rest framework

rest architecture

api = application programming interface -> we can run multiple app using one database. Example facebook whatsapp insta server down if the organisation use one database but they are interconnected each other through api. if server down then 3 application will not work properly

convert database in api

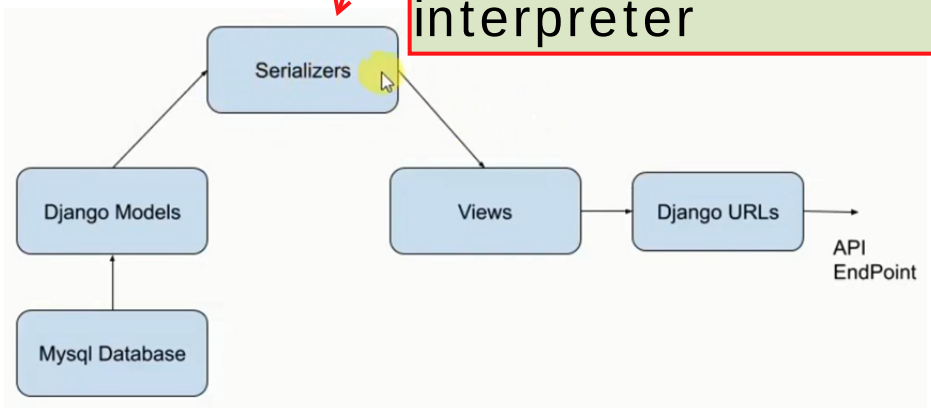
Rest = Representational state transfer

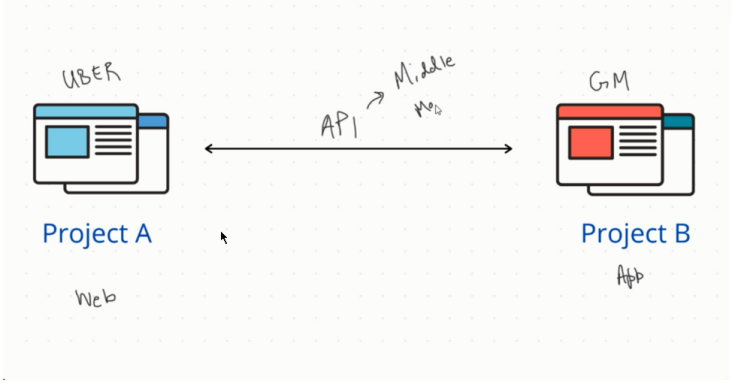
is a type of architecture (make of such kinds of algo) thats makes the data in one kind of standard data form JSON

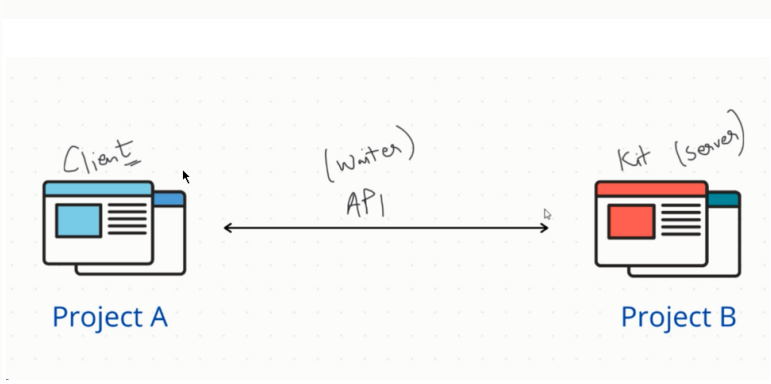
take data from model and convert in json.example chineese interpreter

json=Java Script Object Notation

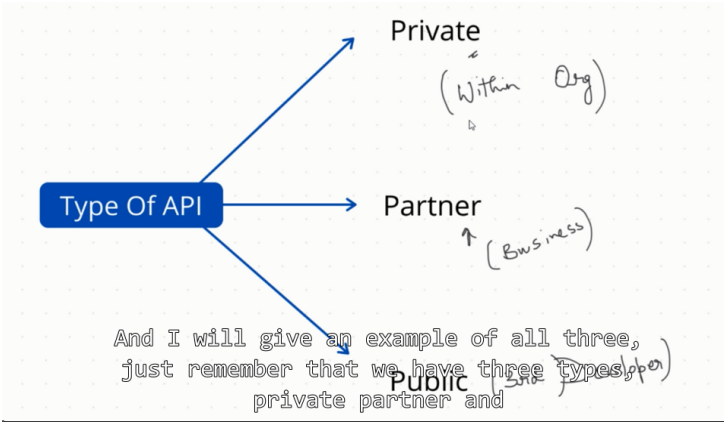
user request-> view
->model->database
->view

