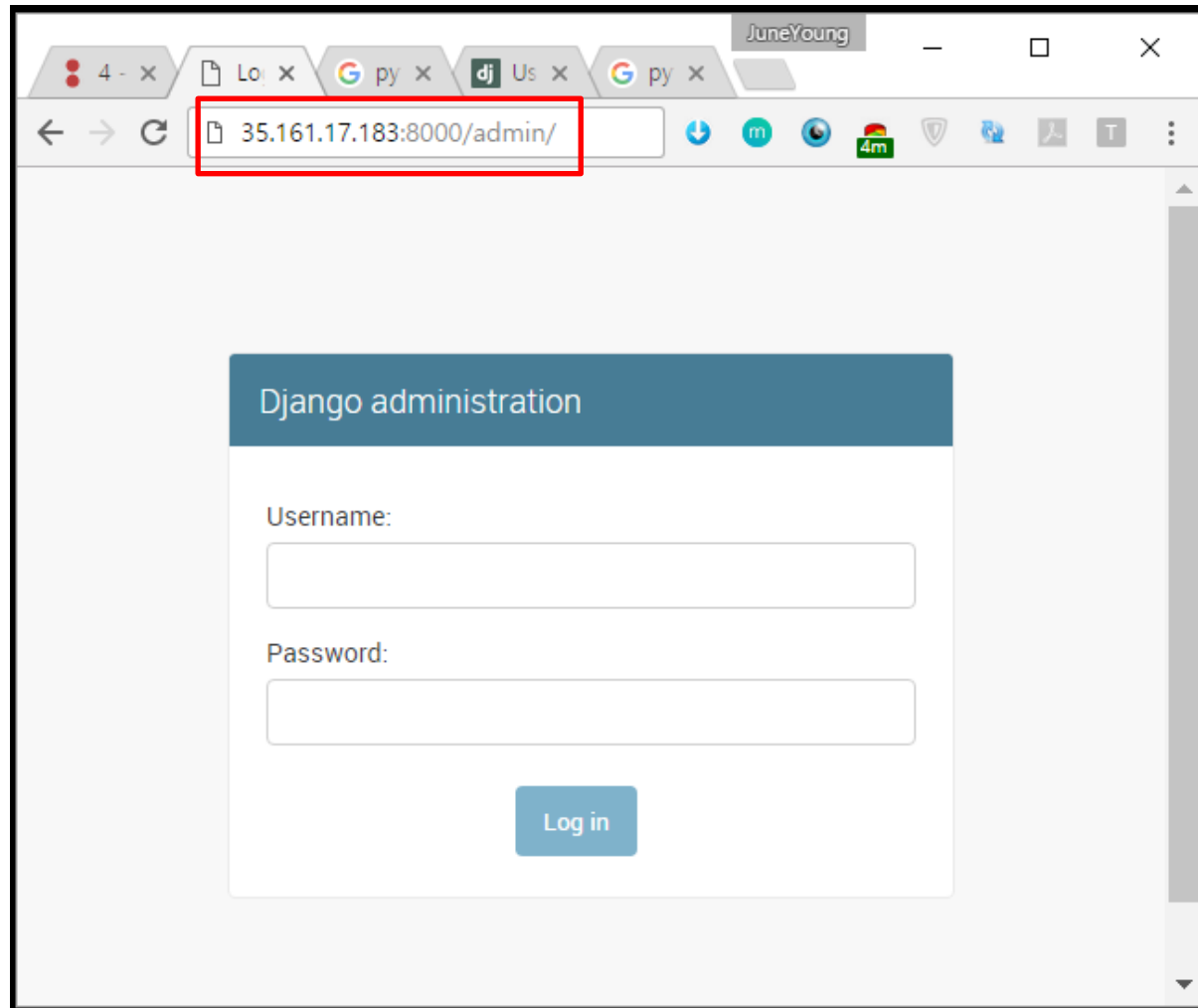
- <http://www.django-rest-framework.org/tutorial/1-serialization/>
- **1-serialization ~ 4-authentication-and-permissions**  
까지
- 구현하는 웹 애플리케이션 : 코드 스니펫을 저장하는 홈페이지
  - \_ Do you know <http://pastebin.com/> ?

