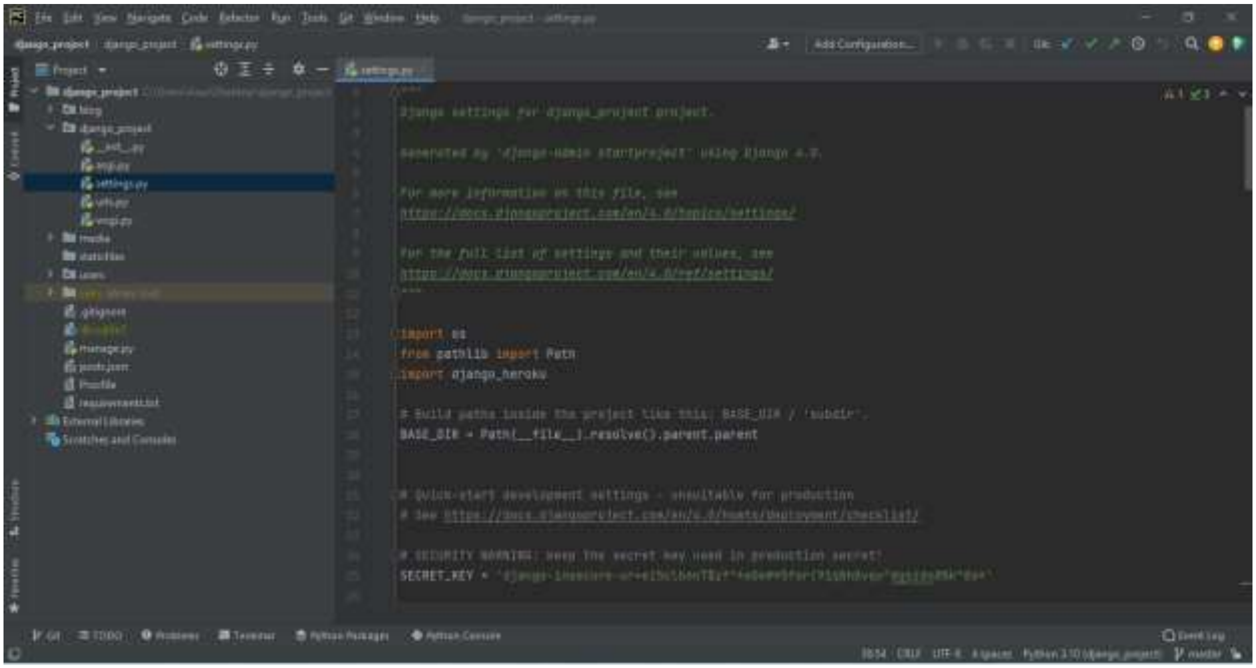
Name: SALAHODIN S. DAGUIT

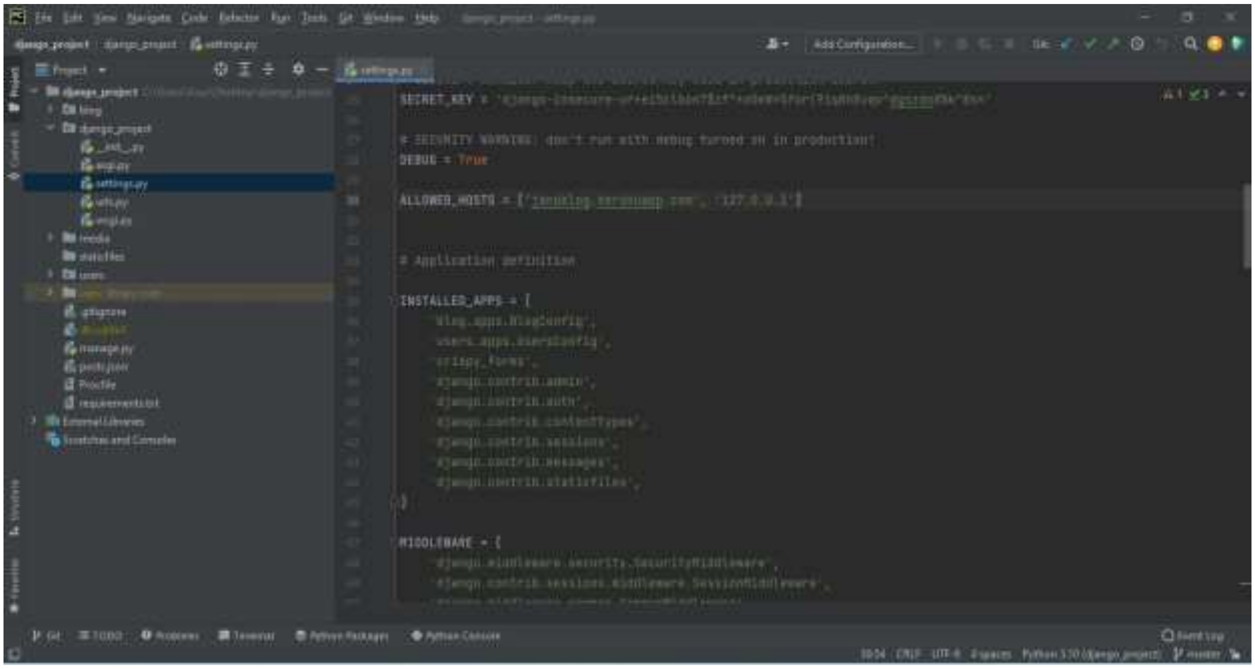
Yr/Course/Section: 3/BSCS/A

ACTIVITY 5 DJANGO

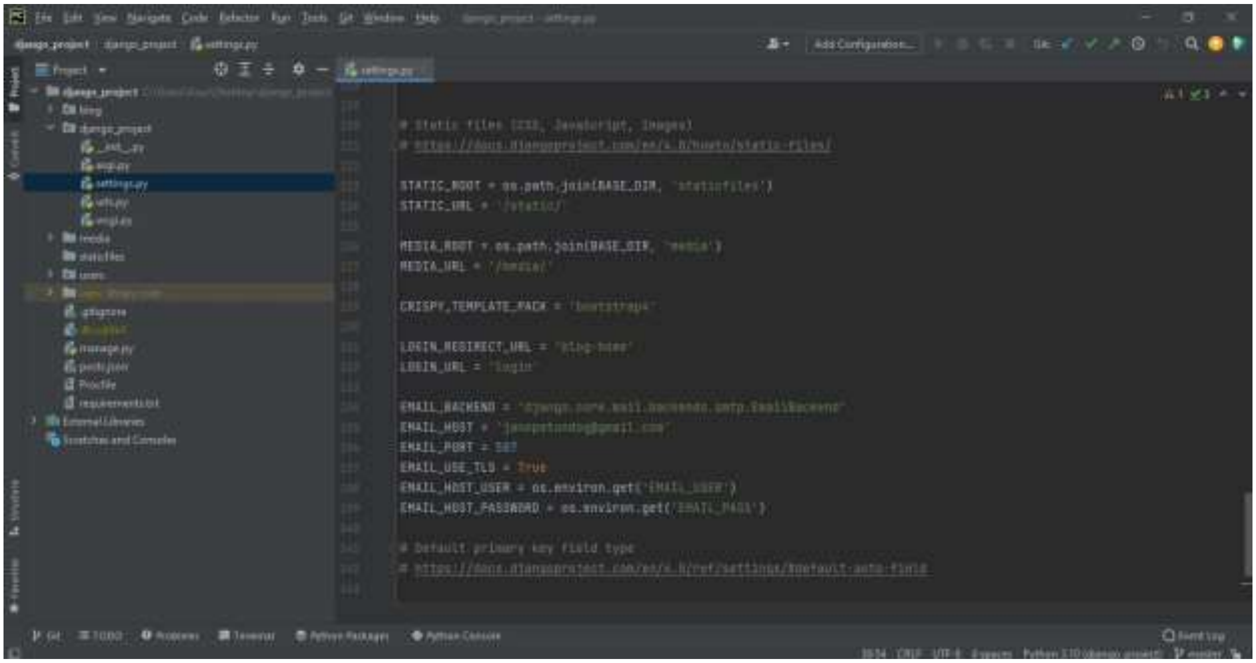
- **django_project > django_project > settings.py**



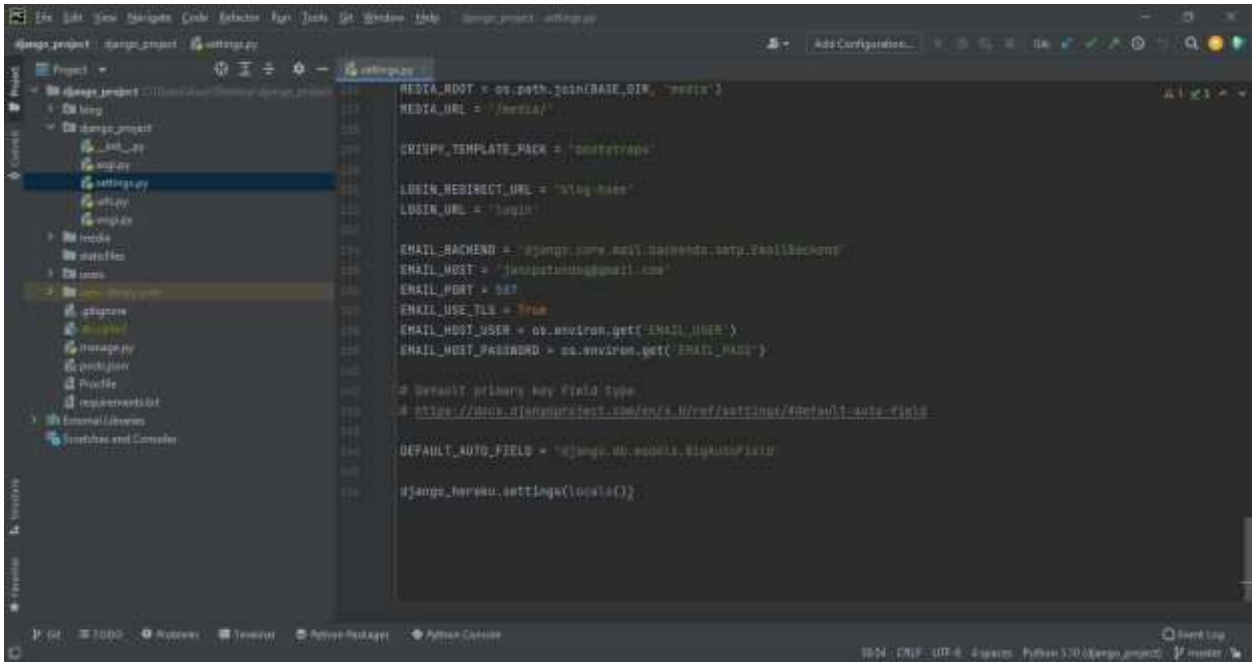
```
"""
Django settings for django_project project.

Generated by 'django-admin startproject' using Django 4.0.

For more information on this file, see
https://docs.djangoproject.com/en/4.0/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/4.0/ref/settings/
"""

import os
from pathlib import Path
import django_heroku

# 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/4.0/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'django-insecure-[illegible]'
```



```
SECRET_KEY = 'django-insecure-[illegible]'

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

ALLOWED_HOSTS = ['[illegible].herokuapp.com', '127.0.0.1']


# Application definition

INSTALLED_APPS = [
    'blog.apps.BlogConfig',
    'users.apps.UsersConfig',
    'crispy_forms',
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
```

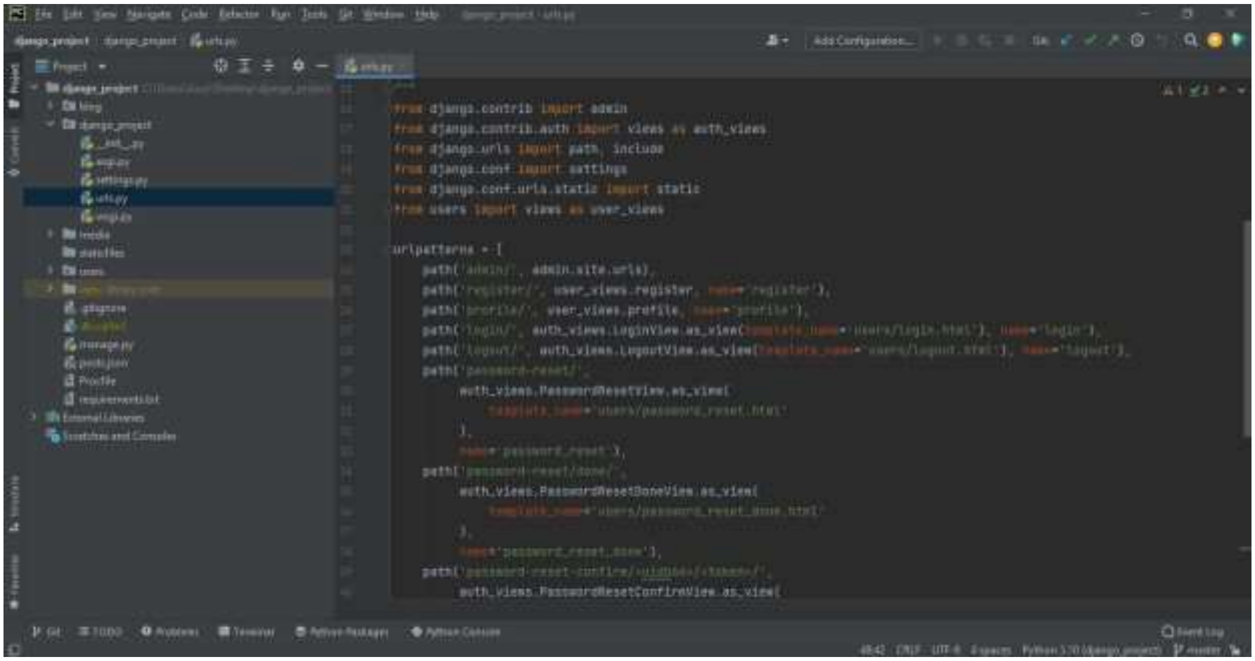


```
# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/4.0/howto/static-files/

STATIC_ROOT = os.path.join(BASE_DIR, 'staticfiles')
STATIC_URL = '/static/'

MEDIA_ROOT = os.path.join(BASE_DIR, 'media')
MEDIA_URL = '/media/'

CRISPY_TEMPLATE_PACK = 'bootstrap4'

LOGIN_REDIRECT_URL = 'blog-home'
LOGIN_URL = 'login'

EMAIL_BACKEND = 'django.core.mail.backends.smtp.EmailBackend'
EMAIL_HOST = 'janopetundo@gmail.com'
EMAIL_PORT = 587
EMAIL_USE_TLS = True
EMAIL_HOST_USER = os.environ.get('EMAIL_USER')
EMAIL_HOST_PASSWORD = os.environ.get('EMAIL_PASS')

# Default primary key field type
# https://docs.djangoproject.com/en/4.0/ref/settings/#default-auto-field
```

