

Защищено:
Гапанюк Ю. Е.

Демонстрация:
Гапанюк Ю. Е.

"__" _____ 2016 г.

"__" _____ 2016 г.

**Отчет по лабораторной работе № 5 по курсу
Разработка интернет-приложений**

Тема работы: "Шаблонизация"

4
(количество листов)
Вариант № 23

ИСПОЛНИТЕЛЬ:

студент группы ИУ5-54

Тимошенко В.А.

(подпись)

"__" _____ 2016 г.

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

1) Листинг

urls.py

```
from bk.views import index, ExampleView, OrdersView, OrderView
```

```
from django.conf.urls import url
urlpatterns = [
    url(r'^$', index),
    url(r'^example$', ExampleView.as_view()),
    url(r'^order$', OrdersView.as_view(), name='orders_url'),
    url(r'^order/(?P<id>\d+)$', OrderView.as_view(), name='order_url'),
]
```

views.py

```
from django.http import HttpResponse
from django.shortcuts import render
from django.views import View
def index(request):
    return HttpResponse("Hello, welcom in our Bukmekerskuy kontoru 'Sorvi Kush'.")
class ExampleView(View):
    def get(self, request):
        return render(request, 'example.html', {'my_variable': 'Здесь объявляется переменная', 'list': [1,2,3]})
class OrdersView(View):
    def get(self, request):
        data = {
            'orders': [
                {'title': 'Первый заказ', 'id': 1},
```

```

        {'title': 'Второй заказ',
'id': 2},
        {'title': 'Третий заказ', 'id': 3},
    ]
}
return render (request, 'orders.html', data)
class OrderView(View):
    def get(self, request, id):
        data = {
            'orders': [
                {'id':int(id)}
            ]
        }
        print(data)
        return render(request, 'orders.html', data)

```

base.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    {% load static %}
    <link rel="stylesheet" type="text/css" href="{% static 'css/bootstrap.css'%}">
    <title>{% block title %}{% endblock %}</title>
</head>
<body>
<nav class="navbar navbar-default">
    <div class="container-fluid">
        <!-- Brand and toggle get grouped for better mobile display -->
        <div class="navbar-header">
            <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
                <span class="sr-only">Toggle navigation</span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </button>
            <a class="navbar-brand" href="#">Brand</a>
        </div>
    </div><!-- /.container-fluid -->
</nav>
<div class="container-fluid">
    <div class="row">
        {% block body%} Переопределяется в наследниках {% endblock %}
    </div>
</div>
</body>
</html>

```

example.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>
    static page => {{my_variable}}
    <ul>
        {% for element in list %}

```

```

        {% if element == 2 %}
            <li>Двойка</li>
        {% elif element == 3 %}
            <li>Тройка</li>
        {% else %}
            <li> {{element}} </li>
        {% endif %}
    {% empty %}
        пустой список
    {% endfor %}
</ul>
</body>
</html>

```

order_element.html

```

<li><a href="{% url 'order_url' id=order.id %}">{{ order.title }}</a></li>
{% block body %} Страница заказа №{{ order.id }}{% endblock %}
{{ conte }}

```

orders.html

```

{% extends 'base.html' %}
{% block title %}Заказы №{{order.id}}{% endblock %}
{% block body %}
<div class="col-md-4">
    <div> Общая часть </div>
    <ul>
        {% for order in orders %}
            {%include 'order_element.html' with order=order %}
        {% empty %}
            <li> пустой список </li>
        {% endfor %}
    </ul>
</div>
{% endblock %}

```

2) Результат



