

Защищено:
Гапанюк Ю.Е.

Демонстрация:
Гапанюк Ю.Е.

"__" _____ 2021 г.

"__" _____ 2021 г.

**Отчет по лабораторной работе № 6 по курсу
Разработка интернет-приложений
ГУИМЦ**

**Тема работы: " Разработка REST API с использованием Django
REST Framework "**

12
(количество листов)

ИСПОЛНИТЕЛЬ:

студент группы ИУ5Ц-74Б

Шанаурина Е.Г.

(подпись)

"__" _____ 2021 г.

СОДЕРЖАНИЕ

1. Описание задания.....	3
2. Текст программы.....	3
3. Результаты выполнения программы.....	8

1. Описание задания

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

2. Текст программы

animals/admin.py

```
from django.contrib import admin
from .models import Animal

# Register your models here.
admin.site.register(Animal)
```

animals/apps.py

```
from django.apps import AppConfig

class AnimalsConfig(AppConfig):
    default_auto_field = 'django.db.models.BigAutoField'
    name = 'animals'
```

animals/models.py

```
from django.db import models

# Create your models here.
class Animal(models.Model):
    animal_type = models.CharField('Вид животного', max_length=50)
    animal_name = models.CharField('Кличка животного', max_length=50)
    animal_photo = models.URLField('Фото животного', max_length=200)

    def __str__(self):
        return f'Тип: {self.animal_type}; Имя: {self.animal_name}; URL фото: {self.animal_photo}'

    class Meta:
        verbose_name = 'Животное'
        verbose_name_plural = 'Животные'
```

animals/tests.py

```
from django.test import TestCase

# Create your tests here.
```

animals/views.py

```
from django.shortcuts import render
from rest_framework import viewsets
from .serializers import AnimalSerializer
from .models import Animal

# Create your views here.
```

```
class AnimalViewSet(viewsets.ModelViewSet):
    """
    API endpoint, который позволяет просматривать и редактировать акции
    компаний
    """
    # queryset всех пользователей для фильтрации по дате последнего изменения
    queryset = Animal.objects.all()
    serializer_class = AnimalSerializer # Сериализатор для модели
```

animals/serializers.py

```
from rest_framework import serializers
from .models import Animal

class AnimalSerializer(serializers.ModelSerializer):
    class Meta:
        # Модель, которую мы сериализуем
        model = Animal
        # Поля, которые мы сериализуем
        fields = ["animal_type", "animal_name", "animal_photo"]
```

LAB_6/settings.py

```
"""
Django settings for LAB_6 project.

Generated by 'django-admin startproject' using Django 3.2.9.

For more information on this file, see
https://docs.djangoproject.com/en/3.2/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/3.2/ref/settings/
"""

from pathlib import Path

# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/3.2/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'django-insecure-m#mp=%7jeczg4o83^1(n%qba5ilwk!_alyu(^grl(pmba-
%t+g'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []

# Application definition
```

```

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'rest_framework',
    'animals',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'LAB_6.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [BASE_DIR / 'templates']
    },
    {
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]

WSGI_APPLICATION = 'LAB_6.wsgi.application'

# Database
# https://docs.djangoproject.com/en/3.2/ref/settings/#databases

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': BASE_DIR / 'db.sqlite3',
    }
}

# Password validation
# https://docs.djangoproject.com/en/3.2/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {

```

```

        'NAME':
'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME':
'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME':
'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME':
'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]

```

```

# Internationalization
# https://docs.djangoproject.com/en/3.2/topics/i18n/

```

```
LANGUAGE_CODE = 'ru-ru'
```

```
TIME_ZONE = 'UTC'
```

```
USE_I18N = True
```

```
USE_L10N = True
```

```
USE_TZ = True
```

```

# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/3.2/howto/static-files/

```

```
STATIC_URL = '/static/'
```

```

# Default primary key field type
# https://docs.djangoproject.com/en/3.2/ref/settings/#default-auto-field

```

```
DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'
```

LAB_6/urls.py

```
"""LAB_6 URL Configuration
```

```

The `urlpatterns` list routes URLs to views. For more information please see:
https://docs.djangoproject.com/en/3.2/topics/http/urls/

```

```
Examples:
```

```
Function views
```

1. Add an import: `from my_app import views`
2. Add a URL to urlpatterns: `path('', views.home, name='home')`

```
Class-based views
```

1. Add an import: `from other_app.views import Home`
2. Add a URL to urlpatterns: `path('', Home.as_view(), name='home')`

```
Including another URLconf
```

1. Import the `include()` function: `from django.urls import include, path`
2. Add a URL to urlpatterns: `path('blog/', include('blog.urls'))`

```

"""
from django.contrib import admin
from django.urls import path, include
from rest_framework import routers
from animals import views as animal_views

router = routers.DefaultRouter()
router.register(r'animals', animal_views.AnimalViewSet)

# 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),
]

