



**Министерство науки и высшего образования Российской
Федерации Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

**Факультет «Радиотехнический»
Кафедра ИУ5 «Системы обработки информации и управления»**

**Рубежный контроль №2
по дисциплине «Разработка интернет-приложений»**

**Выполнил:
студент группы РТ5-51Б
В. И. Плешаков**

**Проверил:
к. т. н., доцент кафедры ИУ5
Ю. Е. Гапанюк**

2021 г.

Модели:

```
from django.db import models
from django.urls import reverse

class CD(models.Model):
    """
    Модель CD
    """
    name = models.CharField(max_length=50, default='')
    capacity = models.IntegerField(default=100)
    lib = models.ForeignKey('Lib', on_delete=models.CASCADE)
    cover = models.ImageField(upload_to='media/')

    def __str__(self):
        return self.name

    def get_absolute_url(self):
        return reverse('CDLib:detail', args=[str(self.id)])

class Lib(models.Model):
    """
    Модель библиотеки
    """
    name = models.CharField(max_length=50, default='')

    def __str__(self):
        return self.name

    def get_absolute_url(self):
        return reverse('CDLib:report')
```

Представления:

```
from django.shortcuts import render
from .models import CD, Lib
from django.views.generic import UpdateView, DeleteView, CreateView, DetailView
from django.urls import reverse_lazy

def report(request):
    queryset = CD.objects.select_related('lib').all().order_by('name').order_by('lib') # select * from
    disk_lib = {cds.lib: [cd for cd in queryset if (cds.lib.name == cd.lib.name)] for cds in queryset}

    return render(request, 'report.html', {'data': disk_lib})

class DetailCD(DetailView):
    model = CD
    template_name = 'detail.html'

class DeleteLib>DeleteView):
    model = Lib
    template_name = 'DeleteObj.html'
    success_url = reverse_lazy('CDLib:report')

class DeleteCD>DeleteView):
    model = CD
    template_name = 'DeleteObj.html'
    success_url = reverse_lazy('CDLib:report')

class UpdateLib>UpdateView):
    model = Lib
    template_name = 'UpdateObj.html'
    fields = ['name']
```

```
class UpdateCD>UpdateView):
    model = CD
    template_name = 'UpdateObj.html'
    fields = ['name', 'capacity', 'lib']

class CreateLib>CreateView):
    model = Lib
    template_name = 'CreateObj.html'
    fields = ['name']

class CreateCD>CreateView):
    model = CD
    template_name = 'CreateObj.html'
    fields = ['name', 'capacity', 'lib']
```

Пример работы программы:

Отчет:

Здесь можно найти много разных дисков!

Создать диск

Создать раздел

Раздел A	Изменить	Удалить	
aa, 100 Mб	Просмотр	Изменить	Удалить
ab, 100 Mб	Просмотр	Изменить	Удалить
aaa, 100 Mб	Просмотр	Изменить	Удалить
aaaa, 100 Mб	Просмотр	Изменить	Удалить
Раздел B	Изменить	Удалить	
bb, 100 Mб	Просмотр	Изменить	Удалить
bab, 100 Mб	Просмотр	Изменить	Удалить
baba, 100 Mб	Просмотр	Изменить	Удалить
bbbbbbbb, 100 Mб	Просмотр	Изменить	Удалить
Раздел C	Изменить	Удалить	
ca, 100 Mб	Просмотр	Изменить	Удалить
cb, 100 Mб	Просмотр	Изменить	Удалить

Создание диска:

Name:

Capacity:

100

Lib:

----- ▾

Cover:

Выберите файл

Файл не выбран

Save

Создание библиотеки:

Name:

Детальная информация о диске:



aa
100

Изменение диска:

Редактировать сведения

Name:

Capacity:

Lib: ▼

Cover: Currently: [media/test.jpg](#)

Change: Файл не выбран

Изменение библиотеки:

Редактировать сведения

Name: