|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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 \_\_\_\_\_ИУ5-53Б\_\_\_\_\_\_ **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_Труфанов В.А.\_\_\_**

(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_Гапанюк Ю.Е.\_\_\_**

(Подпись, дата) (И.О.Фамилия)

*2019 г.*

**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
5. **Выполнение задания и исходный код проекта**

**Работа с MySQL:**

1. Создание пользователя

CREATE USER 'dbuser'@'localhost' IDENTIFIED BY '123';

1. Создание и использование базы данных

CREATE DATABASE sample CHARACTER SET utf8 COLLATE utf8\_general\_ci;

USE sample;

1. Привилегии пользователю:

GRANT ALL PRIVILEGES ON sample.\* TO 'dbuser'@'localhost';

1. 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`)

);

1. Вставка записи в таблицу:

INSERT INTO users VALUES(1, 'Вадим', 0, 'ruby@maiil.com');

SELECT \* from users;

**Подключение к базе данных из Python:**

DATABASES = {

'default': {

'ENGINE': 'django.db.backends.mysql',

'OPTIONS': {

'read\_default\_file': '/etc/mysql/my.cnf',

'charset': 'utf8mb4',

'use\_unicode': True,

},

}

}

Файл /etc/mysql/my.cnf

[client]

database = Coursera

user = courser

password = courser123

default-character-set = utf8mb4

**Работа с Class Based Views:**

models.py

class Course(models.Model):

name = models.CharField(max\_length=100)

description = models.TextField()

image = models.ImageField(upload\_to='courses')

pub\_date = models.DateTimeField('Дата добавления')

subscribers = models.ManyToManyField(CustomUser, blank=True)

class Meta:

verbose\_name = \_("Курс")

verbose\_name\_plural = \_("Курсы")

def \_\_str\_\_(self):

return self.name

views.py

class CourseDescView(DetailView):

model = Course'