Московский государственный технический университет им. Н.Э. Баумана
Факультет «Информатика и системы управления» Кафедра «Системы обработки информации и управления»

Отчет
по домашнему заданию
По курсу
“Разработка интернет-приложений”

Выполнила:
Студентка группы ИУ5-51
Соболева Е.Д.

Вариант 22
Субъект Пользователь
Отношение Ставка
Объект Команда

Views.py

```
from django.shortcuts import render
from django.http import HttpResponseRedirect
from django.views.generic import View, DetailView
from .models import *
from django import forms
from django.contrib.auth import authenticate, login, logout
from django.contrib.auth.decorators import login_required
from datetime import datetime
from django.core.paginator import Paginator, EmptyPage, PageNotAnInteger


def main(request):
    par = {
        'header': 'Main page'
    }
    return render(request, 'MainPage.html', context=par)


class TeamForm(forms.ModelForm):
    class Meta(object):
        model = Team
        fields = ['team_name', 'rating', 'sport', 'number_of_players',
'picture', 'TeamId']

    def save(self):
        team = Team()
        team.team_name = self.cleaned_data.get('name')
        team.rating = self.cleaned_data.get('price')
        team.sport = self.cleaned_data.get('type')
        team.number_of_players = self.cleaned_data.get('quantity')
        picture = self.cleaned_data.get('picture')
        team.picture = picture
        team.save()


class RegistrationForm(forms.Form):
    username = forms.CharField(min_length=5, label='Логин')
    password = forms.CharField(min_length=6, widget=forms.PasswordInput,
label='Пароль')
    password2 = forms.CharField(min_length=6, widget=forms.PasswordInput,
label='Повторите ввод')
    last_name = forms.CharField(label='Фамилия')
    first_name = forms.CharField(label='Имя')
    email = forms.EmailField(label='Email')


def registration_1(request):
    form = RegistrationForm(request.POST or None)
    if request.method == 'POST':
        form = RegistrationForm(request.POST)
        is_val = form.is_valid()
        data = form.cleaned_data
        if data['password'] != data['password2']:
            is_val = False
            form.add_error('password2', ['Пароли должны совпадать'])
        if User.objects.filter(username=data['username']).exists():
            form.add_error('username', ['Такой логин уже занят'])
            is_val = False
        if is_val:
            new_user = User.objects.create_user(data['username'],
data['email'], data['password'])
            print(new_user)
            user1 = User1()
            user1.user1 = new_user
            user1.email = data['email']
            user1.last_name = data['last_name']
            user1.first_name = data['first_name']
            user1.save()
            return HttpResponseRedirect('/login1')
        else:
            form = RegistrationForm()
    return render(request, 'registration_1.html', {'form': form})


def registration_form(request):
    errors = {}
    if request.method == 'POST':
        last_name = request.POST.get('last_name')
        if not last_name:
            errors['last_name'] = 'Введите Фамилию'

        first_name = request.POST.get('first_name')
        if not first_name:
            errors['first_name'] = 'Введите имя'

        email = request.POST.get('Email')
        if not email:
            errors['Email'] = 'Введите Email'

        username = request.POST.get('username')
        if not username:
            errors['username'] = 'Введите логин'
        elif len(username) < 8:
            errors['username'] = 'Логин должен превышать 5 символов'
        if User.objects.filter(username=username).exists():
            errors['username'] = 'Данный логин занят'

        password = request.POST.get('password')
        if not password:
            errors['password'] = 'Введите пароль'
        elif len(password) < 8:
            errors['password'] = 'Длина пароля должна превышать 6 символов'
        password_repeat = request.POST.get('password2')
        if password != password_repeat:
            errors['password_repeat'] = 'Пароли должны совпадать'
        print(username, password, "1")

        if not errors:
            new_user = User.objects.create_user(username, email, password)
            print(new_user)
            user1 = User1()
            user1.user1 = new_user
            user1.email = email
            user1.last_name = last_name
            user1.first_name = first_name
            user1.save()
            return HttpResponseRedirect('/login2')
        else:
            context = {'errors': errors, 'username': username, 'email': email,
'last_name': last_name,
                       'first_name': first_name, 'password': password,
'password_repeat': password_repeat}
            return render(request, 'registration.html', context)
    return render(request, 'registration.html', {'errors': errors})


class LoginForm(forms.Form):
    username = forms.CharField(label='Логин')
    password = forms.CharField(widget=forms.PasswordInput, label='Пароль')


def log_in(request):
    errors = {}
    if request.method == 'POST':
        username = request.POST.get('username')
        if not username:
            errors['username'] = 'Введите логин'
        elif len(username) < 5:
            errors['username'] = 'Слишком короткий логин. Минимальная длина-5
знаков'

        password = request.POST.get('password')
        if not password:
            errors['password'] = 'Введите пароль'
        elif len(password) < 8:
            errors['password'] = 'Слишком короткий пароль. Минимальная длина-8
знаков'

        user = authenticate(request, username=username, password=password)
        print(user)
        if user is None and 'username' not in errors.keys() and 'password' not
in errors.keys():
            errors['login'] = 'Логин или пароль введены неверно'

        if not errors:
            login(request, user)
            return HttpResponseRedirect('/items')
        else:
            context = {'errors': errors}
            return render(request, 'UserLogin.html', context)
    return render(request, 'UserLogin.html', {'errors':errors})


def log_in1(request):
    if request.method == 'POST':
        form = LoginForm(request.POST)
        print(form)
        data = form.cleaned_data

        if form.is_valid():
            user = authenticate(request, username=data['username'],
password=data['password'])
            print(len(data['username']), len(data['password']))
            if user is not None:
                login(request, user)
                return HttpResponseRedirect('/items')
            else:
                form.add_error('username', ['Неверный логин или пароль'])
    else:
        form = LoginForm()
    return render(request, 'UserLogin_1.html', {'form': form})


def logout_view(request):
    logout(request)
    return render(request, 'MainPage.html')


@login_required(login_url='/login2')
def logged_in(request):
    return render(request, 'items.html')


def logged_in_view(request):
    if request.user.is_authenticated:
        return render(request, 'items.html')
    else:
        return HttpResponseRedirect('/login1')


def new_item(request):
    errors = {}
    if request.method == 'POST':
        team_name = request.POST.get('team_name')
        if not team_name:
            errors['team_name'] = 'Введите Название команды'
        if Team.objects.filter(team_name=team_name).exists():
            errors['team_name'] = 'Данная команда уже существует'

        rating = request.POST.get('rating')
        if not rating:
            errors['rating'] = 'Введите рейтинг'

        sport = request.POST.get('sport')
        if not sport:
            errors['sport'] = 'Введите вид спорта'

        number_of_players = request.POST.get('number_of_players')
        if not number_of_players:
            errors['number_of_players'] = 'Введите кол-во участников команды'

        picture = request.FILES.get('picture')
        if not picture:
            errors['picture'] = 'Загрузите фото'
        if not errors:
            team = Team(team_name=team_name, rating=rating, sport=sport,
number_of_players=number_of_players,
                        picture=picture)
            team.save()
            TeamId = team.TeamId
            return HttpResponseRedirect('/item/' + str(TeamId))
        else:
            context = {'errors': errors, 'team_name': team_name, 'rating':
rating, 'sport': sport,
                       'number_of_players': number_of_players, 'picture':
picture}

    return render(request, 'new_item.html', locals())


class ItemsView(View):
    def get(self, request):
        dict_users = {}
        teams = Team.objects.all()
        teams1 = Team.objects.all()
        form = TeamForm()
        paginator = Paginator(teams, 5)
        page = request.GET.get('page')
        try:
            teams = paginator.page(page)
        except PageNotAnInteger:
            teams = paginator.page(1)
        except EmptyPage:
            teams = paginator.page(paginator.num_pages)
        return render(request, 'items.html', context={'teams': teams, 'users':
dict_users, 'form': form,
                                                      'teams1': teams1})


class TeamObject(DetailView):
    model = Team
    context_object_name = 'TeamObject'
    template_name = 'obj.html'
```

```
def new_bet(request, team):
    errors = {}
    if request.method == 'POST':
        bet = Bet()
        uid = request.user.id
        user = User1.objects.get(id=uid)
        bet.user = user
        bet_id = bet.id
        team1 = Team.objects.get(TeamId=int(team))
        amount = request.POST.get('amount')
        date = datetime.today()
        bet.date = date
        bet.amount = amount
        if not errors:
            bet.save()
            bet2 = BetTeam()
            bet2.team_id = team1.TeamId
            bet2.id = bet_id
            bet2.bet_id = bet.id
            bet2.user_id = uid
            bet2.save()
            return HttpResponseRedirect('/item/'+str(team1.TeamId)+'/')
        else:
            context = {'errors': errors, 'amount': amount}

    return render(request, 'new_bet.html', locals())


def bets_users(request, team):
    bets = []
    team1 = Team.objects.get(TeamId=int(team))
    users = User1.objects.all()
    bet3 = []
    bet = BetTeam.objects.filter(team_id=team1.TeamId).all()
    for b in bet:
        bet1 = Bet.objects.filter(id=b.bet_id).all()
        if len(bet1) != 0:
            bet3.append(b.id)
    for i in bet3:
        bets.append(Bet.objects.get(id=i))
    return render(request, 'bets.html', context={"bets": bets, "users": users,
'team': team})
```

