Московский государственный технический университет им. Н.Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Автоматизированные системы обработки информации и управления»



**Отчет**

**Рубежный контроль № 2**

**По курсу «Разработка интернет приложений»**

**ИСПОЛНИТЕЛЬ:**

Эдилов Адам

Группа ИУ5-52

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

"\_\_"\_\_\_\_\_\_\_\_\_\_\_2019 г.

**ПРЕПОДАВАТЕЛЬ:**

Гапанюк Ю.Е.

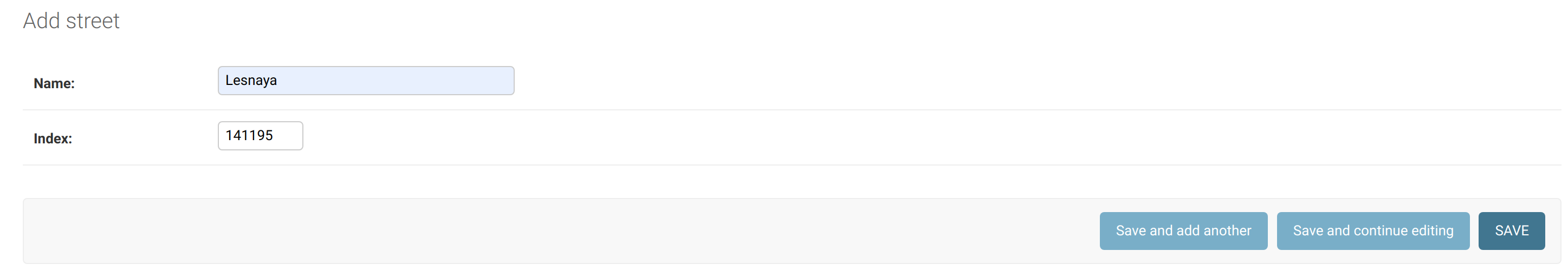
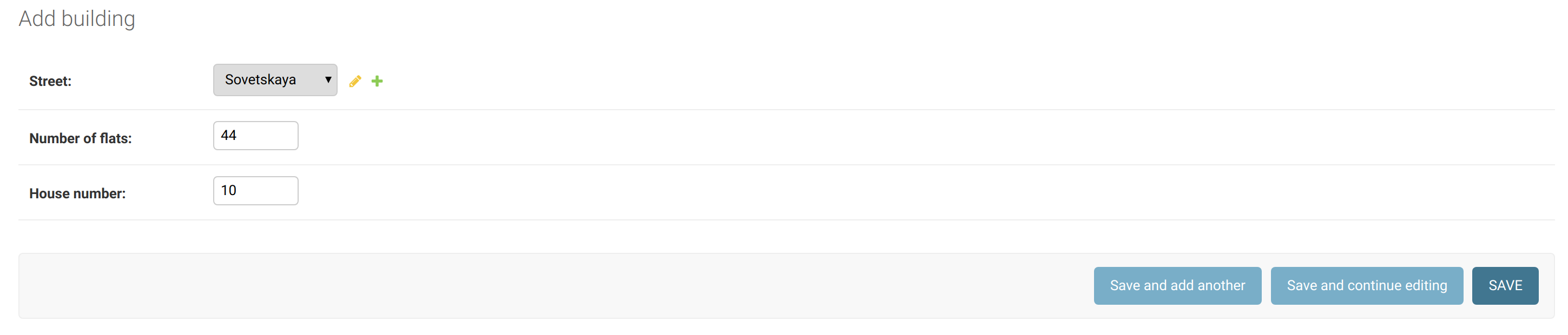
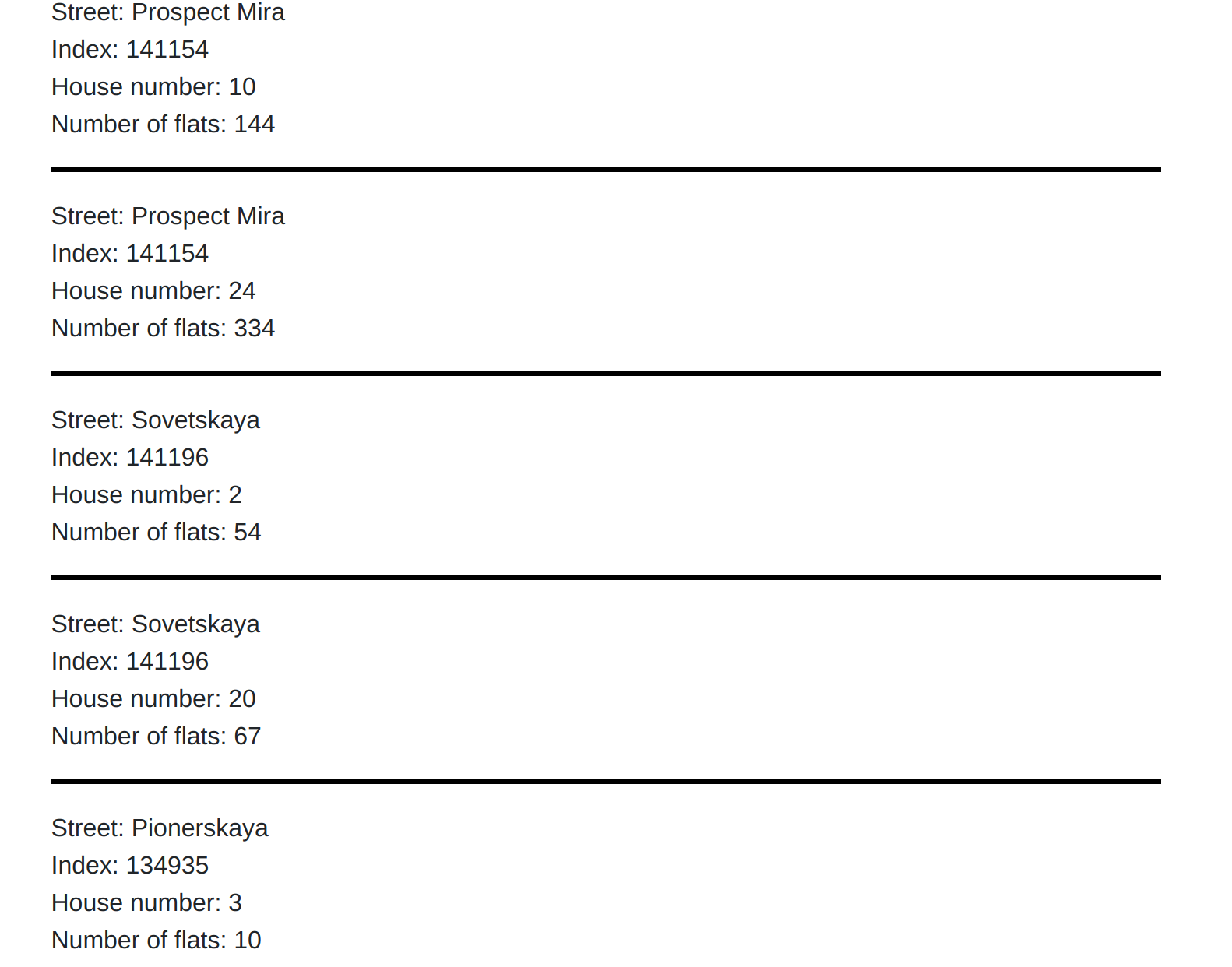
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

