

**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 ИУ5-53Б
Ветошкин Артём

Подпись и дата:
20.12.21

Проверил:
Юрий Евгеньевич Гапанюк

Подпись и дата:

Москва, 2021 г.

Цель лабораторной работы: изучение возможностей СУБД PostgreSQL для разработки веб-приложений. Изучение возможностей обработки данных с использованием Django ORM.

Задание:

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.

Выполнение задания лабораторной работы

Текст программы:

views.py

```
from django.shortcuts import render
from main.models import Icecreams, Company

# Create your views here.

def index(request):
    return render(request, 'master.html', {'icecreams':
Icecreams.objects.get_short_des()})

def icecream(request, pk):
    return render(request, 'details.html', {'icecream':
Icecreams.objects.get_current(pk)})
```

urls.py

```
from django.contrib import admin
from django.urls import path
from main import views
from django.conf.urls.static import static
from django.conf import settings

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.index, name="root"),
    path('details/<int:pk>', views.icecream, name="details")
] + static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

models.py

```
from django.db import models

# Create your models here.

class IcecreamManager(models.Manager):
    def get_short_des(self):
        return self.all().only('image', 'name')

    def get_current(self, id):
        return self.prefetch_related('company').get(id=id)
```

```

class Company(models.Model):
    id = models.BigAutoField(primary_key=True)
    name = models.TextField()

    class Meta:
        managed = False
        db_table = 'company'
        verbose_name = 'Компания'
        verbose_name_plural = 'Компании'

class Icecreams(models.Model):
    id = models.BigAutoField(primary_key=True)
    name = models.TextField()
    description = models.TextField()
    image = models.ImageField('картинка', upload_to='icecreams/')
    company = models.ForeignKey(Company, on_delete=models.CASCADE,
verbose_name='компания',
                                related_name='icecream_company')
    price = models.DecimalField(max_digits=100, decimal_places=2, blank=True,
null=True)

    objects = IcecreamManager()

    class Meta:
        managed = False
        db_table = 'icecreams'
        verbose_name = 'Мороженное'
        verbose_name_plural = 'Мороженное'

```

base.html

```

{% load static %}

<!doctype html>
<html class="h-100">
    {% include "include/head.html" %}

    <body class="d-flex flex-column h-100">
        {% include "include/navbar.html" %}

        <!-- Begin page content -->
        <main class="flex-shrink-0">
            <div class="container main-body">
                {% block content %}
                {% endblock content %}
            </div>

        </main>
        {% include "include/footer.html" %}
        <script src="{% static 'js/bootstrap.bundle.min.js' %}"></script>
    </body>
</html>

```

master.html

```

{% extends "include/base.html" %}
{% load static %}

```

```
{% block content %}
<h1 class="body-head"> Мороженное </h1>
<ul class="list-group list-group-flush ">
    {% for icecream in icecreams %}
        <li class="float-left clear-top-bot-padding clear-padding list-group-item row">
            
            <a class="float-left list-icecream nav-link blue-link icecream-size
shadow-text col-9"
href="{% url 'details' pk=icecream.id %}"> {{ icecream.name }} </a>
        </li>
    {% endfor %}
</ul>
{% endblock content %}
```

detail.html

```
{% extends "include/base.html" %}
{% load static %}

{% block content %}
    <h1 class="body-head"> {{ icecream.name }} </h1>
    
    <ul class="list-group list-group-flush ">
        <li class="float-left clear-top-bot-padding clear-padding list-group-item
row">
            <div class="float-left icecream-size col-4 fix-
display">Описание</div>
            <div class="float-left list-icecream text-size col-8"> {{
icecream.description }} </div>
        </li>
        <li class="float-left clear-top-bot-padding clear-padding list-group-item
row">
            <div class="float-left icecream-size col-4 fix-
display">Производитель</div>
            <div class="float-left list-icecream text-size col-8"> {{
icecream.company.name }} </div>
        </li>
        <li class="float-left clear-top-bot-padding clear-padding list-group-item
row">
            <div class="float-left icecream-size col-4 fix-display">Цена</div>
            <div class="float-left list-icecream text-size col-8"> {{
icecream.price }} руб </div>
        </li>
    </ul>
{% endblock content %}
```

footer.html

```
{% block footer %}
<footer class="footer mt-auto py-3 bg-dark">
  <div class="container">
    <span class="text-muted">© 2017–2021</span> <span class="">© 2017–
2021</span>
  </div>
</footer>
{% endblock footer %}
```

head.html

```
{% load static %}

<head>
  <meta charset="utf-8">
  <title>Cool Icecream</title>

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

  <link href="{% static 'css/my_css.css' %}?20181314" rel="stylesheet">
</head>
```

navbar.html

```
{% load static %}

<header>
  <!-- Fixed navbar -->
  <nav class="navbar navbar-expand-md navbar-dark fixed-top bg-navbar">
    <div class="container-fluid">
      <div class="navbar-head">
        <a class="navbar-brand text-head align-self-start" href="{% url
'root' %}"><i><span class="color-head">Cool</span>Icecream</i></a>
      </div>
    </div>
  </nav>
</header>
```

Данные в таблицах базы данных:

	id	name
1	1	Магнат
2	2	Корова из коровники
3	3	Пинолатта

WHERE		ORDER BY				
id	name	description	company_id	price	image	
1	3 Мороженое Пинолатта 33 пингвина (Стаканчик)	Пломбир классический с цельным кедровым орехом и ...	3	100.00	icecreams/pingvin.png	
2	Мороженое пломбир Коровка из Кореновки ваниль БЗМ...	Мороженое понравится любителям сладких холодных д...	2	408.45	icecreams/korovka.png	
3	Мороженое сливочное Магнат шоколадный трифель в ш...	Магнат шоколадный трифель - это утонченный десерт...	1	93.35	icecreams/magnat.png	

Результат работы программы:

CoolIcecream

Мороженное

Мороженое Тинолатта **33** пингвина (Стаканчик)



Мороженое пломбир Коровка из Кореновки ваниль БЗМЖ **1** кг



Мороженое сливочное Магнат шоколадный трюфель в шоколадной глазури **72** г



CoolIcecream

Мороженое Пинолатта 33 пингвина (Стаканчик)



Описание	Пломбир классический с цельным кедровым орехом и шоколадным соусом Состав: молоко цельное, молоко цельное сгущенное с сахаром (молоко цельное, молоко обезжиренное, сахар), масло сладко- сливочное, сливки из коровьего молока, сахар, вода, сыворотка молочная сухая, натуральный стабилизатор (молочный белок, камедь рожкового дерева), шоколадный топпинг (сахар, масло растительное, рафинированное, дезодорированное, какао- порошок, эмульгатор – лецитин соевый, ароматизатор натуральный – ванилин), орех кедровый, ароматизатор натуральный – ванилин.
Производитель	Пинолатта
Цена	100.00 руб