settings.py

```
import os

# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
# BASE_DIR = os.path.dirname(os.path.dirname(__file__))


# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/1.11/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = '*-^fly(c-6*fig9t%of)-&%khbu2s&z895#)$k&eo0nw%$))$r'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []


# Application definition
```

```
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'app.apps.AppConfig',

]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'untitled8.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [os.path.join(BASE_DIR, 'templates')]
        ,
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]

WSGI_APPLICATION = 'untitled8.wsgi.application'


# Database
# https://docs.djangoproject.com/en/1.11/ref/settings/#databases

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'db5',
        'USER': 'dbuser',
        'PASSWORD': '123',
        'HOST': 'localhost',
        'PORT': 3306,
        'OPTIONS': {'charset': 'utf8'},
        'TEST_CHARSET': 'utf8',
    }
}


# Password validation
# https://docs.djangoproject.com/en/1.11/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME':
'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME':
'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME':
'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME':
'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]


# Internationalization
# https://docs.djangoproject.com/en/1.11/topics/i18n/

LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'UTC'

USE_I18N = True

USE_L10N = True

USE_TZ = True


# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/1.11/howto/static-files/

STATIC_URL = '/static/'
MEDIA_URL = '/pics/'
MEDIA_ROOT='C:/Users/Елизавета/PycharmProjects/untitled8/'
```

urls.py

```
from django.conf.urls import url
from app.views import *
from django.conf.urls.static import static
from django.conf import settings
from django.contrib import admin
urlpatterns = [
    url(r'^admin/', admin.site.urls),
    url(r'^$', main),
    url(r'^registration$', registration_form, name='registration'),
    url(r'^registration_1$', registration_1, name='registration_1'),
    url(r'^login1$', log_in, name='login1'),
    url(r'^login2$', log_in1, name='login2'),
    url(r'^logout$', logout_view, name='logout'),
    url(r'^logged_in$', logged_in_view, name='logged_in'),
    url(r'^logged_in_1$', logged_in, name='logged_in_1'),
    url(r'^new_item$', new_item, name='new_item'),
    url(r'^items$', ItemsView.as_view(), name='items'),
    url(r'^item/(?P<pk>\d+)/$', TeamObject.as_view(), name='TeamObject'),
    url(r'^new_bet/(?P<team>\d+)$', new_bet, name='new_bet'),
    url(r'^bets/(?P<team>\d+)$', bets_users, name='bets'),
]
urlpatterns += static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)

admin.py
from django.contrib import admin
from .models import *
from django.contrib.auth.models import User
# Register your models here.
```

```
@admin.register(User1)
class UserAdmin(admin.ModelAdmin):
    empty_value_display = 'null'
    list_display = ('last_name', 'first_name', 'email', 'phone', 'passport',
'birthday')
    list_filter = ('last_name',)
    search_fields = ['last_name', 'first_name', 'email']


class BetTeam(admin.TabularInline):
    model = BetTeam
    extra = 1


@admin.register(Team)
class TeamAdmin(admin.ModelAdmin):
    empty_value_display = 'null'
    list_display = ('team_name','rating','sport','number_of_players')
    list_filter = ('team_name',)
    search_fields = ['team_name','sport']

    inlines = (BetTeam,)

    def bets(self, request):
        bets = []
        for s in BetTeam.objects.filter(team=request.name):
            bets.append(s)
        return bets


@admin.register(Bet)
class BetAdmin(admin.ModelAdmin):
    empty_value_display = 'null'

    def username(self, obj):
        return "{}".format(obj.user)

    inlines = (BetTeam,)

    list_display = ('id', 'username', 'date', 'amount')
    list_filter = ('id',)
    search_fields = ['uname', 'date', 'amount']
```

models.py

```
from django.db import models
from django.contrib.auth.models import User

# Create your models here.


class User1(models.Model):
    user1 = models.OneToOneField(User, on_delete=models.CASCADE)
    first_name = models.CharField(max_length=20, null=True)
    last_name = models.CharField(max_length=20, null=True)
    phone = models.PositiveIntegerField(unique=True, null=True)
    email = models.EmailField(max_length=50, unique=True)
    birthday = models.DateField(blank=True, null=True)
    passport = models.PositiveIntegerField(unique=True, null=True)

    def __str__(self):
        return str(self.last_name)


class Team(models.Model):
    TeamId = models.AutoField(primary_key=True)
    team_name = models.CharField(max_length=30, unique=True, null=True)
    rating = models.PositiveSmallIntegerField(null=True, unique=True)
    sport = models.CharField(max_length=30, null=True)
    number_of_players = models.PositiveSmallIntegerField(null=True)
    picture = models.ImageField(upload_to='pics/', null=True, blank=True)

    def __str__(self):
        return str(self.team_name)


class Bet(models.Model):
    user = models.ForeignKey(User1, on_delete=models.CASCADE)
    team = models.ManyToManyField(Team, through="BetTeam")
    date = models.DateField()
    amount = models.FloatField(null=True)

    def __str__(self):
        return str(self.id)


class BetTeam(models.Model):
    team = models.ForeignKey(Team, on_delete=models.CASCADE)
    bet = models.ForeignKey(Bet, on_delete=models.CASCADE)
    id = models.AutoField( primary_key=True)

    def __str__(self):
        return str(self.id)
```

base.html

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>{% block title %}{% endblock %}</title>
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css">
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap-
theme.min.css">
<body>
    <nav class="navbar navbar-inverse navbar-static-top">
        <div class="container-fluid">
            <div class="navbar-header">
              <a class="navbar-brand" href="/">Главная</a>
            </div>
            <div class="collapse navbar-collapse" id="myNavbar">
              <ul class="nav navbar-nav">
                <li><a href="/registration">Регистрация</a></li>
                <li><a href="/login1">Вход</a></li>
                <li><a href="/logout">Выход</a></li>
{#                <li><a href="/registration_1">Регистрация1</a></li>#}
{#                <li><a href="/ulogin_1">Вход1</a></li>#}
              </ul>
            </div>
            <div>
                {% block content %}{% endblock %}
            </div>
        </div>
    </nav>
</body>
</html>
```

UserLogin.html

```
{% extends 'base.html' %}
{% block title %}{% endblock %}
{% block content %}
    <div class="jumbotron">
        <div class="container">
            <form method="POST" class="form-horizontal">
                <h1 style="text-align:center">Вход</h1>
                {% csrf_token %}
                {% for err in errors.values %}
                    <div class="alert alert-danger" >{{ err }}</div>
                {% endfor %}
                <div class="col-sm-offset-3 form-group">
                    <label class="col-sm-3 control-label">Логин:</label>
                    <div class="col-sm-8">
                        <input type="text" name="username">
                    </div>
                </div>
                <div class="col-sm-offset-3 form-group">
                    <label  class="col-sm-3 control-label">Пароль:</label>
                    <div class="col-sm-8">
                        <input  type="password" name="password">
                    </div>
                </div>
                <div class="form-group">
                    <div class="col-sm-offset-3 col-sm-9">
                        <button  style="text-align:center" class="col-sm-1 btn
btn-primary  control-label" type="submit">Войти</button>
                        <input   style="text-align:center" class="col-sm-3  btn
col-sm-offset-1  btn-primary control-label" type="button"
value="Зарегистрироваться" onClick='location.href="/registration_1"'>
                    </div>
                </div>
            </form>
        </div>
    </div>
{% endblock %}
```

Registration_1.html

```
{% extends 'base.html' %}
{% block title %}registration{% endblock %}
{% block content %}
    <div class="main_block_elem">
        <div class="jumbotron">
            <div class="container">
                <form method="POST" class="form-horizontal">
                    <h1>Регистрация </h1>
                    {% csrf_token %}
                    {% for field in form %}
                        {% for error in field.errors %}
                            <div class="alert alert-danger" role="alert">{{
field.errors }}</div>
                        {% endfor %}
                    {% endfor %}
                    {% for field in form %}
                        <div class="form-group">
                            <label for="{{ field.id_for_label }}" class="col-
sm-3 control-label">{{ field.label }}:</label>
                            <div class="col-sm-9">
                                {{ field }}
                            </div>
                        </div>
                    {% endfor %}
                    <div class="form-group">
                        <div class="col-sm-offset-3 col-sm-9">
                            <button type="submit" class="btn btn-
primary">Зарегистрировать</button>
                        </div>
                    </div>
                </form>
            </div>
        </div>
    </div>
{% endblock %}
```

Obj.html

