

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

Факультет «Информатика и системы управления»

Кафедра ИУ5. Курс «Разработка интернет-приложений»

Отчет по рубежному контролю №2

Выполнил:

студент группы ИУ5-51Б  
Абдрашитова Анжела Наилевна

Проверил:

преподаватель каф. ИУ5  
Гапанюк Юрий Евгеньевич

Подпись и дата:

Подпись и дата:

Москва, 2021 г.

## Описание задания:

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

№ варианта	Класс 1	Класс 2
1	Студент	Группа

## Текст программы:

*urls.py*

```
from django.contrib import admin
from django.urls import path
from RK_app.views import StudentsGroups

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', StudentsGroups, name='StudentsGroups'),
]
```

*views.py*

```
from django.shortcuts import render
from .models import Student, Group

def StudentsGroups(request):
    students = Student.objects.all()
    group = Group.objects.all()
    context = {
        'students': students,
        'group': group
    }
    return render(request, 'index.html', context)
```

### *models.py*

```
from django.db import models

class Group(models.Model):
    number = models.CharField(max_length=100)
    course = models.IntegerField()

class Student(models.Model):
    name = models.CharField(max_length=100)
    avg_mark = models.CharField(max_length=100)
    group = models.ForeignKey(Group, on_delete=models.CASCADE)
    def __str__(self):
        return self.name
```

### *admin.py*

```
from django.contrib import admin

from .models import Student, Group

admin.site.register(Student)
admin.site.register(Group)
```

### *index.html*

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Группы и студенты</title>
  </head>
  <body>
    {%for gr in group%}
    <ul>
      <li>ID Группы: {{gr.id}} <br>
        Номер группы: {{gr.number}}<br>
        Курс:{{gr.course}}<br>
      <ol>
        {%for stud in students%}
        {%if stud.group == gr%}
        <li>ID студента:{{stud.id}}<br>
          Имя:{{stud.name}}<br>
          Средняя оценка:{{stud.avg_mark}}<br>
        </li>
        {%endif%}
      {%endfor%}
      </ol>
    </li>
    </ul>
    {%endfor%}
```

```
</body>  
</html>
```

## Результат выполнения программы:

С помощью администрирования создадим несколько записей в БД:

### Django administration

#### Site administration

##### AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#) [Change](#)

Users [+ Add](#) [Change](#)

##### RK\_APP

Groups [+ Add](#) [Change](#)

Students [+ Add](#) [Change](#)

#### Recent actions

##### My actions

- + Илья  
Student
- + Ольга  
Student
- + Анжела  
Student
- + Виталий  
Student
- + Алексей  
Student
- + Group object (3)  
Group
- + Group object (2)  
Group
- + Group object (2)  
Group
- + Group object (1)  
Group

## Вывод записей:

- ID Группы: 1  
Номер группы: ИУ-51Б  
Курс:3
  - 1. ID студента:2  
Имя:Виталий  
Средняя оценка:5
  - 2. ID студента:3  
Имя:Анжела  
Средняя оценка:5
- ID Группы: 2  
Номер группы: ИУ-52Б  
Курс:3
  - 1. ID студента:1  
Имя:Алексей  
Средняя оценка:4,5
  - 2. ID студента:5  
Имя:Илья  
Средняя оценка:4,7
- ID Группы: 3  
Номер группы: ИУ6-32Б  
Курс:2
  - 1. ID студента:4  
Имя:Ольга  
Средняя оценка:4