**Методические указания по выполнению практического задания**

**Целью** занятия является практическое освоение теоретических знаний, полученных в процессе изучения темы 4 «Разработка современных веб-приложений на языке программирования Python с использованием Django».

**Задачи**, реализуемые на практическом занятии:

* Закрепление представления о понятиях, введенных на лекциях 5 и 6.
* Освоение технологии создания веб-проекта Django в среде Visual Studio 2017.

**Методические указания по выполнению практического задания 1**

1. Прежде чем приступить к выполнению задания 1, необходимо просмотреть видео, изучить презентации, конспекты и глоссарий к лекции 5 и 6.
2. Для закрепления представления о понятиях, введенных на лекции 5, найдите в конспекте к лекции понятия MVС-шаблон разработки веб-приложения, модель данных, контроллер, представление. По материалам лекции 5 изучите структуру веб-проекта Django.
3. Для закрепления представления о понятиях, введенных на лекции 6, найдите в конспекте к лекции понятия базовый шаблон, наследование шаблонов. По материалам лекции 6 изучите структуру Мастер-страницы.
4. Для освоения технологии создания веб-проекта Django в среде Visual Studio 2017:

* в конспекте к лекции 5 найдите описание структуру веб-проекта Django;
* в конспекте к лекции 6 найдите описание структуры Мастер-страницы (файл **layout.html**);
* прочитайте теорию для практического занятия;
* выполните [**задание 1**](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+fall_2019/courseware/4c1a673e46674051b9d3f29af1d7051c/cbea87ffb54e41a89b46e91a98de05bf/?child=first#Задание_1).

**Теория для практического занятия**

Для создания **веб-проекта Django** в среде **Visual Studio 2017** необходимо:

1. Запустить Visual Studio 2017.
2. В меню **Файл** выбрать команду **Создать** далее выбрать **Проект**. В появившемся окне «Создание проекта»:
   * в левой части окна выбрать **Python** –> **Web**
   * в центральной части окна выбрать **Веб-проект Django**
   * внизу окна ввести имя проекта и выбрать папку для проекта
   * нажать кнопку ОК
3. Установить внешние пакеты в проект: для этого выбрать пункт «Установить в виртуальном окружении». В окне «Добавление виртуального окружения» нажать кнопку «Создать».

Для того, чтобы запустить веб-приложение на выполнение в среде Visual Studio 2017, необходимо:

1. При первом запуске можно в настройке способа отладочного запуска выбрать браузер для запуска приложения – например, Internet Explorer.
2. Запустить веб-приложение, нажав кнопку 

Для того, чтобы заменить меню, заголовки и тексты стандартных макетов страниц шаблона сайта на соответствующие русскоязычные названия и тексты, необходимо:

1. В файле settings.py заменить значение параметра LANGUAGE\_CODE на 'ru-ru'
2. Сохранить все веб-страницы (все html-файлы в папке \app\templates\app ) в кодировке **UTF-8** одним из двух альтернативных способов:
   * либо  в среде Visual Studio 2017 с помощью команды **Файл > Сохранить имя.html  как...** сохранить каждую веб-страницу в кодировке UTF-8 (в диалоговом окне **Сохранить файл как** необходимо выбрать из списка **Сохранить** команду **Сохранить с кодировкой** и далее выбрать в диалоговом окне  **Дополнительные параметры сохранения** кодировку: **Юникод(UTF-8**, с сигнатурой) и нажать **ОК**).
   * либо с помощью текстового редактора **Блокнот** все **html-страницы** сохранить в кодировке UTF-8.
3. Отредактировать код контроллера **views.py** – заменить англоязычные заголовки страниц на русскоязычные.
4. Отредактировать макеты всех страниц сайта в папке \app\templates\app. При создании разметки страниц используйте информацию по тегам HTML5 из справочника: <https://webref.ru/html>

Справочник по библиотеке **Bootstrap** доступен по ссылке: <https://webref.ru/layout/bootstrap>

Для того, чтобы поместить графический логотип сайта в верхней области страниц и, чтобы клик по логотипу приводил к переходу на главную страницу сайта, необходимо:

1. Добавить картинку логотипа сайта в папку **app\static\app\content**
2. Добавить в файле **templates\app\layout.html** в блоке <div class="navbar-header"> после блока <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">

вместо кода:

<a href="/" class="navbar-brand">Application name</a>

следующий код:

<a href="/" class="navbar-brand"> <img src="{% static 'app/content/имя\_файла\_с\_логотипом ' %}" alt="logo" /> </a>

где вместо имя\_файла\_с\_логотипом укажите имя файла картинки.

1. В файле стилей **app\static\app\content\site.css** для тега **body** подберите значение параметра **padding-top**, которое обеспечит отсутствие перекрытия шапки сайта с содержанием страниц.