```

LAB_6/wsgi.py

```

"""
WSGI config for LAB_6 project.

It exposes the WSGI callable as a module-level variable named ``application``.

For more information on this file, see
https://docs.djangoproject.com/en/3.2/howto/deployment/wsgi/
"""

import os

from django.core.wsgi import get_wsgi_application

os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'LAB_6.settings')

application = get_wsgi_application()

```


3. Результаты выполнения программы

Лог

```
"C:\Program Files\JetBrains\PyCharm 2021.2\bin\runnerw.exe" "A:\Liza
\YandexDisk\МГТУ\7-й семестр\РИП\ЛР\Py\LAB_6\venv\Scripts\python.exe"
"A:/Liza/YandexDisk/МГТУ/7-й семестр/РИП/ЛР/Py/LAB_6/manage.py"
runserver 8000
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
November 27, 2021 - 21:19:59
Django version 3.2.9, using settings 'LAB_6.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
[27/Nov/2021 21:20:02] "GET / HTTP/1.1" 200 5546
[27/Nov/2021 21:20:02] "GET /static/rest_framework/js/jquery-
3.5.1.min.js HTTP/1.1" 304 0
[27/Nov/2021 21:20:02] "GET /static/rest_framework/js/ajax-form.js
HTTP/1.1" 304 0
[27/Nov/2021 21:20:02] "GET /static/rest_framework/js/csrf.js HTTP/1.1"
304 0
[27/Nov/2021 21:20:02] "GET /static/rest_framework/js/bootstrap.min.js
HTTP/1.1" 304 0
[27/Nov/2021 21:20:02] "GET /static/rest_framework/js/prettify-min.js
HTTP/1.1" 304 0
[27/Nov/2021 21:20:02] "GET /static/rest_framework/js/default.js
HTTP/1.1" 304 0
[27/Nov/2021 21:20:03] "GET /static/rest_framework/css/prettify.css
HTTP/1.1" 200 817
[27/Nov/2021 21:20:03] "GET /static/rest_framework/css/bootstrap-
tweaks.css HTTP/1.1" 200 3385
[27/Nov/2021 21:20:03] "GET /static/rest_framework/css/default.css
HTTP/1.1" 200 1152
[27/Nov/2021 21:20:03] "GET /static/rest_framework/css/bootstrap.min.css
HTTP/1.1" 200 121457
.....
```

Главная страница (API root)

 127.0.0.1:8000 

Django REST framework admin ▾

Api Root

Api Root OPTIONS GET ▾

The default basic root view for DefaultRouter

GET /

HTTP 200 OK
Allow: GET, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

```
{
  "animals": "http://127.0.0.1:8000/animals/"
}
```


API животных (/animals)

127.0.0.1:8000/animals/



Django REST framework

admin ▾

Api Root / Animal List

Animal List

OPTIONS

GET ▾

API endpoint, который позволяет просматривать и редактировать акции компаний

GET /animals/

```
HTTP 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

[
  {
    "animal_type": "Собака",
    "animal_name": "Альбус",
    "animal_photo": "http://pc.sccraft.ru/SharedFiles/LAB_6/albus.jpg"
  },
  {
    "animal_type": "Кошка",
    "animal_name": "Барса",
    "animal_photo": "http://pc.sccraft.ru/SharedFiles/LAB_6/barsa.jpg"
  },
  {
    "animal_type": "Кошка",
    "animal_name": "Стеша",
    "animal_photo": "http://pc.sccraft.ru/SharedFiles/LAB_6/stesha.jpg"
  },
  {
    "animal_type": "Тест",
    "animal_name": "Тест",
    "animal_photo": "http://localhost/iisstart.png"
  }
]
```

Raw data

HTML form

Вид животного

Кличка животного

Фото животного

POST

Администрирование (/admin)

← → ↻ 🏠 127.0.0.1:8000/admin/login/?next=/admin/

Администрирование Django

Имя пользователя:

Пароль:

Войти

← → ↻ 🏠 127.0.0.1:8000/admin/ 📄 ☆ ⬇️ ||| 📄 👤 🔒 🔔 >> ☰

Администрирование Django

ДОБРО ПОЖАЛОВАТЬ, ADMIN. [ОТКРЫТЬ САЙТ](#) / [ИЗМЕНИТЬ ПАРОЛЬ](#) / [ВЫЙТИ](#)

Администрирование сайта

ANIMALS		
Животные	+ Добавить	✎ Изменить
ПОЛЬЗОВАТЕЛИ И ГРУППЫ		
Группы	+ Добавить	✎ Изменить
Пользователи	+ Добавить	✎ Изменить

Последние действия

Мои действия

- ✗ Тип: Test; Имя: Test; URL фото: <http://localhost/iisstart.png>
Животное
- ✗ Тип: Test; Имя: Test; URL фото: <http://localhost/iisstart.png>
Животное
- ✎ Тип: Кошка; Имя: Барса; URL фото: http://pc.sccraft.ru/SharedFiles/LAB_6/barsa.jpg
Животное
- + Тип: Кошка; Имя: Стеша; URL фото: http://pc.sccraft.ru/SharedFiles/LAB_6/stesha.jpg
Животное
- + Тип: Кошка; Имя: Барса; URL фото: https://pc.sccraft.ru/SharedFiles/LAB_6/barsa.jpg
Животное
- + Тип: Собака; Имя: Альбус; URL фото: http://pc.sccraft.ru/SharedFiles/LAB_6/albus.jpg
Животное