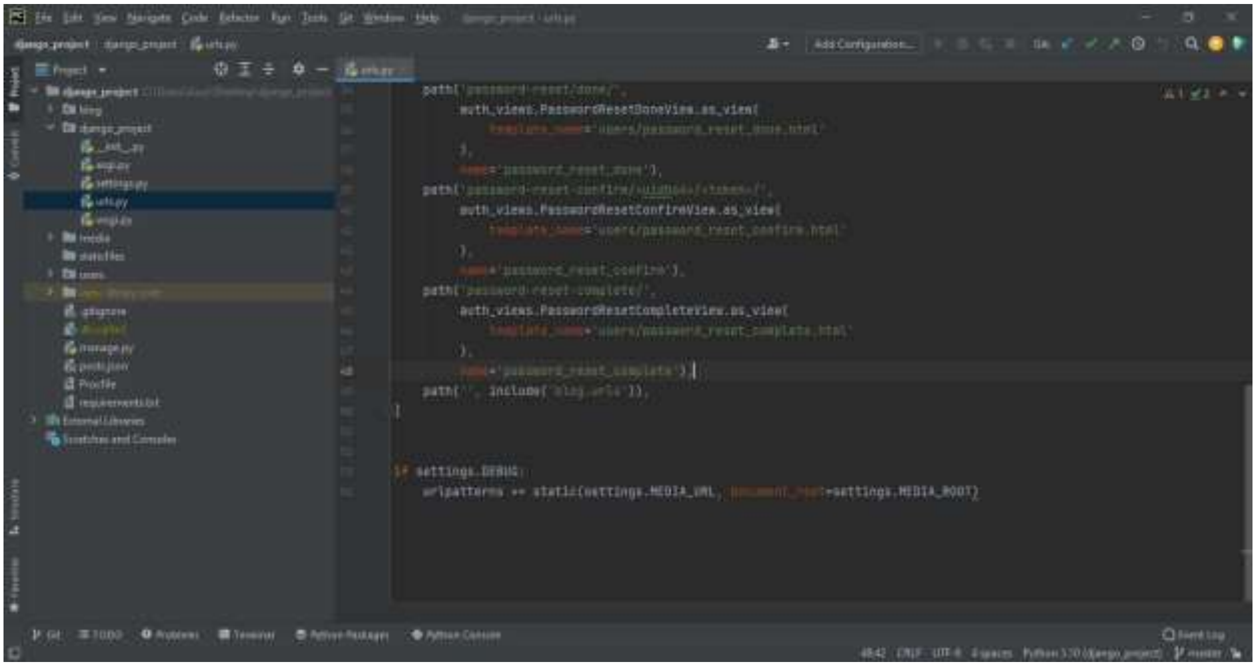


```
MEDIA_ROOT = os.path.join(BASE_DIR, 'media')
MEDIA_URL = '/media/'

CRISPY_TEMPLATE_PACK = 'bootstrap4'

LOGIN_REDIRECT_URL = 'blog-home'
LOGIN_URL = 'login'

EMAIL_BACKEND = 'django.core.mail.backends.smtp.EmailBackend'
EMAIL_HOST = 'janopetundo@gmail.com'
EMAIL_PORT = 587
EMAIL_USE_TLS = True
EMAIL_HOST_USER = os.environ.get('EMAIL_USER')
EMAIL_HOST_PASSWORD = os.environ.get('EMAIL_PASS')

# Default primary key field type
# https://docs.djangoproject.com/en/4.0/ref/settings/#default-auto-field

DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'

django_heroku.settings(locals())
```

- **django_project > django_project > urls.py**



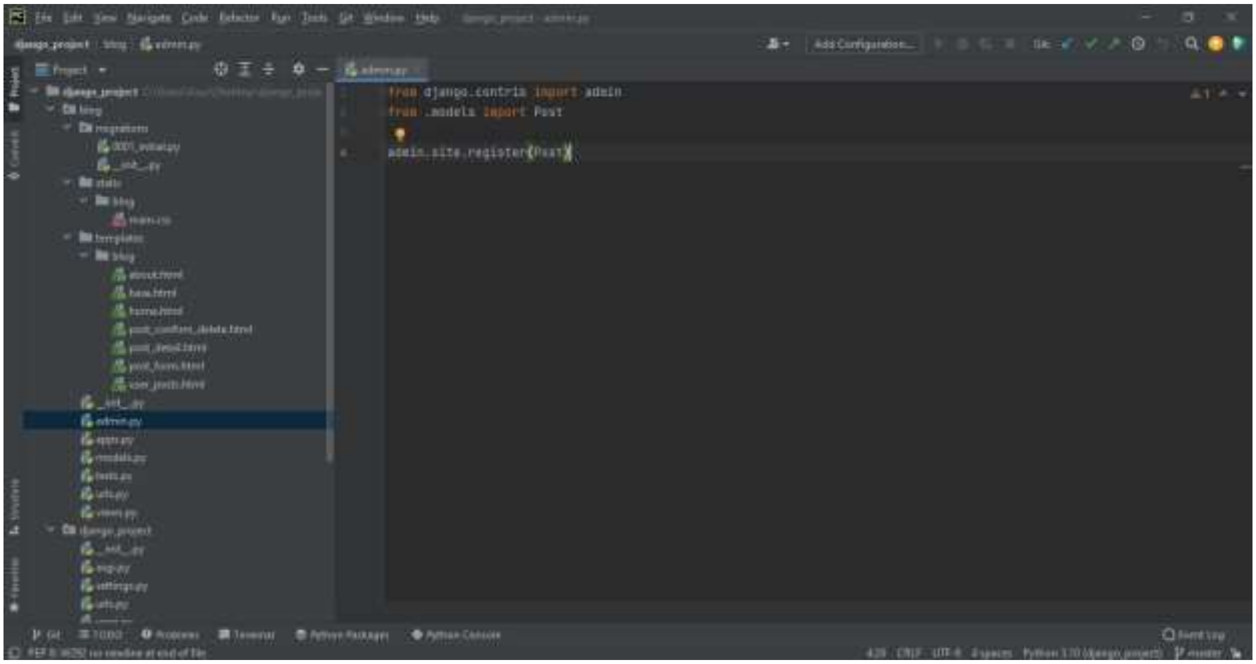
```python
from django.contrib import admin
from django.contrib.auth import views as auth_views
from django.urls import path, include
from django.conf import settings
from django.conf.urls.static import static
from users import views as user_views

urlpatterns = [
    path('admin/', admin.site.urls),
    path('register/', user_views.register, name='register'),
    path('profile/', user_views.profile, name='profile'),
    path('login/', auth_views.LoginView.as_view(template_name='users/login.html'), name='login'),
    path('logout/', auth_views.LogoutView.as_view(template_name='users/logout.html'), name='logout'),
    path('password-reset/',
         auth_views.PasswordResetView.as_view(
             template_name='users/password_reset.html'
         ),
         name='password_reset'),
    path('password-reset/done/',
         auth_views.PasswordResetDoneView.as_view(
             template_name='users/password_reset_done.html'
         ),
         name='password_reset_done'),
    path('password-reset-confirm/<uidb64>/<token>/',
         auth_views.PasswordResetConfirmView.as_view(
```



```python
    path('password-reset/done/',
         auth_views.PasswordResetDoneView.as_view(
             template_name='users/password_reset_done.html'
         ),
         name='password_reset_done'),
    path('password-reset-confirm/<uidb64>/<token>/',
         auth_views.PasswordResetConfirmView.as_view(
             template_name='users/password_reset_confirm.html'
         ),
         name='password_reset_confirm'),
    path('password-reset-complete/',
         auth_views.PasswordResetCompleteView.as_view(
             template_name='users/password_reset_complete.html'
         ),
         name='password_reset_complete'),
    path('', include('blog.urls')),
]


if settings.DEBUG:
    urlpatterns += static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

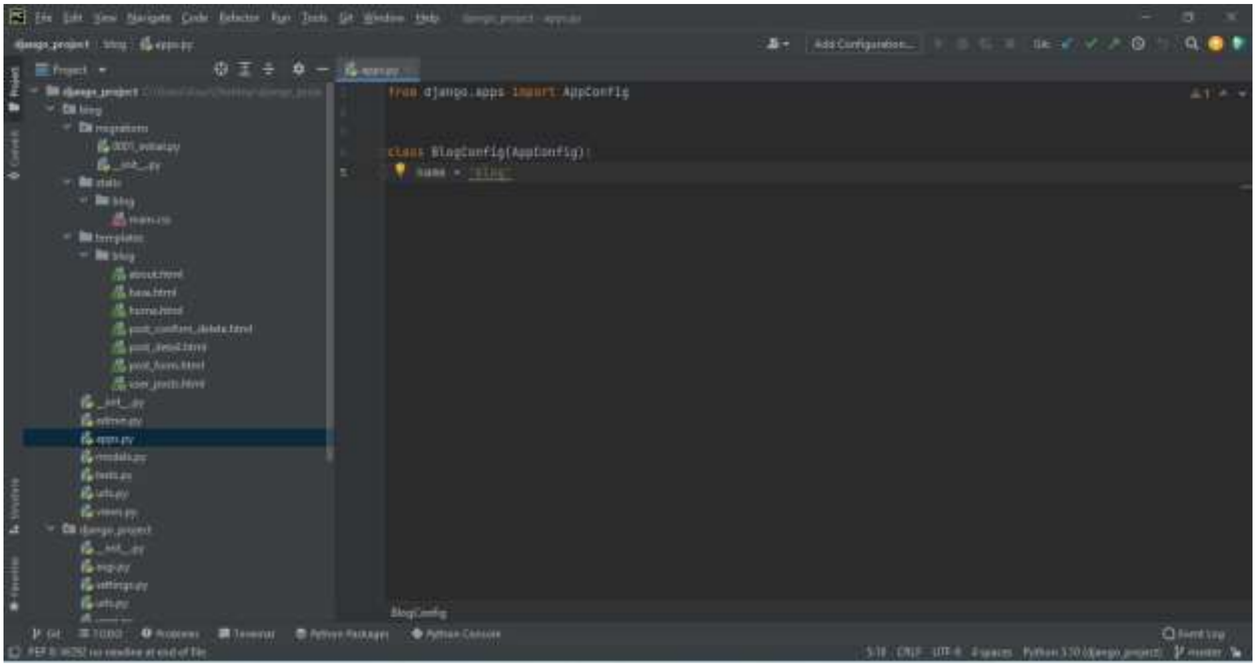
BlOg APP

- **django_project > blog > admin.py**



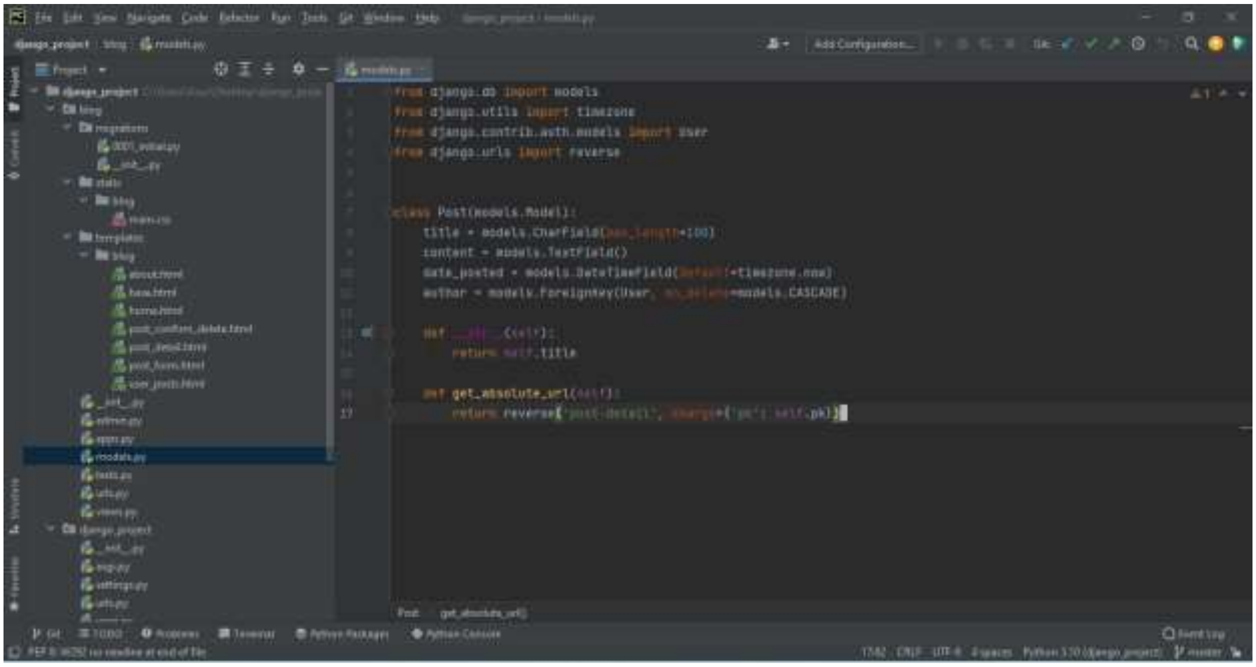
```
from django.contrib import admin
from .models import Post

