

Защищено:
Гапанюк Ю.Е.

Демонстрация:
Гапанюк Ю.Е.

"__" _____ 2021 г.

"__" _____ 2021 г.

**Отчет по лабораторной работе № 5 по курсу
Разработка интернет-приложений
ГУИМЦ**

**Тема работы: " Работа с СУБД. Обработка данных с
использованием Django ORM. "**

17

(количество листов)

ИСПОЛНИТЕЛЬ:

студент группы ИУ5Ц-74Б

Карпов И.Е.

(подпись)

"__" _____ 2021 г.

СОДЕРЖАНИЕ

1. Описание задания	3
2. Текст программы	3
3. Результаты выполнения программы	13

1. Описание задания

В этой лабораторной работе Вы познакомитесь с популярной СУБД MySQL, создадите свою базу данных. Также Вам нужно будет дополнить свои классы предметной области, связав их с созданной БД. После этого Вы создадите свои модели с помощью Django ORM, отобразите объекты из БД с помощью этих моделей.

1. Создайте сценарий с подключением к БД и несколькими запросами, примеры рассмотрены в [методических указаниях](#).
2. Реализуйте модели Вашей предметной области из предыдущей ЛР (минимум две модели, т.е. две таблицы).
3. Создайте представления и шаблоны Django для отображения списка данных по каждой из сущностей.

2. Текст программы

LAB_5/asgi.py

```
"""
ASGI config for LAB_5 project.

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

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

import os

from django.core.asgi import get_asgi_application

os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'LAB_5.settings')

application = get_asgi_application()
```

LAB_5/settings.py

```
"""
Django settings for LAB_5 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
import os

# 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-u3dv^@aurnum@$_-i43e$5ui$4dlmy9v^ff)+-  
z+y&l!l9(=%x4'
```

```
# 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',  
    'App.apps.AppConfig',  
]
```

```
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_5.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_5.wsgi.application'
```

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

```

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'rip_lab_4',
        'USER': 'admin',
        'PASSWORD': '1234567890',
        'HOST': 'SRV2022',
        'PORT': 3306, # Стандартный порт MySQL
        'OPTIONS': {'charset': 'utf8'},
        'TEST_CHARSET': 'utf8',
    }
}

# 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/'

STATICFILES_DIRS = [
    BASE_DIR / "static",
    os.path.join(BASE_DIR, "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_5/urls.py

```

"""LAB_5 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

urlpatterns = [
    path('', include('App.urls')),
    path('admin/', admin.site.urls)
]

```

LAB_5/wsgi.py

```

"""
WSGI config for LAB_5 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_5.settings')

application = get_wsgi_application()

```

App/admin.py

```

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

```

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

App/apps.py

```
from django.apps import AppConfig

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

App/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'

class Animal_review(models.Model):
    animal = models.ForeignKey(Animal, on_delete=models.CASCADE)
    author = models.CharField('Имя автора', max_length=50)
    review_text = models.CharField('Текст отзыва', max_length=200)

    def __str__(self):
        return f'Автор: {self.author}, Отзыв: {self.review_text}'

    class Meta:
        verbose_name = 'Отзыв'
        verbose_name_plural = 'Отзывы'
```

App/tests.py

```
from django.test import TestCase

# Create your tests here.
```

App/urls.py

```
from django.urls import path
from . import views
```

```

app_name = 'App'
urlpatterns = [
    path('', views.index, name='index'),
    path('<int:animal_id>', views.detail, name='detail'),
    path('<int:animal_id>/leave_review', views.leave_review,
name='leave_review')
]

```

App/views.py

```

from django.http import Http404, HttpResponseRedirect
from django.shortcuts import render
from django.urls import reverse
from .models import Animal

def index(request):
    a_list = Animal.objects.all()
    return render(request, 'App/list.html', dict([('a_list', a_list)]))

def detail(request, animal_id):
    try:
        a = Animal.objects.get(id=animal_id)
    except:
        raise Http404('Животное не найдено!')
    reviews = a.animal_review_set.order_by('-id')
    return render(request, 'App/detail.html', dict([('animal', a),
('review_list', reviews)]))

def leave_review(request, animal_id):
    try:
        a = Animal.objects.get(id=animal_id)
    except:
        raise Http404('Животное не найдено!')
    a.animal_review_set.create(author=request.POST['name'],
review_text=request.POST['text'])
    return HttpResponseRedirect(reverse('App:detail', args=(a.id,)))

```


static/albus.jpg



static/barsa.jpg



static/stesha.jpg



templates/base.html

