

**Московский государственный технический университет  
им. Н.Э. Баумана.**

**Факультет «Информатика и управление»**

**Кафедра ИУ5. Курс «Разработка интернет приложений»**

**Отчет по лабораторной работе №6**

**«Авторизация, работа с формами, Django admin»**

**Выполнил:**

студент группы ИУ5-52  
Афанасьев Л. Е.

**Проверил:**

преподаватель каф. ИУ5  
Гапанюк Ю.Е.

Москва, 2017 г.

## 1 Задание

1. Создайте view, которая возвращает форму для регистрации.

Поля формы:

- Логин
- Пароль
- Повторный ввод пароля
- Email
- Фамилия
- Имя

2. Создайте view, которая возвращает форму для авторизации.

Поля формы:

- Логин
- Пароль

3. При отправке формы регистрации во view проверять каждый параметр по правилам валидации, если валидация всех полей пройдена, то создавать пользователя и делать перенаправление на страницу логина, а ошибки, если они есть, выводить над формой.

Правила валидации:

- Логин не меньше 5 символов
- Пароль не меньше 8 символов
- Пароли должны совпадать
- Все поля должны быть заполнены
- Логин – уникален для каждого пользователя

4. При возникновении ошибок в момент отправки формы, введенные значения в полях ввода, кроме пароля, не должны исчезать.
5. Переписать view регистрации с использованием Django Form, правила валидации удалить из view, использовать встроенный механизм валидации полей.
6. Во view авторизации реализовать логин при POST запросе. При успешной авторизации должен происходить переход на страницу успешной авторизации.
7. Страница успешной авторизации должна проверять, что пользователь авторизован. Иначе делать перенаправление на страницу авторизации.
8. Реализовать view для выхода из аккаунта.
9. Заменить проверку на авторизацию на декоратор login\_required
10. Добавить superuser'a через команду manage.py
11. Подключить django.contrib.admin и войти в панель администрирования.
12. Зарегистрировать все свои модели в django.contrib.admin

13. Для выбранной модели настроить страницу администрирования:

- Настроить вывод необходимых полей в списке
- Добавить фильтры
- Добавить поиск
- Добавить дополнительное поле в список

## 2 Листинг

### Forms:

```
# регистрация вручную def
registration_form(request):
    errors = {}
    request.encoding = 'utf-8'
    if request.method == 'POST':
        username = request.POST.get('username')
        if not username:
            errors['uname']='Введите логин'
        elif len(username) < 5:
            errors['uname']='Длина логина должна быть не меньше 5 символов'
        if
        User.objects.filter(username=username).exists():
            errors['uname']='Такой логин уже занят'
        password =
        request.POST.get('password') if not
        password:
            errors['psw']='Введите пароль'
        elif len(password) < 8:
            errors['psw']='Длина пароля должна быть не меньше 8 символов'

        password2 = request.POST.get('password2')
        if password != password2:
            errors['psw2']='Пароли должны совпадать'

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

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

            birthday = request.POST.get('birthday')
            if not birthday:
                errors['bday'] = 'Введите дату рождения'

            sex = request.POST.get('sex')
            if not sex:
                errors['sex'] = 'Введите пол'
            if not
        errors:
```

```

        user = User.objects.create_user(username, email, password)
        cust = Customer()
        cust.user = user
        cust.last_name = last_name
        cust.first_name = first_name
        cust.birthday = birthday
        cust.sex = sex
        cust.save()

        return HttpResponseRedirect('/authorization')
    else:
        context = {'errors': errors, 'username': username, 'email':
email, 'last_name': last_name,
                    'first_name': first_name, 'birthday': birthday, 'sex':
sex}

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

    return render(request, 'registration_form.html', {'errors': errors })

# форма регистрации 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')
    last_name = forms.CharField(label='Фамилия')
    first_name = forms.CharField(label='Имя')
    birthday = forms.DateField(label='День рождения')
    #sex = forms.CharField(label='Пол')
    choices = (('м', 'мужской'), ('ж', 'женский'))
    sex = forms.ChoiceField(label='Пол', widget=forms.RadioSelect,
choices=choices)

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

# регистрация def registration(request):
    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:
            data = form.cleaned_data
            user = User.objects.create_user(data['username'], data['email'],
data['password'])
            cust = Customer()
            cust.user = user
            cust.first_name = data['first_name']
            cust.last_name = data['last_name']
            cust.birthday = data['birthday']
            cust.sex = data['sex']
            cust.save()
            return HttpResponseRedirect('/authorization')
        else:
            form = RegistrationForm()

