

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

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

Курс «Разработка интернет приложений»

Рубежный контроль №2

Выполнил:

**студент группы ИУ5-53Б
Балабас Анна**

Проверил:

**преподаватель каф. ИУ5
Гапанюк Ю.Е.**

Москва, 2021 г.

Описание задания:

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.

Сущности: “Компьютер”, “Дисплей”.

models.py:

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

```
class PC(models.Model):
```

```
    id = models.IntegerField(primary_key=True)
```

```
    name = models.CharField(max_length=20)
```

```
    class Meta:
```

```
        db_table = 'pc'
```

```
class Display(models.Model):
```

```
    id = models.IntegerField(primary_key=True)
```

```
    matrix_type = models.TextField()
```

```
diagonal = models.FloatField()
```

```
pc = models.ForeignKey(PC, on_delete=models.PROTECT)
```

```
class Meta:
```

```
    db_table = 'display'
```

urls.py:

```
from rest_framework import routers
```

```
from django.urls import path, include
```

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

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

```
from computers import views as views
```

```
router = routers.DefaultRouter()
```

```
router.register('display', views.PCViewSet)
```

```
router.register('pc', views.DisplayViewSet)
```

```
urlpatterns = [
```

```
    path("", include(router.urls)),
```

```
    path('report/', views.report),
```

```
    path('api-auth/', include('rest_framework.urls', namespace='rest_framework')),
```

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

```
]
```

Settings.py

```
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  
    'stocks',  
]
```

views.py:

```
from django.shortcuts import render
```

```
from rest_framework import viewsets
```

```
from stocks.models import PC, Display
```

```
from stocks.serializers import PCSerializer, DisplaySerializer
```

```
class PCViewSet(viewsets.ModelViewSet):
```

```
    queryset = PC.objects.all()
```

```
serializer_class = PCSerializer
```

```
class DisplayViewSet(viewsets.ModelViewSet):
```

```
    queryset = Display.objects.all()
```

```
    serializer_class = DisplaySerializer
```

```
def report(request):
```

```
    return render(request, 'report.html', {'data': {
```

```
        'display': Display.objects.select_related('pc')
```

```
    })
```

Serializers.py

```
from rest_framework import serializers
```

```
from stocks.models import Display, PC
```

```
class DisplaySerializer(serializers.ModelSerializer):
```

```
    class Meta:
```

```
        model = Display
```

```
        fields = "__all__"
```

```
class PCSerializer(serializers.ModelSerializer):
```

```
    class Meta:
```

```
        model = PC
```

```
fields = "__all__"
```

base.html:

```
<!doctype html>
```

```
<html lang="en" class="h-100">
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<title>{% block title %}{% endblock %}</title>
```

```
</head>
```

```
<body>
```

```
{% block content %}{% endblock %}
```

```
</body>
```

```
</html>
```

Report.html

```
{% extends 'base.html' %}
```

```
{% block title %}Компьютер{% endblock %}
```

```
{% block content %}
```

```
<div>
```

```
<h1>Дисплей:</h1>
```

```
</div>
```

```
<ul>
```

```
{% for disp in data.display %}
```

```
<li>Матрица: <i>{{ disp.matrix_type }}</i></li>
```

Цена: <i>{{ disp.diagonal }}</i>

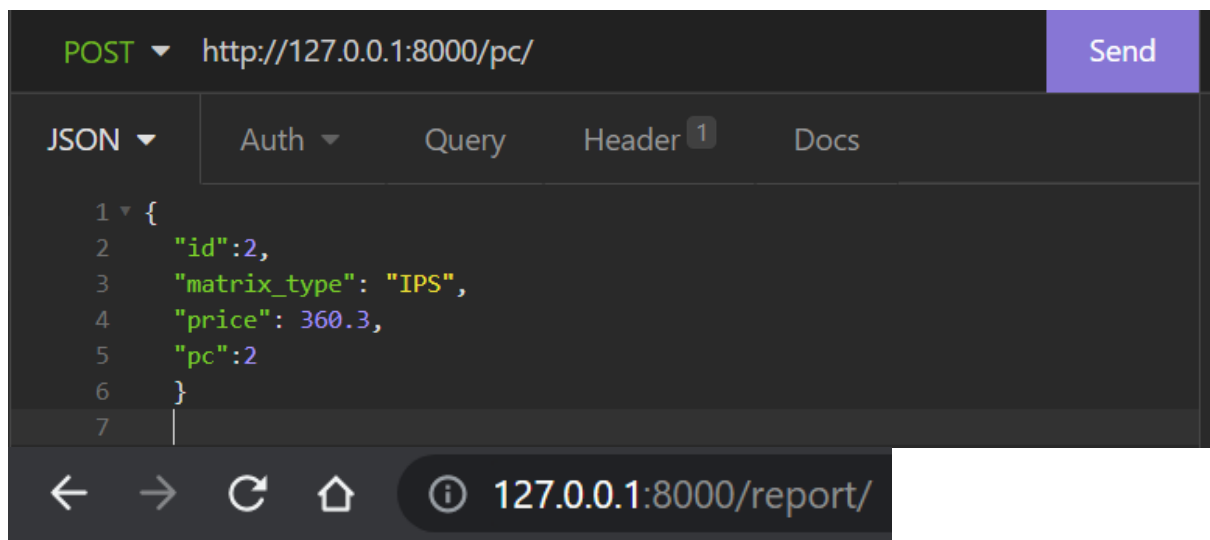
Модель компьютера: <i>{{ disp.pc.name }}</i>

{% empty %}

Список пуст

{% endfor %}

{% endblock %}



Дисплеи:

- Матрица: *TN*
- Цена: *15.3*
- Модель компьютера: *Huawei*

- Матрица: *IPS*
- Цена: *360.3*
- Модель компьютера: *Lenovo*