```
{% extends 'base.html' %}
{% csrf_token %}
{% block title %}Team{% endblock %}
{% block content %}
    <div class="jumbotron">
        <div class="container">
            <div class="blog-header">
                <h1 >{{ TeamObject.team_name }}</h1>
            </div>
            <div>
                <div class="col-lg-3">
                    <img src="{{ TeamObject.picture.url }}" width="50%">
                </div>
                <div class="col-lg-3">
                    <h4>Вид спорта: {{TeamObject.sport }}</h4>
                    <h4>Кол-во игроков: {{ TeamObject.number_of_players }}</h4>
                    <h4>Рейтинг: {{ TeamObject.rating }}</h4>
                    <br>
                </div>
                <div class="col-lg-3 ">
                    {% if user.is_authenticated %}
                        <a role="button" class="btn btn-default btn-lg btn-
block" href="/items">Назад к списку</a>
                        <a role="button" class="btn btn-default btn-lg btn-
block"
                           onClick='location.href="/bets/{{
TeamObject.TeamId}}"'>Просмотр ставок</a>
                        <a role="button" class="btn btn-default btn-lg btn-
block"
                           onClick='location.href="/new_bet/{{
TeamObject.TeamId}}"'>Добавить ставку</a>
                        {% endif %}
                </div>
            </div>
        </div>
    </div>
{% endblock %}
```

MainPage.html

```
{% extends 'base.html' %}
{% block title %}{% endblock %}
{% block content %}

{% endblock %}
```

Registration.html

```
{% extends 'base.html' %}
{% block title %}{% endblock %}
{% block content %}
    <div class="jumbotron">
        <div class="container">
            <div class="page-header">
                <h1>Регистрация</h1>
                {% for err in errors.values %}
                    <div class="alert alert-danger" >{{ err }}</div>
                {% endfor %}
            </div>
            <form method="POST" class="form-horizontal">{% csrf_token %}
                <div class="col-sm-offset-3 form-group">
                    <label class="col-sm-3 control-label">Фамилия:</label>
                    <div class="col-sm-8">
                        <input type="text" name="last_name">
                    </div>
                </div>
                <div class="col-sm-offset-3 form-group">
```

```
                    <label class="col-sm-3 control-label">Имя:</label>
                    <div class="col-sm-8">
                        <input type="text" name="first_name">
                    </div>
                </div>
                <div class="col-sm-offset-3 form-group">
                    <label class="col-sm-3 control-label">Email:</label>
                    <div class="col-sm-8">
                        <input type="text" name="Email">
                    </div>
                </div>
                <div class="col-sm-offset-3 form-group">
                    <label class="col-sm-3 control-label">Логин:</label>
                    <div class="col-sm-8">
                        <input type="text" name="username">
                    </div>
                </div>
                <div class="col-sm-offset-3 form-group">
                    <label class="col-sm-3 control-label">Пароль:</label>
                    <div class="col-sm-8">
                        <input type="password" name="password">
                    </div>
                </div>
                <div class="col-sm-offset-3 form-group">
                    <label class="col-sm-3 control-label">Повторите
ввод:</label>
                    <div class="col-sm-8">
                        <input type="password" name="password2">
                    </div>
                </div>

                <button style="text-align:center" class="col-sm-offset-3 col-
sm-2 btn btn-primary  control-label" type="submit">Зарегистрировать</button>

            </form>
        </div>
    </div>
{% endblock %}
```

New_bet.html

```
{% extends "base.html" %}
{% block title %}Add new{% endblock %}
{% block styles %}
{% endblock %}
{% block content %}
        <div class="jumbotron">
            <div class="container">
                <div class="col-md-8">
                    <h3>Сделать ставку</h3>
                    {% for err in errors.values %}
                        <div class="alert alert-danger" >{{ err }}</div>
                    {% endfor %}
                    <form method="POST" enctype="multipart/form-data">
                        {% csrf_token %}
                        <div class="form-group">
                            <label for="exampleInputLogin1">СУММА</label>
                            <input type="text" name="amount" class="form-
control" id="exampleInputLogin1" placeholder="amount">
                        </div>
                        <button type="submit" class="btn btn-default btn-lg 
btn-block " >Готово</button>
                        <a role="button" class="btn btn-default btn-lg btn-
block" href="/items">Назад к списку</a>
                    </form>
                </div>
            </div>
        </div>
{% endblock %}
```

UserLogin_1.html

```
{% extends 'base.html' %}
{% block title %}login{% endblock %}
{% block content %}
    <div class="jumbotron">
        <div class="container">
            <form method="POST" class="form-horizontal">
                <h1 style="text-align:center">Вход</h1>
                {% csrf_token %}
                {% for f in form %}
                    {% for error in f.errors %}
                        <div class="alert " role="alert">{{ f.errors }}</div>
                    {% endfor %}
                {% endfor %}
                {% for f in form %}
                    <div class="form-group">
                        <label for="{{ f.id_for_label }}" class="col-sm-3 
control-label">{{ f.label }}:</label>
                        <div class="col-sm-9">
                            {{ f }}
                        </div>
                    </div>
                {% endfor %}
                <div class="form-group">
                    <div class="col-sm-offset-3 col-sm-9">
                        <button  style="text-align:center" class="col-sm-1 btn-
primary control-label" type="submit">Войти</button>
                        <input style="text-align:center" class="col-sm-3  col-
sm-offset-1  btn-primary control-label"
                               type="button"  value="Зарегистрироваться"
onClick='location.href="/registration_1"'>
                    </div>
                </div>
            </form>
        </div>
    </div>
{% endblock %}
```

Items.html

```
{% extends 'base.html' %}
{% csrf_token %}
{% block title %}Teams{% endblock %}
{% block content %}
    <div class="jumbotron">
        <div class="container-fluid">
            <div class="blog-header">
                <h1  class="col-lg-offset-4">Список команд</h1>
            </div>
            <div class="row">
                <div class="col-md-9">
                    {% include 'item.html' with item=item %}
                </div>
                <div class="col-md-offset-9">
{#                          {% if user.is_superuser %}#}
                            <a role="button" class="btn btn-default btn-lg btn-
block"
                                onClick='location.href="/new_item"'>Добавить 
команду</a>
{#                          {% endif %}#}
                </div>
            </div>
        </div>
        <div class="pagination">
            <span class="step-links">
                {% if teams.has_previous %}
```

```
                    <a href="?page={{ teams.previous_page_number
}}">previous</a>
                {% endif %}

