

Московский государственный технический университет им. Н.Э. Баумана
Факультет «Информатика и системы управления» Кафедра
«Автоматизированные системы обработки информации и управления»



Студент Андреев К.А.

Отчет

по выполнению лабораторной работы По курсу

“Разработка интернет-приложений”

Лабораторная работа № 5

Задание: 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

Файлы:

Models.py

```
from django.db import models
from django.contrib.auth.models import User
from django.contrib import admin

class films(models.Model):
    name = models.CharField(max_length=30)
    description = models.CharField(max_length=255)

    objects = models.Manager()

    def __unicode__(self):
        return self.name

class User(models.Model):
    name = models.CharField(max_length=30)
    email = models.CharField(max_length=255)
```

Views.py

```
from django.views.generic import
from polls.models import films
class filmsView(ListView):
    model = films
    template_name = 'lab5.html'
```

из файла Settings.py

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

```

    }
}

Lab5.html
{% extends 'base.html' %}

{% block title %}Товары{% endblock %}

{% block body %}
    <div class="jumbotron">
        <div class="container">
            <div class="page-header">
                <h1>ФИЛЬМЫ</h1>
            </div>
            <ul class="list-group">
                {% for film in object_list %}
                    <li>
                        <div class="main_block_elem">
                            <h4>{{ film.name }}</h4>
                            Описание: {{ film.description }}<br>
                        </div>
                    </li>
                {% empty %}
                    <li>пустой список</li>
                {% endfor %}
            </ul>
        </div>
    </div>
{% endblock %}

```

Classes.py

```

import MySQLdb

class Connection:
    def __init__(self, user, password, db, host='localhost'):
        self.user = user
        self.host = host
        self.password = password
        self.db = db
        self._connection = None

    @property
    def connection(self):
        return self._connection

    def __enter__(self):
        self.connect()

    def __exit__(self, exc_type, exc_val, exc_tb):
        self.disconnect()

    #! Открытие соединения
    def connect(self):
        if not self._connection:
            self._connection = MySQLdb.connect(
                host=self.host,
                user=self.user,
                passwd=self.password,
                db=self.db
            )
            self._connection.set_character_set('utf8')

    #! Закрытие соединения

```

```

def disconnect(self):
    if self._connection:
        self._connection.close()

class Film:

    def __init__(self, db_connection, name, description):
        self.db_connection = db_connection.connection
        self.name = name
        self.description = description

    def save(self):
        c = self.db_connection.cursor()
        c.execute("insert into films(name, description) values(%s, %s);",
                  (self.name, self.description))
        self.db_connection.commit()
        c.close()

    def show(self):
        c = self.db_connection.cursor()
        c.execute("select * from films;")
        entries = c.fetchall()
        for e in entries:
            print(e)
        c.close()
conn = Connection("dbuser", "123", "first_db")

with conn:
    film = Film(conn, 'Матрица', "Добро пожаловать в реальный мир")
    film.show()

```

Example.py

```

import MySQLdb

#! Открытие соединения с базой данных
db = MySQLdb.connect(
    host="localhost",
    user="dbuser",
    passwd="123",
    db="first_db"
)
db.set_character_set('utf8')
#! Получить курсор для работы с базой данных
c=db.cursor()

#! Выполнить вставку
# c.execute("insert into films (name, description) VALUES (%s, %s);",
# ('Аватар', 'Это новый мир'))
# #! Фиксирование изменений
# db.commit()

#! Выполнить выборку
c.execute("select * from films;")

#! Забрать все полученные записи
entries = c.fetchall()

#! Распечатать записи
for e in entries:
    print(e)

```

```
#!/ Закрытие курсора
c.close()
#!/ Закрытие соединения
db.close()
```

Результаты:

```
C:\Users\kiril\AppData\Local\Programs\Python\Python36-32\python.exe C:/Users/kiril/Dropbox/РИП/lab5/python/classes.py
(1, '2012', 'Узнай правду...Если сможешь!')
(2, 'Аватар', 'Это новый мир')
(3, 'Интерстеллар', 'Следующий шаг человечества станет величайшим')

Process finished with exit code 0
```

