

МГТУ им. Н. Э. Баумана, кафедра ИУ5  
курс “Разработка интернет-приложений”

## Лабораторная работа №5

### Django ORM

ВЫПОЛНИЛ:

Матюнин да Вейга Р.А.

Группа: ИУ5-51Б

ПРОВЕРИЛ:

Гапанюк Ю.Е.

Москва 2019

## Задание и порядок выполнения

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

## Выполненная работа

### Работа с MySQL:

1. Создание пользователя

```
CREATE USER 'dbuser'@'localhost' IDENTIFIED BY '41';
```

2. Создание и использование базы данных

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

3. Привилегии пользователю:

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

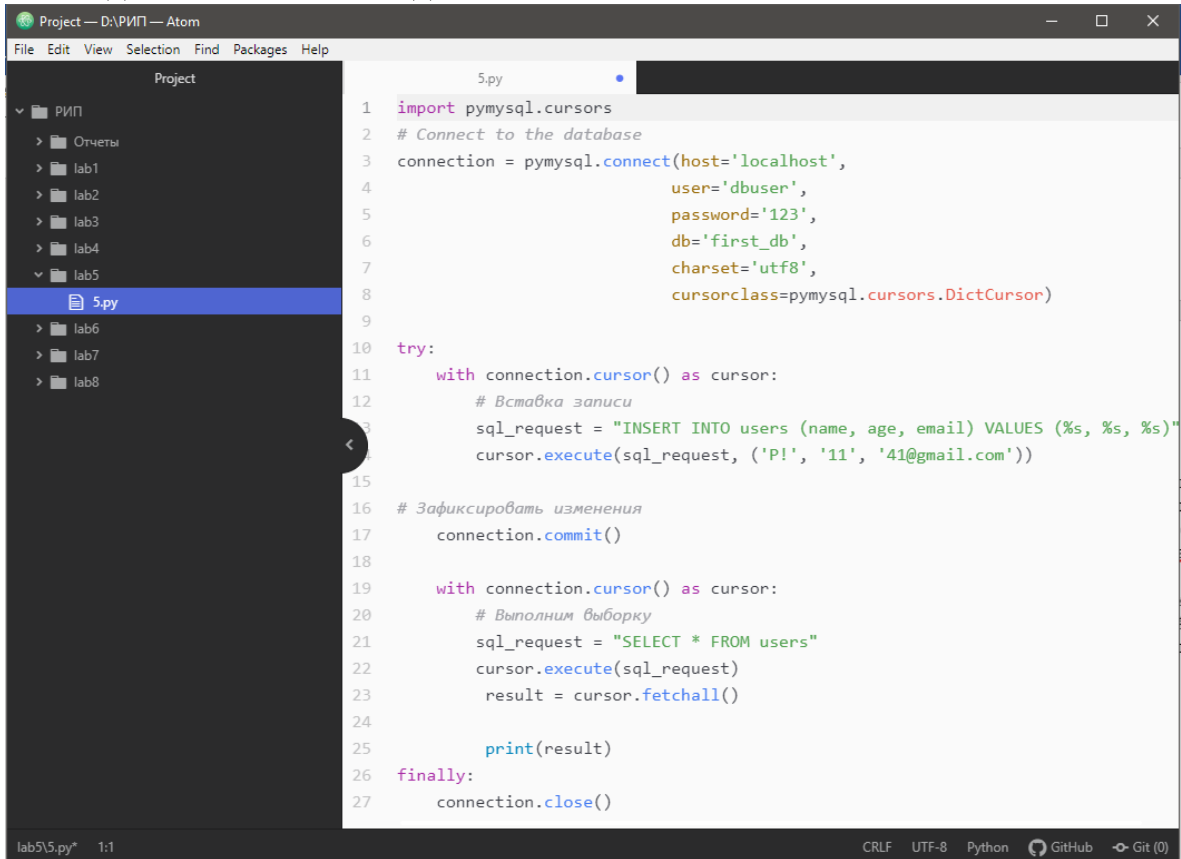
4. Создание таблицы:

```
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 записи в таблицу:

```
INSERT INTO users VALUES(1, 'Пати', 0, '1@gmail.com');  
SELECT * from users;
```

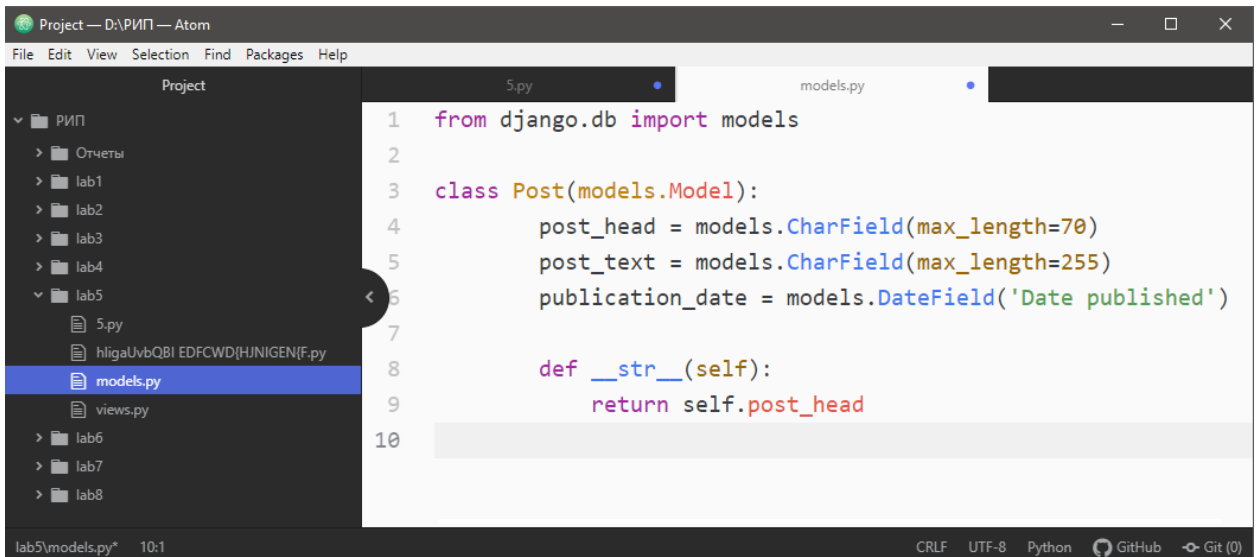
## Подключение к базе данных:



The screenshot shows the Atom editor interface with a project named 'D:\РИП'. The left sidebar displays a file tree with folders 'lab1' through 'lab8' and a file '5.py' selected. The main editor window shows the code for '5.py'.