**Задание 1**

Цель задания – освоить технологию создания веб-проекта Django в среде Visual Studio 2017.

1. Создать веб-проект Django (см. теорию в подразделе [**Теория для практического занятия**](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+fall_2019/courseware/4c1a673e46674051b9d3f29af1d7051c/cbea87ffb54e41a89b46e91a98de05bf/?child=first#03)).
2. Запустить веб-приложение на выполнение (см. теорию в подразделе [**Теория для практического занятия**](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+fall_2019/courseware/4c1a673e46674051b9d3f29af1d7051c/cbea87ffb54e41a89b46e91a98de05bf/?child=first#03)). Завершить работу веб-приложения.
3. Изучить структуру веб-проекта в «Обозревателе решений» (см. материалы лекции 5).
4. Заменить меню, заголовки и тексты стандартных макетов страниц шаблона веб-сайта на соответствующие русскоязычные названия и тексты в соответствии с выбранной темой (см. теорию в подразделе [**Теория для практического занятия**](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+fall_2019/courseware/4c1a673e46674051b9d3f29af1d7051c/cbea87ffb54e41a89b46e91a98de05bf/?child=first#03)).
5. Найти или создать самостоятельно графический файл (с расширением jpg или png) для использования в качестве логотипа Вашего веб-сайта. Поместить графический логотип сайта в верхней области страниц. Клик по логотипу должен приводить к переходу на главную страницу сайта. (см. теорию в подразделе [**Теория для практического занятия**](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+fall_2019/courseware/4c1a673e46674051b9d3f29af1d7051c/cbea87ffb54e41a89b46e91a98de05bf/?child=first#03)).
6. Проверить правильность отображения контента веб-страниц в браузере и возможность перехода на главную страницу при нажатии на логотип.

**Разбор задач, пошаговые примеры**

**Пошаговый пример решения задания**

Рассмотрим пример решения задания 1 практического занятия по теме 4.

1. Из теории в подразделе **Теория для практического занятия** следует, что первоначально необходимо:

а. Запустить на своем компьютере **Visual Studio 2017**.

b. Далее в меню **Файл** выбрать команду **Создать**, далее выбрать **Проект**. В появившемся окне «**Создание проекта**» (см. рисунок 1):

* в левой части окна выбрать **Python** –> **Web**
* в центральной части окна выбрать **Веб-проект Django**
* внизу окна ввести **имя проекта** и выбрать **папку** для проекта
* нажать кнопку **ОК**

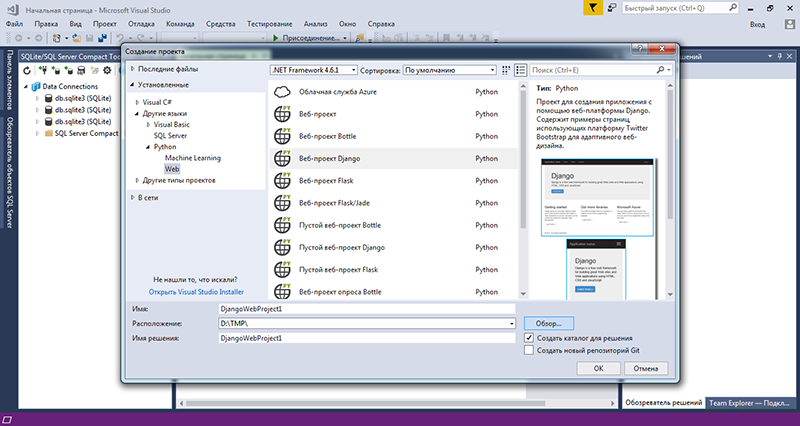


Рисунок 1 – Окно «Создание проекта»

с. Далее установить внешние пакеты в проект: для этого выбрать пункт «Установить в виртуальном окружении» (см. рисунок 2). В окне «Добавление виртуального окружения» нажать кнопку «Создать» (см. рисунок 3).

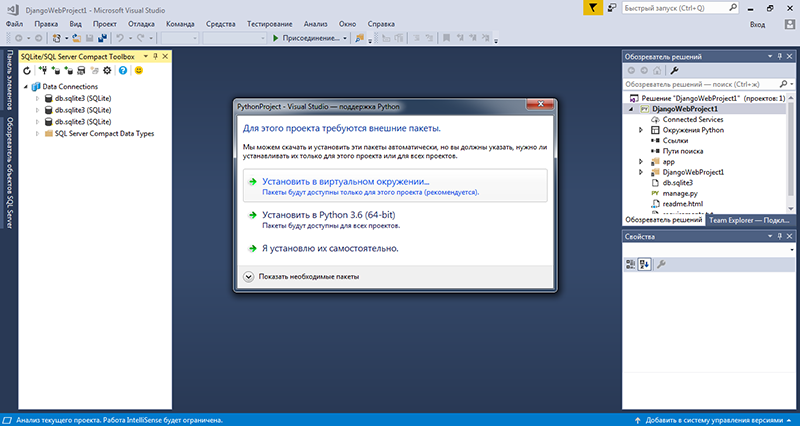


Рисунок 2 – Установка внешних пакетов в виртуальном окружении

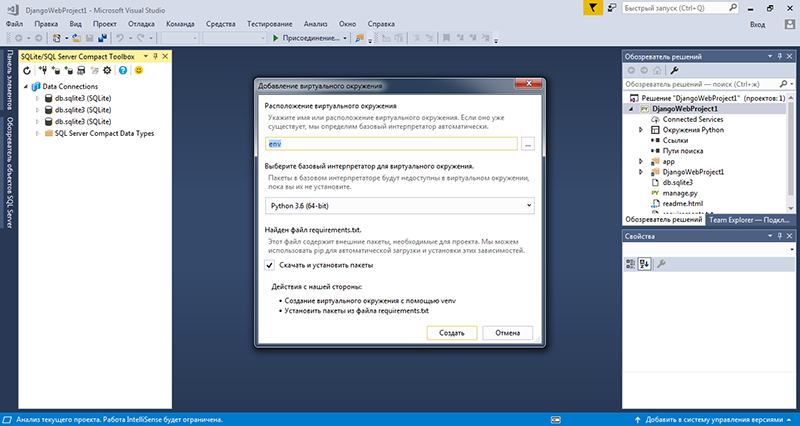


Рисунок 3 –Добавление виртуального окружения

2. Из теории в подразделе **Теория для практического занятия** следует, что при первом запуске веб-приложения можно в настройке способа отладочного запуска **выбрать браузер** для запуска приложения – например, Internet Explorer (см. рисунок 4).

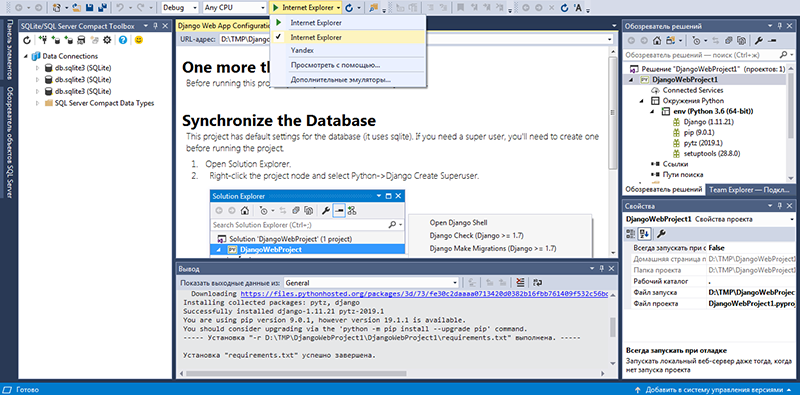


Рисунок 4 – Выбор браузера

Затем запустить веб-приложение, нажав кнопку .

Для завершения работы веб-приложения требуется нажать кнопку закрытия окна браузера, а затем на панели инструментов Visual Studio 2017 нажать кнопку **Остановить отладку** (см. рисунок 5).

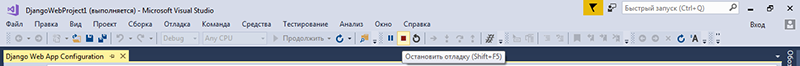


Рисунок 5 – Остановить отладку

3. На основе материалов конспекта лекции 5, презентации к лекции 5 в окне «Обозреватель решений» посмотреть структуру веб-проекта Django. Открыть и посмотреть содержание файлов:

* + модели models.py;
  + контроллера views.py;
  + представлений в папке \app\templates\app – about.html, contact.html, index.html;
  + Мастер-страницы – layout.html;
  + urls.py;
  + settings.py

4. Из теории в подразделе **Теория для практического занятия** следует, что для того, чтобы заменить меню, заголовки и тексты стандартных макетов страниц шаблона сайта на соответствующие русскоязычные названия и тексты, необходимо:

a. В файле **settings.py** заменить:

LANGUAGE\_CODE = 'en-us'

на

LANGUAGE\_CODE = 'ru-ru'

b. Был выбран путь сохранения html-файлов в стандартном приложении Windows **Блокнот** в кодировке UTF-8, поэтому необходимо закрыть веб-проект, выбрав в меню **Файл** команду **Закрыть решение**.