                <span class="current">
                    Page {{ teams.number }} of {{ teams.paginator.num_pages }}.
                </span>
                {% if teams.has_next %}
                    <a href="?page={{ teams.next_page_number }}">next</a>
                {% endif %}
            </span>
        </div>
    </div>
{% endblock %}
```

Item.html

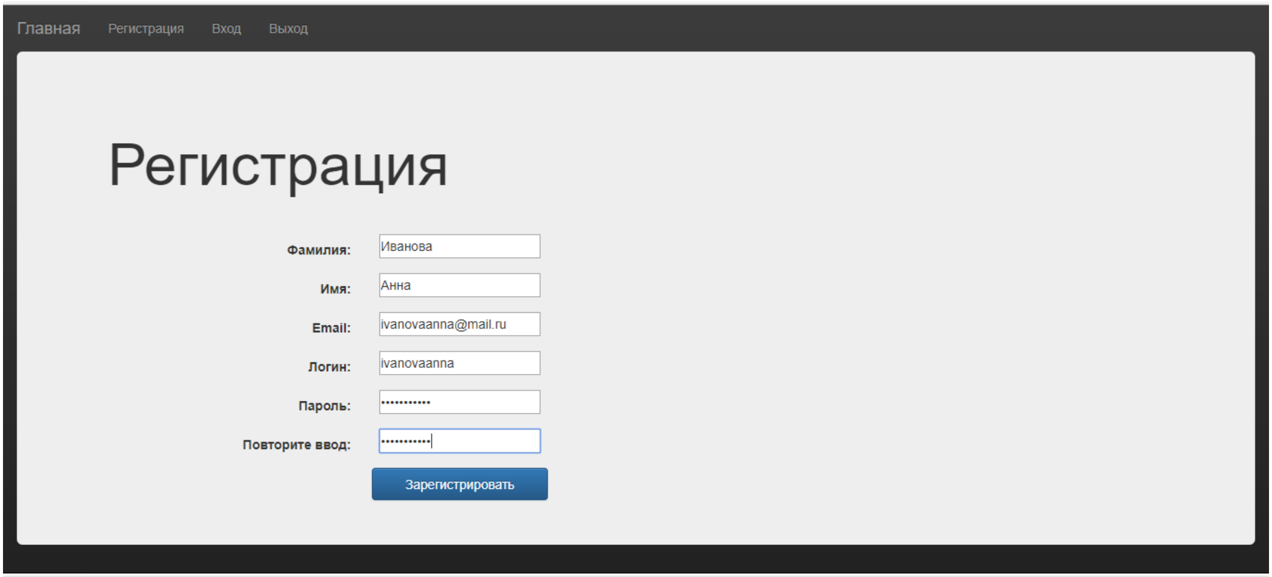
```
{% for item in teams %}
    <section class="panel ">
            <div class="panel-title">
                <h2>
                    <div class="title">
                        <a href="/item/{{ item.TeamId }}/">{{
item.team_name}}</a>
                    </div>
                </h2>
            </div>
        <div class="panel-body ">
            <div class="row">
                <div class="col-lg-4">
                    <img src=" {{item.picture.url}}">
                </div>
                <div class="col-lg-4">
                    <h4>Вид спорта:{{ item.sport}}</h4>
                    <h4>Количество игроков:{{ item.number_of_players }}</h4>
                    <h4>Рейтинг:{{ item.rating }}</h4>
                </div>
            </div>
        </div>
    </section>
{% empty %}
   ПУСТОЙ СПИСОК
{% endfor %}
```

Bets.html

```
{% extends 'base.html' %}
{% csrf_token %}
{% block title %}Bets{% endblock %}
{% block content %}
    <div class="jumbotron">
        <div class="container">
            <div class="blog-header">
                <h1 class="blog-title col-lg-offset-3">Ставки на команду</h1>
            </div>
            <table id ='orders' class="table table-striped table-hover ">
                <thead>
                <tr>
                    <th>ID ставки</th>
                    <th>Дата</th>
                    <th>Сумма</th>
                    <th>Пользователь</th>
                </tr>
                </thead>
                <tbody>
                    {% for bet in bets %}
                        <tr>
                            <td>{{ bet.id}}</td>
                            <td>{{ bet.date}}</td>
                            <td>{{ bet.amount}}</td>
                            <td>
                                  {% for u in users  %}
                                      {% if u.id == bet.user.id %}
                                          {{ u.last_name }}
                                          {{ u.first_name }}
                                      {% endif %}
                                  {% endfor %}
                            </td>
                        </tr>
                    {% endfor %}
                </tbody>
            </table>
            <a role="button" class="btn btn-default btn-lg btn-block" href="/item/{{ team }}">Назад к команде</a>
            <a role="button" class="btn btn-default btn-lg btn-block" href="/items">Назад к списку</a>
        </div>
    </div>
{% endblock %}
```

New_item.html

```
{% extends "base.html" %}
{% block title %}Add new{% endblock %}
{% block styles %}
{% endblock %}
{% block content %}
        <div class="jumbotron">
            <div class="container">
                <div class="page-header">
                    <h3>Добавить команду</h3>
                    {% for err in errors.values %}
                        <div class="alert alert-danger" >{{ err }}</div>
                    {% endfor %}
                </div>
                <div class="col-md-8">
                    <form method="POST" enctype="multipart/form-data" >
                        {% csrf_token %}
                        <div class="form-group">
                            <label>Название</label>
                            <input type="text" name="team_name" class="form-control"  placeholder="название">
                        </div>
                        <div class="form-group">
                            <label>Рейтинг</label>
                            <input type="text" name="rating" class="form-control"  placeholder="рейтинг">
                        </div>
                        <div class="form-group">
                            <label>Вид спорта</label>
                            <input type="text" name="sport" class="form-control"  placeholder="вид спорта">
                        </div>
                        <div class="form-group">
                            <label>Кол-во игроков</label>
                            <input type="text" name="number_of_players" class="form-control"  placeholder="кол-во игроков">
                        </div>
                        <div class="form-group">
                            <label>Picture</label>
                            <input type="file" name="picture" >
                        </div>
                        <button type="submit" class="btn btn-default btn-lg btn-block">Добавить</button>
                        <a role="button" class="btn btn-default btn-lg btn-block" href="/items">Назад к списку</a>
                    </form>
                </div>
            </div>
        </div>
{% endblock %}
```



Регистрация

Фамилия: Иванова

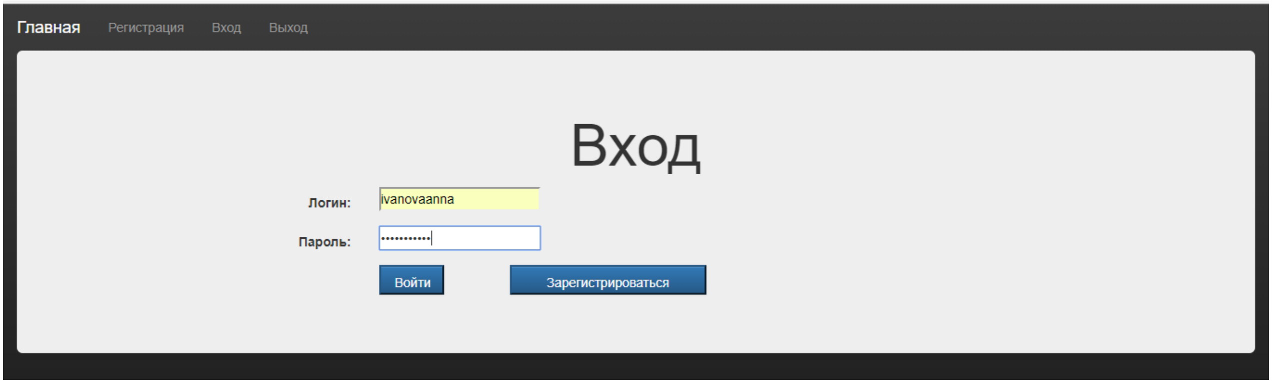
Имя: Анна

Email: ivanovaanna@mail.ru

Логин: ivanovaanna

Пароль: •••••••••••

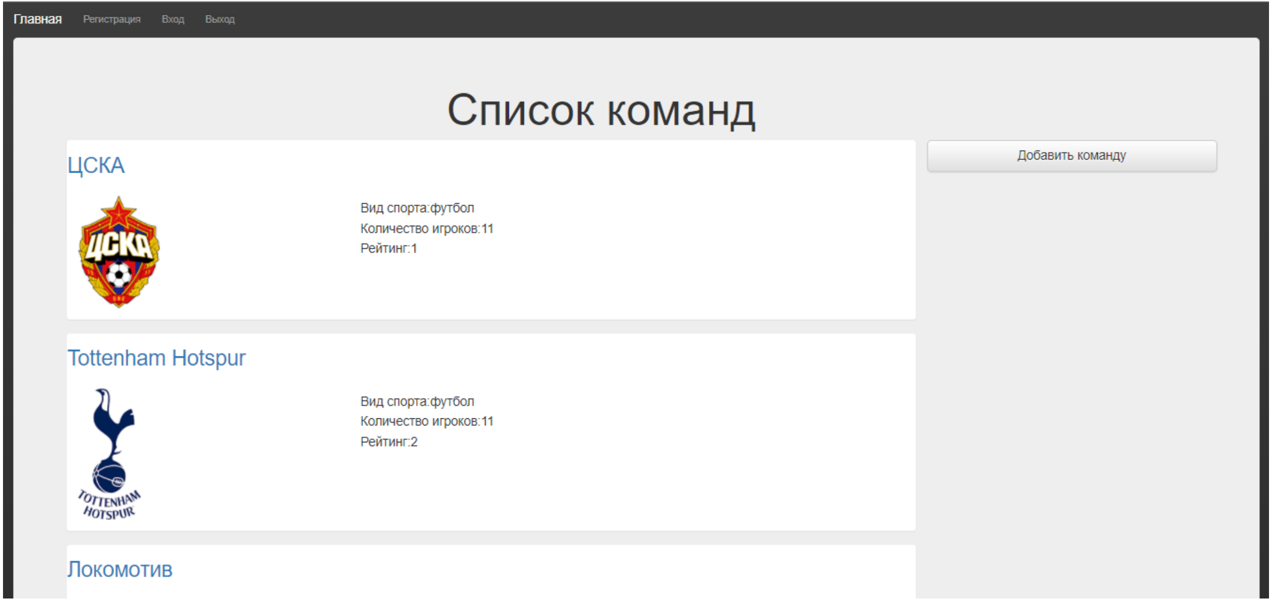
Повторите ввод: •••••••••••



Вход

Логин: ivanovaanna

Пароль: •••••••••••



Список команд

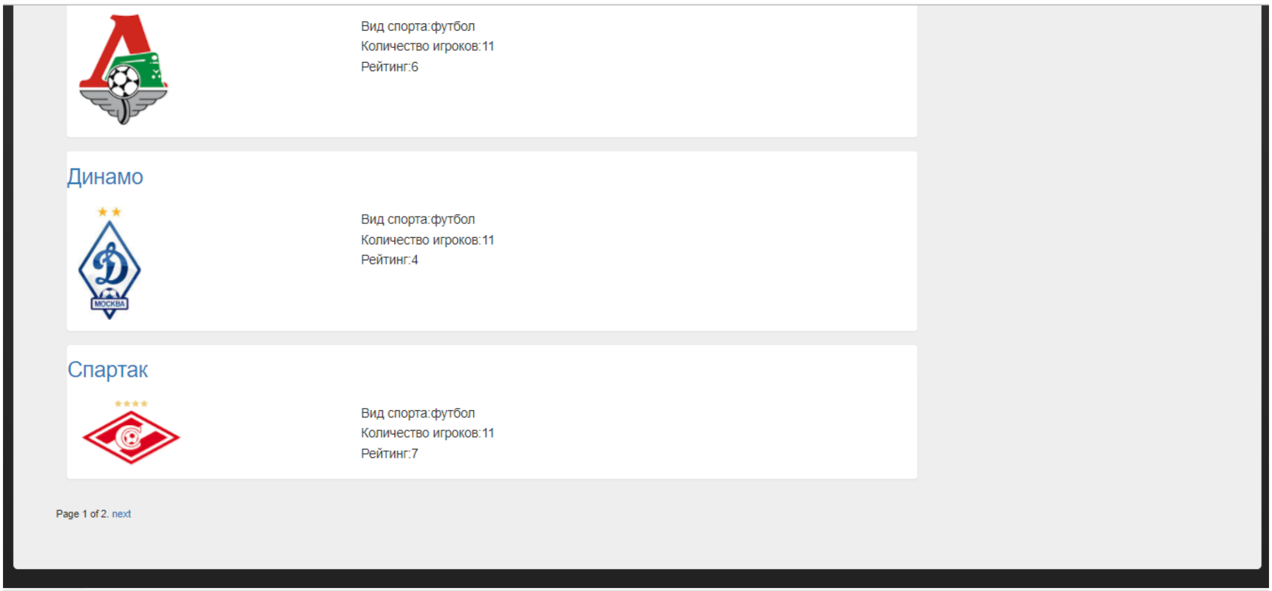
ЦСКА

Вид спорта:футбол
Количество игроков:11
Рейтинг:1

Tottenham Hotspur

Вид спорта:футбол
Количество игроков:11
Рейтинг:2

Локомотив