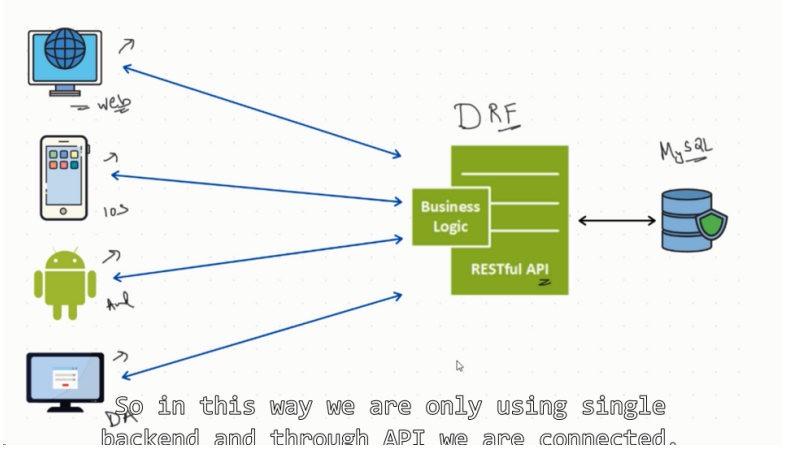


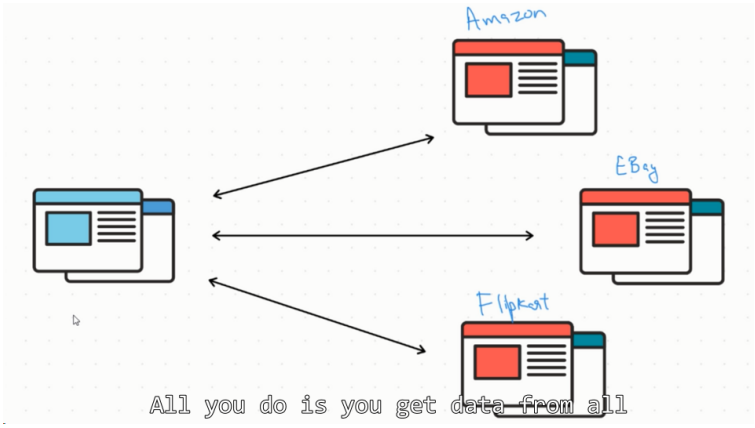


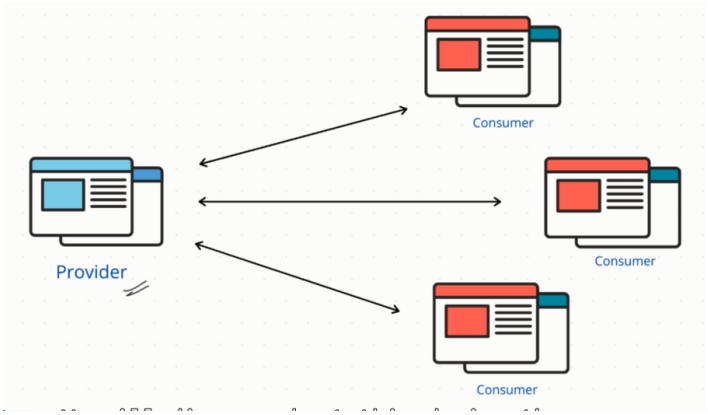


api act as a middle man between client and server









Understanding URL

https://www.api.movielist.com/movies/
https://www.api.movielist.com/movies/list/

https://www.api.movielist.com/movies/127/
https://www.api.movielist.com/movies/127/reviews/
https://www.api.movielist.com/movies/127/reviews/?limit=20

https://www.api.movielist.com/account/login/
https://www.api.movielist.com/account/register/

It can be Spider-Man, Superman or any other movie.