admin.site.register(Post)
```

- django_project > blog > **apps.py**



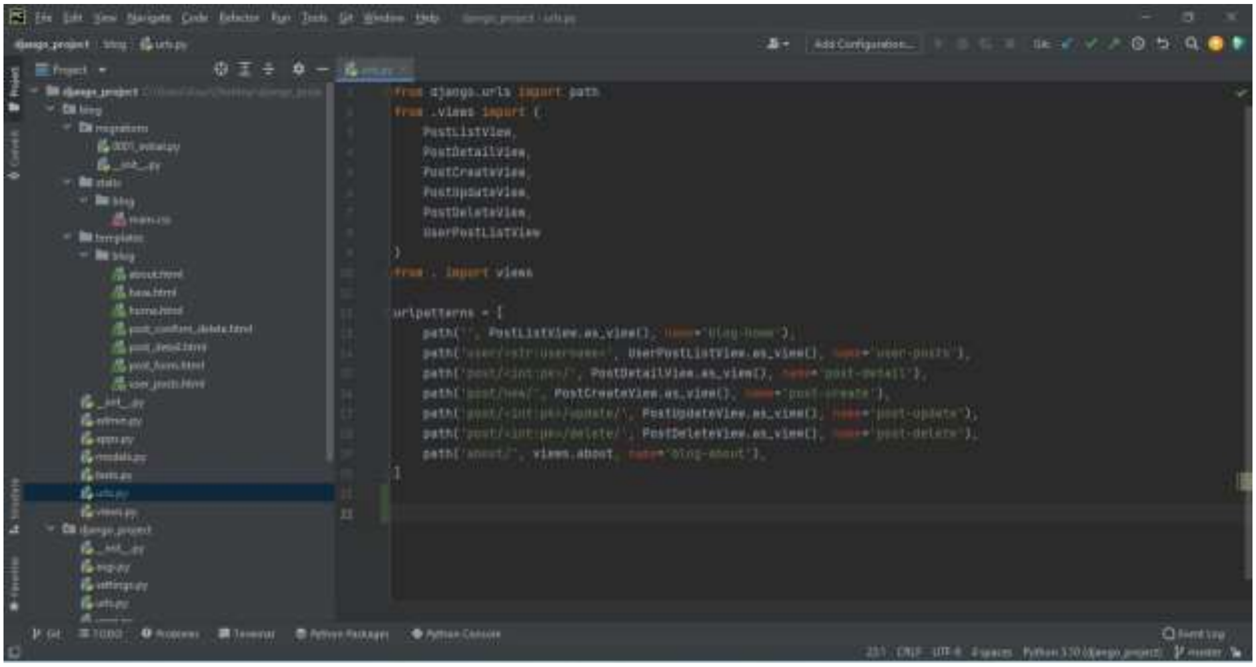
```
from django.apps import AppConfig


class BlogConfig(AppConfig):
    name = 'blog'
```

- django_project > blog > **models.py**



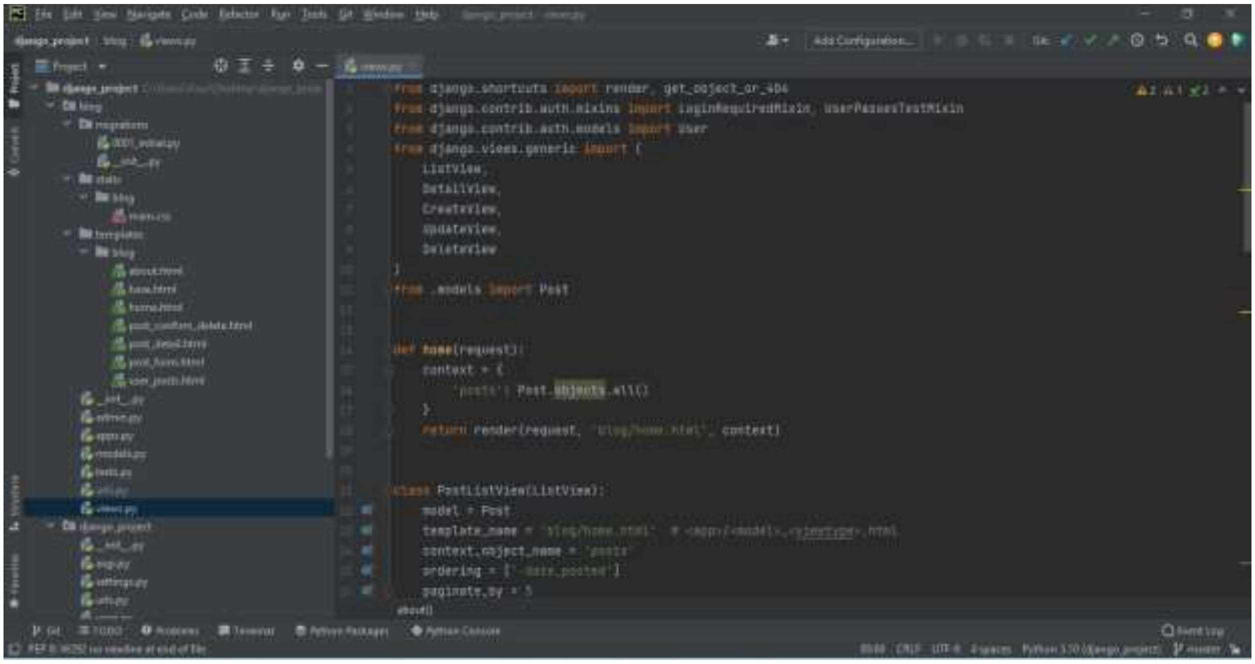
```python
from django.db import models
from django.utils import timezone
from django.contrib.auth.models import User
from django.urls import reverse


class Post(models.Model):
    title = models.CharField(max_length=100)
    content = models.TextField()
    date_posted = models.DateTimeField(default=timezone.now)
    author = models.ForeignKey(User, on_delete=models.CASCADE)

    def __str__(self):
        return self.title

    def get_absolute_url(self):
        return reverse('post-detail', kwargs={'pk': self.pk})
```

- django_project > blog > **urls.py**

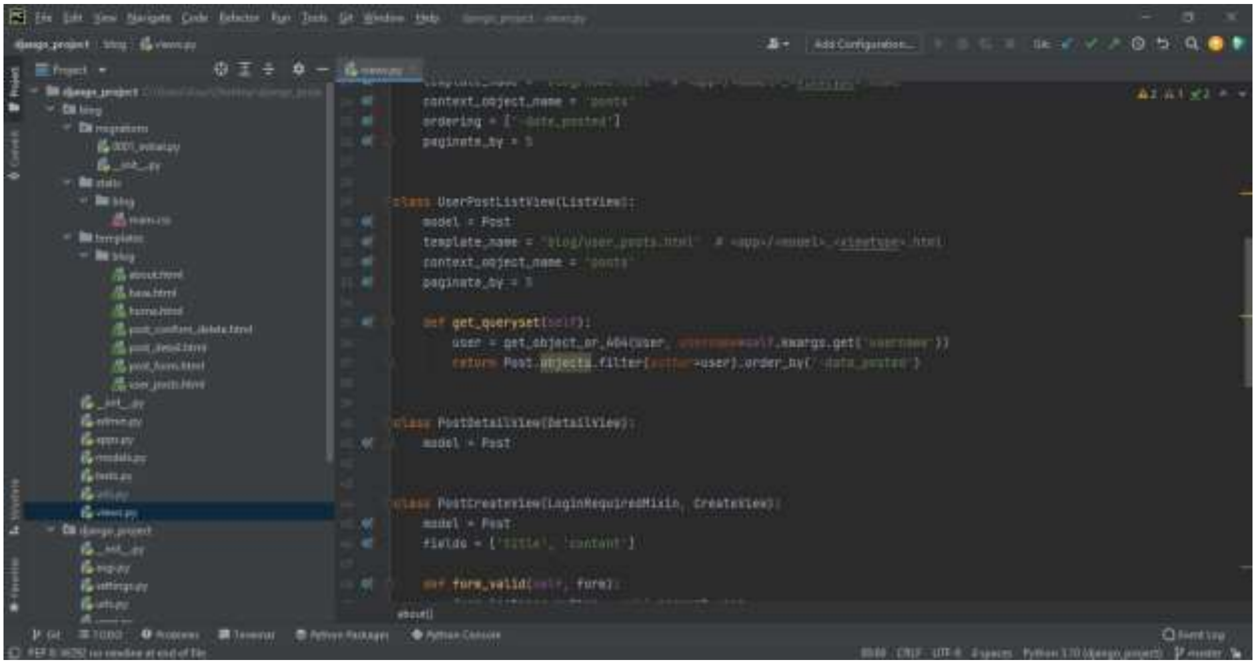


```python
from django.urls import path
from .views import (
    PostListView,
    PostDetailView,
    PostCreateView,
    PostUpdateView,
    PostDeleteView,
    UserPostListView
)
from . import views

urlpatterns = [
    path('', PostListView.as_view(), name='blog-home'),
    path('user/<str:username>', UserPostListView.as_view(), name='user-posts'),
    path('post/<int:pk>/', PostDetailView.as_view(), name='post-detail'),
    path('post/new/', PostCreateView.as_view(), name='post-create'),
    path('post/<int:pk>/update/', PostUpdateView.as_view(), name='post-update'),
    path('post/<int:pk>/delete/', PostDeleteView.as_view(), name='post-delete'),
    path('about/', views.about, name='blog-about'),
]
```

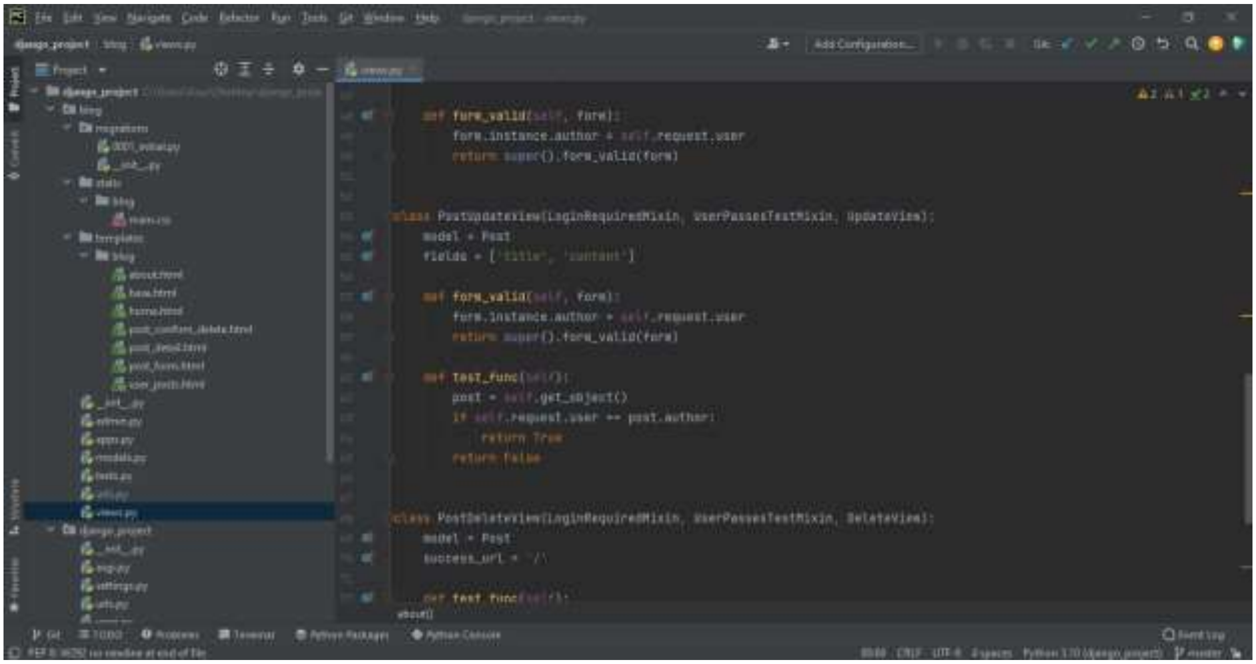
- django_project > blog > **views.py**



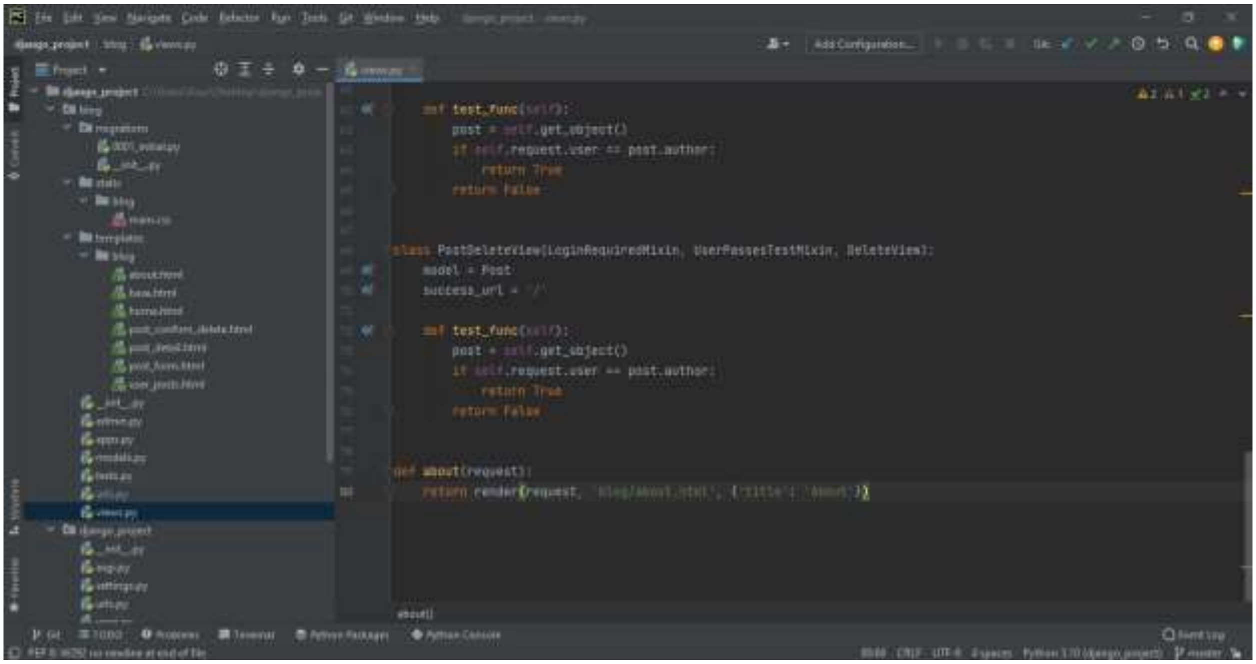
```python
from django.shortcuts import render, get_object_or_404
from django.contrib.auth.mixins import LoginRequiredMixin, UserPassesTestMixin
from django.contrib.auth.models import User
from django.views.generic import (
    ListView,
    DetailView,
    CreateView,
    UpdateView,
    DeleteView
)
from .models import Post