```
<!DOCTYPE html>
<html lang="ru">
<head>
    <meta charset="UTF-8">
    <title>{% block title %} Лабораторная работа № 4{% endblock %}</title>
</head>
<body>

{% block content %}{% endblock %}

<hr>
<center>Карпов И.Е. ИУ5Ц-74Б</center>

</body>
</html>
```

templates/App/list.html

```
{% extends 'base.html' %}

{% block title %}ЛР №5{% endblock %}

{% block content %}

    <center><h1>СПИСОК ЖИВОТНЫХ</h1></center><hr><br>

    {% if a_list %}
        {% for a in a_list %}
            <a href="{% url 'App:detail' a.id %}">{{ a.animal_name }}</a><br>
        {% endfor %}
    {% else %}
        Список пуст
    {% endif %}


```

```
{% endblock %}
```

templates/App/detail.html

```
{% extends 'base.html' %}
```

```
{% block title %}{{ animal.animal_name }}{% endblock %}
```

```
{% block content %}
```

```
    <center><h1>{{ animal.animal_name }}</h1></center><hr><br>
    {# Вид животного: {{ animal.animal_type }}<br>#}
    {# Имя животного: {{ animal.animal_name }}<br>#}
    {# <br>#}
    {# Фото животного<br>#}
    {# #}
    <table border="1" cellpadding="5">
    <tr>
        <th>Вид животного</th>
        <th>{{ animal.animal_type }}</th>
    </tr>
    <tr>
        <th>Имя животного</th>
        <th>{{ animal.animal_name }}</th>
    </tr>
    <tr>
        <th>Фото животного</th>
        <th></th>
    </tr>
    </table>
    <br>
    <form action="{% url 'App:leave_review' animal.id %}" method="post">
    {% csrf_token %}
    <input type="text" required placeholder="Имя" name="name"><br>
    <textarea name="text" required="" placeholder="Текст отзыва" cols="30"
rows="10"></textarea>
    <button type="submit">Отправить отзыв</button>

    </form><hr>

    {% if review_list %}
        <table border="1" cellpadding="5">
        <tr>
            <th>Автор</th>
            <th>Отзыв</th>
        </tr>
        {% for r in review_list %}
            <tr>
                <th>{{ r.author }}</th>
                <th>{{ r.review_text }}</th>
            </tr>
        {% endfor %}
        </table>
    {% else %}
    Отзывы не найдены! Оставьте первый!
    {% endif %}
```

```
<hr><br>
<a href="/">Назад</a>

{% endblock %}
```

3. Результаты выполнения программы

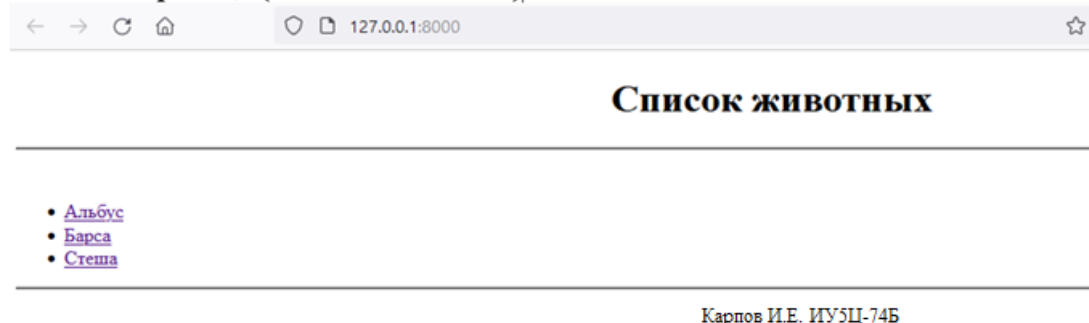
Лог

```
"C:\Program Files\JetBrains\PyCharm 2021.2\bin\runnerw.exe" "A:\Igor
\YandexDisk\МГТУ\7-й семестр\РИП\ЛР\Py\LAB_5\Scripts\python.exe"
"A:/Игорь/YandexDisk/МГТУ/7-й семестр/РИП/ЛР/Py/LAB_5/manage.py"
runserver 8000
Performing system checks...
```

```
Watching for file changes with StatReloader
System check identified no issues (0 silenced).
November 18, 2021 - 11:47:43
Django version 3.2.9, using settings 'LAB_5.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
[18/Nov/2021 11:47:47] "GET / HTTP/1.1" 200 485
Not Found: /favicon.ico
[18/Nov/2021 11:47:47] "GET /favicon.ico HTTP/1.1" 404 2655
[18/Nov/2021 11:49:30] "GET /2 HTTP/1.1" 200 2167
[18/Nov/2021 11:49:30,347] - Broken pipe from ('127.0.0.1', 56007)

