Отчет по лабораторной работе № 5 по курсу
"Разработка Интернет-Приложений"

Выполнил:
Студентка группы
ИУ5-55Б
Зубарева Антонина Михайловна

Москва, МГТУ – 2021

Задание:
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

Для выполнения лабораторной работы была выбрана база данных Postgres. Чтобы избежать проблем с локальной установкой и настройкой базы данных приложение запускается в Docker контейнере.

Текст программы:

docker-compose.yml

```yaml
version: "3.9"

services:
  ripdb:
    image: postgres
    container_name: ripdb
    volumes:
      - ./data/ripdb:/var/lib/postgresql/data
    environment:
      - POSTGRES_USER=root
      - POSTGRES_PASSWORD=root
      - POSTGRES_DB=root
  api:
    build:
      context: ./server
      dockerfile: Dockerfile
    command: "sh entripoint.sh"
    volumes:
      - "./server:/code"
    ports:
      - "8000:8000"
    environment:
      - POSTGRES_NAME=root
      - POSTGRES_USER=root
      - POSTGRES_PASSWORD=root
    depends_on:
      - ripdb
  client:
    container_name: client
    build:
      context: ./client
      dockerfile: Dockerfile
    ports:
      - "5000:80"
    environment:
      - NODE_ENV=production
    depends_on:
      - api
```

docker-compose.dev.yml

docker-compose.dev.yaml

lab4 > docker-compose.dev.yaml

You, 3 weeks ago | 1 author (You)

```yaml
version: "3.9"      You, 3 weeks ago

services:
  client:
    build:
      dockerfile: Dockerfile.dev
    volumes:
      - "./client:/usr/src/app"
    ports:
      - "3000:3000"
    environment:
      - NODE_ENV=development

```

server/Dockerfile

Dockerfile

lab4 > server > Dockerfile > ...

You, 3 weeks ago | 1 author (You)

```
# syntax=docker/dockerfile:1
FROM python:3 as build-deps
ENV PYTHONDONTWRITEBYTECODE=1
ENV PYTHONUNBUFFERED=1
WORKDIR /code
COPY requirements.txt .      You, 3 weeks
RUN pip install -r requirements.txt

```

settings.py

```
"""
Django settings for lab4 project.

Generated by 'django-admin startproject' using Django 3.1.4.

For more information on this file, see
https://docs.djangoproject.com/en/3.1/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/3.1/ref/settings/
"""

from pathlib import Path
```

