



*«Московский государственный технический университет
имени Н.Э. Баумана»*

(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ

Информатика и системы управления

КАФЕДРА

Системы обработки информации и управления

О т ч ё т п о л а б о р а т о р н о й р а б о т е

п о к у р с у

« Р а з р а б о т к а и н т е р н е т - п р и л о ж е н и й »

Шаблонизация

Исполнитель: студентка группы **РТ5-51**

Галичий Д.А.

Преподаватель: **Гапанюк Ю.Е.**

Цель работы: создание Django-проекта, отображение статичной страницы, знакомство с конструкциями шаблонизаторов: переменными, тегами, наследованием шаблонов.

Содержание файла «settings.py»:

```
"""
Django settings for lab5 project.

Generated by 'django-admin startproject' using Django 1.11.7.

For more information on this file, see
https://docs.djangoproject.com/en/1.11/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/1.11/ref/settings/
"""

import os

# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))

# Quick-start development settings - unsuitable for production
# See
https://docs.djangoproject.com/en/1.11/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = '^_bbs-m=1=5n4&r5nv^0e%=77d&!0d#**7@#s-0*^imrs1+4-h'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'lab5.apps.my_app',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
```

```

        'django.contrib.auth.middleware.AuthenticationMiddleware',
        'django.contrib.messages.middleware.MessageMiddleware',
        'django.middleware.clickjacking.XFrameOptionsMiddleware',
    ]

    ROOT_URLCONF = 'lab5.urls'

    TEMPLATES = [
        {
            'BACKEND': 'django.template.backends.django.DjangoTemplates',
            'DIRS': [os.path.join(BASE_DIR, 'lab5/apps/my_app/templates')]
            ,
            'APP_DIRS': True,
            'OPTIONS': {
                'context_processors': [
                    'django.template.context_processors.debug',
                    'django.template.context_processors.request',
                    'django.contrib.auth.context_processors.auth',
                    'django.contrib.messages.context_processors.messages',
                ],
            },
        ],
    ]

    WSGI_APPLICATION = 'lab5.wsgi.application'

    # Database
    # https://docs.djangoproject.com/en/1.11/ref/settings/#databases

    DATABASES = {
        'default': {
            'ENGINE': 'django.db.backends.sqlite3',
            'NAME': os.path.join(BASE_DIR, 'db.sqlite3'),
        }
    }

    # Password validation
    # https://docs.djangoproject.com/en/1.11/ref/settings/#auth-password-validators

    AUTH_PASSWORD_VALIDATORS = [
        {
            'NAME':
            'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
        },
        {
            'NAME':
            'django.contrib.auth.password_validation.MinimumLengthValidator',
        },
        {
            'NAME':
            'django.contrib.auth.password_validation.CommonPasswordValidator',
        },
        {
            'NAME':

```

```
'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]
```

```
# Internationalization
# https://docs.djangoproject.com/en/1.11/topics/i18n/
```

```
LANGUAGE_CODE = 'ru-ru'
```

```
TIME_ZONE = 'UTC'
```

```
USE_I18N = True
```

```
USE_L10N = True
```

```
USE_TZ = True
```

```
# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/1.11/howto/static-files/
```

```
STATIC_URL = '/static/'
```

Содержание файла «lab5\urls.py»:

```
"""lab5 URL Configuration

The `urlpatterns` list routes URLs to views. For more information
please see:
    https://docs.djangoproject.com/en/1.11/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, include
from django.contrib import admin

urlpatterns = [
    url(r'^admin/', admin.site.urls),
    url(r'^my_app/', include('lab5.apps.my_app.urls')),
]
```

Содержание файла «my_app\urls.py»:

```
from django.conf.urls import url
from django.contrib import admin
```

```

from lab5.apps.my_app import views
from lab5.apps.my_app.views import OrderView, OrdersView

urlpatterns = [
    url(r'^main/', views.main, name='main_url'),
    url(r'^orders/', OrdersView.as_view(), name='orders_url'),
    url(r'^order/(?P<id>\d+)', OrderView.as_view(), name='order_url'),
]

```

Содержание файла «views.py»:

```

from django.shortcuts import render, HttpResponseRedirect
from django.views import View
import random

# Create your views here.

def main(request):
    return render(request, 'main_page.html')

class OrdersView(View):
    def get(self, request):
        data = {
            'orders': [
                {'title': 'The first order', 'id': 1},
                {'title': 'The second order', 'id': 2},
                {'title': 'The third order', 'id': 3},
            ]
        }
        return render(request, 'orders\orders.html', data)

class OrderView(View):
    def get(self, request, id):
        data = {
            'order': {
                'id': id
            }
        }
        return render (request, 'orders\order.html', data)