# Django REST – Tutorial

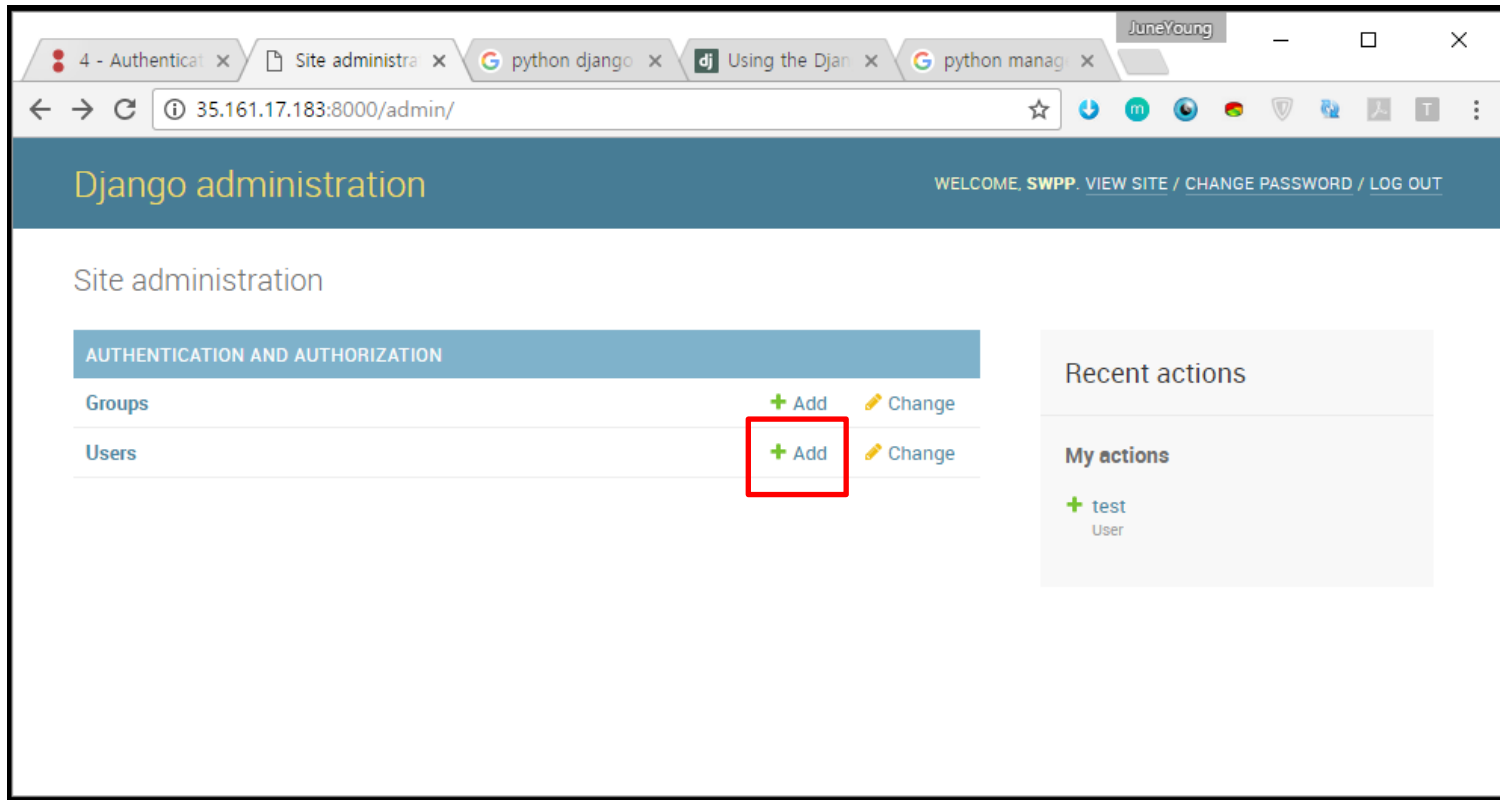
	GET	POST	PUT	DELETE
.../snippets/	스니펫 목록	새로운 스니펫을 추가	에러	에러
.../snippets/1/	1 번째 스니펫	에러	내용 수정	삭제
.../users/	유저 목록	에러	에러	에러
.../users/2	2 번째 유저	에러	에러	에러

\* 유저는 어떻게 등록을 하나요 ?

# Django REST –Add User?



# Django REST –Add User?





# Django REST -Add User?

---

- **Or.. 터미널을 이용해서 .**

```
(env) swpp@ip-172-31-30-158:~/resttutorial/tutorial$ python manage.py shell
Python 3.5.2 (default, Nov 17 2016, 17:05:23)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
>>> from django.contrib.auth.models import User
>>> user = User.objects.create_user('test3', password='test2212343')
>>> user.save()
>>> quit()
(env) swpp@ip-172-31-30-158:~/resttutorial/tutorial$
```

# Django REST – Miscellaneous Setting

- 혹시 **AWS** 가 너무 느리면 ..
  - \_ **Local** 에서 작업한 뒤 **scp** 를 사용해 보세요 .

```
scp -P< 포트 > a.txt  
user@your.server.example.com:/path/to/b.txt
```

- **Vim** 세팅 중 도움이 될 만한 것

```
set ai  
set expandtab  
set ts=4  
set shiftwidth=4  
set number  
set whichwrap+=<,>,h,l[,]  
set background=dark
```

# Django REST – Tutorial 1. Serialization

- <http://www.django-rest-framework.org/tutorial/1-serialization/>
- “**Serialization**” : 입 / 출력을 위해 데이터의 요소에 순서를 부여
- **Snippet** 이라는 데이터타입을 정의 및 여러 연산 구현
- 참고 :
  - \_ **virtualenv -p python3** 을 쓰세요
  - \_ **pip** 대신 **pip3** 을 사용하세요
  - \_ **Vim** 에서 붙여넣기 할 때 - **`:set paste`**

# Django REST – Tutorial 2. Requests and Responses

- <http://www.django-rest-framework.org/tutorial/2-requests-and-responses/>
- 여러 연산을 Django REST 에서 제공하는 클래스로 구현
- **Browsable** 구현
  - \_ **Browsable** : 웹 브라우저에서 열었을 때 예쁘게 나오도록 함

# Django REST – Tutorial 3. Class-based Views

- <http://www.django-rest-framework.org/tutorial/3-class-based-views/>
- **Django REST 에서 제공하는 클래스를 상속함으로써 , 비슷한 패턴의 코드를 없앴**

```
class SnippetDetail(mixins.RetrieveModelMixin,  
                   mixins.UpdateModelMixin,  
                   mixins.DestroyModelMixin,  
                   generics.GenericAPIView):  
    queryset = Snippet.objects.all()
```

} 4 개의 class 를 상속

# Django REST – Tutorial 4. Authentication & Permissions

- <http://www.django-rest-framework.org/tutorial/4-authentication-and-permissions/>
- **Snippet** 을 작성한 **User** 를 정의함

```
linenos = self.linenos and 'table' or False  
options = self.title and {'title': self.title} or {}
```

`X and Y or Z` 는 C 에서 “X ? Y : Z” 와 동일