



**Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное  
учреждение высшего образования «Московский государственный  
технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)**

**Факультет «Радиотехнический»**

**Кафедра ИУ5 «Системы обработки информации и управления»**

**Рубежный контроль №2**

**по дисциплине «Разработка интернет-приложений»**

**Выполнил:  
студент группы РТ5-51Б  
Н.Н. Васянин**

**Проверил:  
к.т.н., доцент кафедры ИУ5  
Ю.Е. Гапанюк**

**2021 г.**

## Задание:

Рубежный контроль представляет собой разработку веб-приложения с использованием фреймворка Django. Веб-приложение должно выполнять следующие функции:

1. Создайте проект Python Django с использованием стандартных средств Django.
2. Создайте модель Django ORM, содержащую две сущности, связанные отношением один-ко-многим в соответствии с Вашим вариантом из условий рубежного контроля №1.
3. С использованием стандартного механизма Django сгенерируйте по модели макет веб-приложения, позволяющий добавлять, редактировать и удалять данные.
4. Создайте представление и шаблон, формирующий отчет, который содержит соединение данных из двух таблиц.

## Текст программы:

Models.py

```
1  from django.db import models
2
3
4  class Orchestra(models.Model):
5      title = models.CharField(max_length=50, verbose_name="Название оркестра")
6      members = models.CharField(max_length=50, verbose_name="Кол-во участников")
7
8      def __str__(self):
9          return self.title
10
11
12  class MembersOrc(models.Model):
13      title_orc = models.CharField(max_length=50, verbose_name="Оркестр")
14      name = models.CharField(max_length=50, verbose_name="Имя")
15      surname = models.CharField(max_length=50, verbose_name="Фамилия")
16
17      def __str__(self):
18          return self.title_orc
```

## Serializers.py

```
1 from main.models import Orcheatra, MembersOrc
2 from rest_framework import serializers
3
4
5 class OrcheatraSerializer(serializers.ModelSerializer):
6     class Meta:
7
8         model = Orcheatra
9
10        fields = "__all__"
11
12
13 class MembersOrcSerializer(serializers.ModelSerializer):
14     class Meta:
15
16         model = MembersOrc
17
18        fields = "__all__"
```

## Views.py

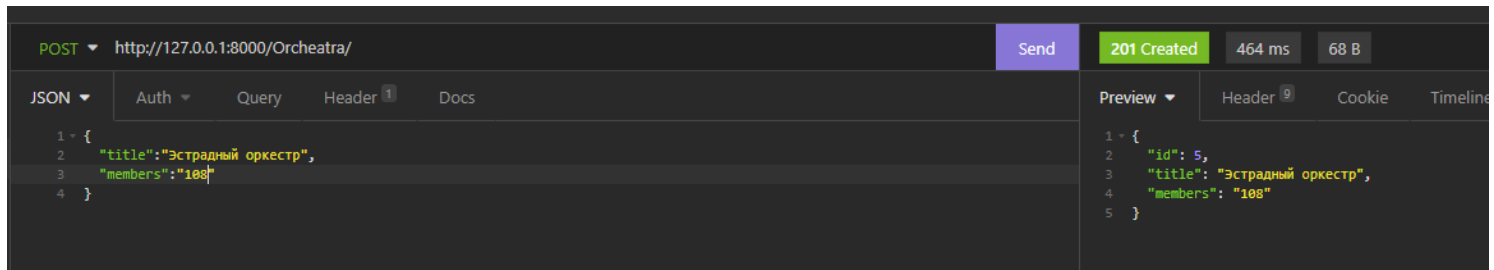
```
1 from django.shortcuts import render
2 from rest_framework import viewsets
3 from main.serializers import OrcheatraSerializer, MembersOrcSerializer
4 from main.models import Orcheatra, MembersOrc
5
6
7 class OrcheatraViewSet(viewsets.ModelViewSet):
8     queryset = Orcheatra.objects.all()
9     serializer_class = OrcheatraSerializer
10
11
12 class MembersOrcViewSet(viewsets.ModelViewSet):
13     queryset = MembersOrc.objects.all()
14     serializer_class = MembersOrcSerializer
15
16
17 def rk2(request):
18     return render(request, 'main/rk2.html', {'Orcheatra': Orcheatra.objects.all(),
19                                             'MembersOrc': MembersOrc.objects.all()
20                                             })
```

