



Федеральное государственное бюджетное образовательное учреждение высшего
профессионального образования
МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. Н. Э. БАУМАНА

Лабораторная работа по курсу:
«Разработка Интернет Приложений»

Шаблонизация_отчет

Выполнил: Кузьмин Анатолий Сергеевич
Проверил: Гапанюк Ю.Е.

Москва - 2017

Цель: Создать Django-проект, показать пользователю стандартную статическую страницу, познакомиться с конструкциями шаблонизаторов

settings.py

```
"""
Django settings for untitled2 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

__PATH = os.path.abspath(os.path.dirname(__file__))

# 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 = 'j3ous-g^%)#%#$dm=hcl!*ljocraxxrm2#0m4ywg^u+_t$tb^2@'

# 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',
    'myapp.apps.MyappConfig',
    'my_app.apps.MyAppConfig',
    'loginsys',
]

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 = 'untitled2.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [os.path.join(BASE_DIR, '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 = 'untitled2.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 = 'en-us'

TIME_ZONE = 'UTC'

USE_I18N = True

USE_L10N = True

USE_TZ = True
STATIC_ROOT = os.path.join(_PATH, 'files', 'static')
STATICFILES_FINDERS = (

    'django.contrib.staticfiles.finders.FileSystemFinder',

    'django.contrib.staticfiles.finders.AppDirectoriesFinder',

)

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

STATIC_URL = '/static/'
MEDIA_URL = '/media/'
```

forms.py

```
from django import forms
from .models import *

class Subscriberforms(forms.ModelForm):
    class Meta:
        model = Subscribers
        exclude = []
```

admin.py

```
from django.contrib import admin
```

```
# Register your models here.
```

tests.py

```
from django.test import TestCase
```

```
# Create your tests here.
```

wsgi.py

```
"""
```

WSGI config for untitled2 project.

It exposes the WSGI callable as a module-level variable named ``application``.

For more information on this file, see

<https://docs.djangoproject.com/en/1.11/howto/deployment/wsgi/>

```
"""
```

```
import os
```

```
from django.core.wsgi import get_wsgi_application
```

```
os.environ.setdefault("DJANGO_SETTINGS_MODULE", "untitled2.settings")
```

```
application = get_wsgi_application()
```


models.py

```
from django.db import models
```

```
# Create your models here.
```

urls.py

```
"""untitled2 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 import settings
```

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

```
from django.contrib import admin
```

```
from myapp.views import *
```

```
admin.autodiscover()
```

```
urlpatterns = [
```

```
url(r'^admin/', admin.site.urls),
```

```
url(r'^main', main, name='main'),
```

```
url(r'^$', main, name='main'),
```

```
url(r'^Gallery/', subs_view.as_view(), name='test_url'),
```

```
url(r'^Nature/', h3, name='h3'),
```

```
url(r'^about/', about, name='about'),
```

```
url(r'^notif/', notif, name='notif'),
```

```
url(r'^reg/', reg.as_view(), name='reg'),
```

```
url(r'^auth/', auth1.as_view(), name='auth'),
```

```
]
```

views.py

```
from django.shortcuts import render
```

```
# Create your views here.
```

apps.py

```
from django.apps import AppConfig
```

```
class MyAppConfig(AppConfig):  
    name = 'myapp'
```