

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
ИТМО**

**Дисциплина:** Основы веб-программирования

Отчет по лабораторной работе №1  
Вариант 3

Создание сайта на Django

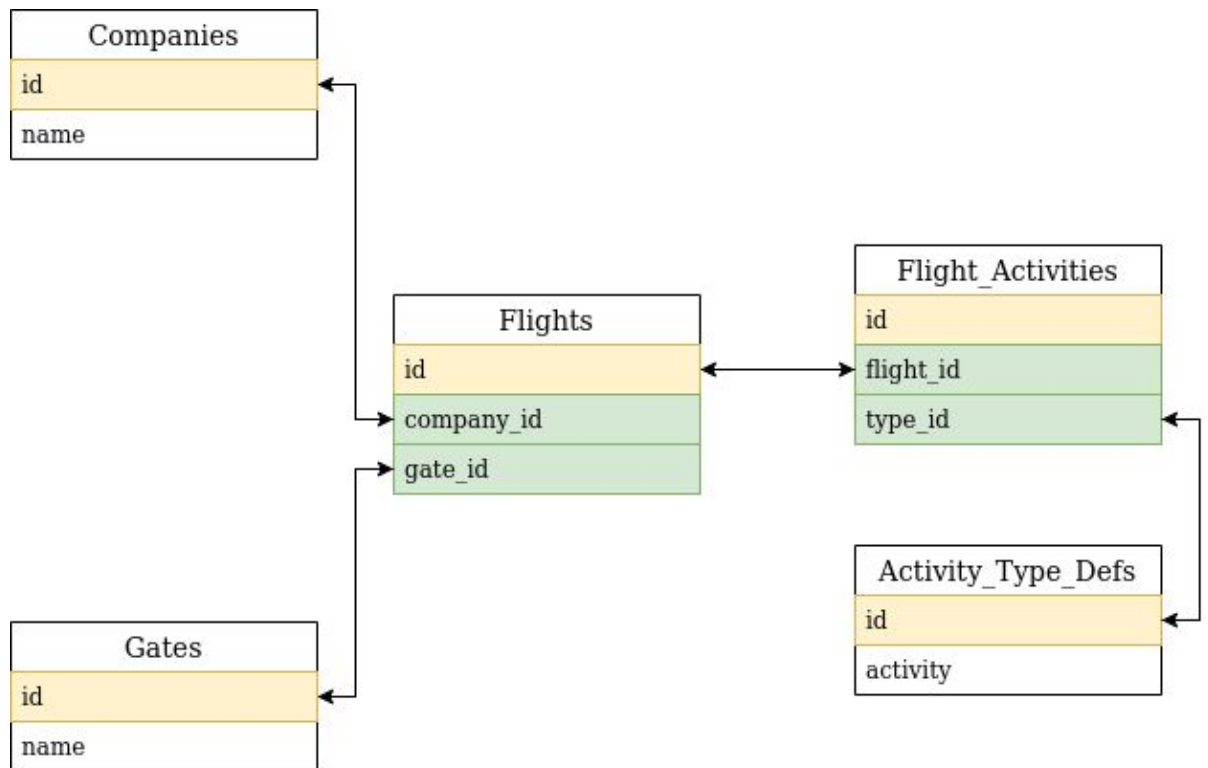
Выполнила:  
Бережнова Марина  
Группа: К3343

Проверил:  
Говоров А. И.

2020 г.

Выполнение работы:

1. Для начала я решила отрисовать модель данных:



Максимальное распределение всех данных по сущностям несколько усложнит процесс разработки, зато всё будет изолировано друг от друга и понятно даже неподготовленному смотрящему.

Как можно понять, главной сущностью я выбрала таблицу Flights, в которой содержится основная информация по авиаперелетам.

## 2. models.py:

```
from django.db import models

# Create your models here.
class Companies(models.Model):
    name = models.CharField(max_length=30)

    def __str__(self):
        return "{}".format(self.name)

class Gates(models.Model):
    name = models.CharField(max_length=30)

    def __str__(self):
        return "{}".format(self.name)

class Flights(models.Model):
    company = models.ForeignKey(Companies, on_delete=models.CASCADE)
    gate = models.ForeignKey(Gates, on_delete=models.CASCADE)

    def __str__(self):
        return "Company: {} | Gate: {}".format(self.company, self.gate)

class FlightActivities(models.Model):
    ACTIVITY = [
        ('0', 'arrival'),
        ('1', 'departure')
    ]

    flight = models.ForeignKey(Flights, on_delete=models.CASCADE)
    activity = models.CharField(choices=ACTIVITY, default='0', max_length=1)
    time = models.DateField()

    def __str__(self):
        return "{} | Arrival/departure: {} | Date {}".format(self.flight, self.get_activity_display(), self.time)
```

Сразу добавим в admin.py.

## 3. Теперь нужно создать суперпользователя, чтобы можно было добавлять данные, через админку:

py manage.py createsuperuser

## 4. Проверяю, что всё создалось:



Всё ок.

5. Теперь создадим view для вывода всех данных по тому или иному полёту.

```
from django.shortcuts import render
from .models import *
from django.http import Http404, HttpResponseRedirect
from django.views.generic.list import ListView

# Create your views here.

class FlightView(ListView):
    model = FlightActivities

    def get(self, request):
        context = {}

        try:
            context["flights"] = FlightActivities.objects.all()
        except Flight.DoesNotExist:
            raise Http404("Flight does not exist")

        return render(request, 'flight_list.html', context)

def show_flight(request, flight_id):
    try:
        flight = FlightActivities.objects.get(pk=flight_id)
    except Flight.DoesNotExist:
        raise Http404("Flight does not exist")

    return render(request, 'flight.html', {'flight': flight})
```

Я уже сделала для них простые шаблоны, теперь осталось добавить форму и связать это всё вместе. Для начала надо добавить модель для комментариев.

6. Она получилась вот такой:

```
class FlightComments(models.Model):
    flight = models.ForeignKey(FlightActivities, on_delete=models.CASCADE)

    COMMENT_TYPE = [
        ('0', 'Gate changing'),
        ('1', 'Lateness'),
        ('2', 'Other')
    ]

    com_type = models.CharField(choices=COMMENT_TYPE, default='0', max_length=1)
    com_text = models.CharField(max_length=1024)
    author = models.EmailField(max_length=254)
```

## 7. Страница с комментариями по полёту:

### Flight:

[Go back](#)

Company: S7 | Gate: Pulkovo1 | Arrival/departure: arrival | Date 2020-04-17

**Author:** [marinabe@gmail.com](mailto:marinabe@gmail.com)

**Comment type:** Gate changing

**Comment text:**

Gate is changing

**Author:** [marinabe@gmail.com](mailto:marinabe@gmail.com)

**Comment type:** Gate changing

**Comment text:**

Gate is changing

• This field is required.

Author:

• This field is required.

Com type:

• This field is required.

Com text:

## 8. Все полёты:

### Flights:

Just click on flight for adding your comment

[Company: S7](#) | [Gate: Pulkovo1](#) | [Arrival/departure: arrival](#) | [Date 2020-04-17](#)