

**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 ИУ5-52Б
Барышников Михаил

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: “Автопарк”, “Водитель”.

RK2/main/models.py:

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

```
class Driver(models.Model):
    """Водитель"""
    id = models.BigAutoField(db_column='ID', primary_key=True)
    fio = models.CharField(db_column='Fio', max_length=150, blank=True)
    sal = models.IntegerField(db_column='Salary', blank=True, null=True)
    park_id = models.ForeignKey('CarPark', models.DO_NOTHING,
db_column='Park_ID')

    class Meta:
        db_table = 'Driver'

    def __str__(self):
        return f'{self.fio}'

class CarPark(models.Model):
    """Автопарк"""
    id = models.BigAutoField(db_column='ID', primary_key=True)
    name = models.CharField(db_column='Name', max_length=100, blank=True)

    class Meta:
```

```

        db_table = 'CarPark'

    def __str__(self):
        return f'{self.name}'

class DriverPark(models.Model):
    """Водители автопарка"""
    id = models.BigAutoField(db_column='ID', primary_key=True)
    park_id = models.ForeignKey('CarPark', models.DO_NOTHING,
db_column='Park_ID')
    driver_id = models.ForeignKey('Driver', models.DO_NOTHING,
db_column='Driver_ID')

    class Meta:
        db_table = 'DriverPark'

```

RK2/RK2/urls.py:

```

from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('main.urls'))
]

```

RK2/main/urls.py:

```

from django.urls import path
from . import views

urlpatterns = [
    path('', views.index),
    path('carpark', views.park_list),
    path('carpark/create', views.CarParkCreate.as_view()),
    path('carpark/<int:id>/update', views.CarParkUpdate.as_view(),
name='park_update'),
    path('carpark/<int:id>/delete', views.CarParkDelete.as_view(),
name='park_delete'),
    path('driver', views.driver_list),

```

```

        path('driver/create', views.DriverCreate.as_view()),
        path('driver/<int:id>/update', views.DriverUpdate.as_view(),
name='driver_update'),
        path('driver/<int:id>/delete', views.DriverDelete.as_view(),
name='driver_delete'),
        path('report', views.report)
]

```

RK2/main_app/views.py:

```

from django.shortcuts import render
from .models import *
from django.views.generic.edit import CreateView, UpdateView, DeleteView
from django import forms
from datetime import datetime

def index(request):
    return render(request, 'main/index.html')

def report(request):
    drivers = Driver.objects.all()
    params = {'drivers': drivers, 'date': datetime.now().strftime("%Y-%m-%d %H:%M:%S")}
    return render(request, 'main/report.html', params)

def driver_list(request):
    drivers = Driver.objects.all().values()
    params = {'entity': 'Driver', 'objects': drivers}
    return render(request, 'main/list.html', params)

def park_list(request):
    parks = CarPark.objects.all().values()
    params = {'entity': 'CarPark', 'objects': parks}
    return render(request, 'main/list.html', params)

class CarParkCreate(CreateView):

```

```
model = CarPark
fields = ['name']
success_url = '/carpark'
template_name = 'main/park_form.html'
```

```
class CarParkUpdate(UpdateView):
    model = CarPark
    fields = ['name']
    pk_url_kwarg = 'id'
    success_url = '/carpark'
    template_name = 'main/park_form.html'
```

```
class CarParkDelete(DeleteView):
    model = CarPark
    pk_url_kwarg = 'id'
    success_url = '/carpark'
    template_name = 'main/park_delete_form.html'
```

```
class DriverCreate(CreateView):
    model = Driver
    fields = ['fio', 'sal', 'park_id']
    success_url = '/driver'
    template_name = 'main/driver_form.html'

    def get_context_data(self, **kwargs):
        context = super(DriverCreate, self).get_context_data(**kwargs)
        context['form'].fields['park_id'] =
forms.ModelFormChoiceField(queryset=CarPark.objects.all(),
empty_label=None, label='Автопарк')
        return context
```

```
class DriverUpdate(UpdateView):
    model = Driver
    fields = ['fio', 'sal', 'park_id']
    pk_url_kwarg = 'id'
    success_url = '/driver'
    template_name = 'main/driver_form.html'
```

```

    def get_context_data(self, **kwargs):
        context = super(DriverUpdate, self).get_context_data(**kwargs)
        context['form'].fields['park_id'] =
forms.ModelFormChoiceField(queryset=CarPark.objects.all(),

empty_label=None, label='Автопарк')
        return context

class DriverDelete(DeleteView):
    model = Driver
    pk_url_kwarg = 'id'
    success_url = '/driver'
    template_name = 'main/driver_delete_form.html'

```

RK2/main/templates/main/base.html:

```

<!DOCTYPE html>
<html>
<head>
    <meta charset='utf-8'>
    <title>
        {% block title %} {% endblock %}
    </title>
    <link rel='stylesheet' type='text/css' media='screen'>
</head>
<body>
    <div>
        <a style='margin: 5px;' href='/'>Главная</a>
        <a style='margin: 5px;' href='/carpark'>Автопарки</a>
        <a style='margin: 5px;' href='/driver'>Водители</a>
        <a style='margin: 5px;' href='/report'>Отчёт</a>
    </div>
    {% block content %} {% endblock %}
</body>
</html>

```

RK2/main/templates/main/index.html:

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

{% block title %}
Главная
{% endblock %}

{% block content %}
<h2>Главная страница</h2>
<p>Рк №2, вариант №3, ПО: Автопарки</p>
<p>Работу выполнил:</p>
<p>Барышников Михаил Игоревич</p>
<p>Группа: ИУ5-52Б</p>
{% endblock %}
```

RK2/main/templates/main/list.html:

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

{% block title %}
{{entity}}
{% endblock %}

{% block content %}
<h2>Сущность <i>{{entity}}</i></h2>

{% if entity == 'Driver' %}
<a href='/driver/create'>Добавить элемент</a>
{% endif %}

{% if entity == 'CarPark' %}
<a href='/carpark/create'>Добавить элемент</a>
{% endif %}

<p>Список элементов:</p>
<ul>
    {% for elem in objects %}
    <hr>
    <li>
        {% if entity == 'Driver' %}
        ФИО: {{elem.fio}}, зарплата: {{elem.sal}}<br>
        <a href='{% url "driver_update" elem.id %}'>Изменить</a>
        <a href='{% url "driver_delete" elem.id %}'>Удалить</a>
        {% endif %}

        {% if entity == 'CarPark' %}
```

```

        Наименование: {{elem.name}} <br>
        <a href='{% url "park_update" elem.id %}'>Изменить</a>
        <a href='{% url "park_delete" elem.id %}'>Удалить</a>
    {% endif %}
</li>
{% endfor %}
<hr>
</ul>
{% endblock %}

```

RK2/main/templates/main/report.html:

```

{% extends 'main/base.html' %}

{% block title %}
Водители
{% endblock %}

{% block content %}
<h2>Список всех водителей:</h2>
<ul>
    {% for driver in drivers %}
        <hr>
        <li>
            <div>
                ФИО: {{driver.fio}} <br>
                Зарплата: {{driver.sal}} p. <br>
                Автопарк: {{driver.park_id}}
            </div>
        </li>
    {% endfor %}
<hr>
</ul>
<i>Время формирования отчёта: </i>{{date}}
{% endblock %}

```

RK2/main/templates/main/park_form.html:

```

{% extends "main/base.html" %}

{% block title %}

```


Изменение

{% endblock %}

{% block content %}

<h2>Параметры записи</h2>

<form action="" method="post">

{% csrf_token %}

<table>

{{ form.as_table }}

</table>

<input type="submit" value="Сохранить" />

</form>

{% endblock %}

RK2/main/templates/main/park_delete_form.html:

{% extends "main/base.html" %}

{% block content %}

<h2>Удаление записи</h2>

<p>Вы уверены, что хотите удалить автопарк {{object.name}}?</p>

<form action="" method="POST">

{% csrf_token %}

<input type="submit" value="Удалить" />

</form>

{% endblock %}

RK2/main/templates/main/driver_form.html:

{% extends "main/base.html" %}

{% block title %}

Изменение

{% endblock %}

```
{% block content %}

<h2>Параметры записи</h2>
<form action="" method="post">
    {% csrf_token %}
    <table>
        {{ form.as_table }}
    </table>
    <br>
    <input type="submit" value="Сохранить" />
</form>

{{ class_text.name }}

{% endblock %}
```

RK2/main/templates/main/driver_delete_form.html:

```
{% extends "main/base.html" %}

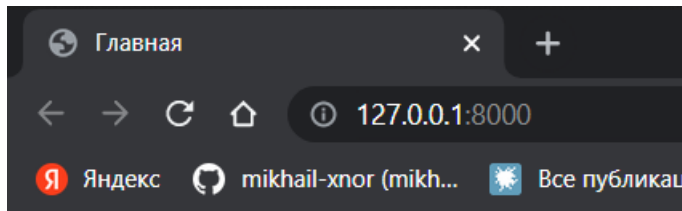
{% block content %}

<h2>Удаление записи</h2>

<p>Вы уверены, что хотите удалить водителя <i>{{ object.fio }}</i>?</p>

<form action="" method="POST">
    {% csrf_token %}
    <input type="submit" value="Удалить" />
</form>
{% endblock %}
```

Скриншоты веб-приложения:



[Главная](#) [Автопарки](#) [Водители](#) [Отчёт](#)

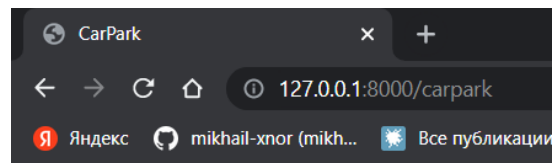
Главная страница

Рк №2, вариант №3, ПО: Автопарки

Работу выполнил:

Барышников Михаил Игоревич

Группа: ИУ5-52Б



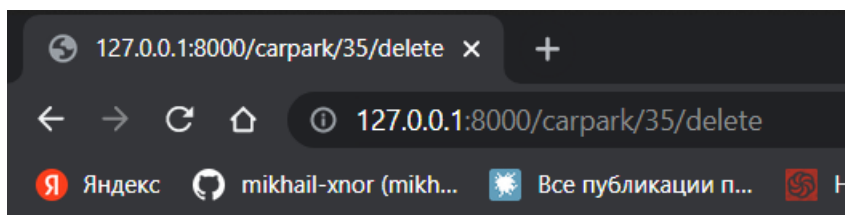
[Главная](#) [Автопарки](#) [Водители](#) [Отчёт](#)

Сущность *CarPark*

[Добавить элемент](#)

Список элементов:

- Наименование: Люблинский
[Изменить](#) [Удалить](#)
- Наименование: Дмитровский
[Изменить](#) [Удалить](#)
- Наименование: Марьинский
[Изменить](#) [Удалить](#)
- Наименование: Чертановский
[Изменить](#) [Удалить](#)
- Наименование: Нахабинский
[Изменить](#) [Удалить](#)
- Наименование: Лобнинский
[Изменить](#) [Удалить](#)
- Наименование: Дмитровский
[Изменить](#) [Удалить](#)
- Наименование: Волоколамский
[Изменить](#) [Удалить](#)



[Главная](#) [Автопарки](#) [Водители](#) [Отчёт](#)

Удаление записи

Вы уверены, что хотите удалить автопарк Волоколамский?