Understanding URL

https://www.api.movielist.com/movies/ List
https://www.api.movielist.com/movies/list/

Ind
https://www.api.movielist.com/movies/127/
https://www.api.movielist.com/movies/127/reviews/
https://www.api.movielist.com/movies/127/reviews/?limit=20

Base URL
End Point

https://www.api.movielist.com/account/login/
https://www.api.movielist.com/account/register/

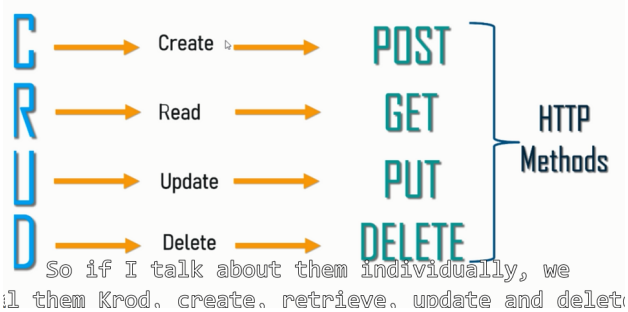
And then this remaining part, this part is known as End Point.

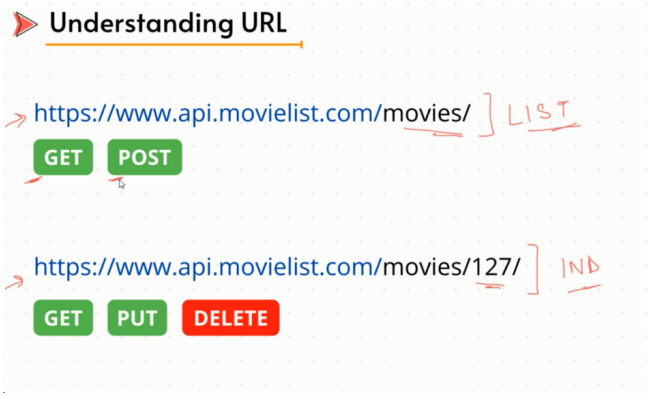
API REST → Rest API

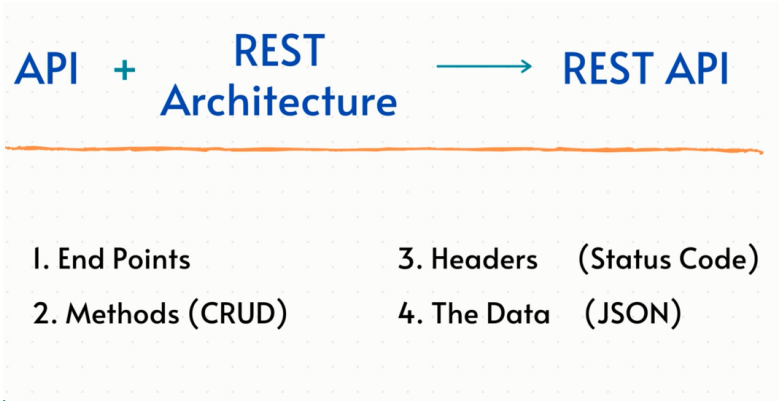
(1) endpoint
(2) method
(3) headers
(4) the data

(status code)