```

```

        return render(request, 'registration.html', {'form': form})

# авторизация вручную def
authorization_form(request):
    errors = {}
    if request.method == 'POST':
        username = request.POST.get('username')
        if not username:
            errors['uname'] = 'Введите логин'

        password = request.POST.get('password')
        if not password:
            errors['psw'] = 'Введите пароль'

        user = authenticate(request, username=username, password=password)
        if user is None and 'uname' not in errors.keys() and 'psw' not in errors.keys():
            errors['login'] = 'Логин или пароль введены неверно'
            if not errors:
                login(request, user)
                return HttpResponseRedirect('/success_authorization_form')
        else:
            context = {'errors': errors}
            return render(request, 'authorization_form.html', context)

    return render(request, 'authorization_form.html', {'errors': errors})

# авторизация django def
authorization(request):
    if request.method == 'POST':
        form = AuthorizationForm(request.POST)
        print(form)
        data = form.cleaned_data
        if form.is_valid():
            user = authenticate(request, username=data['username'], password=data['password'])
            if user is not None:
                login(request, user)
                return HttpResponseRedirect('/success_authorization')
            else:
                form.add_error('username', ['Неверный логин или пароль'])
            else:
                form = AuthorizationForm()
                return render(request, 'authorization.html', {'form': form})

# успешная авторизация вручную def
success_authorization_form(request):
    if request.user.is_authenticated:
        return HttpResponseRedirect('/')
    else:
        return HttpResponseRedirect('/authorization')

```

```

# успешная авторизация django
@login_required(login_url='/authorization')
def success_authorization(request):
    return HttpResponseRedirect('/')
# ВЫХОД def
logout_view(request):
    logout(request)
    return HttpResponseRedirect('/')
admin.py
from .models import *

# Register your models here.
@admin.register(Customer) class
CustomerAdmin(admin.ModelAdmin):
    #fields = ('first_name', 'last_name')
    list_display = ('username', 'full_name', 'birthday', 'count_of_orders',)
    list_filter = ('sex',)
    search_fields = ['last_name', 'first_name']
    def full_name(self,
obj):
        return "{} {}".format(obj.last_name, obj.first_name)
    def username(self,
obj):
        return "{}".format(obj.user.username)
    def count_of_orders(self,
obj):
        ord = Order.objects.filter(user=obj)
    return len(ord)

@admin.register(Product) class
ProductAdmin(admin.ModelAdmin):
    empty_value_display = '-empty-'
    list_display = ('product', 'price', 'description',)
    list_filter = ('price',) search_fields =
['product_name']
    def product(self,
obj):
        return "{}".format(obj.product_name)
    def price(self,
obj):
        return "{}".format(obj.price)
    def description(self,
obj):
        return "{}".format(obj.description)

@admin.register(Order) class
OrderAdmin(admin.ModelAdmin):
    empty_value_display = '-empty-'
    list_display = ('username', 'product', 'date')
    list_filter = ('product__product_name',) search_fields
= ['product__product_name']
    def username(self,
obj):
        return "{}".format(obj.user.user)
    def date(self,
obj):

```

```
return "{}".format(obj.order_date)
```

## registration\_form.html

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

{% block title %}authorization{% endblock %}

{% block content %}
    <div class="main_block_elem">
    <h3 style="text-align:center">Регистрация</h3>
    {% for error in errors.values %}
        <div class="alert alert-danger" role="alert">{{ error }}</div>
    {% endfor %}

