

**Московский государственный технический  
университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»  
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Разработка интернет приложений»

Отчет по лабораторной работе №5  
«Работа с СУБД. Обработка данных с использованием Django ORM.»

Выполнил:  
студент группы ИУ5-54Б  
Кравцов Андрей

Проверил:  
преподаватель каф. ИУ5  
Гапанюк Ю.Е.

Москва, 2021 г.

**Описание задания:**

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 БД. После этого Вы создадите свои модели с помощью Django ORM, отобразите объекты из БД с помощью этих моделей.

1. Создайте сценарий с подключением к БД и несколькими запросами, примеры рассмотрены в [методических указаниях](#).
2. Реализуйте модели Вашей предметной области из предыдущей ЛР (минимум две модели, т.е. две таблицы).
3. Создайте представления и шаблоны Django для отображения списка данных по каждой из сущностей.

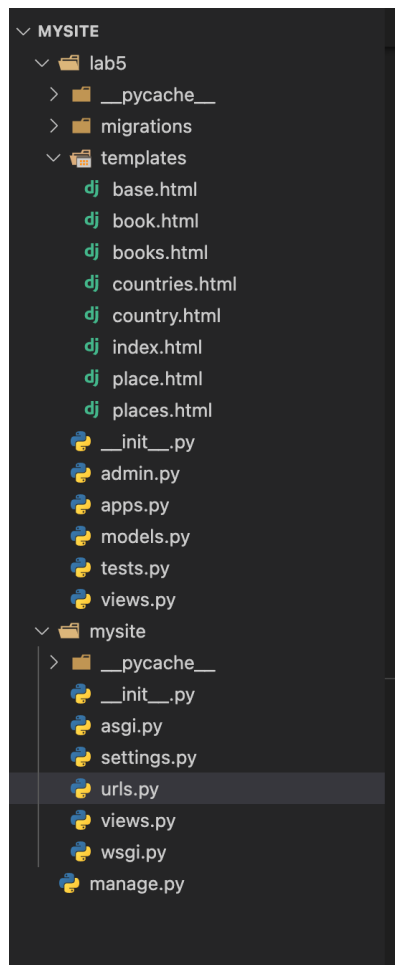
### Создание двух таблиц базы данных:

```
CREATE TABLE Places
```

```
(  
    `id`      INT(11) NOT NULL AUTO_INCREMENT,  
    `name`    CHAR(255) NOT NULL,  
    `description` CHAR(255) NOT NULL,  
    PRIMARY KEY (`id`)  
)
```

```
CREATE TABLE Countries
```

```
(  
    `id`      INT(11) NOT NULL AUTO_INCREMENT,  
    `name`    CHAR(255) NOT NULL,  
    `description` CHAR(255) NOT NULL,  
    PRIMARY KEY (`id`)  
)
```



## Lab5/Views.py

```
from lab5.models import *
from django.shortcuts import render
from datetime import date
```

```
def bookList(request):
    return render(request, 'books.html', {'data': {
        'current_date': date.today(),
        'books': Book.objects.all()
    }})
```

```
def GetBook(request, id):
    return render(request, 'book.html', {'data': {
        'current_date': date.today(),
        'book': Book.objects.filter(id=id)[0],
        'description': Book.objects.filter(id=id)[0]
    }})
```

```
def GetCountry(request, id):
    return render(request, 'country.html', {'data': {
```

```
'current_date': date.today(),
'country': Countries.objects.filter(id=id)[0],
'description': Countries.objects.filter(id=id)[0]
}}
```

```
def GetPlace(request, id):
    return render(request, 'place.html', {'data': {
        'current_date': date.today(),
        'place': Places.objects.filter(id=id)[0],
        'description': Places.objects.filter(id=id)[0]
    }})
```

```
def Countries_list(request):
    return render(request, 'countries.html', {'data': {
        'current_date': date.today(),
        'countries': Countries.objects.all()
    }})
```

```
def Places_list(request):
    return render(request, 'places.html', {'data': {
        'current_date': date.today(),
        'places': Places.objects.all()
    }})
```

```
def index(request):
    return render(request, "index.html")
```

