

Отчет по лабораторной работе № 6 по курсу
"Разработка Интернет-Приложений"

Выполнил:
Студент группы
ИУ5-55Б
Хижняков Вадим Максимович

Москва, МГТУ – 2021

Задание:

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

Текст программы:

models.py

```
from django.db import models

class Manufacturer(models.Model):
    name = models.CharField(max_length=200)

    def __str__(self):
        return self.name

class Car(models.Model):
    name = models.CharField(max_length=200, verbose_name="Name")
    price = models.PositiveIntegerField(verbose_name="Price")
    description = models.TextField(verbose_name="Description")
    main_image = models.ImageField(
        upload_to='car_images/',
        blank=True
    )
    manufacturer = models.ForeignKey(
        'Manufacturer',
        on_delete=models.CASCADE,
    )
    score = models.PositiveSmallIntegerField(verbose_name="Score")

    def __str__(self):
        return self.name
```

serializers.py

```
from django.db import models
from .models import Car, Manufacturer
from rest_framework.serializers import ModelSerializer

class ManufacturerSerializer(ModelSerializer):
    class Meta:
        model = Manufacturer
        fields = ["pk", "name"]
```

```
class CarSerializer(ModelSerializer):
    class Meta:
        model = Car
        fields = ["pk", "name", "price", "description",
                  "main_image", "manufacturer", "score"]
```

views.py

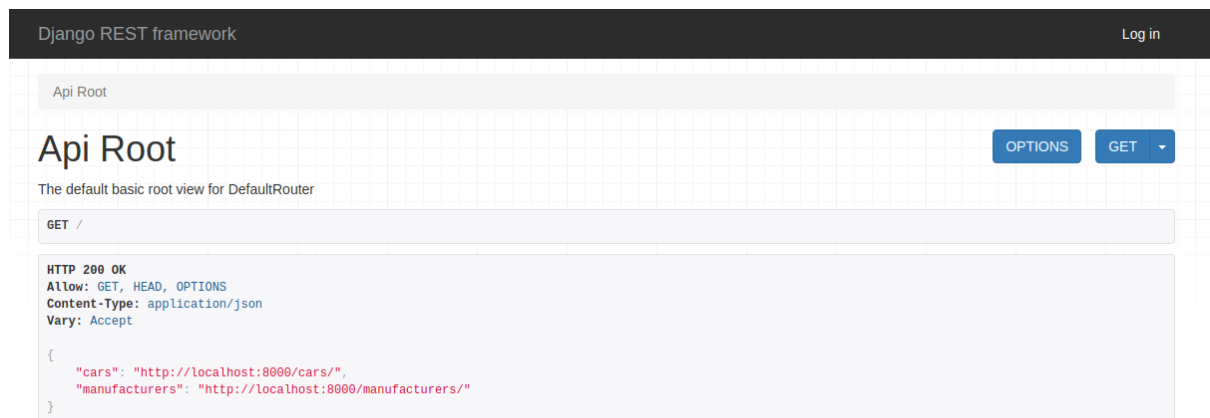
```
from rest_framework.viewsets import ModelViewSet
from .serializers import CarSerializer, ManufacturerSerializer
from .models import Car, Manufacturer

class CarsViewSet(ModelViewSet):
    queryset = Car.objects.all()
    serializer_class = CarSerializer

class ManufacturerViewSet(ModelViewSet):
    queryset = Manufacturer.objects.all()
    serializer_class = ManufacturerSerializer
```

Результат работы программы:

<http://localhost:8000/>



Далее приведены скриншоты работы с апи через приложение Postman

rip / New Request

GET http://localhost:8000/cars/ Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (9) Test Results

Status: 200 OK Time: 40 ms Size: 586 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "pk": 1,
3   "name": "zhiga",
4   "price": 123123,
5   "description": "sdfsdfsdfsdf",
6   "main_image": "http://localhost:8000/media/car_images/zhiga_mqlv5Kj.jpg",
7   "manufacturer": 1,
8   "score": 123
9 }
10
11 {
12   "pk": 9,
13   "name": "asdasd",
14   "price": 99999,
15   "description": "asdasdasdasdasd",
16   "main_image": null,
17   "manufacturer": 2,
18   "score": 4564
19 }
20
```

rip / New Request

GET http://localhost:8000/cars/ Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (9) Test Results

Status: 200 OK Time: 24 ms Size: 479 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "pk": 1,
3   "name": "zhiga",
4   "price": 123123,
5   "description": "sdfsdfsdfsdf",
6   "main_image": "http://localhost:8000/media/car_images/zhiga_mqlv5Kj.jpg",
7   "manufacturer": 1,
8   "score": 123
9 }
```

rip / New Request

POST http://localhost:8000/cars/ Send

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

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

```
1 {
2   "name": "test",
3   "price": 456,
4   "description": "test test test",
5   "manufacturer": 1,
6   "score": 10
7 }
```

Body Cookies Headers (9) Test Results Status: 201 Created Time: 28 ms Size: 414 B Save Response

Pretty Raw Preview Visualize **JSON**

```
1 {
2   "pk": 10,
3   "name": "test",
4   "price": 456,
5   "description": "test test test",
6   "main_image": null,
7   "manufacturer": 1,
8   "score": 10
9 }
```

rip / New Request

PATCH http://localhost:8000/cars/10/ Send

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

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

```
1 {
2   "name": "test",
3   "price": 456,
4   "description": "updated value",
5   "manufacturer": 1,
6   "score": 10
7 }
```

Body Cookies Headers (9) Test Results Status: 200 OK Time: 26 ms Size: 422 B Save Response

Pretty Raw Preview Visualize **JSON**

```
1 {
2   "pk": 10,
3   "name": "test",
4   "price": 456,
5   "description": "updated value",
6   "main_image": null,
7   "manufacturer": 1,
8   "score": 10
9 }
```

rip / New Request

DELETE http://localhost:8000/cars/10/ Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "name": "test",
3   "price": 456,
4   "description": "updated value",
5   "manufacturer": 1,
6   "score": 10
7 }
```

Body Cookies Headers (9) Test Results

Status: 200 OK Time: 22 ms Size: 586 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   {
3     "pk": 1,
4     "name": "zhiga",
5     "price": 123123,
6     "description": "sdifsdfsdf",
7     "main_image": "http://localhost:8000/media/car_images/zhiga_mqlvSKl.jpg",
8     "manufacturer": 1,
9     "score": 123
10  },
11  {
12    "pk": 9,
13    "name": "asdasd",
14    "price": 99999,
15    "description": "asdasdasdasd",
16    "main_image": null,
17    "manufacturer": 2,
18    "score": 4564
19  }
20 }
```