    <form method="POST" class="form-horizontal">
        {% csrf_token %}
        <div class="form-group">
            <label for="uname" class="col-sm-3
controllabel">Логин:</label>
            <div class="col-sm-9">
                <input type="text" name="username" value="{{ username }}"
id="uname">
            </div>
        </div>
        <div class="form-group">
            <label for="psw" class="col-sm-3 controllabel">Пароль:</label>
            <div class="col-sm-9">
                <input type="password" name="password" id="psw">
            </div>
        </div>
        <div class="form-group">
            <label for="psw2" class="col-sm-3 control-label">Повторите
ВВОД:</label>
            <div class="col-sm-9">
                <input type="password" name="password2" id="psw2">
            </div>
        </div>
        <div class="form-group">
            <label for="email" class="col-sm-3
controllabel">Email:</label>
            <div class="col-sm-9">
                <input type="email" name="email" value="{{ email }}"
id="email">
            </div>
        </div>
        <div class="form-group">
            <label for="lname" class="col-sm-3
controllabel">Фамилия:</label>
            <div class="col-sm-9">
                <input type="text" name="last_name" value="{{ last_name }}"
id="ls">
            </div>
        </div>
        <div class="form-group">
            <label for="fname" class="col-sm-3 controllabel">Имя:</label>
            <div class="col-sm-9">
                <input type="text" name="first_name" value="{{ first_name
}} " id="fs">

```

```

    </div>
  </div>
  <div class="form-group">
    <label for="bday" class="col-sm-3 control-label">Дата
рождения:</label>
    <div class="col-sm-9">
      <input type="date" name="birthday" value="{{ birthday }}"
id="bday">
    </div>
  </div>
  <div class="form-group">
    <label for="sex" class="col-sm-3 control-label">Пол:</label>
    <div class="col-sm-9">
      <input type="text" name="sex" value="{{ sex }}" id="sex">
      <!--<input type="radio" name="sex" value="{{ sex }}"
id="sex">М
      <input type="radio" name="sex" value="{{ sex }}"
id="sex">Ж-->
    </div>
  </div>
  <div class="form-group">
    <div class="col-sm-offset-3 col-sm-9">
      <button type="submit" class="btn btn-
default">Зарегистрировать</button>
    </div>
  </div>
</form>
</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>
{% endblock %}

```

## authorization\_form.html

```

{% extends 'base.html' %}

{% block title %}authorization_form{% endblock %}
{% block content %}
<div class="main_block_elem">
    <h3 style="text-align:center">Авторизация</h3>
    {% for error in errors.values %}
        <div class="alert alert-danger" role="alert">{{ error }}</div>
    {% endfor %}
    <form method="POST" class="form-horizontal">
        {% csrf_token %}
        <div class="form-group">
            <label for="uname" class="col-sm-3 control-label">Логин:</label>
            <div class="col-sm-9">
                <input type="text" name="username" value="{{ username }}"
id="uname">
            </div>
        </div>
        <div class="form-group">
            <label for="uname" class="col-sm-3 control-label">Пароль:</label>
            <div class="col-sm-9">
                <input type="password" name="password" id="psw">
            </div>
        </div>
        <div class="form-group">
            <div class="col-sm-offset-3 col-sm-9">
                <button type="submit" class="btn btn-default">Войти</button>
            </div>
        </div>
    </form>
</div>
{% endblock content %}

```

## authorization.html

```

{% extends 'base.html' %}

{% block title %}authorization{% endblock %}

{% block content %}
<div class="main_block_elem">
<form method="POST" class="form-horizontal">
    <h3 style="text-align:center">Авторизация пользователя</h3>
    {% csrf_token %}
    {% for field in form %}
        {% for error in field.errors %}
            <div class="alert alert-danger" role="alert">{{ field.errors
}}</div>
        {% endfor %}
    {% 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">Войти</button>
  </div>
</div>
</form>
</div>
{% endblock %}

```

## urls.py

```

from django.conf.urls import url, include
from django.contrib import admin
from labApp.views import *

urlpatterns = [
    url(r'^$', home),
    url(r'^product/$', ProductView.as_view(), name='products_url'),
    url(r'^customers/$', CustomerView.as_view(), name='customers_url'),
    url(r'^registration_form/$', registration_form, name='registration_form'),
    url(r'^registration/$', registration, name='registration'),
    url(r'^authorization_form/$', authorization_form,
        name='authorization_form'),
    url(r'^authorization/$', authorization, name='authorization'),
    url(r'^logout/$', logout_view, name='logout'),
    url(r'^success_authorization_form/$', success_authorization_form,
        name='success_authorization_form'),
    url(r'^success_authorization/$', success_authorization,
        name='success_authorization'),
    url(r'^admin/',
        include(admin.site.urls))
]

```

## Результат

### Регистрация вручную

### Переход на авторизацию вручную:

Главная Фильмы

Войти Войти(вручную) Зарегистрироваться(вручную) Зарегистрироваться

### Регистрация

Логин:

Пароль:

Повторите ввод:

Email:

Фамилия:

Имя:

Дата рождения:

Пол:

Главная Фильмы

Войти Войти(вручную) Зарегистрироваться(вручную) Зарегистрироваться

### Авторизация

Логин:

Пароль:

**Переход на главную страницу:**

Главная Фильмы

user2 Выйти

# Home

**Возможные ошибки:**

Введите логин

Введите пароль

Введите email

Введите фамилию

Введите имя

Введите дату рождения

Введите пол

Логин:

Пароль:

Повторите ввод:

Email:

Фамилия:

Имя:

Дата рождения:

Пол:

### Регистрация через Django form:

Главная Фильмы

#### Регистрация пользователя

Логин:

Пароль:

Повторите ввод:

Email:

Фамилия:

Имя:

День рождения:

Пол:   
☒ мужской   
☐ женский

### Переход на авторизацию через Django form:

Главная Фильмы

Войти Войти(вручную) Зарегистрироваться(вручную) Зарегистрироваться

### Авторизация

Логин: leonid

Пароль: .....

Войти

#### Переход на главную страницу:

Главная Фильмы

leonid Выйти

# Home

#### Возможные ошибки:

Главная Фильмы

Войти Войти(вручную) Зарегистрироваться(вручную) Зарегистрироваться

### Регистрация пользователя

Логин: 123

! Минимально допустимое количество символов: 5. Длина текста сейчас: 3.

Повторите ввод: ...

Email:

Фамилия:

Имя:

День рождения:

Пол: ☐ мужской ☐ женский

Зарегистрировать

## Создание суперпользователя:

### Django admin:

Администрирование Django

ДОБРО ПОЖАЛОВАТЬ, ASUS. ОТКРЫТЬ САЙТ / ИЗМЕНИТЬ ПАРОЛЬ

Администрирование сайта

LABAPP

Customers	+ Добавить	✎ Изменить
Orders	+ Добавить	✎ Изменить
Products	+ Добавить	✎ Изменить

ПОЛЬЗОВАТЕЛИ И ГРУППЫ

Группы	+ Добавить	✎ Изменить
Пользователи	+ Добавить	✎ Изменить

Последние действия

Мои действия

Недоступно

Администрирование Django

ДОБРО ПОЖАЛОВАТЬ, ASUS. ОТКРЫТЬ САЙТ / ИЗМЕНИТЬ ПАРОЛЬ / ВЫЙТИ

Начало &gt; Пользователи и группы &gt; Пользователи

Выберите пользователь для изменения

ДОБАВИТЬ ПОЛЬЗОВАТЕЛЯ +

ФИЛЬТР

статус персонала

Все

Да

Нет

статус суперпользователя

Все

Да

Нет

активный

Все

Да

Нет

Q

Найти

Действие: 

-----

Выполнить

 Выбрано 0 объектов из 4

4 пользователи