

Московский государственный технический
университет им. Н. Э. Баумана

Факультет «Радиотехнический»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Разработка интернет приложений»
Отчет по лабораторной
работе № 5
«Работа с СУБД. Обработка данных
с использованием Django ORM.»

Выполнил:

студент группы РТ5-51
Робертс Д.А.

Подпись и дата:

Проверил:

преподаватель каф. ИУ5
Гапанюк Ю. Е.

Подпись и дата:

Москва, 2021 г.

Полученное задание:

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.

Создайте сценарий с подключением к БД и несколькими запросами, примеры рассмотрены в методических указаниях.

Реализуйте модели Вашей предметной области из предыдущей ЛР (минимум две модели, т.е. две таблицы).

Создайте представления и шаблоны Django для отображения списка данных по каждой из сущностей.

Текст программы:

Файл models.py

```
from django.db import models

# Create your models here.
class Book(models.Model):
    name = models.CharField(max_length=30)
    dicription = models.CharField(max_length=255)

    class Meta:
        managed = False
        db_table = 'books'

class Doge(models.Model):
    name = models.CharField(max_length=30)
    dicription = models.CharField(max_length=255)

    class Meta:
        managed = False
        db_table = 'doge'

class Kit(models.Model):
    name = models.CharField(max_length=30)
    dicription = models.CharField(max_length=255)

    class Meta:
        managed = False
        db_table = 'kit'
```

файл views.py

```
from doge.models import Book
from datetime import date
from doge.models import Doge
from doge.models import Kit
def bookList(request):
    return render(request, 'doge/books.html', {'data' : {
        'current_date': date.today(),
        'books': Book.objects.all(),
        'dogs': Doge.objects.all(),
        'cats': Kit.objects.all()
    }})
def GetBook(request, id):
    return render(request, 'doge/book.html', {'data' : {
        'current_date': date.today(),
        'book': Book.objects.filter(id=id)[0]
    }})
def GetDog(request, id):
    return render(request, 'doge/dog.html', {'data' : {
        'dog': Doge.objects.filter(id=id)[0]
```

```

    })
def GetKit(request, id):
    return render(request, 'doge/kit.html', {'data' : {

        'cat': Kit.objects.filter(id=id)[0]
    }})

```

Файл books.html (Главная страница)

```

{% extends 'doge/base.html' %}
{% load static %}
{% block title1 %}
5 Лабораторная работа
{% endblock %}
{% block title %}
5 Лабораторная работа
{% endblock %}
{% block content %}
<h1>Книги</h1>
    {% for book in data.books %}
    <table>
        <td> <a href="book/{{book.id}}">{{ book.name }}</a></td>
    </table>
    {% empty %}
        <li>Список пуст</li>
    {% endfor %}
<hr>
<h1>Собаки</h1>
{% for dog in data.dogs %}
<table>
    <td> <a href="dog/{{dog.id}}">{{ dog.name }}</a></td>
</table>
    {% empty %}
        <li>Список пуст</li>
    {% endfor %}</hr>
<hr><h1>Коты</h1>
{% for cat in data.cats %}
<table>
    <td> <a href="kit/{{cat.id}}">{{ cat.name }}</a></td>
</table>
    {% empty %}
        <li>Список пуст</li>
    {% endfor %}</hr>
{% endblock %}

```

Файл book.html

```

{% extends 'doge/base.html' %}
{% load static %}
{% block title1 %}
Книга
{% endblock %}
{% block title %}

```

Книга

```
{% endblock %}
{% block content %}
    <div>Название: {{ data.book.name }}</div>
    <div>Описание: {{ data.book.dicription }}</div>
{% endblock %}
```

Файл kit.html

```
{% extends 'doge/base.html' %}
{% load static %}
{% block title1 %}
```

Кот

```
{% endblock %}
{% block title %}
```

Кот

```
{% endblock %}
{% block content %}
    <div>Название породы: {{ data.cat.name }}</div>
    <div>Описание: {{ data.cat.dicription }}</div>
{% endblock %}
```

Файл dog.html

```
{% extends 'doge/base.html' %}
{% load static %}
{% block title1 %}
```

Собака

```
{% endblock %}
{% block title %}
```

Собака

```
{% endblock %}
{% block content %}
    <div>Название породы: {{ data.dog.name }}</div>
    <div>Описание: {{ data.dog.dicription }}</div>
{% endblock %}
```

Файл settings.py (строки подключения бд)

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': "first_db",
        'USER': "dbuser",
        'PASSWORD': "123",
        'HOST': "localhost",
        'PORT': 3306, # Стандартный порт MySQL
        'OPTIONS': {'charset': 'utf8'},
        'TEST_CHARSET': 'utf8',
    }
}
```

Экранные формы с примерами выполнения программы:



Книги

[Мастер и Маргарита](#)

[Book](#)

[Book](#)

[Book1](#)

Собаки

[Сибя-Ину](#)

[Чихуахуа](#)

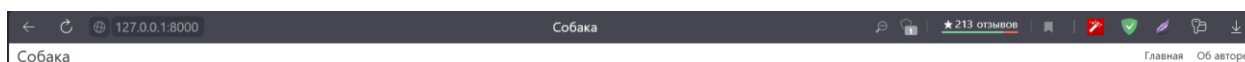
Коты

[Сфинкс](#)

[Манчкин](#)

Так выглядит главная страница лабораторной работы.

При переходе по любой ссылке будет видна следующая страница:



Скрины таблиц из mysql

Командная строка - "C:\Program Files\MySQL\MySQL Server 8.0\bin\mysql" -u root -p

```
mysql> select * from books;
```

id	name	dicription
1	Мастер и Маргарита	Крутая книга
2	Book	Description
3	Book	Description
4	Book1	Description

```
4 rows in set (0.00 sec)
```

```
mysql> select * from kit
```

```
-> ;
```

id	name	dicription
1	Сфинкс	Лысый кот
2	Манчкин	У него маленькие лапки

```
2 rows in set (0.01 sec)
```

```
mysql> select * from doge;
```

id	name	dicription
1	Сиба-Ину	Японская охотничья собака
2	Чихуахуа	Маленький дьявол

```
2 rows in set (0.00 sec)
```

Вывод

Я научился подключать базу данных MySQL к среде разработки python, создавать базы данных в MySQL . Также получил опыт в создании сайтов с использованием баз данных, расположенных на отдельном сервере.