Московский государственный технический университет им. Н.Э. Баумана

Разработка интернет-приложений Лабораторная работа № 5 "Шаблонизация"

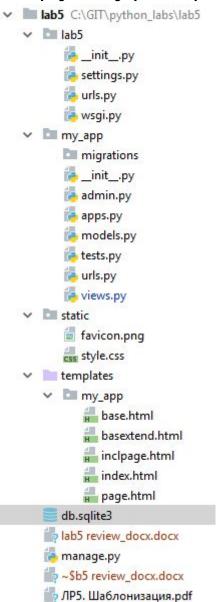
Выполнил: студент группы ИУ5-53 Степанов Д. Г. Подпись: Дата:

Задание и порядок выполнения

В этой ЛР вы создадите Django-проект, покажете пользователю статичную страницу, познакомитесь с конструкциями шаблонизаторов: переменные, теги, наследование шаблонов.

- Создать проект
- Реализовать view, в которых генерируются html-страницы
- В шаблонах должны быть использованы рассмотренные конструкции: переменные, вложенные значения, циклы, условия
- Все шаблоны должны расширять базовый шаблон
- Для элементов списка использовать тег include
- По нажатии на элемент списка должна открываться страница информации об элементе
- Для верстки необходимо использовать Bootstrap

Структура проекта



Файл views.py

```
from django.shortcuts import render
   from django.http import HttpRequest, HttpResponse
   import datetime
  from django.views import View
 def basepage (request):
      return render(request, "my_app/base.html")
  def basextend(request):
      colors = {"black": "#000000", "white": "#FFFFFF", "red": "#FF0000", "green": "#00FF00", "blue": "#0000FF"}
      return render (request, "my_app/basextend.html", colors)
   def testpage (request):
      pages = {'pages':[
                {'title': "Первая страница", "id": 1},
                {'title': "Вторая страница", "id": 2},
                {'title': "Третья страница", "id": 3}
      server time = datetime.datetime.now()
      return render (request, "my app/index.html", pages)
   def function_view(request):
      return HttpResponse ("Response from fucntion view")
   class ExampleClassBased(View):
       def get (self, request):
           return HttpResponse("response from class based view")
   class PageView(View):
      def get(self, request, id):
           data = {'page': {'id': id}}
           return render(request, 'my_app/page.html', data)
  class IncludePage (View):
       def get(self, request):
           return render(request, 'my_app/inclpage.html')
Файл urls.pv
from django.conf.urls import url, include
from django.contrib import admin
from my app.views import function_view, testpage, ExampleClassBased, basepage, basextend, PageView, IncludePage
urlpatterns = [
    url(r'^basextend/', basextend, name='basextend'),
    url(r'^basepage/', basepage, name='basepage'),
    url(r'^testpage/', testpage, name='testpage'),
    url(r'^function_view/', function_view, name='function_view'),
    url(r'^class based view/', ExampleClassBased.as_view(), name='class based view'),
    url(r'^page/(?P<id>\d+)', PageView.as_view(), name='page'),
    url(r'^inclpage/', IncludePage.as_view(), name= "includepage"),
 url(r'^$', testpage, name='testpage')
```

Базовый шаблон base.html

```
🎁 views.py 🗶 🐞 urls.py 🗶 🙀 base.html 🗶 🕌 index.html 🗶 🔒 p
       <! DOCTYPE html>
2
     <html lang="en">
3
     <head>
           <meta charset="UTF-8">
5 1
           <title>{% block title %}{% endblock %}</title>
6 0
           {% block head %}{% endblock %}
      </head>
7
      <body>
8
9 1
      {% block content %}{% endblock %}
10
       {% block footer %} {% endblock %}
11
     ⊕</body>
12
      </html>
```

Начальная страница (пример использования тегов, наследования, bootstrap, подключения static-файлов, include)

```
{% extends 'my app/base.html' %}
{% load static %}
{% block title %} My app {% endblock %}
{% block head %}
    <meta charset="UTF-8">
    <link rel = "icon" type = "image/png" href="{\s static 'favicon.png' \s}">
    <link rel = "stylesheet" href ="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/</pre>
    <link rel = "stylesheet" href ="{\{\partial static '/style.css' \{\partial \}}">
{ % endblock % }
{ * block content *}
<div class = "container">
   <div class="row">
        <div class = "col-lg-4 col-lg-offset-4">
            <h2 class = "title">Начальная Страница</h2>
            <h2 class = "title">{{ server time }}</h2>
            {% for page in pages %}
            <a href = "{\star url 'page' page.id \star ">{{ page.title }}</a>
            { \ empty \ \ }
            Пустая ссылка
            {% endfor %}
        </div>
    </div>
</div>
    <l
        {% for page in pages %}
            {\$ include 'my app/inclpage.html' with element=page \$}
        { * empty * }
            Iycтой список
        {% endfor %}
    <script src ="https://code.jquery.com/jquery-2.1.4.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="</pre>
{\ endblock \ \}
```

Использование циклов

Страница с include

Результат выполнения программы

← → C ① 127.0.0.1:8000

Начальная Страница Nov. 3, 2017, 7:46 p.m.

- Первая страница
- Вторая страница
- Третья страница

- · Первая страница included page
- Вторая страница included page
- Третья страница included page



response from class based view



Page number 1



Response from fucntion view