def home(request):
    context = {
        'posts': Post.objects.all()
    }
    return render(request, 'blog/home.html', context)


class PostListView(ListView):
    model = Post
    template_name = 'blog/home.html'  # <app>/<model>_<viewtype>.html
    context_object_name = 'posts'
    ordering = ['-date_posted']
    paginate_by = 5
```



```python
    context_object_name = 'posts'
    ordering = ['-date_posted']
    paginate_by = 5


class UserPostListView(ListView):
    model = Post
    template_name = 'blog/user_posts.html'  # <app>/<model>_<viewtype>.html
    context_object_name = 'posts'
    paginate_by = 5

    def get_queryset(self):
        user = get_object_or_404(User, username=self.kwargs.get('username'))
        return Post.objects.filter(author=user).order_by('-date_posted')


class PostDetailView(DetailView):
    model = Post


class PostCreateView(LoginRequiredMixin, CreateView):
    model = Post
    fields = ['title', 'content']

    def form_valid(self, form):
```



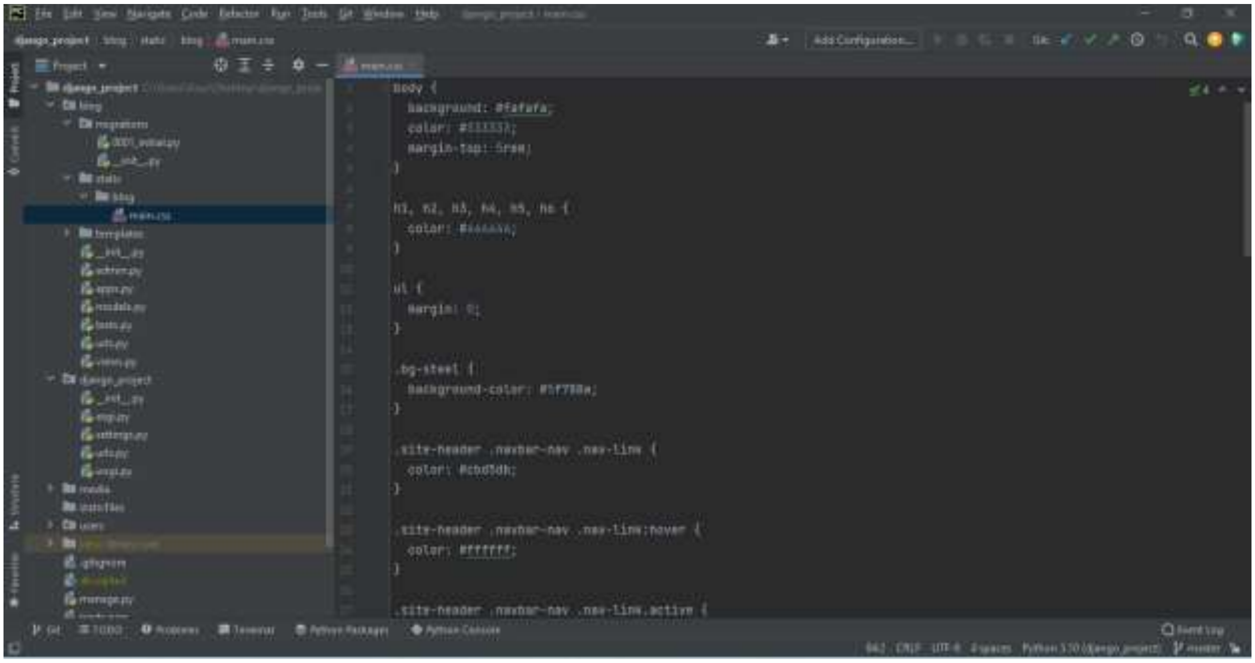
```
    def form_valid(self, form):
        form.instance.author = self.request.user
        return super().form_valid(form)


class PostUpdateView(LoginRequiredMixin, UserPassesTestMixin, UpdateView):
    model = Post
    fields = ['title', 'content']

    def form_valid(self, form):
        form.instance.author = self.request.user
        return super().form_valid(form)

    def test_func(self):
        post = self.get_object()
        if self.request.user == post.author:
            return True
        return False


class PostDeleteView(LoginRequiredMixin, UserPassesTestMixin, DeleteView):
    model = Post
    success_url = '/'

    def test_func(self):
```



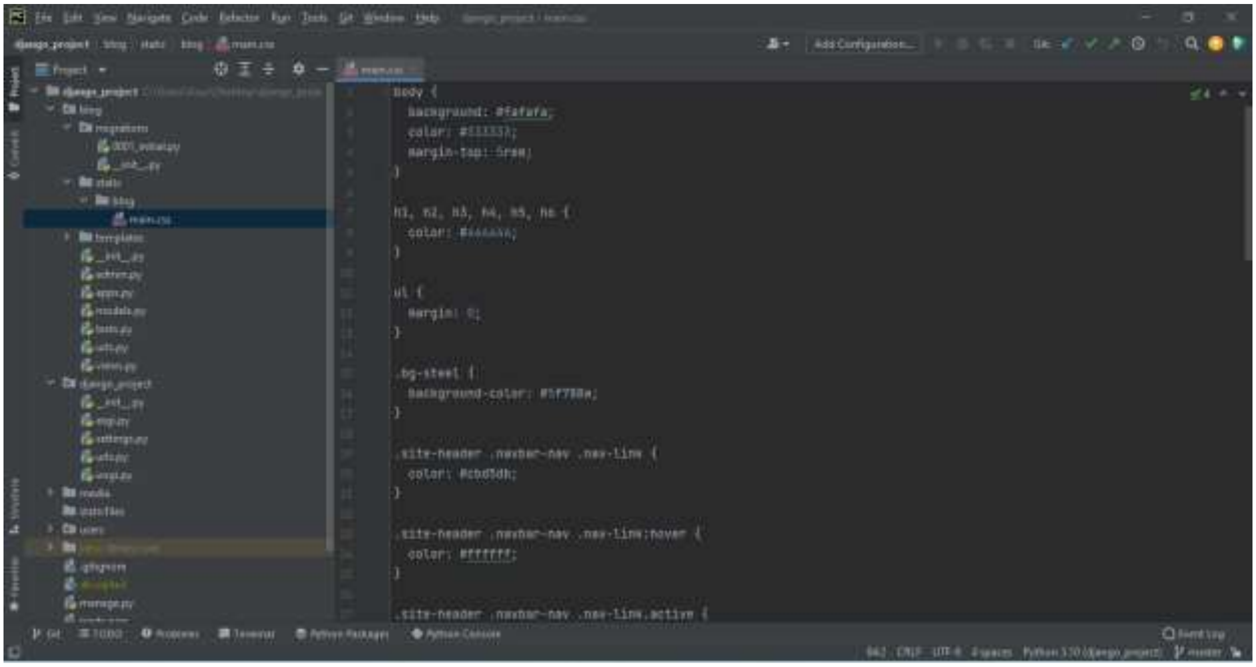
```
    def test_func(self):
        post = self.get_object()
        if self.request.user == post.author:
            return True
        return False


class PostDeleteView(LoginRequiredMixin, UserPassesTestMixin, DeleteView):
    model = Post
    success_url = '/'

    def test_func(self):
        post = self.get_object()
        if self.request.user == post.author:
            return True
        return False


def about(request):
    return render(request, 'blog/about.html', {'title': 'About'})
```

Blog App Static

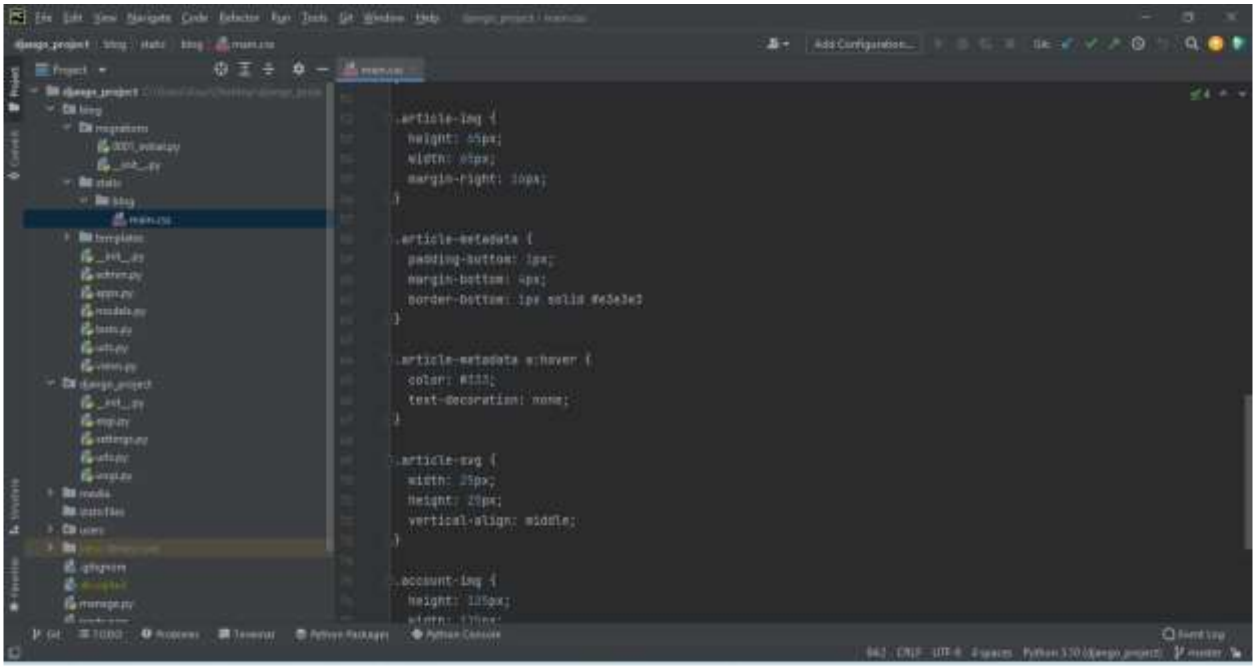
- django_project > blog > static > blog > **main.css**



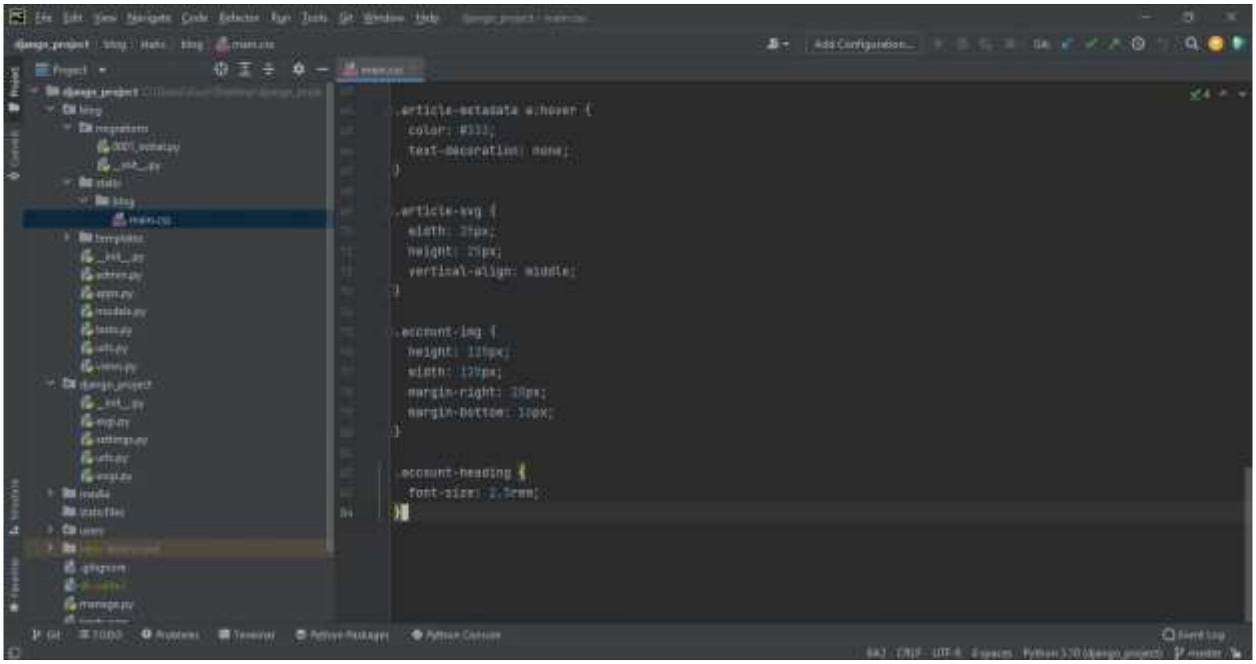
```css
body {
  background: #fafafa;
  color: #333333;
  margin-top: 5rem;
}

