DIRECTORY H:\ZZZ.python PROJECTS\2. thug coding blog\ThugCodingBlog

ThugCodingBlog\blogproject\blog\templates\blog\base.html  
  
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>{% block title %}Thug Coding® Blog{% endblock %}</title>

    <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" rel="stylesheet">

</head>

<body>

    <nav class="navbar navbar-expand-lg navbar-light bg-light">

        <a class="navbar-brand" href="{% url 'home' %}">Thug Coding® Blog</a>

        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span>

        </button>

        <div class="collapse navbar-collapse" id="navbarNav">

            <ul class="navbar-nav ml-auto">

                <li class="nav-item">

                    <a class="nav-link" href="{% url 'home' %}">Home</a>

                </li>

                {% if user.is\_authenticated %}

                    <li class="nav-item">

                        <a class="nav-link" href="{% url 'logout' %}">Logout</a>

                    </li>

                {% else %}

                    <li class="nav-item">

                        <a class="nav-link" href="{% url 'login' %}">Login</a>

                    </li>

                {% endif %}

            </ul>

        </div>

    </nav>

    <div class="container mt-4">

        {% block content %}

        {% endblock %}

    </div>

    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>

    <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.5.4/dist/umd/popper.min.js"></script>

    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>

</body>

</html>

ThugCodingBlog\blogproject\blog\templates\blog\file\_list.html

<!DOCTYPE html>

<html>

<head>

    <title>Uploaded Files</title>

</head>

<body>

    <h1>Uploaded Files</h1>

    <ul>

        {% for file in files %}

        <li>

            <a href="{{ file.file.url }}">{{ file.title }}</a>

        </li>

        {% endfor %}

    </ul>

    <a href="{% url 'upload\_file' %}">Upload New File</a>

</body>

</html>

ThugCodingBlog\blogproject\blog\templates\blog\home.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Thug Coding® Blog</title>

*<!-- Bootstrap CSS -->*

    <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" rel="stylesheet">

    <style>

        .thumbnail-img {

            max-width: 150px; */\* Adjust as needed \*/*

            height: auto; */\* Maintain aspect ratio \*/*

        }

        .media-body {

            padding-left: 15px; */\* Space between image and text \*/*

        }

    </style>

</head>

<body>

    <nav class="navbar navbar-expand-lg navbar-light bg-light">

        <a class="navbar-brand" href="{% url 'home' %}">Thug Coding® Blog</a>

        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span>

        </button>

        <div class="collapse navbar-collapse" id="navbarNav">

            <ul class="navbar-nav ml-auto">

                <li class="nav-item active">

                    <a class="nav-link" href="{% url 'home' %}">Home</a>

                </li>

                <li class="nav-item">

                    <a class="nav-link" href="#">About</a>

                </li>

                <li class="nav-item">

                    <a class="nav-link" href="#">Contact</a>

                </li>

                {% if user.is\_authenticated %}

                    <li class="nav-item">

                        <a class="nav-link" href="{% url 'create\_post' %}">New Post</a>

                    </li>

                    <li class="nav-item">

                        <a class="nav-link" href="{% url 'logout' %}">Logout</a>

                    </li>

                {% else %}

                    <li class="nav-item">

                        <a class="nav-link" href="{% url 'login' %}">Login</a>

                    </li>

                {% endif %}

            </ul>

        </div>

    </nav>

    <div class="container">

        <h1 class="my-4">Blog Home</h1>

        {% for post in posts %}

            <div class="card mb-4">

                <div class="media">

                    {% if post.image %}

                        <img src="{{ post.image.url }}" alt="{{ post.title }}" class="thumbnail-img align-self-start">

                    {% endif %}

                    <div class="media-body">

                        <h2 class="mt-0"><a href="{% url 'post\_detail' post.pk %}">{{ post.title }}</a></h2>

                        <p class="card-text">{{ post.content|truncatewords:30 }}</p>

                        <a href="{% url 'post\_detail' post.pk %}" class="btn btn-primary">Read More &rarr;</a>

                    </div>

                </div>

            </div>

        {% endfor %}

*<!-- Pagination controls here -->*

        <h2 class="my-4">Uploaded Files</h2>

        <ul>

            {% for file in files %}

            <li>

                <a href="{{ file.file.url }}">{{ file.title }}</a>

            </li>

            {% endfor %}

        </ul>

    </div>

*<!-- Bootstrap JS and dependencies -->*

    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>

    <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.5.4/dist/umd/popper.min.js"></script>

    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>

</body>

</html>

ThugCodingBlog\blogproject\blog\templates\blog\login.html

{% extends "blog/base.html" %}

{% block title %}Login{% endblock %}

{% block content %}

    <h2>Login</h2>

    <form method="post">

        {% csrf\_token %}

        {{ form.as\_p }}

        <button type="submit" class="btn btn-primary">Login</button>

    </form>

{% endblock %}

ThugCodingBlog\blogproject\blog\templates\blog\post\_detail.html

{% load static %}

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>{{ post.title }}</title>

    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">

    <link rel="stylesheet" type="text/css" href="{% static 'blog/styles.css' %}">

    <style>

        .post-img {

            max-width: 100%; */\* Adjust as needed \*/*

            height: auto; */\* Maintain aspect ratio \*/*

        }

        .media {

            margin-top: 20px;

        }

        .media-body {

            padding-left: 15px; */\* Space between image and text \*/*

        }

    </style>

</head>

<body>

    <nav class="navbar navbar-expand-lg navbar-light bg-light">

        <a class="navbar-brand" href="#">My Blog</a>

        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span>

        </button>

        <div class="collapse navbar-collapse" id="navbarNav">

            <ul class="navbar-nav">

                <li class="nav-item active">

                    <a class="nav-link" href="{% url 'home' %}">Home <span class="sr-only">(current)</span></a>

                </li>

                <li class="nav-item">

                    <a class="nav-link" href="#">About</a>

                </li>

                <li class="nav-item">

                    <a class="nav-link" href="#">Contact</a>

                </li>

            </ul>

        </div>

    </nav>

    <div class="container my-4">

        <h1>{{ post.title }}</h1>

        <p><strong>Author:</strong> {{ post.author }}</p>

        <p><strong>Created:</strong> {{ post.created\_at }}</p>

        <div class="media">

            {% if post.image %}

                <img src="{{ post.image.url }}" alt="{{ post.title }}" class="post-img align-self-start">

            {% endif %}

            <div class="media-body">

                <p>{{ post.content }}</p>

            </div>

        </div>

        <a href="{% url 'home' %}" class="btn btn-secondary mt-3">Back to Home</a>

    </div>

    <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"></script>

    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js"></script>

    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>

</body>

</html>

ThugCodingBlog\blogproject\blog\templates\blog\post\_form.html

<!DOCTYPE html>

<html>

<head>

    <title>{{ form.instance.pk|default:"New" }} Post</title>

</head>

<body>

    <h1>{{ form.instance.pk|default:"New" }} Post</h1>

    <form method="post" enctype="multipart/form-data">

        {% csrf\_token %}

        {{ form.as\_p }}

        <button type="submit">Save</button>

    </form>

    <a href="{% url 'home' %}">Back to home</a>

</body>

</html>

ThugCodingBlog\blogproject\blog\templates\blog\upload.html

<!DOCTYPE html>

<html>

<head>

    <title>Upload File</title>

</head>

<body>

    <h1>Upload File</h1>

    <form method="post" enctype="multipart/form-data">

        {% csrf\_token %}

        {{ form.as\_p }}

        <button type="submit">Upload</button>

    </form>

    <a href="{% url 'file\_list' %}">View Files</a>

</body>

</html>

ThugCodingBlog\blogproject\blog\admin.py

from django.contrib import admin

from .models import Post, UploadedFile

admin.site.register(Post)

admin.site.register(UploadedFile)

*# Register your models here.*

ThugCodingBlog\blogproject\blog\apps.py

from django.apps import AppConfig

class BlogConfig(AppConfig):

    default\_auto\_field = 'django.db.models.BigAutoField'

    name = 'blog'

ThugCodingBlog\blogproject\blog\forms.py

from django import forms

from .models import Post, UploadedFile

class PostForm(forms.ModelForm):

    class Meta:

        model = Post

        fields = ['title', 'author', 'content', 'image']

class UploadFileForm(forms.ModelForm):

    class Meta:

        model = UploadedFile

        fields = ['title', 'file']

ThugCodingBlog\blogproject\blog\models.py  
  
from django.db import models

from django.utils import timezone

from django.contrib.auth.models import User

class Post(models.Model):

    title = models.CharField(*max\_length*=200)

    author = models.ForeignKey(User, *on\_delete*=models.CASCADE)

    content = models.TextField()

    created\_at = models.DateTimeField(*default*=timezone.now)

    updated\_at = models.DateTimeField(*auto\_now*=True)

    image = models.ImageField(*upload\_to*='images/', *blank*=True, *null*=True)

    def \_\_str\_\_(*self*):

        return *self*.title

class UploadedFile(models.Model):

    title = models.CharField(*max\_length*=255)

    file = models.FileField(*upload\_to*='uploads/')

    uploaded\_at = models.DateTimeField(*auto\_now\_add*=True)

    def \_\_str\_\_(*self*):

        return *self*.title

*# Create your models here.*

*# Create your models here.*

ThugCodingBlog\blogproject\blog\tests.py  
  
from django.test import TestCase

*# Create your tests here.*

ThugCodingBlog\blogproject\blog\urls.py

from django.urls import path

from . import views

urlpatterns = [

    path('', views.home, *name*='home'),

    path('post/<int:pk>/', views.post\_detail, *name*='post\_detail'),

    path('post/new/', views.create\_post, *name*='create\_post'),

    path('post/<int:pk>/edit/', views.edit\_post, *name*='edit\_post'),

    path('logout/', views.logout\_view, *name*='logout'),

    path('upload/', views.upload\_file, *name*='upload\_file'),

    path('files/', views.file\_list, *name*='file\_list'),

]