```

Содержание файла «base.html»:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    {% block title1 %}<title>Home page</title>{% endblock %}
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-
beta.2/css/bootstrap.min.css"
integrity="sha384-
PsH8R72JQ3SOdhVi3uxftmaW6Vc51MKb0q5P2rRUppVrszuE4W1povHYgTpBfshb"
crossorigin="anonymous">

```

```

<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"
integrity="sha384-
KJ3o2DKtIkVYIK3UENzmM7KCKRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
crossorigin="anonymous"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.3/umd/popper
r.min.js"
integrity="sha384-
vFJXuSJphROIrBnz7yo7oB41mKfc8JzQZiCq4NCceLEaO4IHwicKwpJf9c9IpFgh"
crossorigin="anonymous"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-
beta.2/js/bootstrap.min.js"
integrity="sha384-
alpBpkhl1PF0epccYVYDB4do5UnbKysX5WZXM3XxPqe5iKTfUKjNkCk9SaVuEZflJ"
crossorigin="anonymous"></script>
{% load static %}
<link rel="stylesheet" href="{% static 'css/style.css' %}"
type="text/css">
</head>
<body>
<div class="jumbotron" style="display:block">
<div class="container text-center">
{% block title %}<h1 class="asap-font">Welcome to a home
page!</h1>{% endblock %}
</div>
<br><br>
<div class="alert alert-success" role="alert">
This is a reference to the <a href="{% url 'orders_url'
%}" class="alert-link">order's list</a>. Give it a click to see an
example.
</div>
<div class="alert alert-success" role="alert">
Click this reference to come back to the <a href="{% url
'main_url' %}" class="alert-link">home page</a>.
</div>
</div>
<br>
{% block body %}{% endblock %}
<br>
</body>
</html>

```

Содержание файла «main_page.html»:

```

{% extends 'base.html' %}
{% load static %}
{% block body %}
<div class="container">
<div id="carouselExampleControls" class="carousel slide" data-
ride="carousel">
<div class="carousel-inner">
<div class="carousel-item active">

</div>
<div class="carousel-item">

    </div>
    <div class="carousel-item">
        
    </div>
</div>
<a class="carousel-control-prev"
href="#carouselExampleControls" role="button" data-slide="prev">
    <span class="carousel-control-prev-icon" aria-
hidden="true"></span>
    <span class="sr-only">Previous</span>
</a>
<a class="carousel-control-next"
href="#carouselExampleControls" role="button" data-slide="next">
    <span class="carousel-control-next-icon" aria-
hidden="true"></span>
    <span class="sr-only">Next</span>
</a>
</div>
</div>
{% endblock %}

```

Содержание файла «orders.html»:

```

{% extends 'base.html' %}
{% block title %}<title>Orders</title>{% endblock %}
{% block title %}
    <h1 class="asap-font">Orders</h1>
{% endblock %}
{% block body %}
    <div class="container1">
        <br>
        <ul>
            {% for order in orders %}
                {% include 'orders/order_element.html' with
element=order %}
            {% empty %}
                <li>Empty list</li>
            {% endfor %}
        </ul>
        <br>
    </div>
{% endblock %}

```

Содержание файла «order_element.html»:

```

<li><a href="{% url 'order_url' element.id %}">{{ element.title
}}</a></li>

```

Содержание файла «order.html»:

```

{% extends 'base.html' %}
{% load static %}
{% block title %}<title>Order {{ order.id }}</title>{% endblock %}

```

```

{% block title %}
    <h1 class="asap-font">The page of order № {{ order.id }}</h1>
{% endblock %}
{% block body %}
    <div class="container">
        
    </div>
{% endblock %}

```

Содержание файла «style.css»:

```

body {
    background-image: url("../imgs/bckg.jpg");
}

.jumbotron {
    background-color: #70cabc;
    font-family: Vivaldi;
    color: white;
}

.alert-success {
    font-family: EuroRoman;
    font-size: 18px;
}

.container {
    width: 50%;
}

.container1 {
    background-color: #70cabc;
    width: 90%;
    margin: 0 auto;
    border-radius: 18px;
}

li {
    color: white;
    text-decoration-color: white;
    font-family: EuroRoman;
    font-size: 18px;
}

```


Результат работы программы:



