

Защищено:

Гапанюк Ю. Е.

"__" _____ 2016 г.

Демонстрация:

Гапанюк Ю. Е.

"__" _____ 2016 г.

**Отчет по лабораторной работе №5
по курсу «Разработка интернет приложений»**

6

(количество листов)

ИСПОЛНИТЕЛЬ:

Студент группы ИУ5-54 _____

Повираева М. Л.

(подпись)

"__" _____ 2016 г.

Оглавление

Описание задания лабораторной работы.....	3
Реализация поставленной задачи	3
Листинг программы example.py	3
Листинг программы views.py	3
Листинг программы urls.py	4
Листинг программы layout.html	4
Листинг программы main.html	5
Листинг программы post.html	5
Листинг программы post_short.html.....	6
Результат работы программы	6

Описание задания лабораторной работы

В этой ЛР нужно создать Django-проект, показать пользователю статичную страницу, познакомиться 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 нажатию на элемент списка должна открываться страница информации об элементе.
- Для верстки необходимо использовать Bootstrap.

Реализация поставленной задачи

Листинг программы example.py

```
posts = [
    {
        'id': i,
        'title': 'Заголовок {}'.format(i),
        'description': 'Краткая информация'.format(i),
        'text': 'Подробная информация'.format(i)
    } for i in range(1, 9)
]

posts_dict = {val.get('id'): val for val in posts}
```

Листинг программы views.py

```
from django.shortcuts import render
from django.http import Http404
from django.views import View
from lab.example import posts, posts_dict

def main(request):
    return render(request, 'main.html', {
        'posts': posts
    })

class PostView(View):
    def get(self, request, id):
        post = posts_dict.get(int(id))

        if post is None:
            raise Http404

        return render(request, 'post.html', {
            'post': post
        })
```

Листинг программы urls.py

```
"""lab5 URL Configuration

The `urlpatterns` list routes URLs to views. For more information please see:
    https://docs.djangoproject.com/en/1.10/topics/http/urls/
Examples:
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  url(r'^$', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  url(r'^$', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.conf.urls import url, include
    2. Add a URL to urlpatterns:  url(r'^blog/', include('blog.urls'))
"""
from django.conf.urls import url
from django.contrib import admin
from lab import views

urlpatterns = [
    url(r'^$', views.main, name='main'),
    url(r'^post/(?P<id>\d+)\$', views.PostView.as_view(), name='post'),
    url(r'^admin/', admin.site.urls)
]
```

Листинг программы layout.html

```
{% load static %}
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>{% block title %}{% endblock %}</title>

    <link href="{% static 'css/bootstrap.min.css' %}" rel="stylesheet">
    <link href="{% static 'css/my.css' %}" rel="stylesheet">
</head>

<body>

<nav class="navbar navbar-static-top navbar-dark bg-inverse">
    <a class="navbar-brand" href="{% url 'main' %}">Разработка интернет приложений</a>
</nav>

<div class="jumbotron">
    <div class="container">
        <h1 class="display-3">{% block title_visible %}Hello, world!{% endblock %}</h1>
    </div>
</div>
```

```

<div class="container">
  {% block body %}{% endblock %}

  <hr>

  <footer>
    <p>&copy; BMSTU 2016</p>
  </footer>
</div>

<script src="{% static 'js/jquery-3.1.1.slim.min.js' %}"></script>
<script src="{% static 'js/bootstrap.min.js' %}"></script>
</body>
</html>

```

Листинг программы main.html

```

{% extends 'layout.html' %}

{% load fltr %}

{% block title %}
  Main page
{% endblock %}

{% block title_visible %}
  Лабораторная работа №5
{% endblock %}

{% block body %}
  {% for p in posts %}
    {% if p.id|divisibleby:2 %}
      <div class="row">
        {% endif %}
        <div class="col-md-6">
          {% include 'post_short.html' with post=p %}
        </div>
        {% if p.id|divisibleby:2 or forloop.last %}
          </div>
        {% endif %}

        {% empty %}
          The list is empty.
        {% endfor %}

  {% endblock %}

```

Листинг программы post.html

```

{% extends 'layout.html' %}

{% block title %}
  Post {{ post.title }}
{% endblock %}

{% block title_visible %}
  {{ post.title }}
{% endblock %}

```

```
{% block body %}
    {{ post.text }}
{% endblock %}
```

Листинг программы post_short.html

```
{% with id=post.id %}
<h2><a href="{% url 'post' id=id %}">{{ post.title }}</a></h2>
<p>{{ post.description }}</p>
{% endwith %}
```

Результат работы программы

