

**Московский государственный технический  
университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»  
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Разработка Интернет-Приложений»

Отчет по лабораторной работе №6  
«Разработка REST API с использованием Django REST Framework»

Выполнил:  
студент группы ИУ5-53Б  
Ветошкин Артём

Подпись и дата:  
20.12.21

Проверил:  
Юрий Евгеньевич Гапанюк

Подпись и дата:

Москва, 2021 г.

**Цель лабораторной работы:** изучение возможностей разработки REST API с использованием Django REST Framework.

**Задание:**

С использованием Django REST Framework разработайте REST API для одной модели (одной таблицы базы данных).

**Выполнение задания лабораторной работы**

Мы создали модель акции в коде (class Stock), и затем, создали такую таблицу при помощи миграции.

```
from django.db import models

class Stock(models.Model):
    company_name = models.CharField(max_length=50, verbose_name="Название компании")
    price = models.DecimalField(max_digits=8, decimal_places=2, verbose_name="Цена акции")
    is_growing = models.BooleanField(verbose_name="Растет ли акция в цене?")
    date_modified = models.DateTimeField(auto_now=True, verbose_name="Когда последний раз обновлялось значение акции?")
```

Затем был создан сериализатор, чтобы преобразовывать нашу модель из базы данных в JSON

```
class StockSerializer(serializers.ModelSerializer):
    class Meta:
        # Модель, которую мы сериализуем
        model = Stock
        # Поля, которые мы сериализуем
        fields = ["pk", "company_name", "price", "is_growing", "date_modified"]
```

Далее в view создан запрос, который будет посылать запросы в базу данных

```
from django.shortcuts import render

from rest_framework import viewsets
from stocks.serializers import StockSerializer
from stocks.models import Stock

class StockViewSet(viewsets.ModelViewSet):
    """
    API endpoint, который позволяет просматривать и редактировать акции компаний
    """
    # queryset всех пользователей для фильтрации по дате последнего изменения
    queryset = Stock.objects.all().order_by('date_modified')
    serializer_class = StockSerializer # Сериализатор для модели
```

Добавили роутер нашего view в URL'ы приложения

```
from django.contrib import admin
from django.urls import include, path
from stocks import views as stock_views
from rest_framework import routers

router = routers.DefaultRouter()
router.register(r'stocks', stock_views.StockViewSet)

urlpatterns = [
    path('', include(router.urls)),
```

```
path('api-auth/', include('rest_framework.urls',
namespace='rest_framework')),
path('admin/', admin.site.urls),
]
```

## Пример работы программы:

http://127.0.0.1:8000/stocks/


GET ▼ http://127.0.0.1:8000/stocks/

Params Authorization Headers (9) Body ● Pre-request Script Tests Settings

Query Params

KEY	VALUE
Key	Value

Body Cookies Headers (9) Test Results

Pretty Raw Preview Visualize JSON ▼ 

```
1  [
2    {
3      "pk": 2,
4      "company_name": "Mail",
5      "price": "1000.00",
6      "is_growing": true,
7      "date_modified": "2021-12-26T18:57:38.599836Z"
8    },
9    {
10     "pk": 3,
11     "company_name": "Google",
12     "price": "1000.00",
13     "is_growing": false,
14     "date_modified": "2021-12-26T18:58:20.714918Z"
15   }
16 ]
```

http://127.0.0.1:8000/stocks/2/

GET



http://127.0.0.1:8000/stocks/2/

Params

Authorization

Headers (9)

Body ●

Pre-request Script

Tests

Settings

Query Params

	KEY	VALUE
	Key	Value

Body

Cookies

Headers (9)

Test Results

Pretty

Raw

Preview

Visualize

JSON



```
1 {  
2   "pk": 2,  
3   "company_name": "Mail",  
4   "price": "1000.00",  
5   "is_growing": true,  
6   "date_modified": "2021-12-26T18:57:38.599836Z"  
7 }
```

http://127.0.0.1:8000/stocks/3/

DELETE ▼ http://127.0.0.1:8000/stocks/3/

Params Authorization Headers (9) Body ● Pre-request Script Tests Settings

Query Params

KEY	VALUE
Key	Value

Body Cookies Headers (8) Test Results

Pretty Raw Preview Visualize Text ▼

1

http://127.0.0.1:8000/stocks/

POST ▼ http://127.0.0.1:8000/stocks/

Params Authorization Headers (9) Body ● Pre-request Script Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL ☐ JSON ▼

```
1 {
2   "company_name": "Google",
3   "price": "1000.00",
4   "is_growing": false,
5   "date_modified": "26/12/2021"
6 }
```

Body Cookies Headers (9) Test Results ⊕ Status: 201 Created Time

Pretty Raw Preview Visualize JSON ▼

```
1 [
2   {
3     "pk": 4,
4     "company_name": "Google",
5     "price": "1000.00",
6     "is_growing": false,
7     "date_modified": "2022-01-18T20:11:31.724891Z"
8   }
9 ]
```