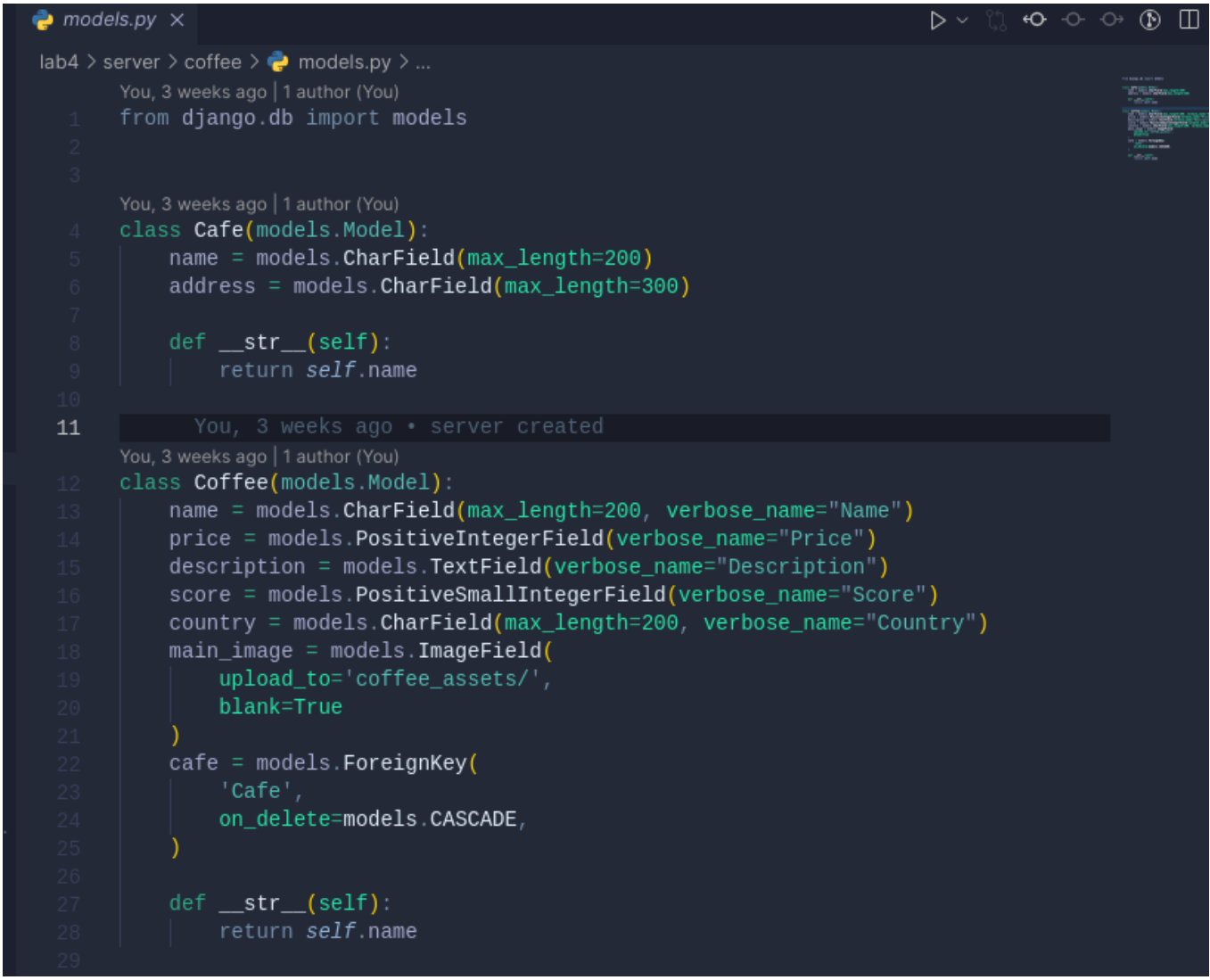
```
import environ
import os

env = environ.Env(DEBUG=(bool, True))

# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent


# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/3.1/howto/deployment/checklist/

# Take environment variables from .env file
environ.Env.read_env(os.path.join(BASE_DIR, '.env'))
SECRET_KEY = env('SECRET_KEY')

DEBUG = env('DEBUG')

ALLOWED_HOSTS = ['localhost', '127.0.0.1', '0.0.0.0']


# Application definition

INSTALLED_APPS = [
    'coffee.apps.CoffeeConfig',
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'corsheaders',
    'rest_framework',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'corsheaders.middleware.CorsMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

# If this is used then `CORS_ALLOWED_ORIGINS` will not have any effect
CORS_ALLOW_ALL_ORIGINS = True
```

```
CORS_ALLOW_CREDENTIALS = True
CORS_ALLOWED_ORIGINS = [
    'http://localhost:3000',
]  # If this is used, then not need to use `CORS_ALLOW_ALL_ORIGINS =
True`
CORS_ALLOWED_ORIGIN_REGEXES = [
    'http://localhost:3000',
]

ROOT_URLCONF = 'lab4.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]

WSGI_APPLICATION = 'lab4.wsgi.application'


# Database
# https://docs.djangoproject.com/en/3.1/ref/settings/#databases

DATABASES = {
    'default': {
        'ENGINE': env('POSTGRES_ENGINE',
default='django.db.backends.sqlite3'),
        'NAME': env('POSTGRES_NAME', default=(os.path.join(BASE_DIR,
'db.sqlite3'))),
        'USER': env('POSTGRES_USER'),
        'PASSWORD': env('POSTGRES_PASSWORD'),
        'HOST': env('POSTGRES_HOST', default='localhost'),
        'PORT': env('POSTGRES_PORT', default='5432'),
    }
}


# Password validation
#
```

```
https://docs.djangoproject.com/en/3.1/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]



# Internationalization
# https://docs.djangoproject.com/en/3.1/topics/i18n/

LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'Europe/Moscow'

USE_I18N = True

USE_L10N = True

USE_TZ = True


# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/3.1/howto/static-files/

STATIC_URL = '/static/'

STATICFILES_DIRS = [
    BASE_DIR / "static",
]

# Media path
MEDIA_URL = '/media/'
```

