**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: |  | Проверил: |
| студент группы ИУ5-53Б |  | Юрий Евгеньевич Гапанюк |
| Ветошкин Артём |  |  |
| Подпись и дата:  20.12.21 |  | 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:**







