|  |  |
| --- | --- |
|  | **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 |

|  |  |
| --- | --- |
| КАФЕДРА | СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ |

Отчет по лабораторной работе №5

Студент Богданова Валерия Вячеславовна

*фамилия, имя, отчество*

Группа ИУ5-55Б.

Студент 22.12.2021 **Богданова В.В.**

*подпись, дата фамилия, и.о.*

Преподаватель 22.12.2021 **Балашов А.Ю.** *подпись, дата фамилия, и.о.*

2021г.

**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.](https://github.com/iu5team/iu5web-fall-2021/blob/main/tutorials/lab5/lab5_tutorial.md)
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.

**Текст программы:**

Работа была выполнена на основе предыдущей ЛР. В данном отчете приведены фрагменты кода, дописанные для текущей ЛР5.

**“insert\_db.py”:**

import psycopg2  
  
db = psycopg2.connect(  
 host="localhost",  
 user="leamur",  
 dbname="lab5\_db"  
)  
  
c = db.cursor()  
c.execute('INSERT INTO "Lab4\_person" (name, experience, technology\_id) VALUES (%s, %s, %s);',  
 ('Natali Harris', 4, 'py'))  
db.commit()  
c.close()  
db.close()

**“models.py”:**

from django.db import models  
  
  
# Create your models here.  
class Technology(models.Model):  
 id = models.CharField(max\_length=5, primary\_key=True)  
 name = models.CharField(max\_length=15)  
 description = models.TextField()  
  
  
class Person(models.Model):  
 name = models.CharField(max\_length=30)  
 experience = models.IntegerField()  
 technology = models.ForeignKey(Technology, on\_delete=models.CASCADE)

**“views.py”:**

from django.shortcuts import render  
from Lab5\_6.models import Technology  
from Lab5\_6.models import Person  
  
  
def get\_langs(request):  
 return render(request, 'langs.html', {  
 'langs': Technology.objects.all(),  
 })  
  
  
def get\_lang(request, lang\_id):  
 return render(request, 'lang.html', {  
 'lang': Technology.objects.filter(id=lang\_id)[0],  
 'persons': Person.objects.all().filter(technology\_id=lang\_id)  
 })

**“lang.html”:**

{% extends 'base.html' %}  
  
{% block title %}{{ lang.name }}{% endblock %}  
  
{% block content %}  
 {% load static %}  
 <div class="container">  
 <div class="card">  
 <div class="card\_\_header" style="text-align: center">  
 {% if lang.id == "py" %}  
 <img src="{% static 'img/' %}{{ lang.id }}.png" alt="{{ lang.id }}" class="card\_\_image" width="600" style="display: inline; max-width: 90%; padding: 20px">  
 {% else %}  
 <img src="{% static 'img/' %}{{ lang.id }}.png" alt="{{ lang.id }}" class="card\_\_image" width="600">  
 {% endif %}  
 </div>  
 <div class="card\_\_body">  
 <h4>{{ lang.name }}</h4>  
 <p>{{ lang.description }}</p>  
 </div>  
 <div class="card\_\_footer">  
 <div class="user">  
 <div class="user\_\_info">  
 <a href="{% url 'master\_url' %}" class="link">  
 <span class="mask">  
 <div class="link-container">  
 <span class="link-title1 title">Back</span>  
 <span class="link-title2 title">Back</span>  
 </div>  
 </span>  
 </a>  
 </div>  
 </div>  
 </div>  
 </div>  
  
 <div class="main-menu">  
 <p3>People worked with this technology: </p3><br><br>  
 <ul>  
 {% for person in persons %}  
 <li>{{ person.name }} Experience: {{ person.experience }}</li><br>  
 {% empty %}  
 <p4>Список пуст</p4>  
 {% endfor %}  
 </ul>  
 </div>  
 </div>  
{% endblock %}

**“settings.py”:**

DATABASES = {  
 'default': {  
 'ENGINE': 'django.db.backends.postgresql',  
 'NAME': 'lab5\_db',  
 'USER': 'leamur',  
 'PASSWORD': '',  
 'HOST': 'localhost',  
 'PORT': 5432,  
 }  
}

**“migrations/0001\_initial.py”:**

# Generated by Django 3.2.9 on 2021-12-18 23:55  
  
from django.db import migrations, models  
import django.db.models.deletion  
  
  
class Migration(migrations.Migration):  
  
 initial = True  
  
 dependencies = [  
 ]  
  
 operations = [  
 migrations.CreateModel(  
 name='Technology',  
 fields=[  
 ('id', models.CharField(max\_length=5, primary\_key=True, serialize=False)),  
 ('name', models.CharField(max\_length=15)),  
 ('description', models.TextField()),  
 ],  
 ),  
 migrations.CreateModel(  
 name='Person',  
 fields=[  
 ('id', models.BigAutoField(auto\_created=True, primary\_key=True, serialize=False, verbose\_name='ID')),  
 ('name', models.CharField(max\_length=30)),  
 ('experience', models.IntegerField()),  
 ('technology', models.ForeignKey(on\_delete=django.db.models.deletion.CASCADE, to='Lab4.technology')),  
 ],  
 ),  
 ]

**Экранные формы с примерами выполнения программы:**