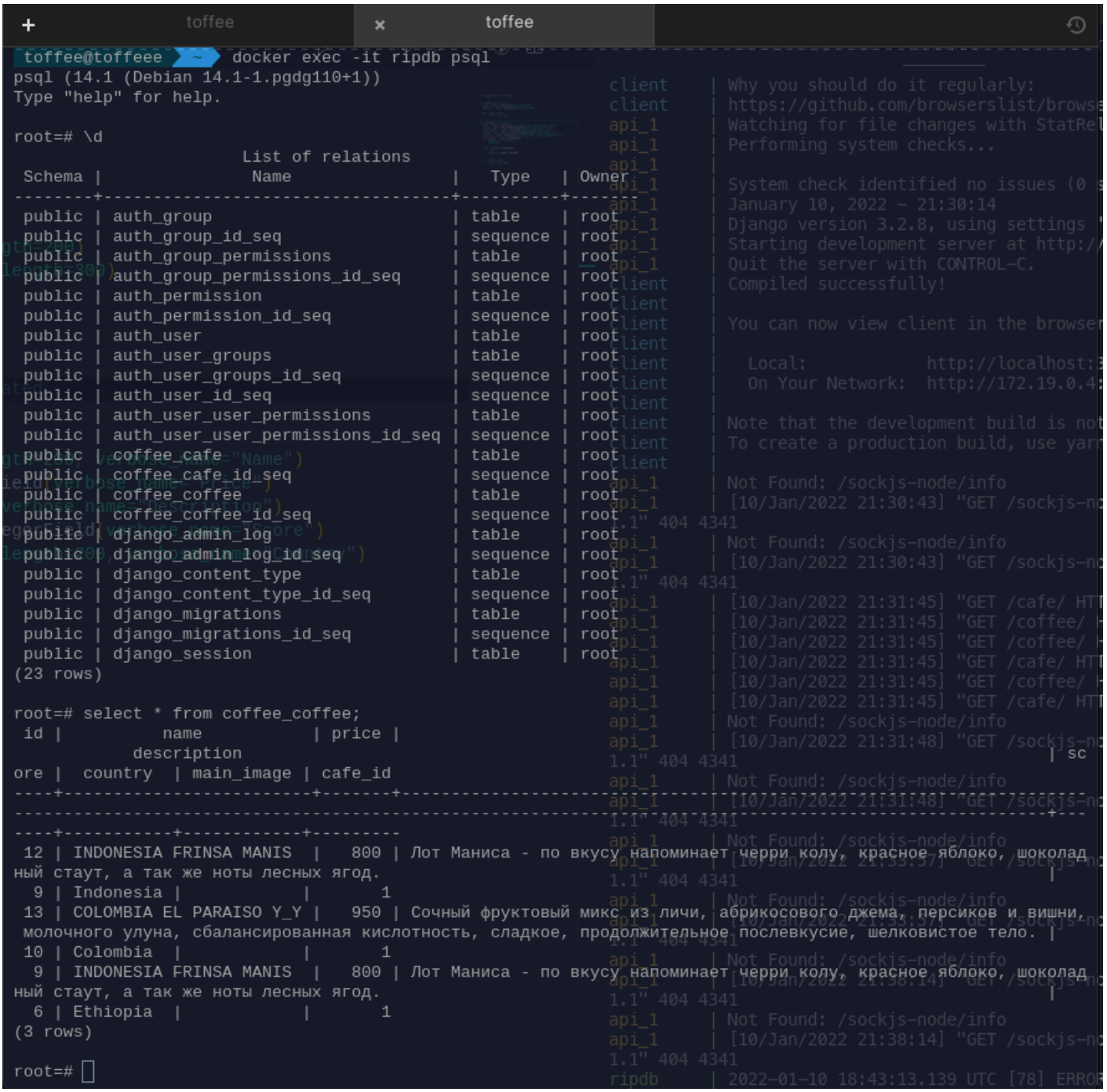
```
MEDIA_ROOT = BASE_DIR / 'media'
```

models.py



Код клиента для лабораторной работы №5 был написан с помощью библиотеки React. В качестве указанной в задании работы выполнен РК2 (код и отчёт находятся в соответствующей директории)