## Urls.py

```
16 from django.urls import path, include
17 from rest_framework import routers
18 from django.contrib import admin
19 from main import views
20
21 router = routers.DefaultRouter()
22 router.register('Orchestra', views.OrchestraViewSet)
23 router.register('MembersOrc', views.MembersOrcViewSet)
24
25 urlpatterns = [
26     path('', include(router.urls)),
27     path('api-auth/', include('rest_framework.urls', namespace='rest_framework')),
28     path('rk2/', views.rk2),
29
30     path('admin/', admin.site.urls),
31 ]
```

## Экранные формы с примерами выполнения программы:

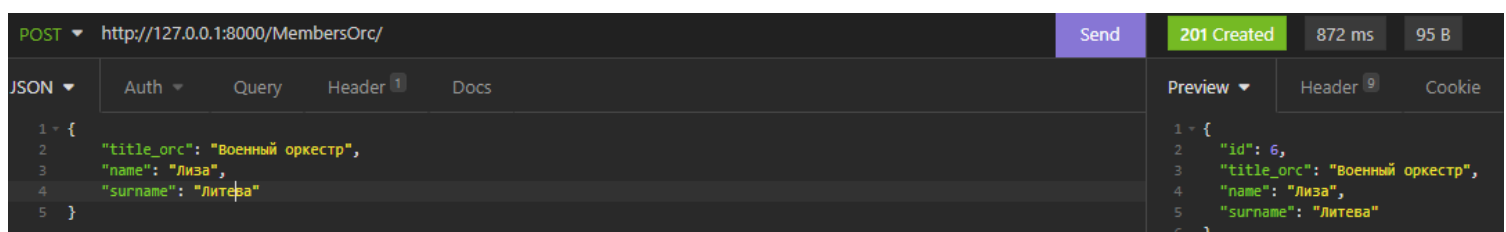
Создание нового оркестра POST:



Запрос списка всех оркестров Get:



Создание нового музыканта POST:



Запрос списка всех оркестров Get:

```
200 OK 7.06 ms 583 B
Preview Header 9 Cookie Timeli
1 [
2 {
3   "id": 1,
4   "title_orc": "Духовой оркестр",
5   "name": "Валера",
6   "surname": "Иванов"
7 },
8 {
9   "id": 2,
10  "title_orc": "Духовой оркестр",
11  "name": "Иван",
12  "surname": "Кипарев"
13 },
14 {
15  "id": 3,
16  "title_orc": "Духовой оркестр",
17  "name": "Иван",
18  "surname": "Патехин"
19 },
20 {
21  "id": 4,
22  "title_orc": "Военный оркестр",
23  "name": "Петя",
24  "surname": "Пякин"
25 },
26 {
27  "id": 5,
28  "title_orc": "Военный оркестр",
29  "name": "Вера",
30  "surname": "Литова"
31 },
32 {
33  "id": 6,
34  "title_orc": "Военный оркестр",
35  "name": "Лиза",
36  "surname": "Литева"
37 }
38 ]
```

Соединение данных из двух таблиц:

**Симфонический оркестр**

Музыкантов: 110

**Духовой оркестр**

Музыкантов: 120

**Военный оркестр**

Музыкантов: 114

**Эстрадный оркестр**

Музыкантов: 108

**Духовой оркестр**

Валера Иванов

Иван Кипарев

Иван Патехин

**Военный оркестр**

Петя Пякин

Вера Литова

Лиза Литева