

**Лабораторная работа №7 по дисциплине
“Разработка интернет приложений”**

ИСПОЛНИТЕЛЬ:

студент группы РТ5-51
Коростелёв В.М.

"__" _____ 2016 г.

Forms.py

```
from django import forms
from django.contrib.auth import authenticate
from django.contrib.auth.hashers import make_password
from django.contrib.auth.models import User
```

```
class RegistrationForm(forms.Form):
    username = forms.CharField(min_length=5, label='Логин')
    password = forms.CharField(min_length=8, widget=forms.PasswordInput, label='Пароль')
    password2 = forms.CharField(min_length=8, widget=forms.PasswordInput, label='Повторите
ВВОД')
    email = forms.EmailField(label='Email')
    first_name = forms.CharField(max_length=30, label='Введите имя')
    last_name = forms.CharField(max_length=30, label='Введите фамилию')

    def clean_username(self):
        username = self.cleaned_data.get('username')
        try:
            user = User.objects.get(username=username)
            raise forms.ValidationError('Логин занят')
        except User.DoesNotExist:
            return username

    def clean_password2(self):
        pass1 = self.cleaned_data['password']
        pass2 = self.cleaned_data['password2']
        if pass1 != pass2:
            raise forms.ValidationError('Пароли не совпадают')

    def save(self):
        user = User()
        data = self.cleaned_data
        user.username = data.get('username')
        user.password = make_password(data.get('password'))
        user.email = data.get('email')
        user.first_name = data.get('first_name')
        user.last_name = data.get('last_name')
        user.is_active = True
        user.is_superuser = False
        user.save()
        return authenticate(username=user.username, password=user.password)
```

```
class AuthorizationForm(forms.Form):
    username = forms.CharField(min_length=5, label='Логин')
    password = forms.CharField(min_length=8, widget=forms.PasswordInput, label='Пароль')

    def clean(self):
        data = self.cleaned_data
        user = authenticate(username=data.get('username'), password=data.get('password'))
        if user is not None:
            if user.is_active:
```

```

        data['user'] = user
    else:
        raise forms.ValidationError('Пользователь неактивен')
    else:
        raise forms.ValidationError('Неверный логин или пароль')

```

views.py

```

from django.contrib.auth import authenticate, login, logout
from django.contrib.auth.decorators import login_required
from django.http import HttpResponseRedirect
from django.shortcuts import render

```

```

from FormsApp.forms import RegistrationForm, AuthorizationForm

```

```

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

```

```

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

```

```

def registration(request):
    if request.method == 'POST':
        form = RegistrationForm(request.POST)
        if form.is_valid():
            user = form.save()
            login(request, user)
            return HttpResponseRedirect('/success/')
    else:
        form = RegistrationForm()
    return render(request, 'registration.html', {'form': form})

```

```

@login_required(login_url='/error/')
def login_success(request):
    # if not request.user.is_authenticated():
    # return HttpResponseRedirect('/')
    return render(request, 'login.html')

```

```

def authorization(request):
    if request.method == 'POST':
        form = AuthorizationForm(request.POST)
        if form.is_valid():
            data = form.cleaned_data
            user = authenticate(username=data.get('username'), password=data.get('password'))
            if user is not None:
                login(request, user)
                return HttpResponseRedirect('/success/')

```

```
else:
    form = AuthorizationForm()
    return render(request, 'authorization.html', {'form': form})
```

```
def logout_view(request):
    logout(request)
    return HttpResponseRedirect('/error/')
```

models.py

```
from django.contrib import admin
from django.db import models
```

```
class Team(models.Model):
    def __str__(self):
        return self.name
    name = models.CharField(max_length=30, default="")
    description = models.TextField(default="")
    coach = models.CharField(max_length=30, default="")
    games = models.IntegerField(default=0)
    win = models.IntegerField(default=0)
```

```
class CountryFilter(admin.SimpleListFilter):
    title = 'Country'
    parameter_name = 'description'

    def lookups(self, request, model_admin):
        return [
            ('Russian', 'Российские клубы')
        ]

    def queryset(self, request, queryset):
        if self.value() == 'Russian':
            return queryset.filter(description='Российский футбольный клуб')
```

```
class TeamAdmin(admin.ModelAdmin):
    list_display = ('name', 'description', 'games', 'win', 'percent_win')
    search_fields = ['name']
    list_filter = (CountryFilter,)

    def percent_win(self, obj):
        return '{0:.1%}'.format(obj.win/obj.games)
```

urls.py

```
"""Lab_7 URL Configuration
```

The `urlpatterns` list routes URLs to views. For more information please see:

<https://docs.djangoproject.com/en/1.10/topics/http/urls/>

Examples:

Function views

1. Add an import: `from my_app import views`
2. Add a URL to `urlpatterns`: `url(r'^$', views.home, name='home')`

Class-based views

1. Add an import: `from other_app.views import Home`
2. Add a URL to `urlpatterns`: `url(r'^$', Home.as_view(), name='home')`

Including another URLconf

1. Import the `include()` function: `from django.conf.urls import url, include`
2. Add a URL to `urlpatterns`: `url(r'^blog/', include('blog.urls'))`

"""

```
from django.conf.urls import url
from django.contrib import admin
```

```
from FormsApp.views import registration, main_page, login_success, authorization, logout_view,
error_auth
```

```
urlpatterns = [
    url(r'^admin/', admin.site.urls),
    url(r'^$', main_page, name='main'),
    url(r'^registration/', registration, name='registration'),
    url(r'^authorization/', authorization, name='authorization'),
    url(r'^success/', login_success),
    url(r'^error/', error_auth, name='error_auth'),
    url(r'^logout/', logout_view, name='logout'),
]
```

Lab 7

Регистрация

Авторизация

Lab 7

Регистрация

Логин:

Пароль:

Повторите ввод:

Email:

Введите имя:

Введите фамилию:

Авторизация

Логин: Пароль: [Авторизоваться](#)

Вы успешно вошли в систему!

[Выйти](#)

Вы не вошли в систему!

[Регистрация](#)[Авторизация](#)

Select team to change

[ADD TEAM](#) Action: 0 of 3 selected

<input type="checkbox"/>	NAME	DESCRIPTION	GAMES	WIN	PERCENT WIN
<input type="checkbox"/>	Барселона	Испанский футбольный клуб	13	10	76.9%
<input type="checkbox"/>	ЦСКА	Российский футбольный клуб	7	2	28.6%
<input type="checkbox"/>	Спартак	Российский футбольный клуб	10	5	50.0%

3 teams

FILTER

By Country

☐ All☐ Российские клубы