Вид спорта:футбол
Количество игроков:11
Рейтинг:6

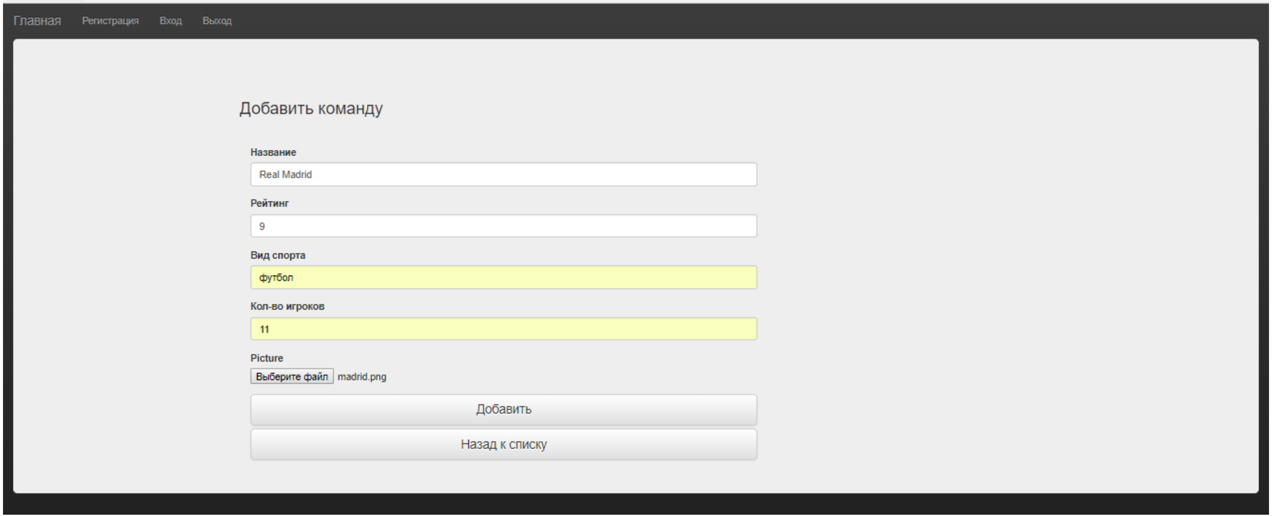
Динамо

Вид спорта:футбол
Количество игроков:11
Рейтинг:4

Спартак

Вид спорта:футбол
Количество игроков:11
Рейтинг:7

Page 1 of 2. next



Добавить команду

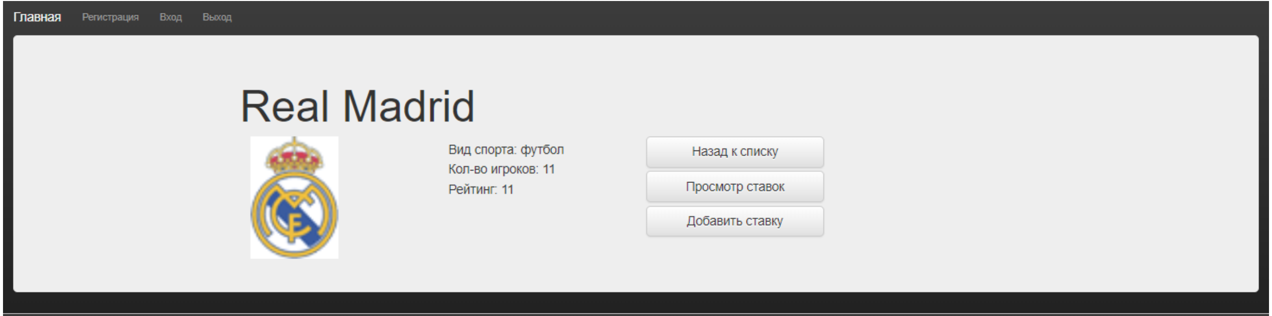
Название: Real Madrid

Рейтинг: 9

Вид спорта: футбол

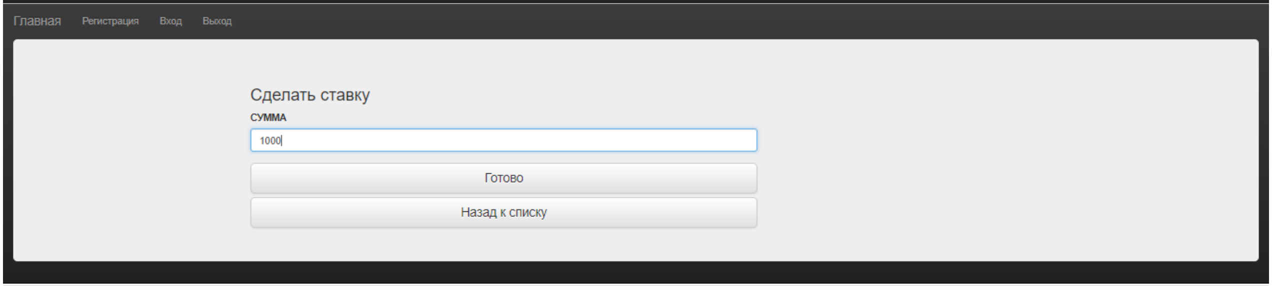
Кол-во игроков: 11

Picture: Выберите файл madrid.png



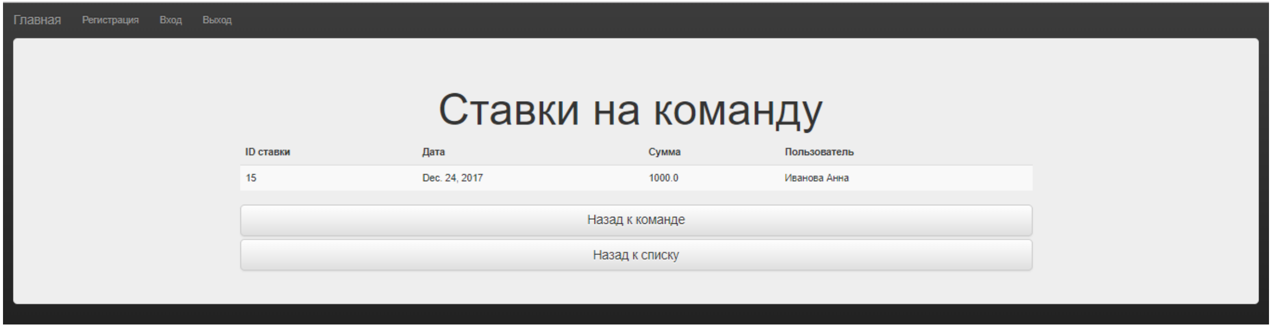
Real Madrid

Вид спорта: футбол
Кол-во игроков: 11
Рейтинг: 11



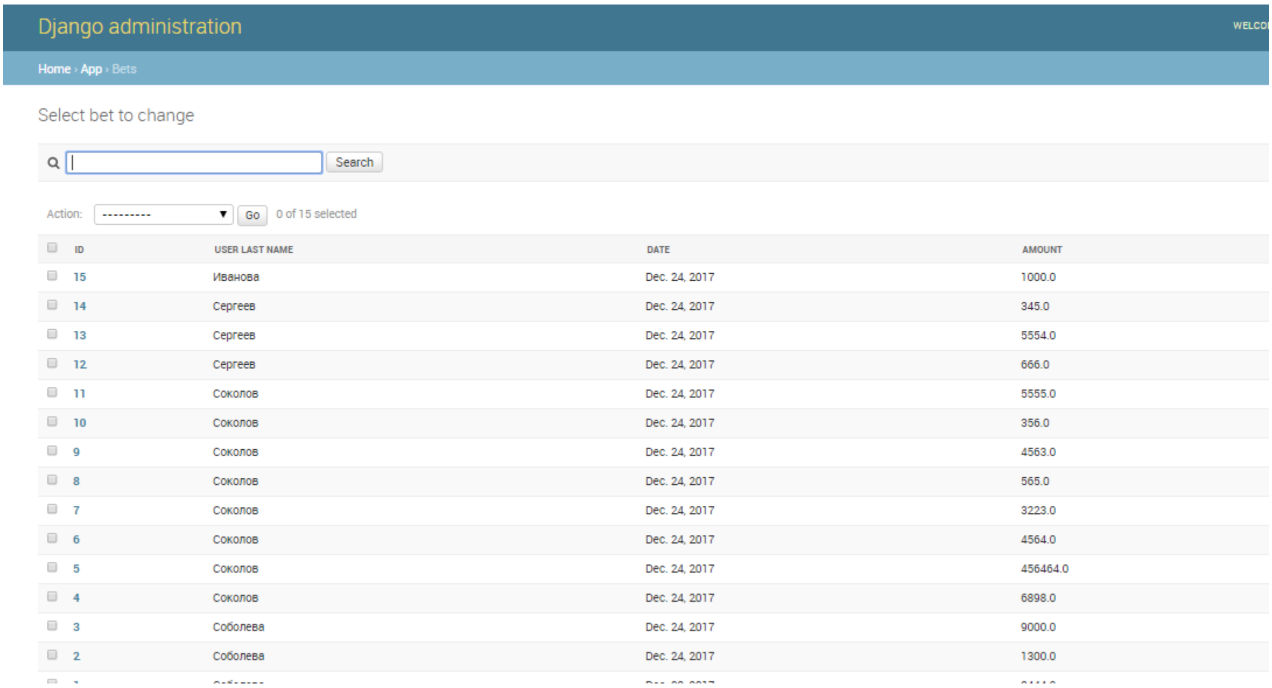
Сделать ставку

СУММА: 1000



Ставки на команду

ID ставки	Дата	Сумма	Пользователь
15	Dec. 24, 2017	1000.0	Иванова Анна



Select bet to change

0 of 15 selected

☐	ID	USER LAST NAME	DATE	AMOUNT
☐	15	Иванова	Dec. 24, 2017	1000.0
☐	14	Сергеев	Dec. 24, 2017	345.0
☐	13	Сергеев	Dec. 24, 2017	5554.0
☐	12	Сергеев	Dec. 24, 2017	666.0
☐	11	Соколов	Dec. 24, 2017	5555.0
☐	10	Соколов	Dec. 24, 2017	356.0
☐	9	Соколов	Dec. 24, 2017	4563.0
☐	8	Соколов	Dec. 24, 2017	565.0
☐	7	Соколов	Dec. 24, 2017	3223.0
☐	6	Соколов	Dec. 24, 2017	4564.0
☐	5	Соколов	Dec. 24, 2017	456464.0
☐	4	Соколов	Dec. 24, 2017	6898.0
☐	3	Соболева	Dec. 24, 2017	9000.0
☐	2	Соболева	Dec. 24, 2017	1300.0
☐	1	[illegible]	[illegible]	[illegible]