

Отчёт по лабораторной работе №5  
по курсу «Разработка интернет-приложений»  
Шаблонизация

Выполнил:

Волобуев В.Н., ИУ5-54

---

Преподаватель:

Гапанюк Ю.Е.

---

2016 г.

## 1) Задание лабораторной работы

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

## 2) Листинг

```
# urls.py

from django.conf.urls import url

from . import views

urlpatterns = [
    url(r'^function_view/', views.function_view),
    url(r'^class_based_view/',
views.ExampleClassBase.as_view()),
    url(r'^example_view/', views.ExampleView.as_view()),
    url(r'^orders_view/', views.OrdersView.as_view()),
    url(r'^empty_orders_view/',
views.EmptyOrdersView.as_view()),
    url(r'^order/(?P<id>\d+)', views.OrderView.as_view(),
name='order_url'),
]
```

```
# views.py

from django.shortcuts import render
from django.http import HttpResponse
from django.views import View

# Create your views here.

def function_view(request):
    return HttpResponse('response form function view')

class ExampleClassBase(View):
    def get(self, request):
        return HttpResponse('response from class based view')

class ExampleView(View):
    def get(self, request):
        return render(request, 'base.html')

class OrdersView(View):
    def get(self, request):
        data = {
            'orders': [
                {'title': 'First order', 'id': 1},
                {'title': 'Second order', 'id': 2},
                {'title': 'Third order', 'id': 3},
            ]
        }
        return render(request, 'orders.html', data)

class EmptyOrdersView(View):
    def get(self, request):
        data = {
            'orders': []
        }
```

```
return render(request, 'orders.html', data)
```

```
class OrderView(View):
```

```
    def get(self, request, id):
```

```
        data = {
```

```
            'order': {
```

```
                'id': id
```

```
            }
```

```
        }
```

```
        return render(request, 'order.html', data)
```

```
# base.html
```

```
{% load static %}
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
    <meta charset="UTF-8">
```

```
    <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">
```

```
        <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">IAD #5</a>
```

```
        </div>
```

```

        <div class="collapse navbar-collapse" id="bs-example-
navbar-collapse-1">
            <ul class="nav navbar-nav">
                <li><a href="/App5/example_view/">Base view</a></li>
                <li><a href="/App5/orders_view/">Orders</a></li>
                <li><a href="/App5/empty_orders_view/">Empty List
Orders</a></li>
            </ul>
        </div><!-- /.navbar-collapse -->
    </div><!-- /.container-fluid -->
</nav>

    <div>
        {% block body %}Redefined by spawnlings{% endblock %}
    </div>
</body>
</html>

```

#### **# inc.html**

```
{{ verb }}
```

#### **# order.html**

```
{% extends 'base.html' %}
```

```
{% block title %}Заказ № {{ order.id }}{% endblock %}
```

```
{% block body %}Страница заказа №{{ order.id }}{% endblock %}
```

#### **# orders.html**

```
{% extends 'base.html' %}
```

```
{% block title %}Заказы{% endblock %}
```

```
{% block body %}
```

```
    <ul class="list-group">
```

```
        {% for order in orders %}
```

```

        <li class="list-group-item">{% include 'inc.html'
with verb="TEST " only %}<a href="{% url 'order_url' order.id
%}">{{ order.title }}</a></li>

        {% empty %}

        <li class="list-group-item disabled">empty
list</li>

        {% endfor %}

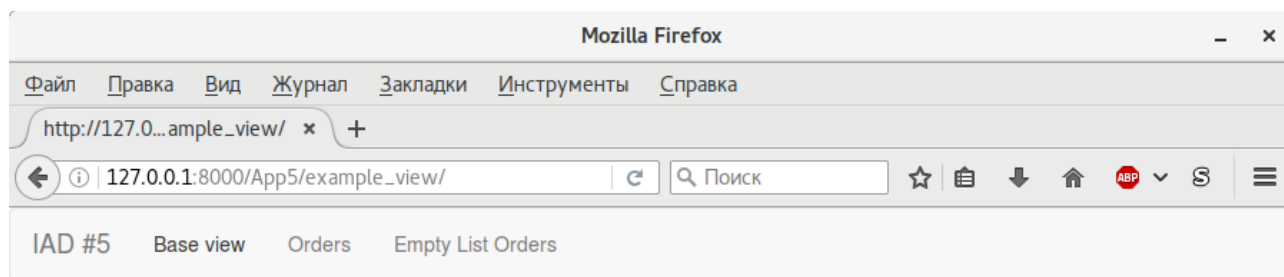
    </ul>

{% endblock %}

```

### 3) Результаты работы

#### 1) Шаблонная страница example\_view (Base view)



Redefined by spawnlings

127.0.0.1:8000/App5/example\_view/

2) Страница заказов orders\_view (Orders)

Заказы

127.0.0.1:8000/App5/orders\_view/Поиск

IAD #5Base viewOrdersEmpty List Orders

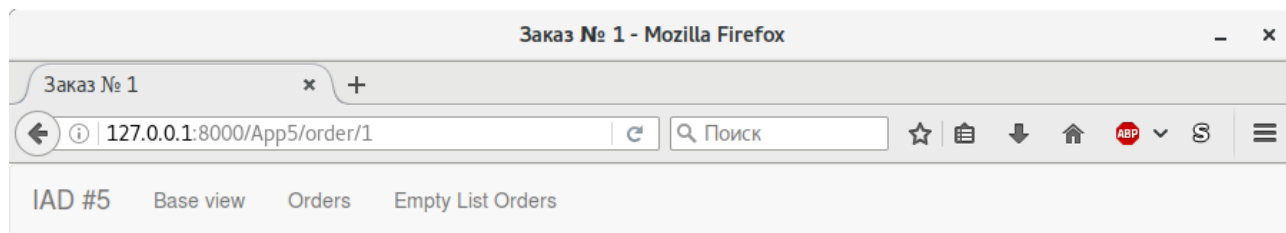
TEST [First order](#)

TEST [Second order](#)

TEST [Third order](#)

127.0.0.1:8000/App5/orders\_view/

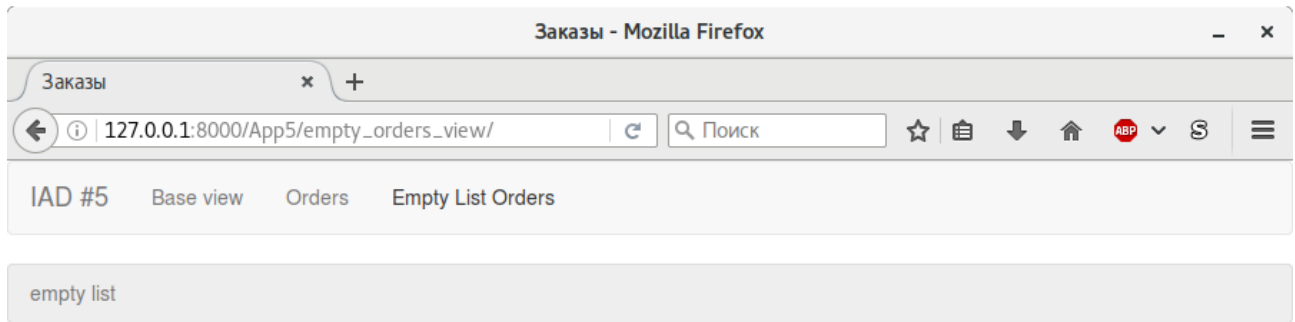
### 3) Детализация заказа order/1



Страница заказа №1



#### 4) Проверка условия empty



127.0.0.1:8000/App5/empty\_orders\_view/