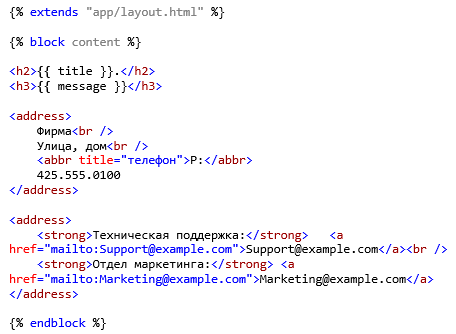
c. Затем необходимо каждый html-файл (about.html, contact.html, index.html, layout.html, login.html, loginpartial.html) из папки веб-проекта  **\app\templates\app** открывать в приложении **Блокнот**. А затем выбирать в меню **Файл** команду **Сохранить как**  и в диалоговом окне выбирать кодировку **UTF-8** и нажимать кнопку **Сохранить**.

d. Снова  открыть свой веб-проект в Visual Studio 2017 .

e. В файле views.py  названия заголовков страниц, сообщений заменить в соответствующих методах действия контроллера на русскоязычные:

* 'title':'Главная',
* 'title':'Контакты',
* 'message':'Страница с нашими контактами.',
* 'title':'О нас',
* 'message':'Сведение о нас.',

f. В html-файлах необходимо заменить англоязычные тексты (названия) на русскоязычные в соответствии с выбранной темой для веб-сайта. Например, код файла contact.html может быть заменен на следующий:



5. Из теории в подразделе **Теория для практического занятия** следует, что для того, чтобы поместить найденный графический логотип сайта в верхней области страниц и, чтобы клик по логотипу приводил к переходу на главную страницу сайта, необходимо:

a. Найти у себя на компьютере или в сети интернет или самостоятельно создать в графическом редакторе, подходящий для логотипа вашего веб-сайта файл с расширением jpg или png. Например, файл **logo.jpg.**

b. Скопировать этот файл в свой веб-проект в папку **app\static\app\content**

c. В файле **templates\app\layout.html** заменить:

<a href="/" class="navbar-brand">Application name</a>

На следующий код:

<a href="/" class="navbar-brand"> <img src="{% static 'app/content/ logo.jpg ' %}" alt="logo" /> </a>

d. В файле стилей **app\static\app\content\site.css** для тега **body** подобрать значение параметра **padding-top**, которое обеспечит отсутствие перекрытия шапки сайта с содержанием страниц.

Например, было изначально:

padding-top: 50px;

требуется изменить на:

padding-top: 80px;

6. Для проверки правильности отображения веб-страниц в браузере необходимо:

* запустить веб-приложение на выполнение;
* посмотреть корректно ли отображается русскоязычный контент веб-страниц в браузере;
* проверить переход по логотипу на главную страницу.

**Дополнительный материал**

**Добавление новой веб-страницы в веб-проект, который создан в среде Visual Studio 2017:**

* 1. Первоначально необходимо добавить макет новой веб-страницы в папку **templates\app** (используя команду **«Добавить > Создать элемент... — HTML-страница»** контекстного меню папки **templates\app**) файл, например, **links.html**. Изменить содержание файла в соответствии с содержанием шаблона Django для страницы представления.
  2. Для добавления изображений на страницу необходимо добавить файлы картинок в папку:  **app\static\app\content**.
  3. Для загрузки статических файлов добавьте в файл **links.html** в блок *content* строку:  
     *{% load staticfiles %}*
  4. Для размещения в файле **links.html** гиперссылок с логотипами веб-ресурсов по теме проекта необходимо добавить записи в следующем формате:  
     *<a href ="URL-ресурса 1" target="\_blank"> <img src="{% static 'app/content/имя\_файла\_с\_логотипом ресурса 1' %}" alt="Logo 1"/></a>*
  5. Добавьте в файл ***views.py*** по аналогии с имеющимися там методами метод с заголовком:  
     *def links (request):*
  6. В файл **templates\app\layout.html** добавьте в список пунктов меню (тег *<ul>*) строку для веб-страницы **links.html**—**Полезные ресурсы**(по аналогии с другими пунктами меню).
  7. В файл **urls.py** необходимо добавить URL для веб-страницы **links.html**, которая содержит гиперссылки с логотипами интернет-ресурсов (по аналогии с имеющимися там URL на другие веб-страницы сайта).

**Задание для самостоятельной работы**

**Цель самостоятельного задания**: Получение навыков подбора интернет-ресурсов по определенной тематике и размещения ссылок на выбранные ресурсы на странице собственного веб-сайта.

1. Выберите, например, три веб-сайта, тематика которых связана с содержанием созданного Вами сайта.
2. Добавьте в веб-проект сайта страницу «Полезные ресурсы», на которой разместите гиперссылки с логотипами выбранных веб-ресурсов (на основе размещенного выше дополнительного материала по теме).
3. Запустить веб-приложение на выполнение. Проверить правильность отображения контента веб-страницы "Полезные ресурсы" в браузере и возможность перехода по ссылкам на интернет-ресурсы.