**Методические указания по выполнению практического занятия**

**Целью** занятия является практическое освоение теоретических знаний, полученных в процессе изучения темы «Разработка онлайн-журнала (часть 2)».

**Задачи**, реализуемые на практическом занятии:

* + Закрепление представления о понятиях, введенных на лекции.
  + Получение практических навыков добавления медиа-контента на сайт, созданного в среде Visual Studio 2017.

**Методические указания по выполнению практического задания 1**

* 1. Прежде чем приступить к выполнению задания 1, необходимо просмотреть видео, изучить презентацию, конспект и глоссарий к лекции .
  2. Для закрепления представления о понятиях, введенных на лекции , найдите в конспекте к лекции понятия позиционирование изображения, кодек, WYSIWYG редактор. По материалам лекции изучите способы позиционирования изображений на веб-странице.
  3. Для получения практических навыков добавления медиа-контента на сайт, созданного в среде Visual Studio 2017:
  + в конспекте к лекции найдите описание действий для реализации загрузки изображений на сайт через форму добавления статьи блога и описание способов добавления видео-файлов на сайт;
  + прочитайте теорию для практического занятия;
  + выполните [задание 1](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+spring_2021/courseware/8a309738de5c42f6b3a7c8d3c435691b/4b8ee56d02784981a8b203690df5660e/?child=first#yakor_1).

**Теория для практического занятия**

Для того, чтобы реализовать на отдельной веб-странице сайта добавление статей блога администратором,необходимо:

* 1. Добавить в начало файла **forms.py** строку импорта модели блога.
  2. Добавить в конец файла **forms.py** код для формы ввода статьи блога по аналогии с формой ввода комментария. Форма должна содержать поля для ввода заголовка, краткого содержания, выбора файла картинки и полного содержания статьи.
  3. Добавить в проект шаблон для html-файла для ввода статьи блога (например, newpost.html).
  4. Добавить отображение формы ввода статьи блога в newpost.html. В тег формы добавьте атрибут **enctype="multipart/form-data"**.
  5. Добавить в начало файла **views.py** код импорта формы для ввода статьи блога.
  6. В файл **views.py** добавьте код метода действия контроллера **def newpost(request):**

Данный метод действия контроллера должен содержать код по обработке формы ввода статьи блога по аналогии с кодом обработки формы ввода комментария в методе действия контроллера страницы поста.

Переменную формы необходимо создавать следующим вызовом:

*form = BlogForm(request.POST, request.FILES)*

(Такой вариант используется при загрузке файлов через форму).

Для переадресации на страницу блога после добавления нового поста, необходимо возвращать метод **redirect**(‘blog’).

7. Добавить строку шаблона URL в файл urls.py для страницы добавления статьи блога.

8. Добавить пункт в меню сайта для страницы добавления статьи блога, доступный только для администратора (метод is\_superuser).

9. Запустить проект на выполнение. Войти на сайт под администратором. Добавить статью блога с картинкой, добавить к ней комментарий. Проверить отображение новой статьи и комментария.

Для того, чтобы в начале страницы статьи блога разместить картинку, загружаемую через форму ввода статьи,необходимо:

1. В файл **blogpost.html** добавьте вывод картинки записи блога перед содержанием статьи:

*{% if post\_1.image and post\_1.image.url != '/media/temp.jpg' %}*

*<img src="{{ post\_1.image.url }}" alt="* *Картинка статьи блога" />*

*{% endif %}*

Для того, чтобы реализовать на отдельной веб-странице сайта отображение и воспроизведение двух видео-фрагментов формата .mp4 и длительностью не более 1 минуты,необходимо:

* 1. Первоначально подготовить (или найти) 2 видео-фрагмента по тематике сайта формата .mp4, длительностью не более 1 минуты и картинки к ним для заставки (poster).
  2. Скопировать подготовленные 2 видео-файла и картинки к ним (poster) в папку media веб-проекта.
  3. Добавить пункт в меню сайта для страницы **Видео**, по аналогии с другими добавленными страницами.
  4. Добавить строку шаблона URL в файл **urls.py** для страницы Видео, по аналогии с другими добавленными страницами.
  5. Добавить в файл **views.py** код метода действия контроллера **def videopost(request):** (по аналогии с методом about).
  6. Добавить в проект **шаблон для html-файла** для видео-контента (например, videopost.html).
  7. Добавить код для отображения и воспроизведения видео-роликов в videopost.html (см. материалы лекции ).
  8. Запустить проект на выполнение. Проверить отображение и воспроизведение видео-роликов на странице сайта.

**Задание 1**

Цель задания – получить практические навыки добавления медиа-контента на сайт, созданного в среде Visual Studio 2017.

1. Добавить в модель **Blog** поле **image** для хранения изображения поста (см. материалы лекции ).
2. Реализовать на отдельной веб-странице сайта добавление статей блога администратором. После добавления статьи должна быть реализована переадресация на страницу блога. На странице статьи блога разместить в начале статьи картинку, загружаемую через форму ввода статьи (см. материалы лекции и теорию в подразделе [Теория для практического занятия](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+spring_2021/courseware/8a309738de5c42f6b3a7c8d3c435691b/4b8ee56d02784981a8b203690df5660e/?child=first#10)).
3. Реализовать на отдельной веб-странице сайта (например, «Видео») отображение и воспроизведение двух видео-фрагментов формата .*mp4* идлительностью не более 1 минуты. Видео-файлы должны быть предварительно загружены в папку *media* веб-проекта (см. материалы лекции и теорию в подразделе [Теория для практического занятия](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+spring_2021/courseware/8a309738de5c42f6b3a7c8d3c435691b/4b8ee56d02784981a8b203690df5660e/?child=first#10)).