



**Московский государственный технический университет
им. Н.Э. Баумана
(МГТУ им. Н.Э. Баумана)
Радиотехнический факультет (РТ)**

Отчёт по лабораторной работе № 5
По дисциплине
«Разработка интернет-приложений»

Проверил:

Преподаватель кафедры ИУ-5

Гапанюк Ю.Е.

Подпись: _____

«___» _____ 201_ г.

Выполнил:

студент группы РТ5-51Б

Ануров Н.С.

Подпись: _____

«___» _____ 201_ г.

Москва, 2019

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

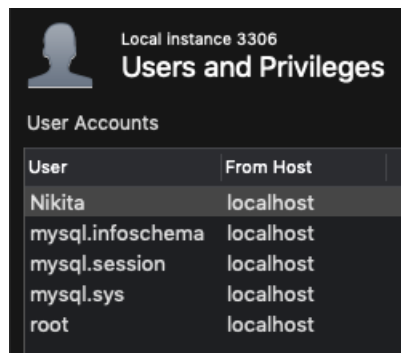
Для сдачи вы должны иметь:

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

2. Выполнение задания и исходный код проекта

Работа с MySQL:

- 1) Создание пользователя



Local Instance 3306
Users and Privileges

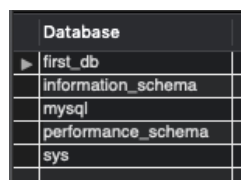
User Accounts

User	From Host
Nikita	localhost
mysql.infoschema	localhost
mysql.session	localhost
mysql.sys	localhost
root	localhost

```
CREATE USER 'Nikita'@'localhost' IDENTIFIED BY '';
```

- 2) Создание и использование базы данных

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



Database
first_db
information_schema
mysql
performance_schema
sys


- 3) Привилегии пользователю:

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

- 4) Создание таблицы:

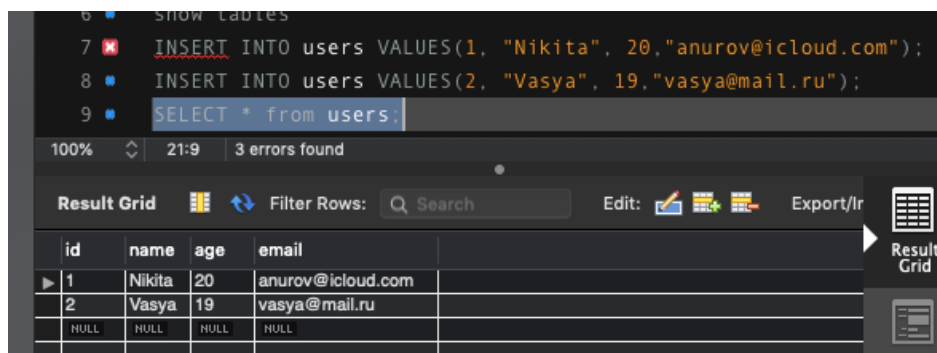
```
CREATE  
`users` (  
  `id` INT(11) NOT NULL AUTO_INCREMENT,  
  `name` CHAR(30) NOT NULL,
```

```
`age` SMALLINT(6) NOT NULL,  
`email` VARCHAR(30) NOT NULL,  
PRIMARY KEY(`id`)  
);  
show tables;
```



Tables_in_first_db	
▶	users

5) Вставка записи в таблицу:



```
6 show tables  
7 ✖ INSERT INTO users VALUES(1, "Nikita", 20,"anurov@icloud.com");  
8 INSERT INTO users VALUES(2, "Vasya", 19,"vasya@mail.ru");  
9 SELECT * from users;
```

100% 21:9 3 errors found

Result Grid Filter Rows: Search Edit: Export/Import

id	name	age	email
▶ 1	Nikita	20	anurov@icloud.com
2	Vasya	19	vasya@mail.ru
HULL	HULL	HULL	HULL

Result Grid

Подключение к базе данных из Python:

```
import pymysql.cursors

# Connect to the database
connection = pymysql.connect(host='lych',
                             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, ('Petr', '17', 'vchanfaew@gmail.com'))

        # 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()
```

Работа с Class Based Views:

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'
```

url-паттерн:

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