Московский Государственный Технический Университет им. Н.Э. Баумана

Разработка интернет-приложений

Отчёт по лабораторной работе №6

«Работа с СУБД»

Выполнил:

студент группы ИУ5-51

Жизневский Павел

1. Цель работы

Подключить СУБД к приложению Djando

2. Листинг

Файл *settings.py* — настройка баз данных

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

Модуль grow/connection.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 pulpit:
    def __init__(self, db_connection, name, year):
        self.db_connection = db_connection.connection
        self.name = name
        self.year = year

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

    def show(self):
        c = self.db_connection.cursor()
        c.execute("select * from teacher;")
        entries = c.fetchall()
        c.close()
        for e in entries:
            print(e)
        return entries


conn = Connection("dbuser", "123", "first_db")

with conn:
    ded = pulpit(conn, 'rt3', 1798)
    # ded.save()
    ded.show()
```

Модуль grow/ models.py – модели

```
from django.db import models


# Create your models here.


class teacher(models.Model):
    name = models.CharField(max_length=30)
   # author = models.ForeignKey("Author")
    second_name = models.CharField(max_length=30)
    third_name = models.CharField(max_length=30)
    phone = models.PositiveIntegerField()
    mail = models.EmailField()

    def __unicode__(self):
        return self.second_name

class pulpit(models.Model):
    name = models.CharField(max_length=3)
    year = models.PositiveIntegerField()

    def __unicode__(self):
        return self.name
```

Модуль lab5/urls.py

```
from django.conf.urls import include, url
from django.contrib import admin

urlpatterns = [
    url(r'^teachers/', include('grow.urls')),
    url(r'^admin/', admin.site.urls),
    url(r'^pulpits/', include('grow.urls')),
]
```

Модуль grow/urls.py

```
from django.conf.urls import url

from . import views

urlpatterns = [
    url(r'^$', views.index, name='index'),
    url(r'^pulpits/', views.Pit, name='pulpit')
    # url(r'^single/', views.Pit, name='pulpit')
]
```

Модуль grow/views.py

```
from django.shortcuts import render
from django.shortcuts import render_to_response
from django.template.loader import render_to_string
from django.http import HttpResponse
from django.template import RequestContext, loader
from .models import teacher
from .models import pulpit
from grow.connection import Connection, pulpit
# Create your views here..


def index(request):
    # conn = Connection("dbuser", "123", "first_db")
    # with conn:
    #     ded = pulpit(conn, 'rt3', 1798)
    #     # ded.save()
    #     teachers = ded.show()
    teachers = teacher.objects.all()
    content = {
        'teachers': teachers
    }
    for a in teachers:
        print(a)
    return render(request, 'teachers.html', content)


def Pit(request):
    pul = pulpit.objects.all()
    content = {
        'pulpits': pul
    }
    return render(request, 'single.html', content)
    pass
```

Шаблон teachers.html

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

{% block content %}

    {% for teacher in teachers %}
        <div class="teacher-body" style="margin: 20px">
                <h2>{{teacher.second_name}} {{ teacher.name }} {{
teacher.third_name }}</h2>
{#                <h1>{{ teacher }}</h1>#}
                <p>{{teacher.mail}}</p>
                <p>{{ teacher.phone }}</p>
        </div>
        {% empty %}
        Список пуст
    {% endfor %}
{% endblock %}
```

Шаблон single.html

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

{% block content %}

        {% for pul in pulpits %}
        <div class="pulpit-body" style="margin: 20px">
                <h2>{{pul.name}}</h2>
{#                <h1>{{ teacher }}</h1>#}
                <p>{{pul.year}}</p>
        </div>
        {% empty %}
        Список пуст
    {% endfor %}

{% endblock %}
```

3. Результат

Страница с преподавателями из базы данных: