**Методические указания по выполнению практического задания**

**Целью** занятия является практическое освоение теоретических знаний, полученных в процессе изучения темы 6 «Административный раздел Django. Аутентификация и авторизация в Django».

**Задачи**, реализуемые на практическом занятии:

* Закрепление представления о понятиях, введенных на лекции.
* Получение практических навыков подключения и работы в административном разделе Django в среде Visual Studio 2017.
* Получение практических навыков регистрации и авторизации пользователей в Django в среде Visual Studio 2017.

**Методические указания по выполнению практических заданий**

1. Прежде чем приступить к выполнению заданий практического занятия, необходимо изучить конспект и глоссарий к лекции.
2. Для получения навыков практической работы в административном разделе сайта:
   * в конспекте к лекции изучите последовательность действий для подключения административного раздела сайта в веб-проект Django;
   * в конспекте к лекции изучите последовательность действий для создания администратора веб-сайта – «**Создание суперпользователя Django**»;
   * в конспекте к лекции изучите последовательность действий для входа в административный раздел веб-сайта;
   * выполните [**задание 1**](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+spring_2021/courseware/e5ca12a14555455490b439be25351ed1/542b3b0c75dc4690ada49301008f39af/?child=first#%D0%97%D0%B0%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5_1).
3. Для закрепления представления о понятиях, введенных на лекции, найдите в конспекте к лекции понятия аутентификации и авторизации.
4. Для получения практических навыков регистрации и авторизации пользователей в Django:
   * прочитайте теорию для практического занятия;
   * выполните [**задание 2**](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+spring_2021/courseware/e5ca12a14555455490b439be25351ed1/542b3b0c75dc4690ada49301008f39af/?child=first#%D0%97%D0%B0%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5_2).

**Теория для практического занятия**

Для того, чтобы в свой веб-проект добавить веб-страницу регистрации пользователей, необходимо:

1. В файле **views.py** необходимо добавить использование встроенной формы регистрации — класс **UserCreationForm**. Для этого добавьте строку в начало файла в соответствующий блок:

*from django.contrib.auth.forms import UserCreationForm*

2. В файле **views.py** добавьте метод действия контроллера **registration** для обработки данных и передачи данных с сервера для отображения шаблона веб-страницы регистрации. Код следующий:

*def registration(request):*

*"""Renders the registration page."""*

 assert isinstance(request, HttpRequest)

*if request.method == "POST": # после отправки формы*

*regform = UserCreationForm (request.POST)*

*if regform.is\_valid(): #валидация полей формы*

*reg\_f = regform.save(commit=False) # не сохраняем автоматически данные формы*

*reg\_f.is\_staff = False # запрещен вход в административный раздел*

*reg\_f.is\_active = True # активный пользователь*

*reg\_f.is\_superuser = False # не является суперпользователем*

*reg\_f.date\_joined = datetime.now() # дата регистрации*

*reg\_f.last\_login = datetime.now() # дата последней авторизации*

*reg\_f.save() # сохраняем изменения после добавления данных*

*return redirect('home') # переадресация на главную страницу после регистрации*

*else:*

*regform = UserCreationForm() # создание объекта формы для ввода данных нового пользователя*

*return render(*

*request,*

*'app/registration.html',*

*{*

*'regform': regform, # передача формы в шаблон веб-страницы*

*'year':datetime.now().year,*

*}*

*)*

3. В файле **views.py** дополните строку импорта функции *render* еще и импортом функции *redirect*:

*from django.shortcuts import render, redirect*

4. Добавьте в папку проекта **app\templates\app\** файл шаблона веб-страницы **registration.html** для отображения веб-страницы с формой регистрации пользователя: в контекстном меню для папки **app** (вызывается по нажатию правой кнопки мыши) выбрать команду **Добавить** – **Создать элемент**…, далее в окне добавления нового элемента выбрать **HTML-страница,**в поле**Имя**ввести название файла **registration.html**и нажать кнопку**Добавить.**Файл будет добавлен в папку для представлений веб-проекта. Для отображения на веб-странице названий на русском языке, необходимо аналогично заданию по теме 4 сохранить страницу в кодировке UTF-8.

5. Содержимое файла **registration.html**необходимо заменить на следующий код отображения формы регистрации на веб-странице:

*{% extends "app/layout.html" %}*

*{% block content %}*

*<h2>Регистрация пользователя</h2>*

*<form method="POST" > {% csrf\_token %}*

*<div>*

*{{ regform.as\_p}}*

*<button type="submit" class="save btn btn-default" > Зарегистрировать </button>*

*</div>*

*</form>*

*{% endblock %}*

6. В файл **urls.py** необходимо добавить URL для веб-страницы **registration:**

*url(r'^registration$', app.views.registration, name='registration'),*

7. Добавьте в файл Мастер-страницы **layout.html** в блок с пунктами меню новый пункт **Регистрация**, который должен быть доступен для не авторизованных пользователей:

*{% if not user.is\_authenticated %}*

*<li><a href="{% url 'registration' %}">Регистрация</a></li>*

*{% endif %}*

Для того, чтобы зарегистрировать тестового пользователя, необходимо:

1. Запустить проект на выполнение.
2. На странице **Регистрация** ввести данные тестового пользователя в форму и нажать кнопку **Зарегистрировать**.

Примечание. Введенные тестовые данные пользователя можно посмотреть (отредактировать) в административном разделе.

Для того, чтобы изменить англоязычные названия на странице **Log in** (**Вход)**  на русскоязычные, необходимо:

1. В файлах **loginpartial.html** и **login.html** заменить англоязычные названия приветствия, ссылок, меток, кнопки "Войти", сообщений на русскоязычные.
2. В файле **urls.py** заменить заголовок **Login**, например, на **Авторизация**.
3. В файле **forms.py** англоязычные надписи полей формы заменить на русскоязычные.

**Задание 1**

Цель задания – получение практических навыков подключения и работы в административном разделе Django в среде Visual Studio 2017.

* 1. Подключить к созданному ранее веб-проекту административный раздел библиотеки Django
  2. Создать администратора сайта
  3. Войти в административный раздел и просмотреть данные администратора сайта. Выйти из административного раздела.

**Задание 2**

Цель задания – получение практических навыков регистрации и авторизации пользователей в Django в среде Visual Studio 2017.

* 1. Добавить в созданный ранее веб-проект веб-страницу регистрации пользователей, содержащую встроенную в библиотеку Django форму UserCreationForm (см. теорию в подразделе [**Теория для практического занятия**](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+spring_2021/courseware/e5ca12a14555455490b439be25351ed1/542b3b0c75dc4690ada49301008f39af/?child=first#Teor_1)).
  2. Зарегистрировать тестового пользователя (см. теорию в подразделе [**Теория для практического занятия**](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+spring_2021/courseware/e5ca12a14555455490b439be25351ed1/542b3b0c75dc4690ada49301008f39af/?child=first#Teor_1)).
  3. Изменить англоязычные названия на веб-странице **Log in** (**Вход)** на русскоязычные (см. теорию в подразделе [**Теория для практического занятия**](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+spring_2021/courseware/e5ca12a14555455490b439be25351ed1/542b3b0c75dc4690ada49301008f39af/?child=first#Teor_1)).
  4. Проверить вход тестового пользователя на сайт.