ThugCodingBlog\blogproject\blog\views.py  
  
from django.shortcuts import render, get\_object\_or\_404, redirect

from django.core.paginator import Paginator

from .models import Post, UploadedFile

from .forms import PostForm, UploadFileForm

from django.contrib.auth.decorators import login\_required

from django.contrib.auth import logout

def home(*request*):

    post\_list = Post.objects.all().order\_by('-created\_at')

    paginator = Paginator(post\_list, 5)  *# Show 5 posts per page*

    page\_number = *request*.GET.get('page')

    posts = paginator.get\_page(page\_number)

    files = UploadedFile.objects.all()  *# Get all uploaded files*

    return render(*request*, 'blog/home.html', {'posts': posts, 'files': files})

def post\_detail(*request*, *pk*):

    post = get\_object\_or\_404(Post, *pk*=*pk*)

    return render(*request*, 'blog/post\_detail.html', {'post': post})

@login\_required

def create\_post(*request*):

    if *request*.method == 'POST':

        form = PostForm(*request*.POST, *request*.FILES)

        if form.is\_valid():

            post = form.save(*commit*=False)

            post.author = *request*.user

            post.save()

            return redirect('post\_detail', *pk*=post.pk)

    else:

        form = PostForm()

    return render(*request*, 'blog/post\_form.html', {'form': form})

@login\_required

def edit\_post(*request*, *pk*):

    post = get\_object\_or\_404(Post, *pk*=*pk*)

    if *request*.method == 'POST':

        form = PostForm(*request*.POST, *request*.FILES, *instance*=post)

        if form.is\_valid():

            form.save()

            return redirect('post\_detail', *pk*=post.pk)

    else:

        form = PostForm(*instance*=post)

    return render(*request*, 'blog/post\_form.html', {'form': form})

def logout\_view(*request*):

    logout(*request*)

    return redirect('home')

@login\_required

def upload\_file(*request*):

    if *request*.method == 'POST':

        form = UploadFileForm(*request*.POST, *request*.FILES)

        if form.is\_valid():

            form.save()

            return redirect('file\_list')

    else:

        form = UploadFileForm()

    return render(*request*, 'blog/upload.html', {'form': form})

@login\_required

def file\_list(*request*):

    files = UploadedFile.objects.all()

    return render(*request*, 'blog/file\_list.html', {'files': files})

ThugCodingBlog\blogproject\blogproject\settings.py

import os

from pathlib import Path

*# Build paths inside the project like this: BASE\_DIR / 'subdir'.*

BASE\_DIR = Path(\_\_file\_\_).resolve().parent.parent

*# Quick-start development settings - unsuitable for production*

*# See https://docs.djangoproject.com/en/5.0/howto/deployment/checklist/*

*# SECURITY WARNING: keep the secret key used in production secret!*

SECRET\_KEY = 'django-insecure-=)8-9\_ivj6=xc%vnno9kigarl27=ssu%kb6bh%t79\_isetcc+5'

*# 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',

    'blog',

]

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 = 'blogproject.urls'

TEMPLATES = [

    {

        'BACKEND': 'django.template.backends.django.DjangoTemplates',

        'DIRS': [],

        '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 = 'blogproject.wsgi.application'

*# Database*

*# https://docs.djangoproject.com/en/5.0/ref/settings/#databases*

DATABASES = {

    'default': {

        'ENGINE': 'django.db.backends.sqlite3',

        'NAME': BASE\_DIR / 'db.sqlite3',

    }

}

*# Password validation*

*# https://docs.djangoproject.com/en/5.0/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/5.0/topics/i18n/*

LANGUAGE\_CODE = 'en-us'

TIME\_ZONE = 'UTC'

USE\_I18N = True

USE\_TZ = True

*# Static files (CSS, JavaScript, Images)*

*# https://docs.djangoproject.com/en/5.0/howto/static-files/*

STATIC\_URL = '/static/'

*# Default primary key field type*

*# https://docs.djangoproject.com/en/5.0/ref/settings/#default-auto-field*

DEFAULT\_AUTO\_FIELD = 'django.db.models.BigAutoField'

*# Media files*

MEDIA\_URL = '/media/'

MEDIA\_ROOT = os.path.join(BASE\_DIR, 'media')

LOGIN\_REDIRECT\_URL = 'home'

LOGIN\_URL = 'login'

LOGOUT\_REDIRECT\_URL = 'home'

ThugCodingBlog\blogproject\blogproject\urls.py

from django.contrib import admin

from django.urls import path, include

from django.conf import settings

from django.conf.urls.static import static

from django.contrib.auth import views as auth\_views

urlpatterns = [

    path('admin/', admin.site.urls),

    path('', include('blog.urls')),

    path('login/', auth\_views.LoginView.as\_view(*template\_name*='blog/login.html'), *name*='login'),

    path('logout/', auth\_views.LogoutView.as\_view(*next\_page*='/'), *name*='logout'),

]

if settings.DEBUG:

    urlpatterns += static(settings.MEDIA\_URL, *document\_root*=settings.MEDIA\_ROOT)