

# ДЗ № 5 - “Авторизация, обработка форм”

Целью домашнего задания является подключение встроенной системы авторизации Django и разработка основных форм добавления данных.

## Авторизация и регистрация пользователей

Поскольку мы используем встроенную в Django систему пользователей, имеет смысл использовать готовые методы для логина и регистрации пользователей. Не забывайте, что при создании модели стандартного пользователя User вы должны так же создать связанный Profile , в котором хранятся доп. поля.

## Необходимые формы авторизации

1. Форма логина. Располагается по URL /login/. Пользователь логин/пароль. Дополнительно в GET параметрах передается параметр continue - URL на который нужно отправить пользователя после успешной авторизации. При неудачной авторизации нужно отображать сообщение об ошибке.
2. Форма регистрации. Располагается по URL /signup/. Пользователь вводит все необходимые поля (email, имя, пароль). После успешной регистрации отправляется на главную страницу.
3. Ссылка “выход”. Располагается на каждой странице в шапке сайта. Видна только авторизованным пользователям. После выхода пользователь остается на текущей странице.
4. Форма редактирования профиля. Располагается по URL /profile/edit. Доступна только для авторизованного пользователя. Пользователь видит все необходимые поля: email, nick, avatar. После сохранения остается на странице.

## Необходимые формы добавления данных

1. Форма добавления вопроса. Располагается по URL /ask/. Пользователь вводит название, текст и тэги вопроса. После успешного добавления отправляется на страницу вопроса.
2. Форма добавления ответа. Располагается на странице вопроса /question/<id>. Пользователь вводит только текст ответа, после успешного добавления необходимо отправить пользователя на нужную старницу ответов данного вопроса и проскролить страницу так что бы добавленный ответ был виден.

## Требования предъявляемые к формам

Ко всем формам предъявляются следующие требования:

1. Обязательна валидация входных данных и вывод сообщений об ошибках

2. При выводе сообщений об ошибках введенные пользователем данные сохраняются, так что нет необходимости вводить их повторно.
3. Все формы отправляются с методом POST.
4. После успешной обработки формы пользователь отправляется на новую страницу с помощью редиректа.
5. Использовать встроенную в Django защиту от CSRF.
6. Для обработки и вывода форм использовать `django.forms`

## Полезные ссылки

1. Документация по системе авторизации  
<http://djbook.ru/rel1.7/topics/auth/default.html#using-the-django-authentication-system>
2. Как войти залогинить пользователя  
<https://docs.djangoproject.com/en/1.7/topics/auth/default/#how-to-log-a-user-in>
3. Как разлогинить пользователя  
<https://docs.djangoproject.com/en/1.7/topics/auth/default/#how-to-log-a-user-out>
4. Как зарегистрировать пользователя  
<https://docs.djangoproject.com/en/1.7/topics/auth/default/#creating-users>
5. Тьюториал по проектированию форм в Django  
<http://djbook.ru/rel1.7/intro/tutorial04.html>