

**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:
Плотников Ф.С. ИУ5-51

Проверил:
Гапанюк Ю. Е.
Балашов А. М.

Москва, 2021 г.

Цель лабораторной работы:

Изучение возможностей разработки REST API с использованием Django REST Framework.

Задание:

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

Текст программы:

settings.py:

```
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'corsheaders',

    # DRF
    'rest_framework',

    # Наше приложение
    'stocks',
]
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'lr6',
        'USER': 'Юзя',
        'PASSWORD': '',
        'HOST': 'localhost',
        'PORT': 3306,
        'OPTIONS': {'charset': 'utf8'},
        'TEST_CHARSET': 'utf8',
    }
}
```

urls.py:

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

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

# Wire up our API using automatic URL routing.
# Additionally, we include login URLs for the browsable API.
urlpatterns = [
    path("", include(router.urls)),
    path('api-auth/', include('rest_framework.urls', namespace='rest_framework')),
    path('admin/', admin.site.urls),
]
```

models.py:

```
# Create your models here.

from django.db import models

class Stock(models.Model):
    name = models.CharField(max_length=50, verbose_name="Название детали")
    price = models.DecimalField(max_digits=8, decimal_places=2, verbose_name="Цена")
    company = models.CharField(max_length=50, verbose_name="Название компании")

    class Meta:
        managed = False
        db_table = 'details'
```

serializers.py:

```
from stocks.models import Stock
from rest_framework import serializers

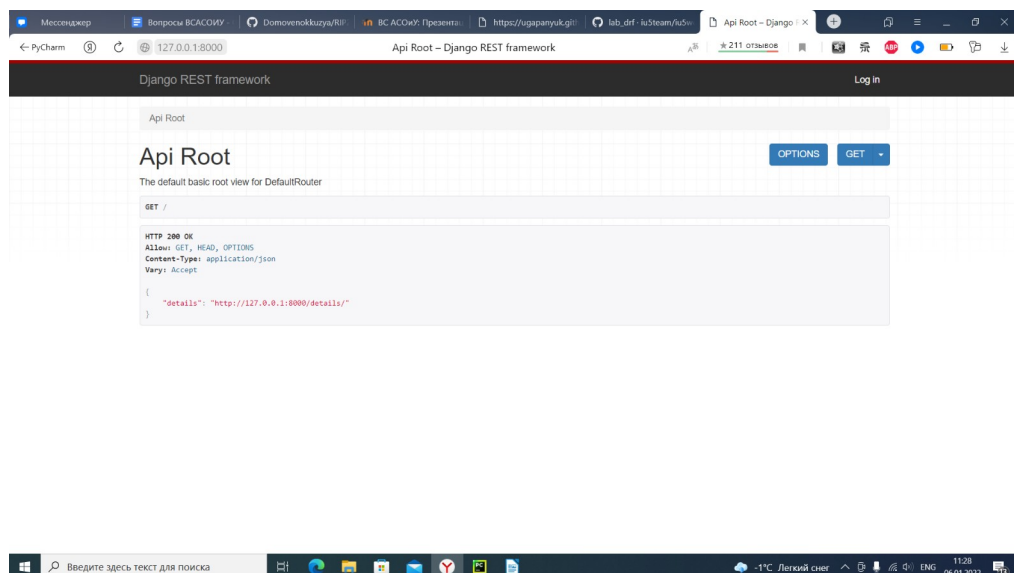
class StockSerializer(serializers.ModelSerializer):
    class Meta:
        # Модель, которую мы сериализуем
        model = Stock
        # Поля, которые мы сериализуем
        fields = ["name", "price", "company"]
```

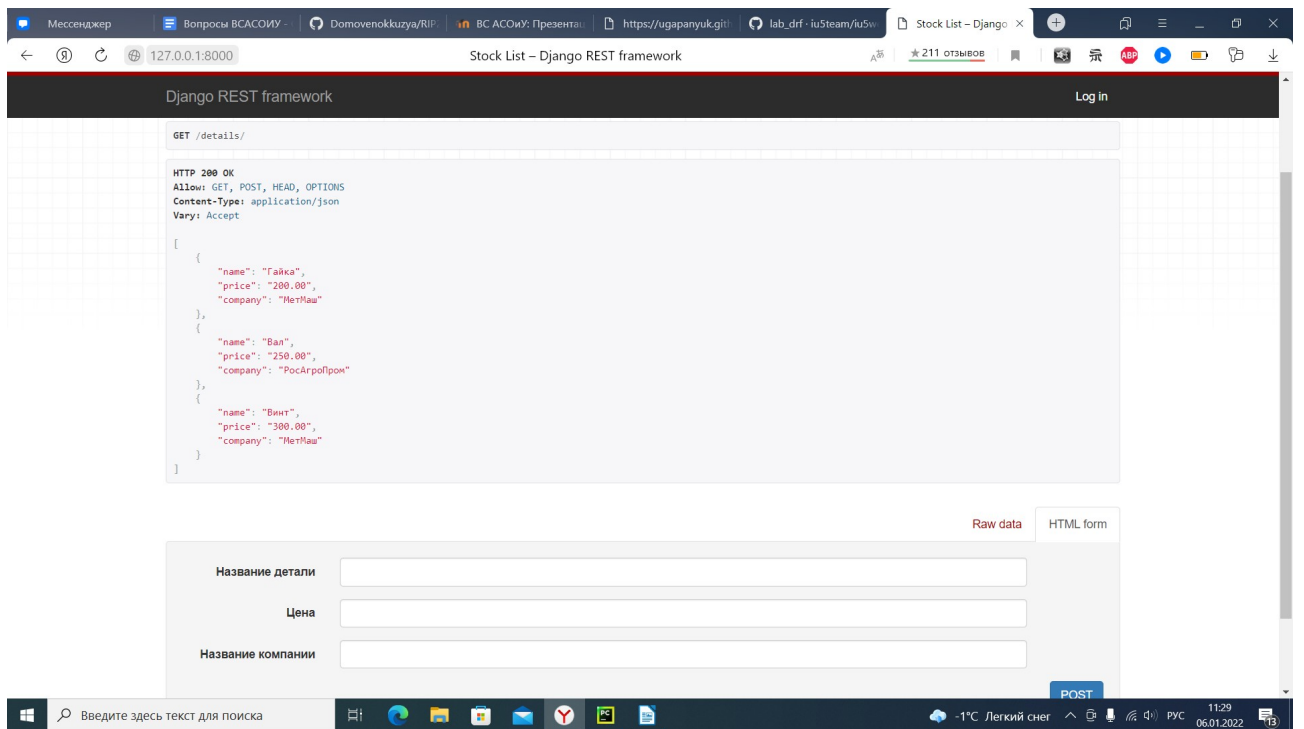
views.py:

```
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('id')
    serializer_class = StockSerializer # Сериализатор для модели
```

Результаты запуска:





Работа в Insomnia, запрос POST и GET:

