

Отчет по РК № 2 по курсу
"Разработка Интернет-Приложений"

Выполнил:
Студент группы
ИУ5-53Б
Кащеев Максим

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.

№ варианта	Класс 1	Класс 2
10	Компьютер	Браузер

Текст программы:

```
# Generated by Django 3.2.9 on 2021-12-14 21:31

from django.db import migrations, models
import django.db.models.deletion

class Migration(migrations.Migration):

    initial = True

    dependencies = [
    ]

    operations = [
        migrations.CreateModel(
            name='Computer',
            fields=[
                ('id', models.BigAutoField(auto_created=True, primary_key=True, serialize=False, verbose_name='ID')),
                ('name', models.CharField(max_length=50, verbose_name='Название браузера')),
            ],
        ),
        migrations.CreateModel(
            name='CD',
            fields=[
                ('id', models.BigAutoField(auto_created=True, primary_key=True, serialize=False, verbose_name='ID')),
                ('name', models.CharField(max_length=50, verbose_name='Название компа')),
                ('capacity', models.CharField(max_length=50, verbose_name='Ёмкость')),
                ('cdLib', models.ForeignKey(null=True, on_delete=django.db.models.deletion.CASCADE, to='cd.cd_lib')),
            ],
        ),
    ]
```

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>edit disk</title>
</head>
<body class="container">
  <form method="POST">
    <{{ csrf_token }}>
    <p>
      <label>Введите название компьютера</label><br>
      <input type="text" name="name" value="{{disk.name}}" />
    </p>
    <p>
      <label>Введите ёмкость компьютера</label><br>
      <input type="text" name="capacity" value="{{disk.capacity}}" />
    </p>
    <p>
      <label>Введите название браузера</label><br>
      <input type="text" name="cdLib" value="{{disk.cdLib.name}}" />
    </p>
    <input type="submit" value="Сохранить" >
  </form>
</body>
</html>

```

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>pk2</title>
</head>
<body class="container">
  <form method="POST" action="create_disk/">
    <{{ csrf_token }}>

    <h2>Добавление нового компьютера</h2>
    <p>
      <label>Введите название компьютера</label><br>
      <input type="text" name="name" />
    </p>
    <p>
      <label>Введите ёмкость памяти</label><br>
      <input type="text" name="capacity" />
    </p>
    <p>
      <label>Введите название браузера</label><br>
      <input type="text" name="cdLib" />
    </p>
    <input type="submit" value="Сохранить" >
  </form>

  <form method="POST" action="create/">
    <{{ csrf_token }}>

    <h2>Добавление нового браузера</h2>
    <p>
      <label>Введите название браузера</label><br>
      <input type="text" name="name_lib" />
    </p>
  </form>

```

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <title>edit lib</title>
</head>
<body class="container">
    <form method="POST">
        <input type="hidden" value="{% csrf_token %}" />
        <p>
            <label>Введите название браузера</label><br>
            <input type="text" name="name" value="{{lib.name}}" />
        </p>
        <input type="submit" value="Сохранить" />
    </form>
</body>
</html>

```

```

from django.shortcuts import render
from django.http import HttpResponseRedirect
from django.http import HttpResponseNotFound
from cd.models import CD
from cd.models import CD_lib

def index(request):
    libs = CD_lib.objects.all()
    disks = CD.objects.all()
    return render(request, "index.html", {"libs": libs, "disks": disks})

# сохранение данных в бд
def create(request):
    if request.method == "POST":
        dl = CD_lib()
        dl.name = request.POST.get("name_lib")
        dl.save()
    return HttpResponseRedirect("/")

def create_disk(request):
    if request.method == "POST":
        d = CD()
        d.name = request.POST.get("name")
        d.capacity = request.POST.get("capacity")
        d.cdLib = CD_lib.objects.get(name=request.POST.get("cdLib"))
        d.save()
    return HttpResponseRedirect("/")

# изменение данных в бд
def edit(request, id):

```

```

def edit_lib(request, id):
    try:
        lib = CD_lib.objects.get(id=id)

        if request.method == "POST":
            lib.name = request.POST.get("name")
            lib.save()
            return HttpResponseRedirect("/")
        else:
            return render(request, "edit_lib.html", {"lib": lib})
    except CD_lib.DoesNotExist:
        return HttpResponseRedirect("<h2>Library not found</h2>")

# удаление данных из бд
def delete(request, id):
    try:
        disk = CD.objects.get(id=id)
        disk.delete()
        return HttpResponseRedirect("/")
    except CD.DoesNotExist:
        return HttpResponseRedirect("<h2>Disk not found</h2>")

def delete_lib(request, id):
    try:
        lib = CD_lib.objects.get(id=id)
        lib.delete()
        return HttpResponseRedirect("/")
    except CD.DoesNotExist:

```

Результат работы:

Введите ёмкость памяти

Введите название браузера

Сохранить

Добавление нового браузера

Введите название браузера

Сохранить

Список браузеров

Название

Opera [Изменить](#) | [Удалить](#)

Chrome [Изменить](#) | [Удалить](#)

Yandex [Изменить](#) | [Удалить](#)

Explorer [Изменить](#) | [Удалить](#)

Список компьютеров

Название Ёмкость Браузер

Maxim 22 Opera [Изменить](#) | [Удалить](#)

Masha 33 Yandex [Изменить](#) | [Удалить](#)

max 333 Chrome [Изменить](#) | [Удалить](#)

Компьютер в браузере Opera:

- Maxim

Компьютер в браузере Chrome:

- max

Компьютер в браузере Yandex:

- Masha

Компьютер в браузере Explorer: