



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 (ИУ)

КАФЕДРА _____ СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ (ИУ5)

О Т Ч Е Т

по лабораторной работе № 5

по дисциплине: Разработка интернет-приложений

на тему: Работа с СУБД

Студент ИУ5-53
(Группа)

(Подпись, дата) А.С. Волков
(И.О.Фамилия)

Руководитель

(Подпись, дата) Ю.Е. Гапанюк
(И.О.Фамилия)

2019 г.

1. 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

2. Исходные коды

2.1. Подключение к БД, добавление записи и вывод всех записей

(test_connect.py)

```
import MySQLdb as pymysql

db = sql.connect(
    host='localhost',
    user='dbadmin',
    password='123321',
    db='first_db'
)

c = db.cursor()

c.execute('INSERT INTO books (name, description) VALUES (%s, %s);', ('Book', 'Book
description'))

db.commit()

c.execute('SELECT * FROM books;')

entries = c.fetchall()

for e in entries:
    print(e)

c.close()
db.close()
```

2.2. Класс Connection и Book (book_class.py)

```
import MySQLdb as pymysql

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()

    def connect(self):
        if not self._connection:
            self._connection = sql.connect(
                host=self.host,
                user=self.user,
                password=self.password,
                db=self.db
            )

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

class Book:

    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 books (name, description) VALUES (%s, %s);',
            (self.name, self.description))
        self.db_connection.commit()
        c.close()

con = Connection('dbadmin', '123321', 'first_db')

with con:
    book = Book(con, 'New book', 'New book description')
    book.save()

```

2.3. views.py

```

from django.shortcuts import render
from django.views.generic import View, ListView
from lab5.models import Book, Review

```

```

# Числок кнуг
class ListbookView(ListView):
    model = Book
    template_name = 'book_list.html'
    context_object_name = 'books'
    paginate_by = 3

```

```

def get(self, request, page=1):

    books = Book.objects.all()

    return render(request, 'book_list.html', {"books": books})

# Страница с информацией о книге и отзывами
class bookView(View):

    def get(self, request, book_id):

        book = Book.objects.get(id=book_id)
        reviews = Review.objects.filter(book_id=book_id)

        return render(request, 'book.html', {"book": book, "reviews": reviews})

```

2.4. book_list.html

```

{% extends 'base.html' %}

{% block title %} {% endblock %}

{% block body %}

    <!-- Cards -->
    <div id="cards" class="container mb-2" style="padding-bottom: 1px">
        <div class="section-title col-lg-7 col-md-4 col-sm-12 mb-4">
            <h3 class="mb-4">Welcome to our book forum!</h3>
            <p>Here is a list of books we have reviews on:</p>
        </div>
        {% for book in books %}
            <div class="col-lg-7 col-md-4 col-sm-12 mb-4">
                <div class="card">
                    <div class="card-body">
                        <h4 class="card-title">{{ book.name }}</h4>
                        <p class="card-text">{{ book.short_description }}...</p>
                        <a href="/book/{{ book.id }}" class="btn btn-
primary">More &rarr;</a>
                    </div>
                </div>
            </div>
        {% endfor %}
    </div>

{% endblock %}

```

2.5. book.html

```

{% extends 'base.html' %}

{% block title %} {% endblock %}

{% block body %}

    <h1>{{ book.name }}</h1>
    <div class="media">
        <div class="media-left">
        </div>
        <div class="media-body">

```

```

        <div class="jumbotron">
            <h2>{{ book.seller }}</h2>
        </div>
        <div class="jumbotron">
            <p>{{ book.description }}</p>
        </div>
        <p><a class="btn btn-lg btn-success" role="button">Like</a></p>
    </div>
</div>
{% for user in reviews %}
    <div class="col-lg-7 col-md-4 col-sm-12 mb-4">
        <div class="card">
            <div class="card-body">
                <h4 class="card-title">{{ user.user }}</h4>
                <p class="card-text">{{ user.description }}</p>
            </div>
        </div>
    </div>
</div>

{% endfor %}
{% endblock %}

```

2.6. base.html

```

{% load static %}

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

        <title>{% block title %}{% endblock %}</title>

        <!-- CSS Dependencies -->
        <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css" integrity="sha384-Y6pD6FV/Vv2HJnA6t+vs1U6fwYXjCFtcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M" crossorigin="anonymous">

        {% load static %}
        <!--<link rel="stylesheet" href="{% static 'css/bootstrap.min.css' %}" --> -->
        <link href="{% static 'css/bootstrap.min.css' %}" rel="stylesheet">

        <!-- Custom styles for this template -->
        <style>
        </style>

    </head>

    <body>

        <nav class="navbar navbar-expand-lg navbar-dark bg-dark mb-4">
            <a class="navbar-brand" href="..">books</a>
            <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavDropdown-1"
                aria-controls="navbarNavDropdown-1"
                aria-expanded="false" aria-label="Toggle navigation">

```

```

        <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarNavDropdown-1">
        {% block navbar-items %}

        {% endblock %}
    </div>
</nav>

<!-- /.container -->
<div class="container">

    {% block body %}

    {% endblock %}

</div>

<!-- JavaScript Dependencies: jQuery, Popper.js, Bootstrap JS, Shards JS -->
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-
KJ3o2DKtIkvYIK3UENzmM7KCKRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
crossorigin="anonymous"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js"
integrity="sha384-b/U6ypiBEHpOf/4+1nzFpr53nxSS+GLCKfwbDfNtXtclqgenISfwAzpKaMNFNmj4"
crossorigin="anonymous"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-
beta/js/bootstrap.min.js" integrity="sha384-
h0AbiXch4ZDo7tp9hKZ4TsHbi047NrKGL03SEJAg45jXxnGIYzk4Si90RDIqNm1"
crossorigin="anonymous"></script>

    {% load static %}
    <!--<script src="{% static 'js/shards.js' %}"></script>-->
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
    <script src="{% static 'js/bootstrap.min.js' %}"></script>
</body>
</html>

```

3. Скриншоты

3.1. Результат выполнения test_connect.py

В таблицу была добавлена новая запись.

```

"C:\Users\Артём\Google Диск\Учёба\5 сем\РИП\lab5_site\venv\Scripts\python.exe" "C:/Users/Артём/Google Диск/Учёба/5 сем/РИП/lab5_site/test_connect.py"
(1, 'War and peace', 'Book by Tolstoy')
(2, 'Crime and punishment', 'Book by Dostoevsky')
(3, 'Roadside picnic', 'Book by Strugatsky')
(4, 'Book', 'Book description')

```

3.2. Результат выполнения book_class.py

В таблицу была добавлена новая запись.

```
mysql> select * from books;
```

id	name	description
1	War and peace	Book by Tolstoy
2	Crime and punishment	Book by Dostoevsky
3	Roadside picnic	Book by Strugatsky
4	Book	Book description
5	New book	New book description

3.3. Главная страница со списком книг

books

Welcome to our book forum!

Here is a list of books we have reviews on:

War and peace
 Book by Tolstoy...
[More →](#)

Crime and punishment
 Book by Dostoevsky...
[More →](#)

Roadside Picnic
 Book by Strugatsky...
[More →](#)

Far Rainbow
 Book by Strugatsky...
[More →](#)

3.4. Страница книги с отзывами

books

Roadside Picnic

FREE-PRESS

Book by Strugatsky

Like

zer0
 Awesome book by Strugatsky brothers

onelvan
 Awful book by Strugatsky brothers