h1, h2, h3, h4, h5, h6 {
  color: #444444;
}

ul {
  margin: 0;
}

.bg-steel {
  background-color: #5f788a;
}

.site-header .navbar-nav .nav-link {
  color: #cbd5db;
}

.site-header .navbar-nav .nav-link:hover {
  color: #ffffff;
}

.site-header .navbar-nav .nav-link.active {
```



```css
body {
  background: #fafafa;
  color: #333333;
  margin-top: 5rem;
}

h1, h2, h3, h4, h5, h6 {
  color: #444444;
}

ul {
  margin: 0;
}

.bg-steel {
  background-color: #5f788a;
}

.site-header .navbar-nav .nav-link {
  color: #cbd5db;
}

.site-header .navbar-nav .nav-link:hover {
  color: #ffffff;
}

.site-header .navbar-nav .nav-link.active {
```



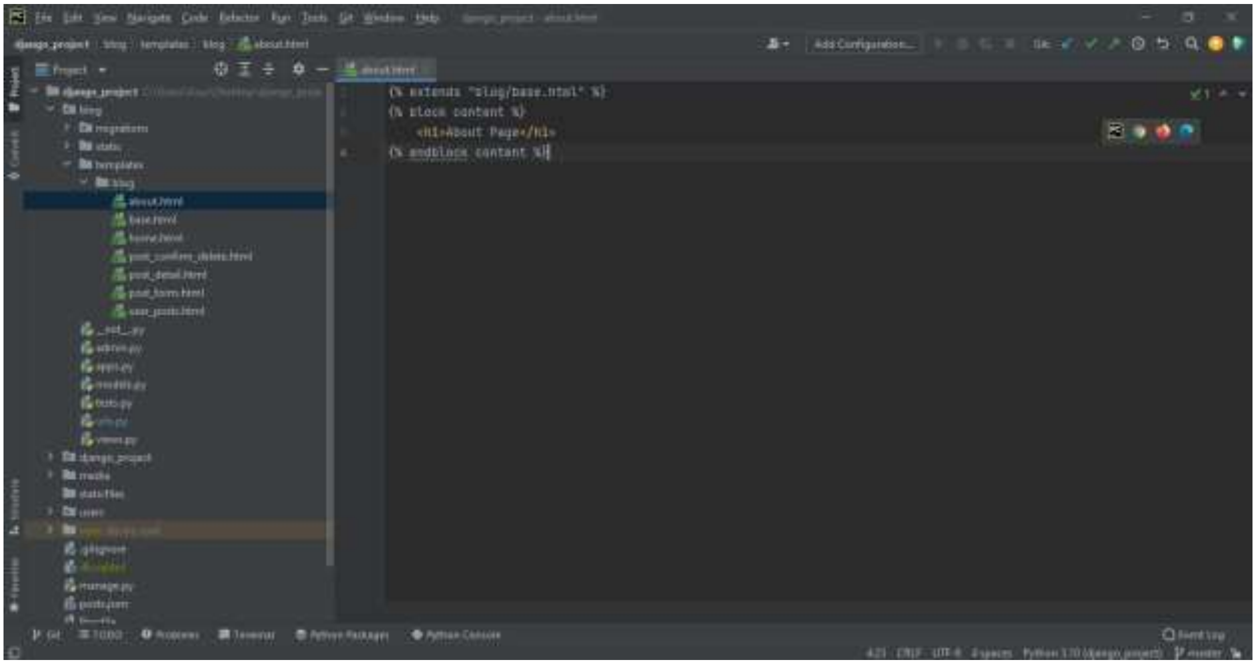
```
.article-img {
  height: 65px;
  width: 65px;
  margin-right: 16px;
}

.article-metadata {
  padding-bottom: 1px;
  margin-bottom: 4px;
  border-bottom: 1px solid #e3e3e3
}

.article-metadata a:hover {
  color: #333;
  text-decoration: none;
}

.article-svg {
  width: 25px;
  height: 25px;
  vertical-align: middle;
}

.account-img {
  height: 125px;
  width: 125px;
```



```
.article-metadata a:hover {
  color: #333;
  text-decoration: none;
}

.article-svg {
  width: 25px;
  height: 25px;
  vertical-align: middle;
}

.account-img {
  height: 125px;
  width: 125px;
  margin-right: 20px;
  margin-bottom: 16px;
}

.account-heading {
  font-size: 2.5rem;
}
```

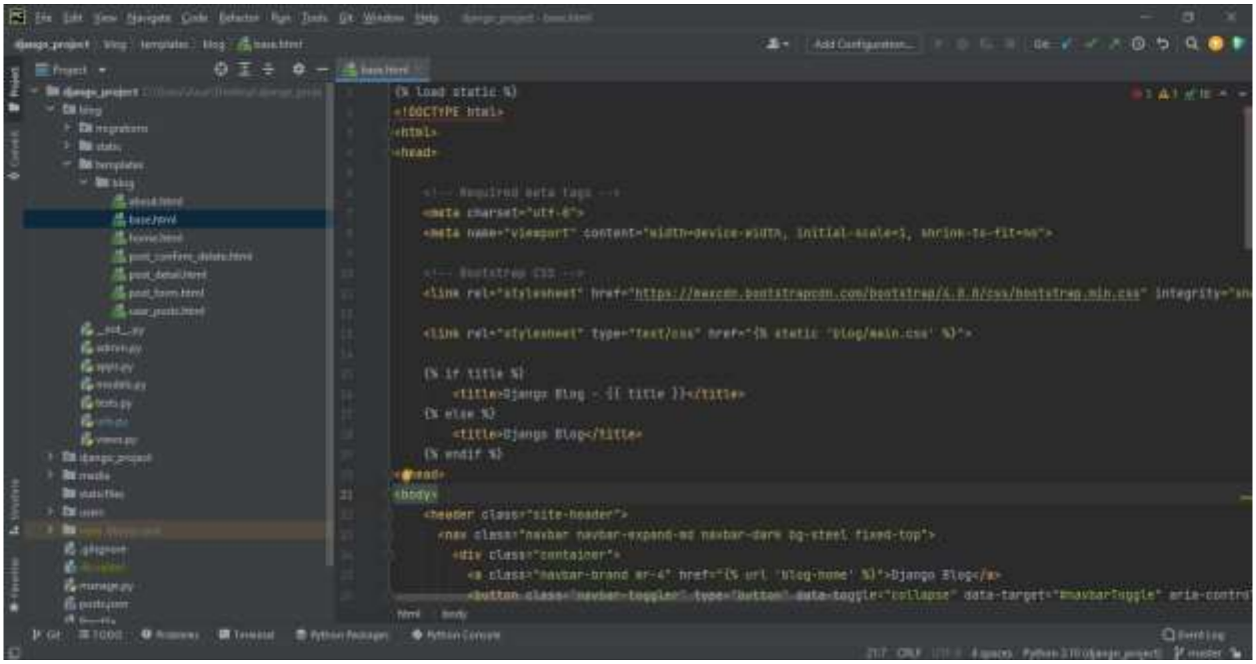
Blog App Templates

- django_project > blog > templates > blog > **about.html**

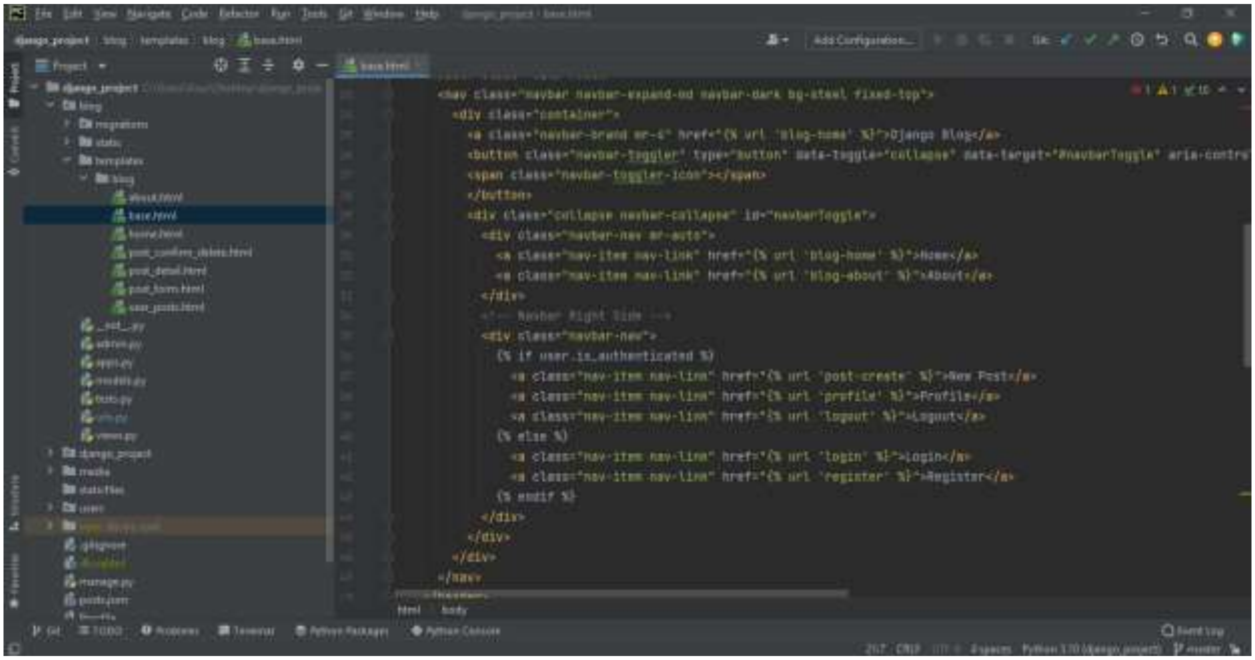


```
{% extends "blog/base.html" %}
{% block content %}
    <h1>About Page</h1>
{% endblock content %}
```

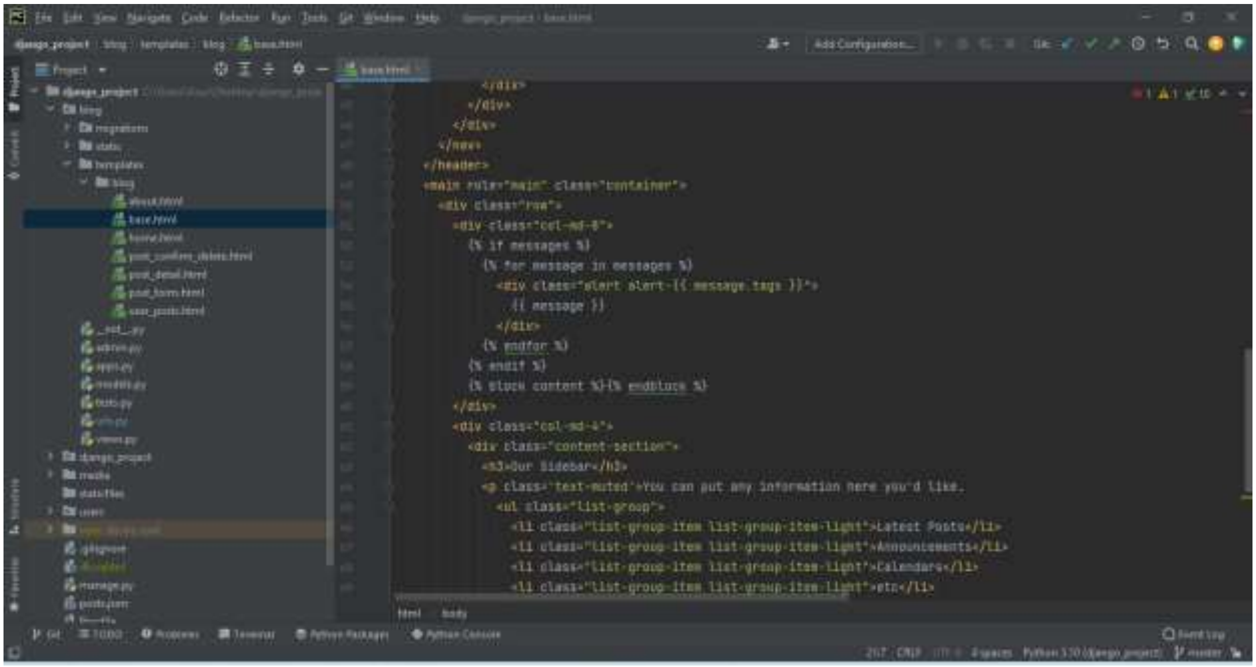
- django_project > blog > templates > blog > **base.html**



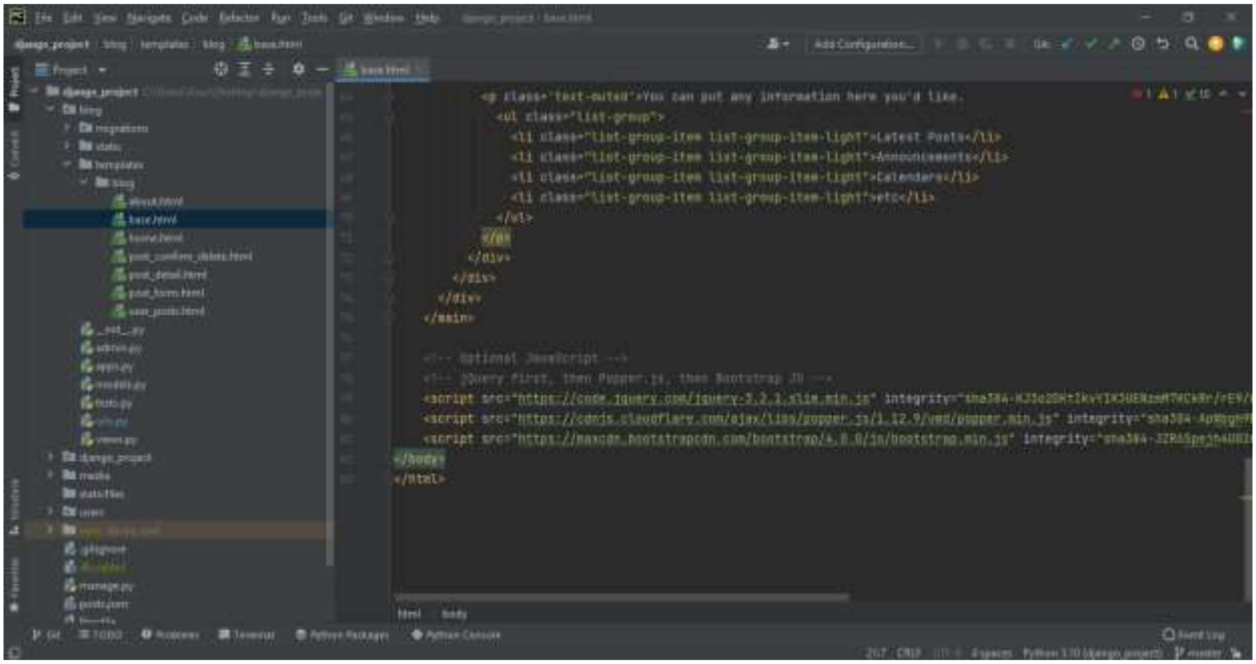
```
{% load static %}
<!DOCTYPE html>
<html>
<head>