Работа программы:


toffee
toffee
toffee@toffeee ~ docker exec -it ripdb psql
psql (14.1 (Debian 14.1-1.pgdg110+1))
Type "help" for help.
root=# \d
List of relations
Schema | Name | Type | Owner
public | auth_group | table | root
public | auth_group_id_seq | sequence | root
public | auth_group_permissions | table | root
public | auth_group_permissions_id_seq | sequence | root
public | auth_permission | table | root
public | auth_permission_id_seq | sequence | root
public | auth_user | table | root
public | auth_user_groups | table | root
public | auth_user_groups_id_seq | sequence | root
public | auth_user_id_seq | sequence | root
public | auth_user_user_permissions | table | root
public | auth_user_user_permissions_id_seq | sequence | root
public | coffee_cafe | table | root
public | coffee_cafe_id_seq | sequence | root
public | coffee_coffee | table | root
public | coffee_coffee_id_seq | sequence | root
public | django_admin_log | table | root
public | django_admin_log_id_seq | sequence | root
public | django_content_type | table | root
public | django_content_type_id_seq | sequence | root
public | django_migrations | table | root
public | django_migrations_id_seq | sequence | root
public | django_session | table | root
(23 rows)
root=# select * from coffee_coffee;
id | name | price |
description
ore | country | main_image | cafe_id
12 | INDONESIA FRINSA MANIS | 800 | Лот Маниса - по вкусу напоминает черри колу, красное яблоко, шоколадный стаут, а так же ноты лесных ягод.
9 | Indonesia | | 1
13 | COLOMBIA EL PARAISO Y_Y | 950 | Сочный фруктовый микс из личи, абрикосового джема, персиков и вишни, молочного улуна, сбалансированная кислотность, сладкое, продолжительное послевкусие, шелковистое тело.
10 | Colombia | | 1
9 | INDONESIA FRINSA MANIS | 800 | Лот Маниса - по вкусу напоминает черри колу, красное яблоко, шоколадный стаут, а так же ноты лесных ягод.
6 | Ethiopia | | 1
(3 rows)
root=#
client | Why you should do it regularly:
client | https://github.com/browserslist/browse
api_1 | Watching for file changes with StatRel
api_1 | Performing system checks...
api_1 |
api_1 | System check identified no issues (0 s
api_1 | January 10, 2022 - 21:30:14
api_1 | Django version 3.2.8, using settings '
api_1 | Starting development server at http://
api_1 | Quit the server with CONTROL-C.
client | Compiled successfully!
client |
client | You can now view client in the browser
client |
client | Local: http://localhost:3
client | On Your Network: http://172.19.0.4:
client |
client | Note that the development build is not
client | To create a production build, use yarn
client |
api_1 | Not Found: /sockjs-node/info
api_1 | [10/Jan/2022 21:30:43] "GET /sockjs-no
1.1" 404 4341
api_1 | Not Found: /sockjs-node/info
api_1 | [10/Jan/2022 21:30:43] "GET /sockjs-no
1.1" 404 4341
api_1 | [10/Jan/2022 21:31:45] "GET /cafe/ HTT
api_1 | [10/Jan/2022 21:31:45] "GET /coffee/ H
api_1 | [10/Jan/2022 21:31:45] "GET /coffee/ H
api_1 | [10/Jan/2022 21:31:45] "GET /cafe/ HTT
api_1 | [10/Jan/2022 21:31:45] "GET /coffee/ H
api_1 | [10/Jan/2022 21:31:45] "GET /cafe/ HTT
api_1 | Not Found: /sockjs-node/info
api_1 | [10/Jan/2022 21:31:48] "GET /sockjs-no
1.1" 404 4341
api_1 | Not Found: /sockjs-node/info
api_1 | [10/Jan/2022 21:31:48] "GET /sockjs-no
1.1" 404 4341
api_1 | Not Found: /sockjs-node/info
api_1 | [10/Jan/2022 21:33:37] "GET /sockjs-no
1.1" 404 4341
api_1 | Not Found: /sockjs-node/info
api_1 | [10/Jan/2022 21:33:37] "GET /sockjs-no
1.1" 404 4341
api_1 | Not Found: /sockjs-node/info
api_1 | [10/Jan/2022 21:38:14] "GET /sockjs-no
1.1" 404 4341
api_1 | Not Found: /sockjs-node/info
api_1 | [10/Jan/2022 21:38:14] "GET /sockjs-no
1.1" 404 4341
ripdb | 2022-01-10 18:43:13.139 UTC [78] ERROR