



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 _____ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ _____

КАФЕДРА _____ СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ (ИУ5) _____

О Т Ч Е Т

по лабораторной работе №5

по дисциплине: Разработка Интернет-приложений _____

на тему: Обработка данных с использованием Django ORM _____

Студент ИУ5-54Б
(Группа)

(Подпись, дата)

(И.О.Фамилия)

Руководитель

(Подпись, дата)

(И.О.Фамилия)

2019 г.

Задание лабораторной работы

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 базой. После этого вы создадите свои модели с помощью Django ORM, отобразите объекты из БД с помощью этих моделей и ClassBasedViews.

Необходимо выполнить следующие пункты:

1. Скрипт с подключением к БД и несколькими запросами.
2. Набор классов вашей предметной области с привязкой к СУБД (класс должен уметь хотя бы получать нужные записи из БД и преобразовывать их в объекты этого класса)
3. Модели вашей предметной области
4. View для отображения списка ваших сущностей

Выполнение лабораторной работы

1. Создание пользователя:

```
CREATE USER 'dbuser'@'localhost'
```

2. Создание и использование базы данных:

```
CREATE DATABASE first_db CHARACTER SET utf8 COLLATE utf8_general_ci;  
USE first_db;
```

3. Выдача привилегий пользователю:

```
GRANT ALL PRIVILEGES ON first_db.* TO 'dbuser'@'localhost';
```

4. Создание таблицы «Пользователи»:

```
CREATE TABLE  
`users` (  
  `id` INT(11) NOT NULL AUTO_INCREMENT,  
  `name` CHAR(30) NOT NULL,  
  `age` SMALLINT(6) NOT NULL,  
  `email` VARCHAR(30) NOT NULL,  
  PRIMARY KEY(`id`)  
);  
show tables;
```

5. Вставка одной записи в таблицу:

```
INSERT INTO users VALUES(1, 'Илья', 0, 'ilia.serg@yandex.ru');
```

Работа с базой данных из Python:

1. Подключение к базе данных:

```
import pymysql.cursors  
  
# Connect to the database  
connection = pymysql.connect(host='localhost',  
                             user='dbuser',  
                             password='123',  
                             db='first_db',  
                             charset='utf8',  
                             cursorclass=pymysql.cursors.DictCursor)
```

```

try:
    with connection.cursor() as cursor:

        # Вставка записи
        sql_request = "INSERT INTO users (name, age, email) VALUES (%s, %s, %s)"
        cursor.execute(sql_request, ('Рома', '3', 'rauger474@mail.ru'))

        # Зафиксировать изменения
        connection.commit()

    with connection.cursor() as cursor:

        # Выполним выборку
        sql_request = "SELECT * FROM users"
        cursor.execute(sql_request)
        result = cursor.fetchall()

        print(result)
finally:
    connection.close()

```

Остальные листинги:

- Models.py

```

from django.db import models

class Post(models.Model):
    post_head = models.CharField(max_length=70)
    post_text = models.CharField(max_length=255)
    publication_date = models.DateField('Date published')

    def __str__(self):
        return self.post_head

```

- Views.py

```

from django.views import generic

class PostView(generic.DeleteView):
    model = Post
    template_name = 'post.html'

```

- Паттерн из urls.py:

```

url(r'^(?P<pk>[0-9]+)/$', views.PostView.as_view(), name='postDetail'

```