    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha

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

    {% if title %}
        <title>Django Blog - {{ title }}</title>
    {% else %}
        <title>Django Blog</title>
    {% endif %}
</head>
<body>
    <header class="site-header">
      <nav class="navbar navbar-expand-md navbar-dark bg-steel fixed-top">
        <div class="container">
          <a class="navbar-brand mr-4" href="{% url 'blog-home' %}">Django Blog</a>
          <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarToggle" aria-contro
```



```
      <nav class="navbar navbar-expand-md navbar-dark bg-steel fixed-top">
        <div class="container">
          <a class="navbar-brand mr-4" href="{% url 'blog-home' %}">Django Blog</a>
          <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarToggle" aria-contro
          <span class="navbar-toggler-icon"></span>
          </button>
          <div class="collapse navbar-collapse" id="navbarToggle">
            <div class="navbar-nav mr-auto">
              <a class="nav-item nav-link" href="{% url 'blog-home' %}">Home</a>
              <a class="nav-item nav-link" href="{% url 'blog-about' %}">About</a>
            </div>
            <!-- Navbar Right Side -->
            <div class="navbar-nav">
              {% if user.is_authenticated %}
                <a class="nav-item nav-link" href="{% url 'post-create' %}">New Post</a>
                <a class="nav-item nav-link" href="{% url 'profile' %}">Profile</a>
                <a class="nav-item nav-link" href="{% url 'logout' %}">Logout</a>
              {% else %}
                <a class="nav-item nav-link" href="{% url 'login' %}">Login</a>
                <a class="nav-item nav-link" href="{% url 'register' %}">Register</a>
              {% endif %}
            </div>
          </div>
        </div>
      </nav>
```

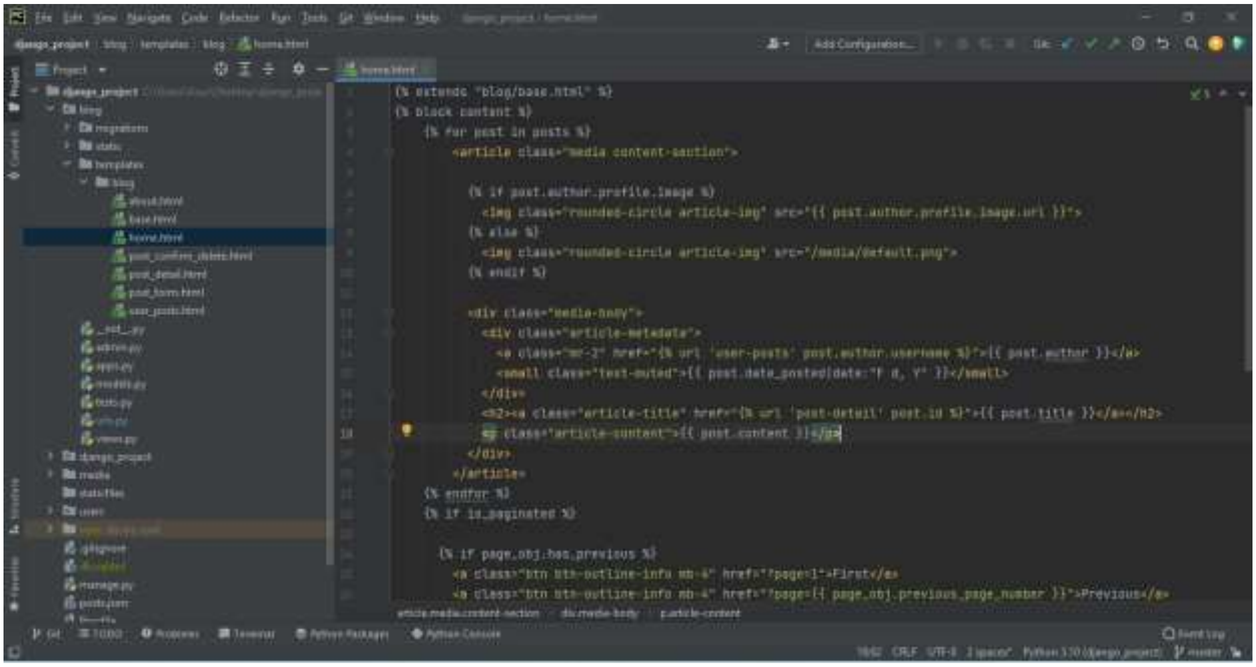


```
            </div>
          </div>
        </div>
      </nav>
    </header>
    <main role="main" class="container">
      <div class="row">
        <div class="col-md-8">
          {% if messages %}
            {% for message in messages %}
              <div class="alert alert-{{ message.tags }}">
                {{ message }}
              </div>
            {% endfor %}
          {% endif %}
          {% block content %}{% endblock %}
        </div>
        <div class="col-md-4">
          <div class="content-section">
            <h3>Our Sidebar</h3>
            <p class='text-muted'>You can put any information here you'd like.
              <ul class="list-group">
                <li class="list-group-item list-group-item-light">Latest Posts</li>
                <li class="list-group-item list-group-item-light">Announcements</li>
                <li class="list-group-item list-group-item-light">Calendars</li>
                <li class="list-group-item list-group-item-light">etc</li>
```

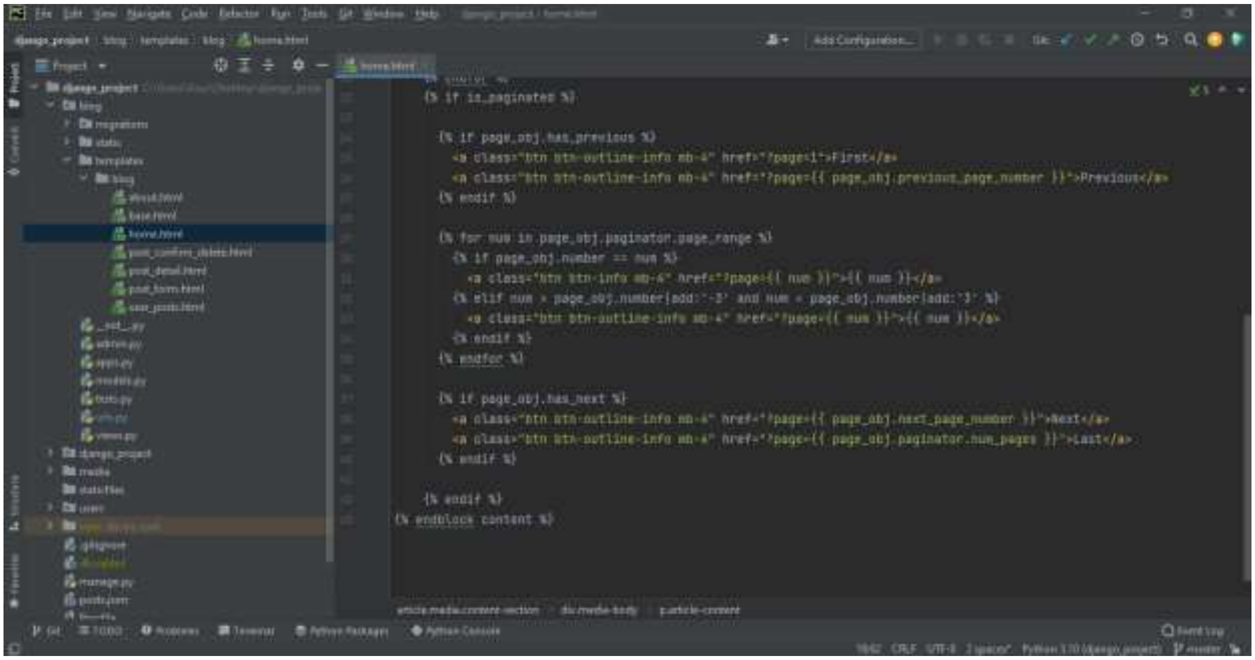


```
            <p class='text-muted'>You can put any information here you'd like.
              <ul class="list-group">
                <li class="list-group-item list-group-item-light">Latest Posts</li>
                <li class="list-group-item list-group-item-light">Announcements</li>
                <li class="list-group-item list-group-item-light">Calendars</li>
                <li class="list-group-item list-group-item-light">etc</li>
              </ul>
            </p>
          </div>
        </div>
      </div>
    </main>

    <!-- Optional JavaScript -->
    <!-- jQuery first, then Popper.js, then Bootstrap JS -->
    <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-ApNbgh9
    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js" integrity="sha384-JZR6Spejh4U02
</body>
</html>
```

- django_project > blog > templates > blog > **home.html**

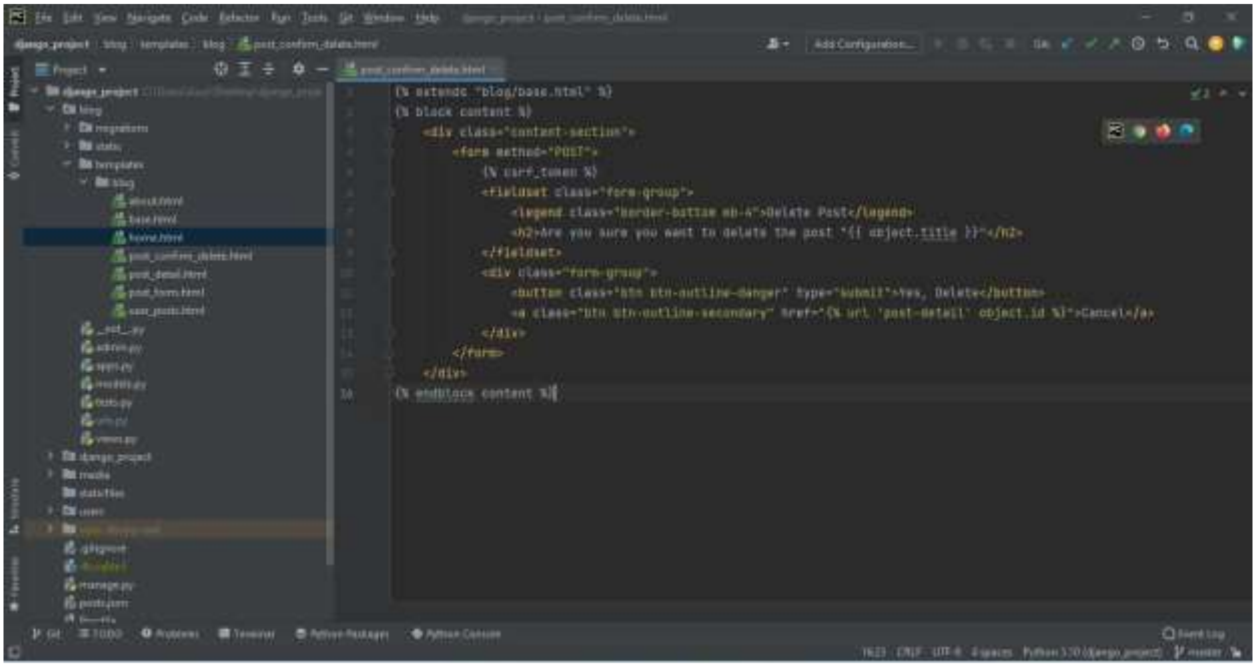


```
{% extends "blog/base.html" %}
{% block content %}
  {% for post in posts %}
    <article class="media content-section">

      {% if post.author.profile.image %}
        <img class="rounded-circle article-img" src="{{ post.author.profile.image.url }}">
      {% else %}
        <img class="rounded-circle article-img" src="/media/default.png">
      {% endif %}