**Файл lab5/models.py с созданием классов моделей таблиц на основе кода, сгенерированного с помощью команды «python**

```
from django.db import models
```

```
# Create your models here.
class Book(models.Model):
    name = models.CharField(max_length=30)
    description = models.CharField(max_length=255)
```

```
class Meta:
    managed = False
    db_table = 'books'
```

```
class Countries(models.Model):
    name = models.CharField(max_length=255)
    description = models.CharField(max_length=255)
```

```
class Meta:
    managed = False
    db_table = 'Countries'
```

```
class Places(models.Model):
    name = models.CharField(max_length=255)
    description = models.CharField(max_length=255)
```

```
class Meta:
    managed = False
    db_table = 'Places'
```

## Файл lab5/base.html:

```
from django.db import models
```

```
# Create your models here.
class Book(models.Model):
    name = models.CharField(max_length=30)
    description = models.CharField(max_length=255)
```

```
class Meta:
    managed = False
    db_table = 'books'
```

```
class Countries(models.Model):
    name = models.CharField(max_length=255)
    description = models.CharField(max_length=255)
```

```
class Meta:
    managed = False
    db_table = 'Countries'
```

```
class Places(models.Model):
    name = models.CharField(max_length=255)
    description = models.CharField(max_length=255)
```

```
class Meta:
```

```
managed = False
db_table = 'Places'
```

## Файл lab5/urls.py:

```
from django.contrib import admin
```

```
from django.urls import path
```

```
from lab5 import views
```

```
urlpatterns = [

    path('admin/', admin.site.urls),

    path("", views.index, name='index'),

    path('book/<int:id>/', views.GetBook, name='book_url'),

    path('place/<int:id>/', views.GetPlace, name='place_url'),

    path('country/<int:id>/', views.GetCountry, name='country_url'),

    path('places/', views.Places_list, name='places_url'),

    path('countries/', views.Countries_list, name='countries_url')

]
```

## Прототип веб-приложения:



[Посмотреть список стран](#)

[Посмотреть список достопримечательностей](#)

1 [Russia](#) Россия – крупнейшая страна мира, расположенная в Восточной Европе и Северной Азии и омываемая водами Тихого и Северного Ледовитого океанов.

2 [Germany](#) Германия – государство в Западной Европе с лесами, реками, горными хребтами и пляжными курортами Северного моря.

3 [USA](#) Соединенные Штаты Америки – государство, состоящее из 50 штатов, занимает значительную часть Северной Америки.

4 [UK](#) Великобритания (официальное название – Соединенное Королевство Великобритании и Северной Ирландии) – островное государство на северо-западе Европы, состоящее из Англии, Шотландии, Уэльса и Северной Ирландии.

[Назад](#)

Название: USA

Описание: Соединенные Штаты Америки – государство, состоящее из 50 штатов, занимает значительную часть Северной Америки.

[Назад](#)

1 [Москва-Сити](#) «Москва-Сити» — это Москва будущего, строящийся международный деловой квартал из ультрасовременных небоскребов.

2 [Воробьевы горы](#) Воробьевы горы — самый высокий из семи холмов, на которых стоит город.

3 [Дворцово-парковый ансамбль Петергоф](#) Дворцово-парковый ансамбль Петергоф — царство фонтанов, феерия играющей воды, дворцы, в которых оживает эпоха Петра Великого.

4 [Аничков мост](#) Аничков мост — один из самых знаменитых мостов Санкт-Петербурга.

[Назад](#)

← → ↺ ⓘ 127.0.0.1:8000/place/2/

Название: Воробьевы горы

Описание: Воробьевы горы — самый высокий из семи холмов, на которых стоит город.

[Назад](#)