[18/Nov/2021 11:51:47] "GET /admin/ HTTP/1.1" 302 0
[18/Nov/2021 11:51:47] "GET /admin/login/?next=/admin/ HTTP/1.1" 200
2300
[18/Nov/2021 11:51:47] "GET /static/admin/css/login.css HTTP/1.1"
200 939
[18/Nov/2021 11:51:47] "GET /static/admin/css/nav_sidebar.css HTTP/1.1"
200 2271
[18/Nov/2021 11:51:47] "GET /static/admin/css/base.css HTTP/1.1" 200
19513
[18/Nov/2021 11:51:47] "GET /static/admin/css/responsive.css HTTP/1.1"
200 18545
[18/Nov/2021 11:51:47] "GET /static/admin/js/nav_sidebar.js HTTP/1.1"
200 1360
[18/Nov/2021 11:51:47] "GET /static/admin/css/fonts.css HTTP/1.1" 200
423
```

Главная страница (Список животных)



Информация о животном (/<int>)

Барса

Вид животного	Кошка
Имя животного	Барса
Фото животного	

Имя
Текст отзыва
<input type="button" value="Отправить отзыв"/>

Автор	Отзыв
Стеша	К сожалению я появилась позже, чем Барса умерла. Мне было бы интересно пережить лидерство.
Александр	Эта кошка известна всей деревне. Она побывала почти на всех участках.
Аслан	Этот белый монстр меня пугает! У меня осыпается шерсть. Хуже этого монстра только дети, от которых приходится прятаться под потолком.

[Назад](#)

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

Администрирование сайта

APP		
Животные	+ Добавить	✎ Изменить
Отзывы	+ Добавить	✎ Изменить
ПОЛЬЗОВАТЕЛИ И ГРУППЫ		
Группы	+ Добавить	✎ Изменить
Пользователи	+ Добавить	✎ Изменить

Последние действия

Мои действия

- ✎

Тип: Собака; Имя: Альбус; URL фото: <http://127.0.0.1:8000/static/albus.jpg>
Животное
- +

Тип: Кошка; Имя: Стеша; URL фото: <http://127.0.0.1:8000/static/stesha.jpg>
Животное
- +

Тип: Кошка; Имя: Барса; URL фото: <http://127.0.0.1:8000/static/barsa.jpg>
Животное
- ✎

Автор: Александр, Отзыв: Большая собака. У неё есть 4 лапы, голова и хвост.
Отзыв
- +

Автор: Александр, Отзыв: Большая собака.
Отзыв
- +

Тип: Собака; Имя: Альюус; URL фото: <http://127.0.0.1:8000/static/albus.jpg>
Животное

← → ↺ 🏠

🔒 127.0.0.1:8000/admin/App/animal/

📄 ☆

☰ 📄 🔍 ⌂ >> ☰

Администрирование Django

ДОБРО ПОЖАЛОВАТЬ, ADMIN. ОТКРЫТЬ САЙТ / ИЗМЕНИТЬ ПАРОЛЬ / ВЫЙТИ

Начало > App > Животные

APP

Животные + Добавить

Отзывы + Добавить

ПОЛЬЗОВАТЕЛИ И ГРУППЫ

Группы + Добавить

Пользователи + Добавить

Выберите Животное для изменения

ДОБАВИТЬ ЖИВОТНОЕ +

Действие:

⌵

 Выполнить

Выбрано 0 объектов из 3

☐ ЖИВОТНОЕ

☐ Тип: Кошка; Имя: Стеша; URL фото: http://127.0.0.1:8000/static/stesha.jpg

☐ Тип: Кошка; Имя: Барса; URL фото: http://127.0.0.1:8000/static/barsa.jpg

☐ Тип: Собака; Имя: Альбус; URL фото: http://127.0.0.1:8000/static/albus.jpg

3 Животные

← → ↺ 🏠

🔒 127.0.0.1:8000/admin/master/animal/2/change/

📄 ☆

📄 ☰ 🔍 ⌂ >> ☰

Администрирование Django

ДОБРО ПОЖАЛОВАТЬ, ADMIN. ОТКРЫТЬ САЙТ / ИЗМЕНИТЬ ПАРОЛЬ / ВЫЙТИ

Начало > ЛР-4 > Животные > Тип: Кошка; Имя: Барса; URL фото: http://127.0.0.1:8000/static/barsa.jpg

ЛР-4

Животные + Добавить

ПОЛЬЗОВАТЕЛИ И ГРУППЫ

Группы + Добавить

Пользователи + Добавить

Изменить Животное