      <div class="media-body">
        <div class="article-metadata">
          <a class="mr-2" href="{% url 'user-posts' post.author.username %}">{{ post.author }}</a>
          <small class="text-muted">{{ post.date_posted|date:"F d, Y" }}</small>
        </div>
        <h2><a class="article-title" href="{% url 'post-detail' post.id %}">{{ post.title }}</a></h2>
        <p class="article-content">{{ post.content }}</p>
      </div>
    </article>
  {% endfor %}
  {% if is_paginated %}

    {% if page_obj.has_previous %}
      <a class="btn btn-outline-info mb-4" href="?page=1">First</a>
      <a class="btn btn-outline-info mb-4" href="?page={{ page_obj.previous_page_number }}">Previous</a>
```



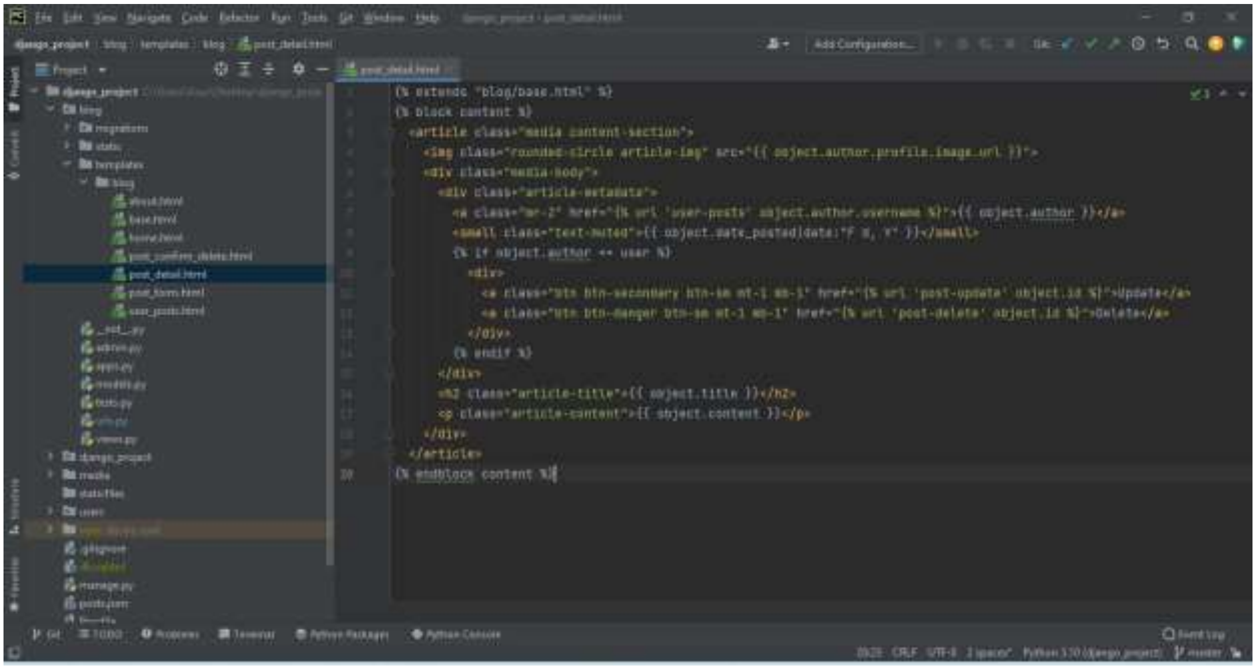
```
  {% if is_paginated %}

    {% if page_obj.has_previous %}
      <a class="btn btn-outline-info mb-4" href="?page=1">First</a>
      <a class="btn btn-outline-info mb-4" href="?page={{ page_obj.previous_page_number }}">Previous</a>
    {% endif %}

    {% for num in page_obj.paginator.page_range %}
      {% if page_obj.number == num %}
        <a class="btn btn-info mb-4" href="?page={{ num }}">{{ num }}</a>
      {% elif num > page_obj.number|add:'-3' and num < page_obj.number|add:'3' %}
        <a class="btn btn-outline-info mb-4" href="?page={{ num }}">{{ num }}</a>
      {% endif %}
    {% endfor %}

    {% if page_obj.has_next %}
      <a class="btn btn-outline-info mb-4" href="?page={{ page_obj.next_page_number }}">Next</a>
      <a class="btn btn-outline-info mb-4" href="?page={{ page_obj.paginator.num_pages }}">Last</a>
    {% endif %}

  {% endif %}
{% endblock content %}
```

- django_project > blog > templates > blog > **post_confirm_delete.html**



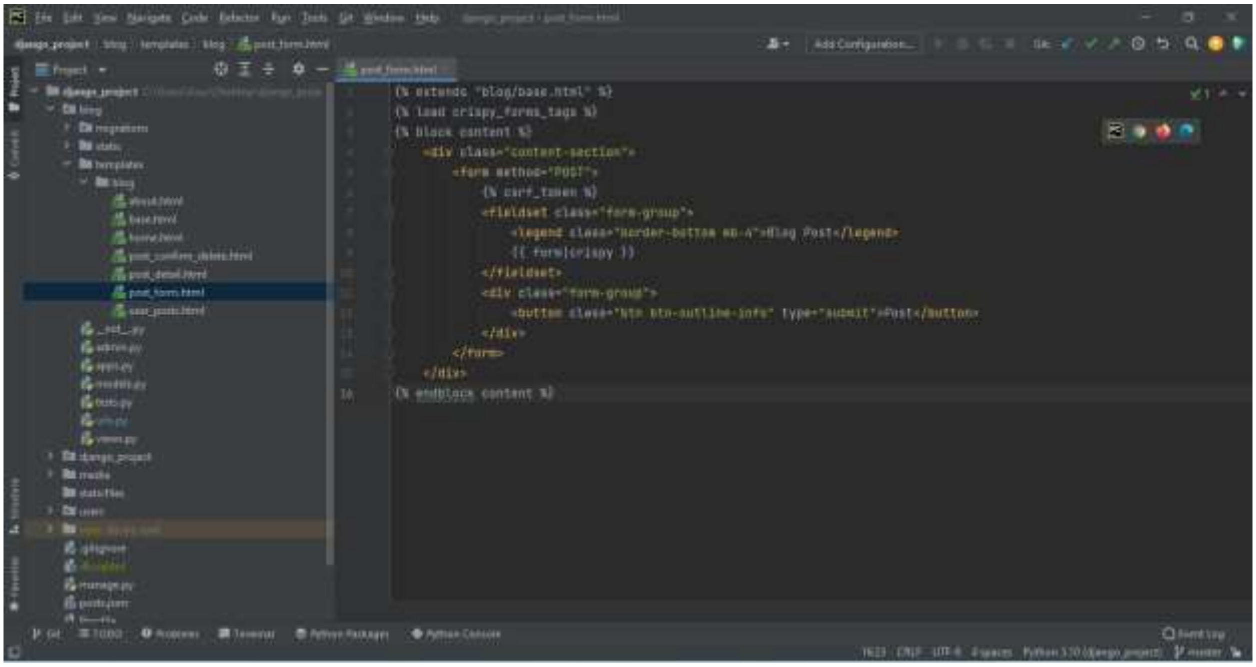
```
{% extends "blog/base.html" %}
{% block content %}
    <div class="content-section">
        <form method="POST">
            {% csrf_token %}
            <fieldset class="form-group">
                <legend class="border-bottom mb-4">Delete Post</legend>
                <h2>Are you sure you want to delete the post "{{ object.title }}"</h2>
            </fieldset>
            <div class="form-group">
                <button class="btn btn-outline-danger" type="submit">Yes, Delete</button>
                <a class="btn btn-outline-secondary" href="{% url 'post-detail' object.id %}">Cancel</a>
            </div>
        </form>
    </div>
{% endblock content %}
```

- django_project > blog > templates > blog > **post_detail.html**



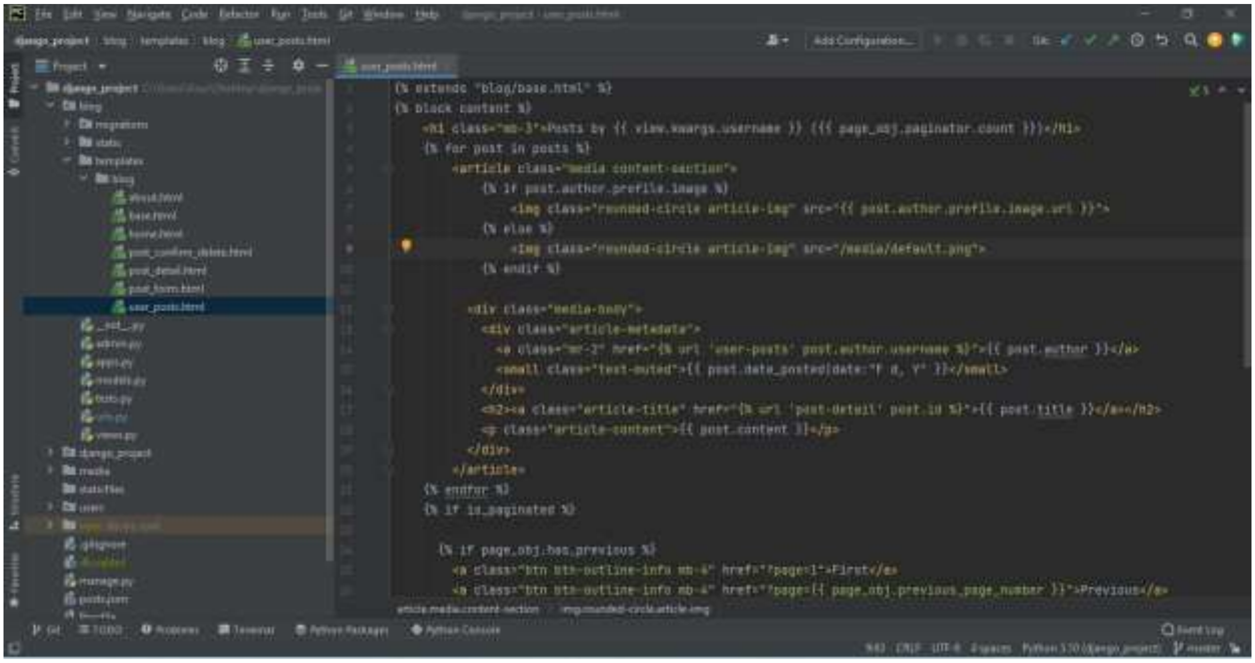
```
{% extends "blog/base.html" %}
{% block content %}
  <article class="media content-section">
    <img class="rounded-circle article-img" src="{{ object.author.profile.image.url }}">
    <div class="media-body">
      <div class="article-metadata">
        <a class="mr-2" href="{% url 'user-posts' object.author.username %}">{{ object.author }}</a>
        <small class="text-muted">{{ object.date_posted|date:"F d, Y" }}</small>
        {% if object.author == user %}
          <div>
            <a class="btn btn-secondary btn-sm mt-1 mb-1" href="{% url 'post-update' object.id %}">Update</a>
            <a class="btn btn-danger btn-sm mt-1 mb-1" href="{% url 'post-delete' object.id %}">Delete</a>
          </div>
        {% endif %}
      </div>
      <h2 class="article-title">{{ object.title }}</h2>
      <p class="article-content">{{ object.content }}</p>
    </div>
  </article>
{% endblock content %}
```

- django_project > blog > templates > blog > **post_form.html**

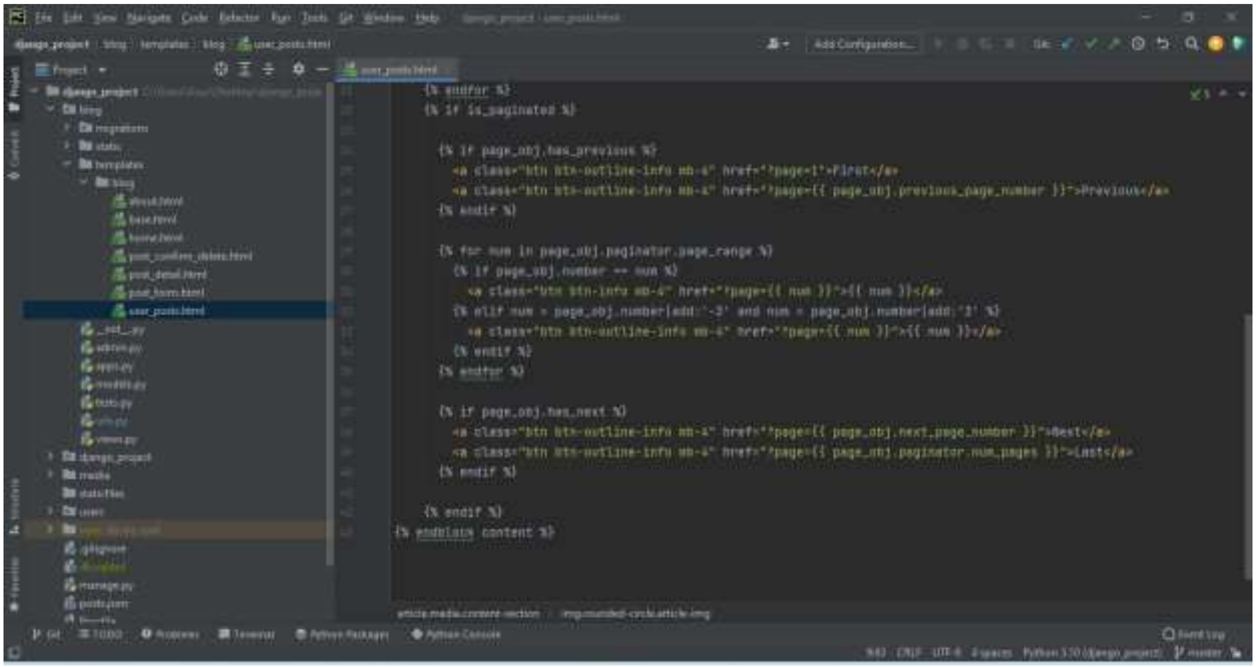