← → ↻ 🏠 127.0.0.1:8000/admin/animals/animal/ 📄 ☆ ⬇️ 📁 📄 👤 🔒 🔥 🔥 >> ☰

Администрирование Django

ДОБРО ПОЖАЛОВАТЬ, ADMIN. [ОТКРЫТЬ САЙТ](#) / [ИЗМЕНИТЬ ПАРОЛЬ](#) / [ВЫЙТИ](#)

Начало · Animals · Животные

ANIMALS

Животные + Добавить

ПОЛЬЗОВАТЕЛИ И ГРУППЫ

Группы + Добавить

Пользователи + Добавить

Выберите Животное для изменения

Действие: Выбрано 0 объектов из 3

- ☐ ЖИВОТНОЕ
- ☐ Тип: Кошка; Имя: Стеша; URL фото: http://pc.sccraft.ru/SharedFiles/LAB_6/stesha.jpg
- ☐ Тип: Кошка; Имя: Барса; URL фото: http://pc.sccraft.ru/SharedFiles/LAB_6/barsa.jpg
- ☐ Тип: Собака; Имя: Альбус; URL фото: http://pc.sccraft.ru/SharedFiles/LAB_6/albus.jpg

3 Животные

← → ↻ 🏠 127.0.0.1:8000/admin/animals/animal/2/change/ 📄 ☆ ⬇️ 📁 📄 👤 🔒 🔥 🔥 >> ☰

Администрирование Django

ДОБРО ПОЖАЛОВАТЬ, ADMIN. [ОТКРЫТЬ САЙТ](#) / [ИЗМЕНИТЬ ПАРОЛЬ](#) / [ВЫЙТИ](#)

Начало · Animals · Животные · Тип: Кошка; Имя: Барса; URL фото: http://pc.sccraft.ru/SharedFiles/LAB_6/barsa.jpg

ANIMALS

Животные + Добавить

ПОЛЬЗОВАТЕЛИ И ГРУППЫ

Группы + Добавить

Пользователи + Добавить

Изменить Животное

Тип: Кошка; Имя: Барса; URL фото: http://pc.sccraft.ru/SharedFiles/LAB_6/barsa.jpg

Вид животного:

Кличка животного:

Фото животного: Сейчас: http://pc.sccraft.ru/SharedFiles/LAB_6/barsa.jpg
Изменить:

Обращение к API из Insomnia

GET http://127.0.0.1:8000/animals/ Send 200 OK 11 ms 372 B

Body Auth Query 1 Header Docs

```

1 {
2   {
3     "animal_type": "Собака",
4     "animal_name": "Альбус",
5     "animal_photo": "http://pc.sccraft.ru/SharedFiles/LAB_6/albus.jpg"
6   },
7   {
8     "animal_type": "Кошка",
9     "animal_name": "Барса",
10    "animal_photo": "http://pc.sccraft.ru/SharedFiles/LAB_6/barsa.jpg"
11  },
12  {
13    "animal_type": "Кошка",
14    "animal_name": "Стеша",
15    "animal_photo": "http://pc.sccraft.ru/SharedFiles/LAB_6/stesha.jpg"
16  }
17 }

```

POST http://127.0.0.1:8000/animals/ Send 201 Created 113 ms 90 B

JSON Auth Query 1 Header 1 Docs

```

1 {
2   "animal_type": "Test",
3   "animal_name": "Test",
4   "animal_photo": "http://localhost/iisstart.png"
5 }

```

Preview Header 9 Cookie Timeline

```

1 {
2   "animal_type": "Test",
3   "animal_name": "Test",
4   "animal_photo": "http://localhost/iisstart.png"
5 }

```

GET http://127.0.0.1:8000/animals/ Send 200 OK 8.45 ms 463 B

Body Auth Query 1 Header Docs

Preview Header 9 Cookie Timeline

```
1 [
2 {
3   "animal_type": "Собака",
4   "animal_name": "Альбис",
5   "animal_photo": "http://pc.sccraft.ru/SharedFiles/LAB_6/albus.jpg"
6 },
7 {
8   "animal_type": "Ковка",
9   "animal_name": "Барса",
10  "animal_photo": "http://pc.sccraft.ru/SharedFiles/LAB_6/barca.jpg"
11 },
12 {
13   "animal_type": "Ковка",
14   "animal_name": "Стеша",
15   "animal_photo": "http://pc.sccraft.ru/SharedFiles/LAB_6/stesha.jpg"
16 },
17 {
18   "animal_type": "Test",
19   "animal_name": "Test",
20   "animal_photo": "http://localhost/iisstart.png"
21 }
22 ]
```