ИСТОРИЯ

Тип: Кошка; Имя: Барса; URL фото: http://127.0.0.1:8000/static/barsa.jpg

Вид животного:

Кошка

Кличка животного:

Барса

Фото животного:

Сейчас: http://127.0.0.1:8000/static/barsa.jpg

Изменить:

http://127.0.0.1:8000/static/barsa.jpg

Удалить

Сохранить и добавить другой объект

Сохранить и продолжить редактирование

СОХРАНИТЬ

← → ↺ 🏠

🔒 127.0.0.1:8000/admin/App/animal/2/change/

📄 ☆

📄 ☰ 🔍 ⌂ >> ☰

Администрирование Django

ДОБРО ПОЖАЛОВАТЬ, ADMIN. ОТКРЫТЬ САЙТ / ИЗМЕНИТЬ ПАРОЛЬ / ВЫЙТИ

Начало > App > Животные > Тип: Кошка; Имя: Барса; URL фото: http://127.0.0.1:8000/static/barsa.jpg

APP

Животные + Добавить

Отзывы + Добавить

ПОЛЬЗОВАТЕЛИ И ГРУППЫ

Группы + Добавить

Пользователи + Добавить

Изменить Животное

ИСТОРИЯ

Тип: Кошка; Имя: Барса; URL фото: http://127.0.0.1:8000/static/barsa.jpg

Вид животного:

Кошка

Кличка животного:

Барса

Фото животного:

Сейчас: http://127.0.0.1:8000/static/barsa.jpg

Изменить:

http://127.0.0.1:8000/static/barsa.jpg

Удалить

Сохранить и добавить другой объект

Сохранить и продолжить редактирование

СОХРАНИТЬ

← → ↺ 🏠

🔒 127.0.0.1:8000/admin/App/animal_review/

📄 ☆

☰ 📄 👤 >> ≡

Администрирование Django

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

Начало · App · Отзывы

APP

Животные + Добавить

Отзывы + Добавить

ПОЛЬЗОВАТЕЛИ И ГРУППЫ

Группы + Добавить

Пользователи + Добавить

Выберите Отзыв для изменения

ДОБАВИТЬ ОТЗЫВ +

Действие:

⌵

 Выполнить Выбрано 0 объектов из 6

☐ ОТЗЫВ

☐ Автор: 111, Отзыв: 222222

☐ Автор: Кот, Отзыв: Из-за наличия собаки я не могу зайти на участок. Приходится ждать "когда же собаку выведут на прогулку"...

☐ Автор: Стеша, Отзыв: К сожалению я появилась позже, чем Барса умерла. Мне было бы интересно перегнать лидерство.

☐ Автор: Александр, Отзыв: Эта кошка известна всей деревне. Она побывала почти на всех участках.

☐ Автор: Аслан, Отзыв: Этот белый монстр меня пугает! У меня осыпается шерсть. Хуже этого монстра только дети, от которых приходится прятаться под потолком.

☐ Автор: Александр, Отзыв: Большая собака. У неё есть 4 лапы, голова и хвост.

6 Отзывов

← → ↺ 🏠

🔒 127.0.0.1:8000/admin/App/animal_review/2/change/

📄 ☆

☰ 📄 👤 >> ≡

Администрирование Django

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

Начало · App · Отзывы · Автор: Аслан, Отзыв: Этот белый монстр меня пугает! У меня осыпается шерсть. Хуже этого монстра только дети, от ...

APP

Животные + Добавить

Отзывы + Добавить

ПОЛЬЗОВАТЕЛИ И ГРУППЫ

Группы + Добавить

Пользователи + Добавить

Изменить Отзыв

Автор: Аслан, Отзыв: Этот белый монстр меня пугает! У меня осыпается шерсть. Хуже этого монстра тол...

ИСТОРИЯ

от которых приходится прятаться под потолком.

Animal:

Тип: Кошка; Имя: Барса; URL фото: http://127.0.0.1:8000/static/barsa.jpg

 ✎ +

Имя автора:

Аслан

Текст отзыва:

Этот белый монстр меня пугает!У меня ос

Удалить

Сохранить и добавить другой объект

Сохранить и продолжить редактирование

СОХРАНИТЬ