**Django REST Framework**

[**REST**](https://ru.wikipedia.org/wiki/REST)(Representational State Transfer) — это программная архитектура интерфейсов REST.

Django REST Framework (DRF) — это мощный и гибкий инструмент для создания веб-интерфейсов (web APIs). Он дает возможность разработчикам не изобретать колесо, самостоятельно создавая сложный и громоздкий REST-интерфейс с нуля. Чем сложнее будут становиться ваши проекты, тем лучше вы будете осознавать необходимость использования [DRF](https://yandex.ru/turbo/pythonist.ru/s/frejmvork-django-rest/?parent-reqid=1605112728926984-1524380700495956673500108-production-app-host-sas-web-yp-191&utm_source=turbo_turbo) или других полезных REST-фреймворков.

curl -H 'Accept: application/json; indent=4' http://127.0.0.1:8000/posts/

curl http://127.0.0.1:8000/hello/ -H 'Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ0b2tlbl90eXBlIjoiYWNjZXNzIiwiZXhwIjoxNTQzODI4NDMxLCJqdGkiOiI3ZjU5OTdiNzE1MGQ0NjU3OWRjMmI0OTE2NzA5N2U3YiIsInVzZXJfaWQiOjF9.Ju70kdcaHKn1Qaz8H42zrOYk0Jx9kIckTn9Xx7vhikY'

curl -H 'Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpZCI6MywiZXhwIjoxNjEwNzEzNjExfQ.\_3xzhAw8Nvjz03\_rgHaMSsUTL29jjZMlcNwYiiiJPCk' -H 'Accept: application/json; indent=4' http://127.0.0.1:8000/api/answer/

<https://yandex.ru/turbo/pythonist.ru/s/razrabotka-restful-interfejsov-pri-pomoshhi-python-django-i-django-rest-framework/>

<https://habr.com/ru/post/160117/>

GET — получить информацию об объекте или списке объектов.

POST — создать объект.

PATCH — изменить параметры существующего объекта. Запросы, выполненные с помощью метода PATCH изменяют только те параметры, которые явно указаны в теле запроса.

DELETE — удалить объект.

# Аутентификация пользователя с помощью Django REST Framework и веб-токенов JSON

https://django.fun/tutorials/autentifikaciya-polzovatelya-s-pomoshyu-django-rest-framework-i-veb-tokenov-json/

https://medium.com/@sebastianojeda/user-authentication-with-django-rest-framework-and-json-web-tokens-747ea4d84b9f

https://simpleisbetterthancomplex.com/tutorial/2018/12/19/how-to-use-jwt-authentication-with-django-rest-framework.html

p.s.: Необходимо в serializers.py добавить импорт:

from django.contrib.auth import Authenticate

До выполнения миграции всего проекта сначала выполняем миграции для приложения authentication:

python manage.py makemigrations authentication

А затем выполняем миграцию для проекта:

python manage.py migrate

**Ошибки**

1. django.db.migrations.exceptions.InconsistentMigrationHistory: Migration admin.0001\_initial is applied before its dependency accounts.0001\_initial on database 'default'.

Решение:

Вам необходимо очистить admin, auth, contenttypes и sessions из истории миграции, а также удалить таблицы. Сначала удалите папки миграции ваших приложений, а затем введите следующее:

python manage.py migrate admin zero

python manage.py migrate auth zero

python manage.py migrate contenttypes zero

python manage.py migrate sessions zero

https://stackoverflow.com/questions/42794212/migrating-from-django-user-model-to-a-custom-user-model

https://askdev.ru/q/filtraciya-cherez-viewsets-v-dzhango-ostalnyh-ramok-256618/

**Ссылки:**

1. https://yandex.ru/dev/connect/directory/api/concepts/query-format.html/
2. https://ilyachch.gitbook.io/django-rest-framework-russian-documentation/overview/navigaciya-po-api/viewsets
3. https://yandex.ru/turbo/pythonist.ru/s/razrabotka-restful-interfejsov-pri-pomoshhi-python-django-i-django-rest-framework/
4. https://webdevblog.ru/sozdanie-django-api-ispolzuya-django-rest-framework-apiview/