

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



“Разработка интернет приложений”

Лабораторная работа №6 «Работа с СУБД»

Студент группы ИУ5-51 _____ Руднев И.В.

Преподаватель _____ Гапанюк Ю.Е.

Москва 2016

1. Задание

Познакомиться с СУБД MySQL, создать свою БД. Дополнить свои классы предметной области, связав их с созданной базой. После этого создать свои модели с помощью Django ORM, отобразить объекты из БД с помощью этих моделей и ClassBasedViews.

2. Порядок выполнения

Создать необходимые модели в файле **models.py**

```
class ServiceManager(models.Manager):
    def new(self):
        return self.order_by('-added_at')

class Service(models.Model):
    title = models.CharField(max_length=128)
    text = models.TextField()
    added_at = models.DateTimeField(default=datetime.datetime.now)
    author = models.ForeignKey(User)
    price = models.DecimalField(max_digits=7, decimal_places=2)
    objects = ServiceManager()
```

Модель пользователя будем брать из `django.contrib.auth.models`

Настраиваем подключение к БД в `settings.py`:

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'django_labs',
        'USER': 'django_client',
        'PASSWORD': '1111',
        'HOST': 'localhost',  # Or an IP Address that your DB is hosted on
        'PORT': '3306',
    }
}
```

Делаем миграции:

`./manage.py makemigrations`

И применяем их, чтобы Django создала нужные таблицы в БД

`./manage.py migrate`

Теперь в views.py мы можем брать реальные данные из базы для отображения их в шаблоне

views.py

```
# -*- coding: utf-8 -*-
from django.shortcuts import render, get_object_or_404
from models import Service
from django.http import HttpResponseRedirect
from django.contrib.auth.models import User
from django.contrib.auth import authenticate, login as auth_login, logout as auth_logout

# Create your views here.

def index(request):
    services = Service.objects.new()
    return render(request, 'index.html', {'services': services})

def service (request, service_id):
    service = get_object_or_404(Service, pk = service_id)
    return render(request, 'service.html', {'service': service})
```

fill_services.py

Скрипт, запуская который мы сохраняем в базу тысячу случайных услуг. Таким образом, можно автоматизировать заполнение базы и проверить работу сайта на большом количестве данных.

```
from django.core.management.base import BaseCommand
from django.contrib.auth.models import User
from my_app.models import Service
from faker import Factory
import random

class Command(BaseCommand):
    def handle(self, *args, **options):
        Service.objects.all().delete()
        fake = Factory.create('en_US')
        user = User.objects.get(pk = 2L)

        for i in range (1000):
            s = Service(
                author=user,
                text = fake.text(),
                title = fake.name(),
                price = random.randint(0, 1000)
            )
            s.save()
```

Скриншоты:


My_APP

Search

Создать услугу


Создать аккаунт

Войти




David Nelson

Harum autem voluptates necessitatibus reiciendis. Sit earum blanditiis doloremque doloremque ad. Sit expedita distinctio iste id.




James Ford

Provident nobis mollitia qui. Tempora a sapiente animi mollitia eaque laboriosam. Molestias dolores quia repellat quis vitae.



Theresa Carter

Cum omnis tempore est corporis deleniti. Iusto ratione harum sapiente pariatur. Velit consectetur adipisci enim molestias. Beatae molestiae maxime velit nam eius libero.



Sara Harris

Nobis omnis quo sequi aliquam quae. Eius inventore vitae unde tenetur vel sed fuqiat. Quae asperiores rerum sed. Accusantium amet quas eos.

Страница с отдельной услугой:

My_APP

Search

Создать услугу

Создать аккаунт

Войти



Theresa Carter

Cum omnis tempore est corporis deleniti. Iusto ratione harum sapiente pariatur. Velit consectetur adipisci enim molestias. Beatae molestiae maxime velit nam eius libero.

Price: 374.00