json data





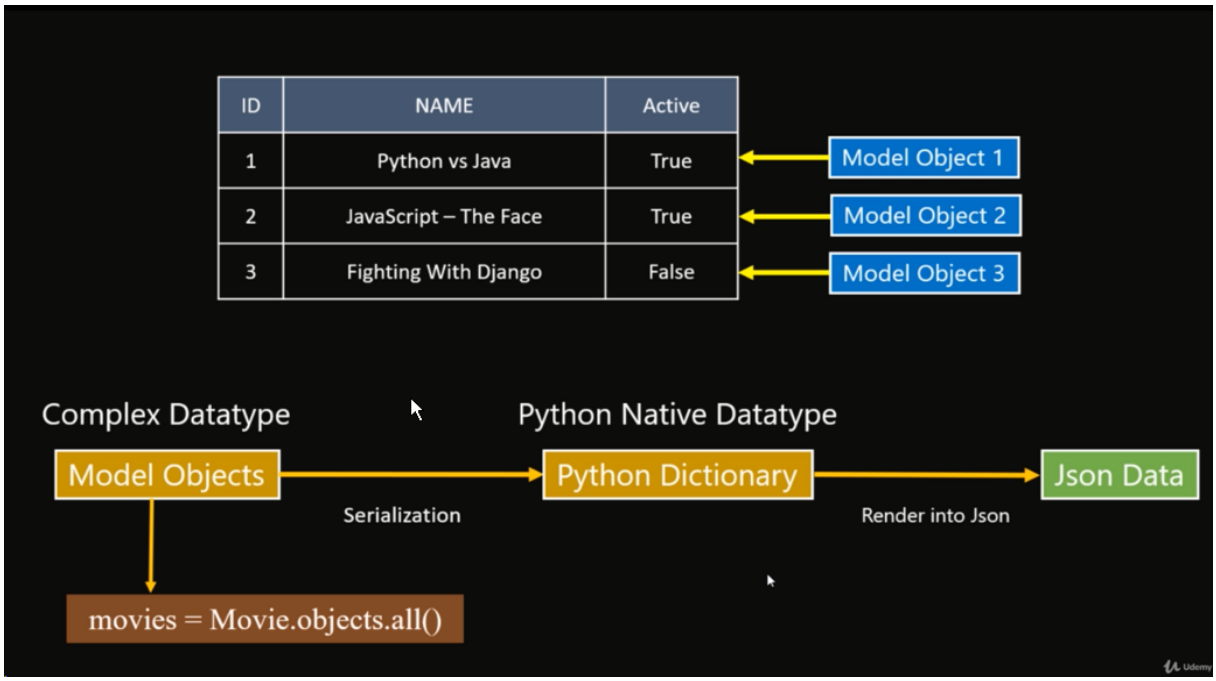


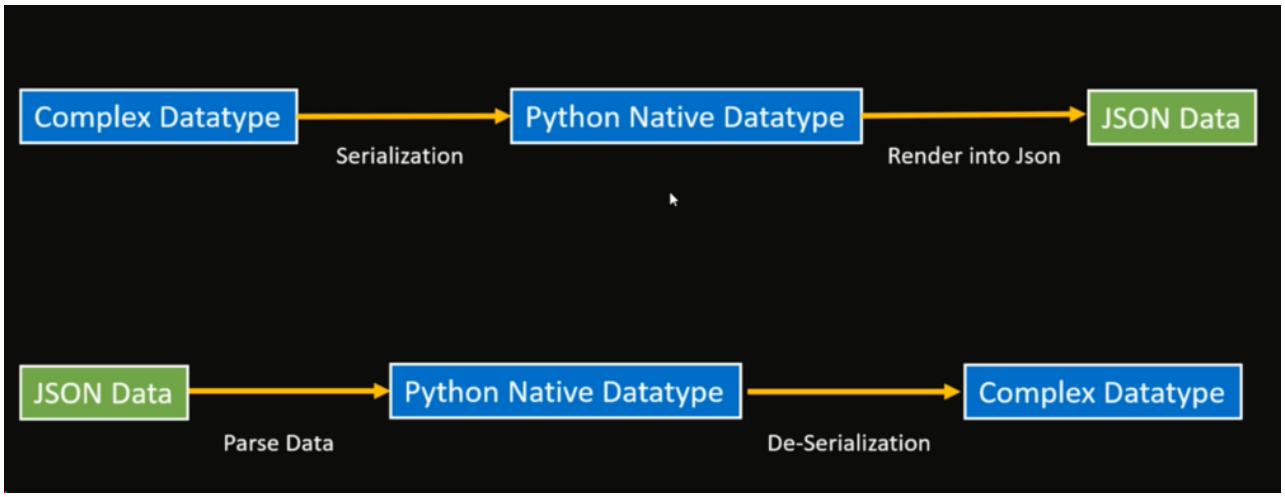

```
def movie(request):
    movies = Movie.objects.all()
    data = {
        'movies': list(movies.values())
    }
    return JsonResponse(data)


    def movie_details(requst,pk):
        movie=Movie.objects.get(id=pk)
        data={
            'name'=movie.name,
            'description'=movie.description
    }
    return JsonResponse(data)
```

queryset -> python dictionary

python dictionary ->json response

serializations in DRF





Type Of Serializers

Type Of Views

Working With API

serializers.Serializer

serializers.ModelSerializer

Serializers

- serializers.Serializer
- serializers.ModelSerializer

Views

Function Based Views

@api_view()

Class Based Views

APIView
- Generic views
- Mixins
- Concrete View Classes
- ViewSets

Working With API

- → DRF Browsable API
- → Postman
- HTTPie