

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

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

Отчет по рубежному контролю №2
по курсу «Разработка интернет-приложений».

Выполнил:
студент группы ИУ5-51Б
Плотников Ф.С.

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю. Е.

Москва, 2021 г

Задание

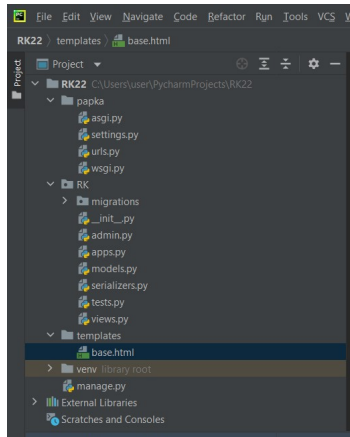
Рубежный контроль представляет собой разработку веб-приложения с использованием фреймворка Django. Веб-приложение должно выполнять следующие функции:

1. Создайте проект Python Django с использованием стандартных средств Django.
2. Создайте модель Django ORM, содержащую две сущности, связанные отношением один-ко-многим в соответствии с Вашим вариантом из условий рубежного контроля №1.
3. С использованием стандартного механизма Django сгенерируйте по модели макет веб-приложения, позволяющий добавлять, редактировать и удалять данные.
4. Создайте представление и шаблон, формирующий отчет, который содержит соединение данных из двух таблиц.

Вариант 19

№ варианта	Класс 1	Класс 2
19	Деталь	Производитель

Текст программы:



settings.py

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'RK',
        'USER': 'ЮзЯ',
        'PASSWORD': '',
        'HOST': 'localhost',
        'PORT': 3306,
        'OPTIONS': {'charset': 'utf8'},
        'TEST_CHARSET': 'utf8',
    }
}

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [BASE_DIR / 'templates'],
        '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',
            ],
        },
    ],
]

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',

    # DRF
    'rest_framework',
]
```

```
# Наше приложение
'RK',
]
```

urls.py

```
from django.contrib import admin
from django.urls import path
from RK import views

urlpatterns = [
    path('admin/', admin.site.urls),
    path("", views.CompanyDetails, name='CD'),
]
```

models.py

```
from django.db import models

# Create your models here.

class Companies(models.Model):
    id_company = models.AutoField(primary_key=True)
    name = models.CharField(max_length=30, blank=True, null=True)

    class Meta:
        managed = False
        db_table = 'companies'

class Details(models.Model):
    name = models.CharField(max_length=30, blank=True, null=True)
    id_company = models.ForeignKey(Companies, models.DO_NOTHING, db_column='id_company', blank=True, null=True)
    price = models.IntegerField(blank=True, null=True)
    quantity = models.IntegerField(blank=True, null=True)

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

serializers.py

```
from RK.models import Companies
from rest_framework import serializers

class StockSerializer(serializers.ModelSerializer):
    class Meta:
        # Модель, которую мы сериализуем
        model = Companies
        # Поля, которые мы сериализуем
        fields = ["name", "id_company"]
```

views.py

```
from django.shortcuts import render
from .models import Companies, Details

def CompanyDetails(request):
    details = Details.objects.all()
    companies = Companies.objects.all()
    context = {
        'details': details,
        'companies': companies
    }
    return render(request, 'base.html', context)
```

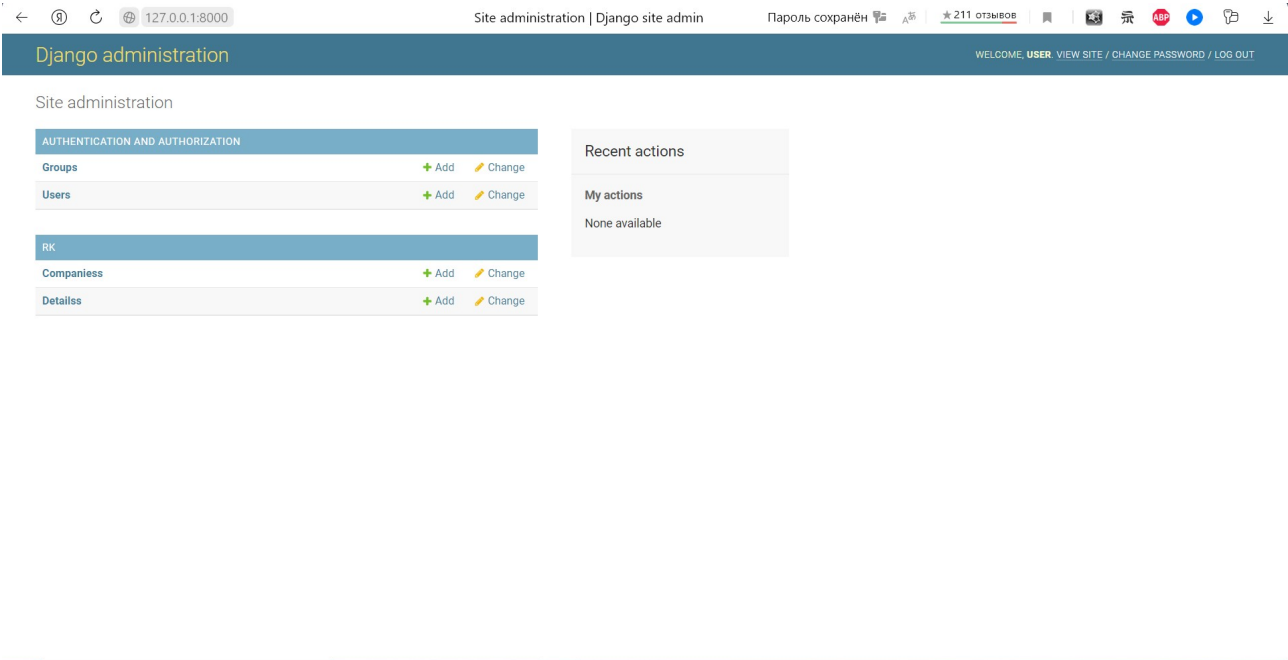
base.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Компании и детали</title>
</head>
<body>
    {%for company in companies%}
    <ul>
        <li>ID Компании: {{company.id_company}}
        <br>  Название компании: {{company.name}} </br>
        <ol>
            {%for detail in details%}
            {%if detail.id_company == company%}
            <li><br>ID Детали: {{detail.id}}</br>
            <br>Название детали: {{detail.name}}</br>
            <br>Цена: {{detail.price}} руб.</br>
            <br>Количество: {{detail.quantity}} шт.</br>
            </li>
            {%endif%}
            {%endfor%}
        </ol>
        </li>
    </ul>
    {%endfor%}
</body>
</html>
```

Результат:



Администрирование:



Добавили несколько записей:

← PyCharm

→

🔍

↻

🌐 127.0.0.1:8000

Компании и детали

★ 211 отзывов

🔍

🏠

📱

🔒

🔗

📄

⬇️

ID Детали: 4

Название детали: Шестерня

Цена: 345 руб.

Количество: 5 шт.

• ID Компании: 2

Название компании: Резьба деталей

1.

ID Детали: 3

Название детали: Винт

Цена: 150 руб.

Количество: 23 шт.

• ID Компании: 3

Название компании: Сарзип

1.

ID Детали: 2

Название детали: Гайка

Цена: 250 руб.

Количество: 56 шт.

• ID Компании: 4

Название компании: РосАгроПром

1.

ID Детали: 5

Название детали: Шуруп

Цена: 400 руб.

Количество: 12 шт.