**Методические указания по выполнению практического задания**

**Целью** занятия является практическое освоение теоретических знаний, полученных в процессе изучения темы 8 «Разработка онлайн-журнала (часть 1)».

**Задачи**, реализуемые на практическом занятии:

* Получение практических навыков добавления комментариев к статьям блога в среде Visual Studio 2017.

**Методические указания по выполнению практического задания 2**

1. Для получения практических навыков добавления комментариев к статьям блога в среде Visual Studio 2017:

* прочитайте теорию для практического занятия;
* выполните [задание 2](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+spring_2021/courseware/ad8be13d20c7444297348c709cb900b5/d0f2166a97d243e0aec5fd5d2625bdda/?child=first#yakor_1).

**Теория для практического занятия**

Для того, чтобы добавить форму ввода комментариев на веб-страницу поста и отобразить на веб-странице поста все её комментарии, необходимо:

1. Добавить в начало файла **forms.py** следующий код:

*from django.db import models*

*from .models import Comment*

2. Добавить в конец файла **forms.py** следующий код для формы ввода комментария:

*class CommentForm (forms.ModelForm):*

*class Meta:*

*model = Comment # используемая модель*

*fields = ('text',) # требуется заполнить только поле text*

*labels = {'text': "Комментарий"} # метка к полю формы text*

3. Отобразить форму ввода комментария в шаблоне веб-страницы **blogpost.html**. Для этого добавить в файл **blogpost.html** после отображения текста статьи блога следующий код:

*{% if user.is\_authenticated %}*

*<h3>Добавить комментарий</h3>*

*<form method="POST" >{% csrf\_token %} {# защита от атак CSRF#}*

*<div>*

*{{form.as\_p}}*

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

*</div>*

*</form>*

*{% else %}*

*<h3> Для добавления своего комментария, пожалуйста, авторизуйтесь </h3>*

*{% endif %}*

4. В файл **blogpost.html** после кода отображения формы ввода комментария добавить код вывода всех комментариев к статье блога:

*{% for comment in comments %}*

*<hr/>*

*<h7> {{comment.date}} </h7>*

*<h5> {{comment.author}} </h5>*

*<p> {{comment.text|linebreaksbr}} </p>*

*{% endfor %}*

5. Добавить в начало файла **views.py** следующий код:

*from .models import Comment # использование модели комментариев*

*from .forms import CommentForm # использование формы ввода комментария*

6. В файле **views.py** в код метода контроллера **def blogpost (request, parametr):** после запроса на выбор конкретной статьи добавить запрос на выбор всех её комментариев:

*comments = Comment.objects.filter(post=parametr)*

7. В файле **views.py** в код метода контроллера **def blogpost (request, parametr):** после запроса на выбор всех комментариев к конкретной статье добавить код работы с формой добавления комментария и добавления недостающих данных в модель:

*if request.method == "POST": # после отправки данных формы на сервер методом POST*

*form = CommentForm(request.POST)*

*if form.is\_valid():*

*comment\_f = form.save(commit=False)*

*comment\_f.author = request.user # добавляем (так как этого поля нет в форме) в модель Комментария (Comment) в поле автор авторизованного пользователя*

*comment\_f.date = datetime.now() # добавляем в модель Комментария (Comment) текущую дату*

*comment\_f.post = Blog.objects.get(id=parametr) # добавляем в модель Комментария (Comment) статью, для которой данный комментарий*

*comment\_f.save() # сохраняем изменения после добавления полей*

*return redirect('blogpost', parametr=post\_1.id) # переадресация на ту же страницу статьи после отправки комментария*

*else:*

*form = CommentForm() # создание формы для ввода комментария*

8. В файле **views.py** в код метода контроллера **def blogpost (request, parametr):** в методе **render** добавить передачу объектов формы добавления комментария и всех комментариев к конкретной статье в шаблон веб-страницы **blogpost.html**:

*'comments': comments, # передача всех комментариев к данной статье в шаблон веб-страницы*

*'form': form, # передача формы добавления комментария в шаблон веб-страницы*

**Задание 2**

Цель задания – получить практические навыки добавления комментариев к статьям блога в среде Visual Studio 2017.

1. Добавить в веб-проект Django модель для комментариев к статьям блога (модель *Comment*) по аналогии с добавленной в практическом задании 7 моделью блога. Создать таблицу в БД веб-проекта db.sqlite3 для сохранения комментариев к статьям блога. Каждый комментарий должен иметь:

a. текст комментария (text);

b. дату добавления комментария (date);

c. имя пользователя, который добавил комментарий (author);

d. статья для комментария (post).

2. Добавить в веб-проект Django форму для ввода комментария к статье блога. Необходимо реализовать на сайте добавление комментариев к статьям блога только авторизованными пользователями. Все оставленные комментарии к статье блога должны отображаться на странице с соответствующей статьей блога в порядке убывания даты.  (см. теорию в подразделе [Теория для практического занятия](https://courses.openedu.ru/courses/course-v1:spbstu+WEBPYT+spring_2021/courseware/ad8be13d20c7444297348c709cb900b5/d0f2166a97d243e0aec5fd5d2625bdda/?child=first#09)).

3. Реализовать индивидуальное оформление отображаемых комментариев.

4. Запустить проект на выполнение. Авторизоваться. Добавить комментарии к статьям блога. Проверить корректность отображения введенных зарегистрированным пользователем комментариев к статьям блога.