```
1 import pymysql.cursors
2 # Connect to the database
3 connection = pymysql.connect(host='localhost',
4                             user='dbuser',
5                             password='123',
6                             db='first_db',
7                             charset='utf8',
8                             cursorclass=pymysql.cursors.DictCursor)
9
10 try:
11     with connection.cursor() as cursor:
12         # Вставка записи
13         sql_request = "INSERT INTO users (name, age, email) VALUES (%s, %s, %s)"
14         cursor.execute(sql_request, ('P!', '11', '41@gmail.com'))
15
16 # Зафиксировать изменения
17 connection.commit()
18
19 with connection.cursor() as cursor:
20     # Выполним выборку
21     sql_request = "SELECT * FROM users"
22     cursor.execute(sql_request)
23     result = cursor.fetchall()
24
25     print(result)
26 finally:
27     connection.close()
```

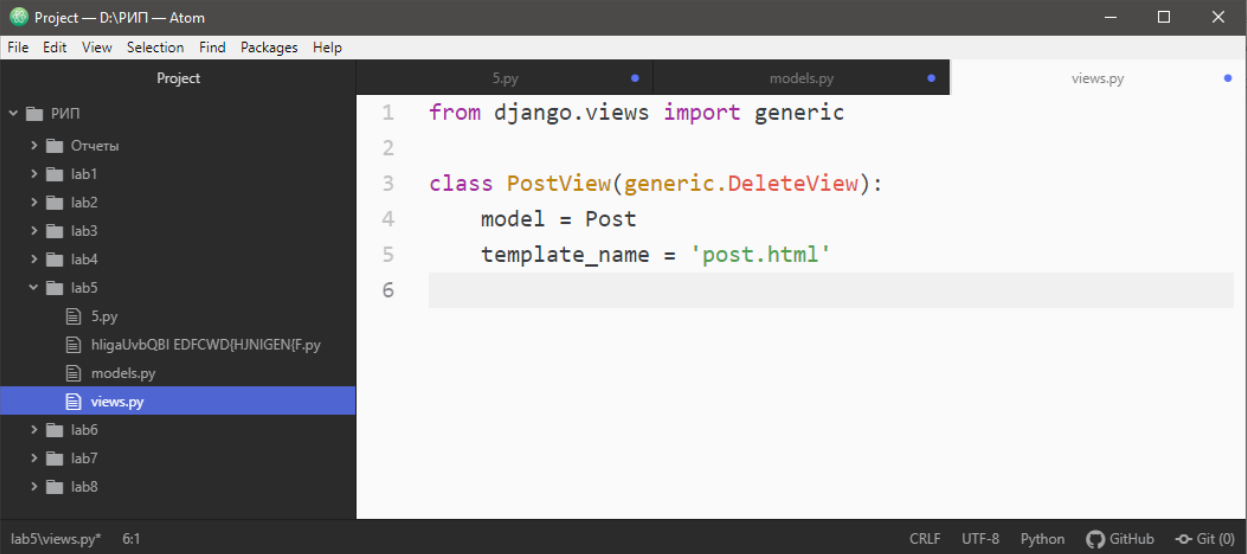
## Based Views: models.py



The screenshot shows the Atom editor interface with the same project. The left sidebar now shows 'models.py' selected under the 'lab5' folder. The main editor window shows the code for 'models.py'.

```
1 from django.db import models
2
3 class Post(models.Model):
4     post_head = models.CharField(max_length=70)
5     post_text = models.CharField(max_length=255)
6     publication_date = models.DateField('Date published')
7
8     def __str__(self):
9         return self.post_head
10
```

## views.py



The screenshot shows the Atom text editor interface. The title bar reads "Project — D:\РИП — Atom". The menu bar includes "File", "Edit", "View", "Selection", "Find", "Packages", and "Help". The left sidebar displays a project tree with a folder named "РИП" containing subfolders "Отчеты", "lab1", "lab2", "lab3", "lab4", "lab5", "lab6", "lab7", and "lab8". Inside "lab5", there are files "5.py", "hlgaUvbQBI EDFCWD(HJNIGEN(F.py", "models.py", and "views.py", which is currently selected. The main editor area shows the content of "views.py" with the following code:

```
1 from django.views import generic
2
3 class PostView(generic.DeleteView):
4     model = Post
5     template_name = 'post.html'
6
```

The status bar at the bottom indicates "lab5\views.py" at line 6, column 1. On the right side of the status bar, there are icons for "CRLF", "UTF-8", "Python", "GitHub", and "Git (0)".

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