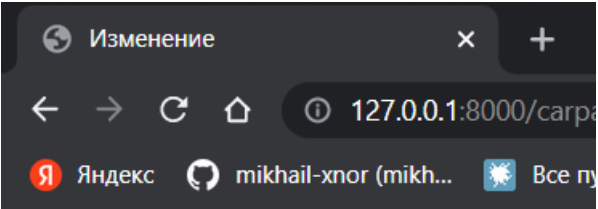
[Удалить](#)

Сущность *CarPark*

[Добавить элемент](#)

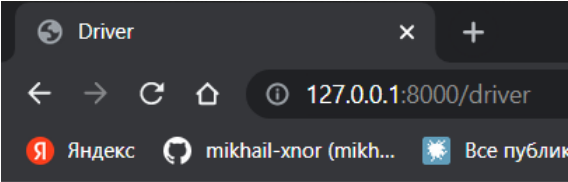
Список элементов:

- Наименование: Люблинский
[Изменить](#) [Удалить](#)
- Наименование: Дмитровский
[Изменить](#) [Удалить](#)
- Наименование: Марьинский
[Изменить](#) [Удалить](#)
- Наименование: Чертановский
[Изменить](#) [Удалить](#)
- Наименование: Нахабинский
[Изменить](#) [Удалить](#)
- Наименование: Лобнинский
[Изменить](#) [Удалить](#)
- Наименование: Дмитровский
[Изменить](#) [Удалить](#)



Параметры записи

Name:



[Главная](#) [Автопарки](#) [Водители](#) [Отчёт](#)

Сущность *Driver*

[Добавить элемент](#)

Список элементов:

- ФИО: Артамонов, зарплата: 35000
[Изменить](#) [Удалить](#)
- ФИО: Петров, зарплата: 40000
[Изменить](#) [Удалить](#)
- ФИО: Иваненко, зарплата: 37500
[Изменить](#) [Удалить](#)
- ФИО: Иванов, зарплата: 30000
[Изменить](#) [Удалить](#)
- ФИО: Иванов, зарплата: 40000
[Изменить](#) [Удалить](#)

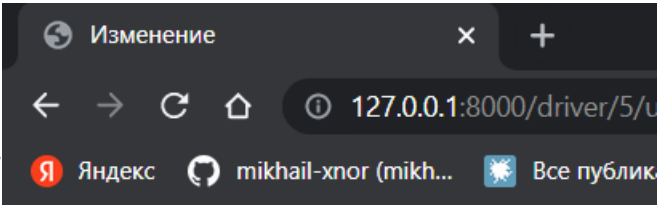
[Главная](#) [Автопарки](#) [Водители](#) [Отчёт](#)

Сущность *Driver*

[Добавить элемент](#)

Список элементов:

- ФИО: Артамонов, зарплата: 35000
[Изменить](#) [Удалить](#)
- ФИО: Петров, зарплата: 40000
[Изменить](#) [Удалить](#)
- ФИО: Иваненко, зарплата: 37500
[Изменить](#) [Удалить](#)
- ФИО: Иванов, зарплата: 30000
[Изменить](#) [Удалить](#)
- ФИО: Иванов, зарплата: 88888
[Изменить](#) [Удалить](#)



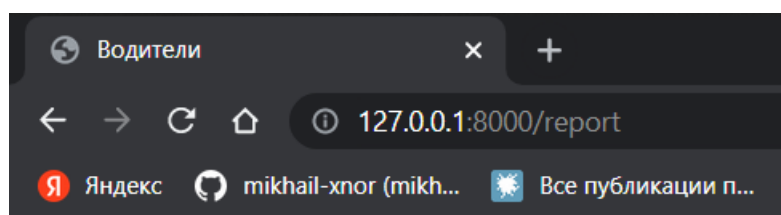
[Главная](#) [Автопарки](#) [Водители](#) [Отчёт](#)

Параметры записи

Фio:

Sal:

Автопарк:



[Главная](#) [Автопарки](#) [Водители](#) [Отчёт](#)

Список всех водителей:

-
- ФИО: Артамонов
Зарплата: 35000 р.
Автопарк: Люблинский
 - ФИО: Петров
Зарплата: 40000 р.
Автопарк: Дмитровский
 - ФИО: Иваненко
Зарплата: 37500 р.
Автопарк: Марьинский
 - ФИО: Иванов
Зарплата: 30000 р.
Автопарк: Марьинский
 - ФИО: Иванин
Зарплата: 888888 р.
Автопарк: Марьинский
-

Время формирования отчёта: 2021-12-15 19:33:55