"\_\_"\_\_\_\_\_\_\_\_\_\_\_2019 г.

Москва 2019

1. **Задание**Рубежный контрольпредставляет собой разработку веб-приложения, которое должно выполнять следующие функции.
2. Создайте проект Python Django с использованием стандартных средств Django.
3. Создайте модель Django ORM, содержащую две сущности, связанные соотношением один-ко-многим в соответствии с вариантом из таблицы. По согласованию с преподавателем предметная область может быть изменена.
4. С использованием стандартного механизма Django сгенерируйте по модели макет веб-приложения, позволяющий добавлять, редактировать и удалять данные.
5. Создайте представление и шаблон, формирующий отчет, который содержит соединение данных из двух таблиц.

|  |  |  |
| --- | --- | --- |
| **Вариант** | **Объект 1** | **Объект 2** |
| 6 | Дом | Улица |

1. **Модель дома  
   class** Building(models.Model):  
    street = models.ForeignKey(**'Street'**, on\_delete=models.CASCADE)  
    number\_of\_flats = models.IntegerField(null=**False**)  
    house\_number = models.IntegerField(null=**False**)  
     
    **def** \_\_str\_\_(self):  
    **return "House number: {0}"**.format(self.house\_number);
2. **Модель улицы  
   class** Street(models.Model):  
    name = models.CharField(max\_length=128, null=**False**, unique=**True**)  
    index = models.IntegerField(null=**False**)  
     
    **def** \_\_str\_\_(self):  
    **return** self.name
3. **Добавление улицы**
4. **Добавление дома**
5. **View для отображения данных  
   def** buildings(request):  
    context = {  
    **'buildings'**: Building.objects.all()  
    }  
    **return** render(request, **'buildings.html'**, context)
6. **Шаблон**{% **extends "extends/base.html"** %}  
     
   {% **block content** %}  
     
    {% **for building in buildings** %}  
    <**div**>  
    Street: {{ **building**.**street**.**name** }}  
    </**div**>  
    <**div**>  
    Index: {{ **building**.**street**.**index** }}  
    </**div**>  
    <**div**>  
    House number: {{ **building**.**house\_number** }}  
    </**div**>  
    <**div**>  
    Number of flats: {{ **building**.**number\_of\_flats** }}  
    </**div**>  
     
    <**hr style="height**: 2**px**; **background-color**: **black"**>  
    {% **endfor** %}  
     
   {% **endblock** %}
7. **Пример работы**