```
{% extends "blog/base.html" %}
{% load crispy_forms_tags %}
{% block content %}
    <div class="content-section">
        <form method="POST">
            {% csrf_token %}
            <fieldset class="form-group">
                <legend class="border-bottom mb-4">Blog Post</legend>
                {{ form|crispy }}
            </fieldset>
            <div class="form-group">
                <button class="btn btn-outline-info" type="submit">Post</button>
            </div>
        </form>
    </div>
{% endblock content %}
```

- django_project > blog > templates > blog > **user_posts.html**



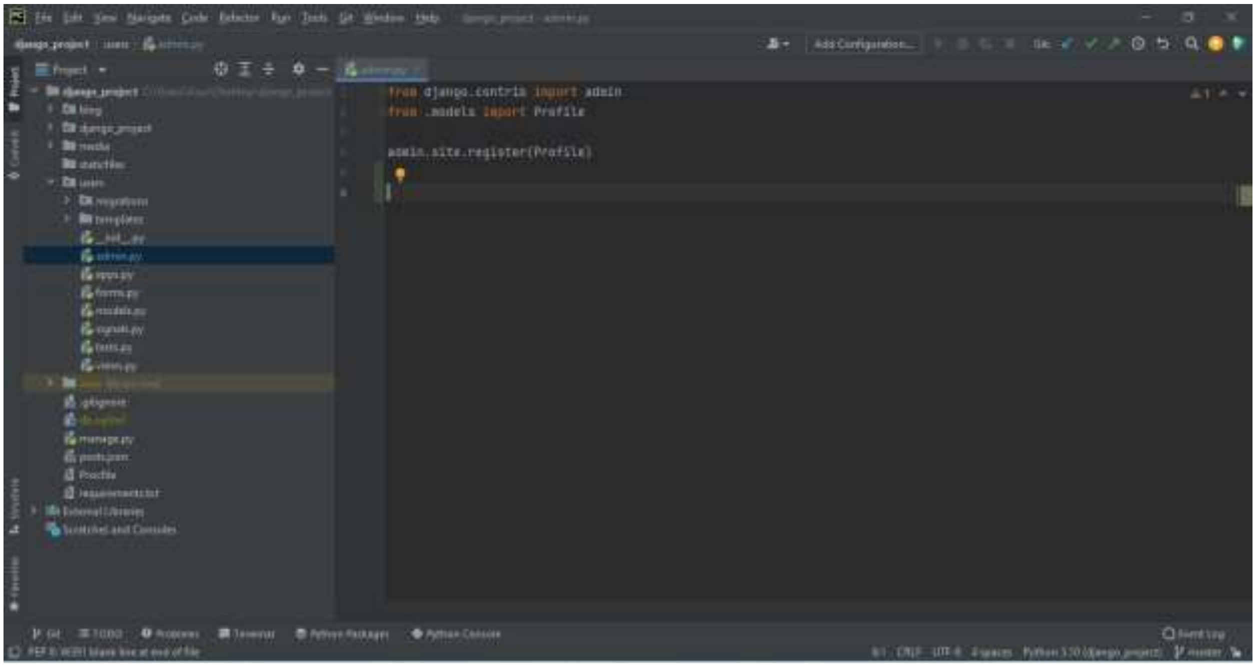
```
{% extends "blog/base.html" %}
{% block content %}
    <h1 class="mb-3">Posts by {{ view.kwargs.username }} ({{ page_obj.paginator.count }})</h1>
    {% for post in posts %}
        <article class="media content-section">
            {% if post.author.profile.image %}
                <img class="rounded-circle article-img" src="{{ post.author.profile.image.url }}">
            {% else %}
                <img class="rounded-circle article-img" src="/media/default.png">
            {% endif %}

          <div class="media-body">
            <div class="article-metadata">
              <a class="mr-2" href="{% url 'user-posts' post.author.username %}">{{ post.author }}</a>
              <small class="text-muted">{{ post.date_posted|date:"F d, Y" }}</small>
            </div>
            <h2><a class="article-title" href="{% url 'post-detail' post.id %}">{{ post.title }}</a></h2>
            <p class="article-content">{{ post.content }}</p>
          </div>
        </article>
    {% endfor %}
    {% if is_paginated %}

      {% if page_obj.has_previous %}
        <a class="btn btn-outline-info mb-4" href="?page=1">First</a>
        <a class="btn btn-outline-info mb-4" href="?page={{ page_obj.previous_page_number }}">Previous</a>
```



```
    {% endfor %}
    {% if is_paginated %}

      {% if page_obj.has_previous %}
        <a class="btn btn-outline-info mb-4" href="?page=1">First</a>
        <a class="btn btn-outline-info mb-4" href="?page={{ page_obj.previous_page_number }}">Previous</a>
      {% endif %}

      {% for num in page_obj.paginator.page_range %}
        {% if page_obj.number == num %}
          <a class="btn btn-info mb-4" href="?page={{ num }}">{{ num }}</a>
        {% elif num > page_obj.number|add:'-3' and num < page_obj.number|add:'3' %}
          <a class="btn btn-outline-info mb-4" href="?page={{ num }}">{{ num }}</a>
        {% endif %}
      {% endfor %}

      {% if page_obj.has_next %}
        <a class="btn btn-outline-info mb-4" href="?page={{ page_obj.next_page_number }}">Next</a>
        <a class="btn btn-outline-info mb-4" href="?page={{ page_obj.paginator.num_pages }}">Last</a>
      {% endif %}

    {% endif %}
{% endblock content %}
```

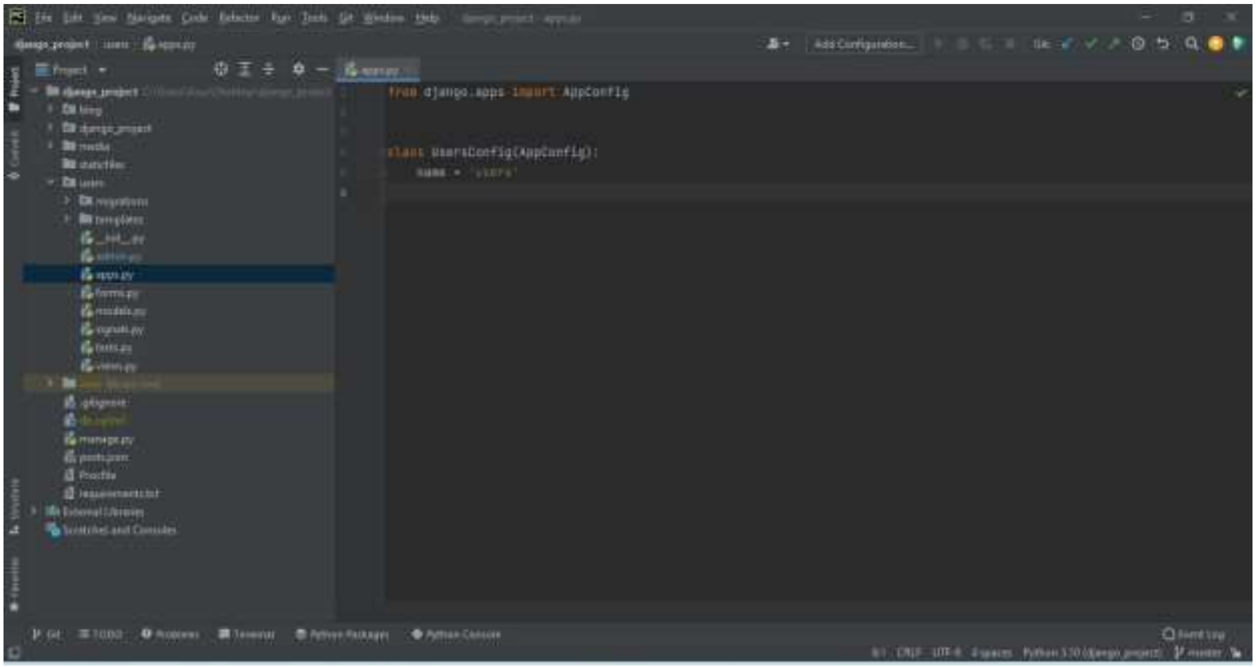
Users APP

- django_project > users > **admin.py**



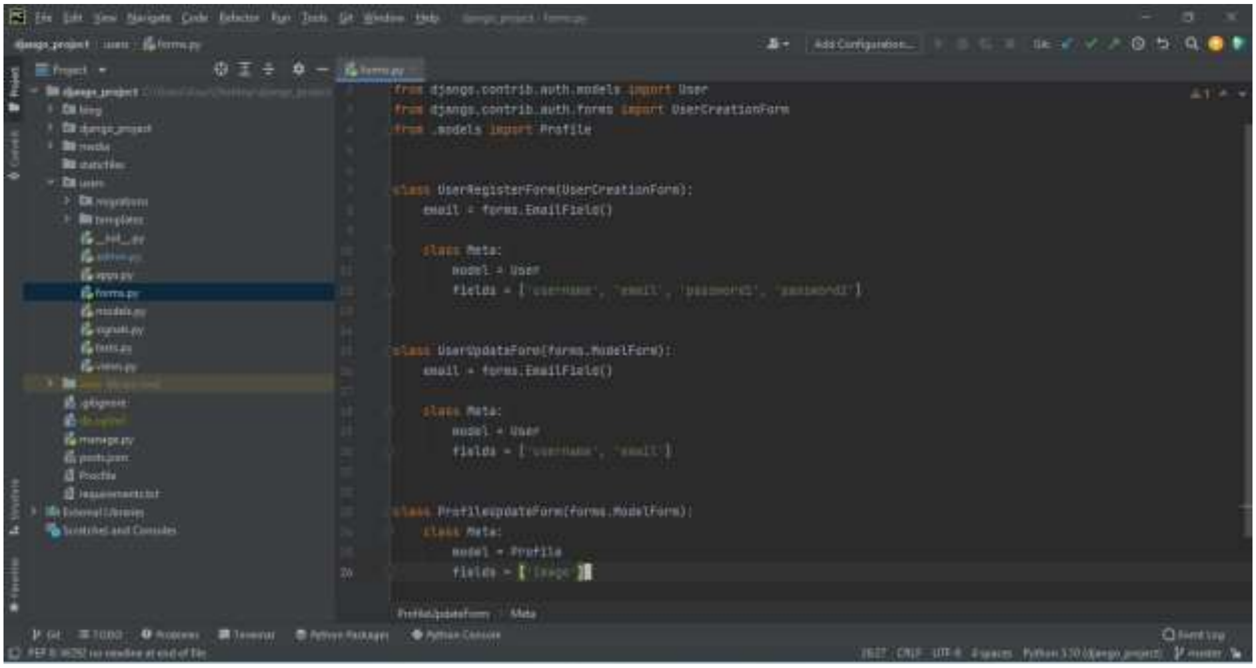
```
from django.contrib import admin
from .models import Profile

admin.site.register(Profile)

```

- django_project > users > **apps.py**



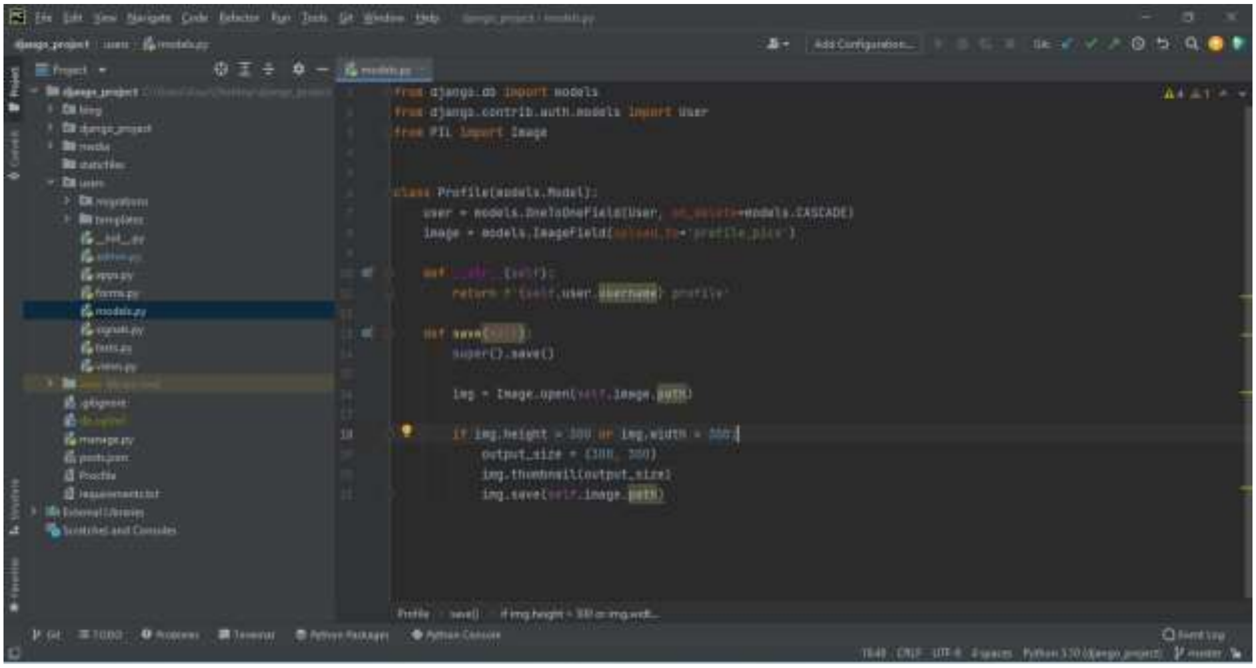
```python
from django.apps import AppConfig


class UsersConfig(AppConfig):
    name = 'users'
```

- django_project > users > **forms.py**



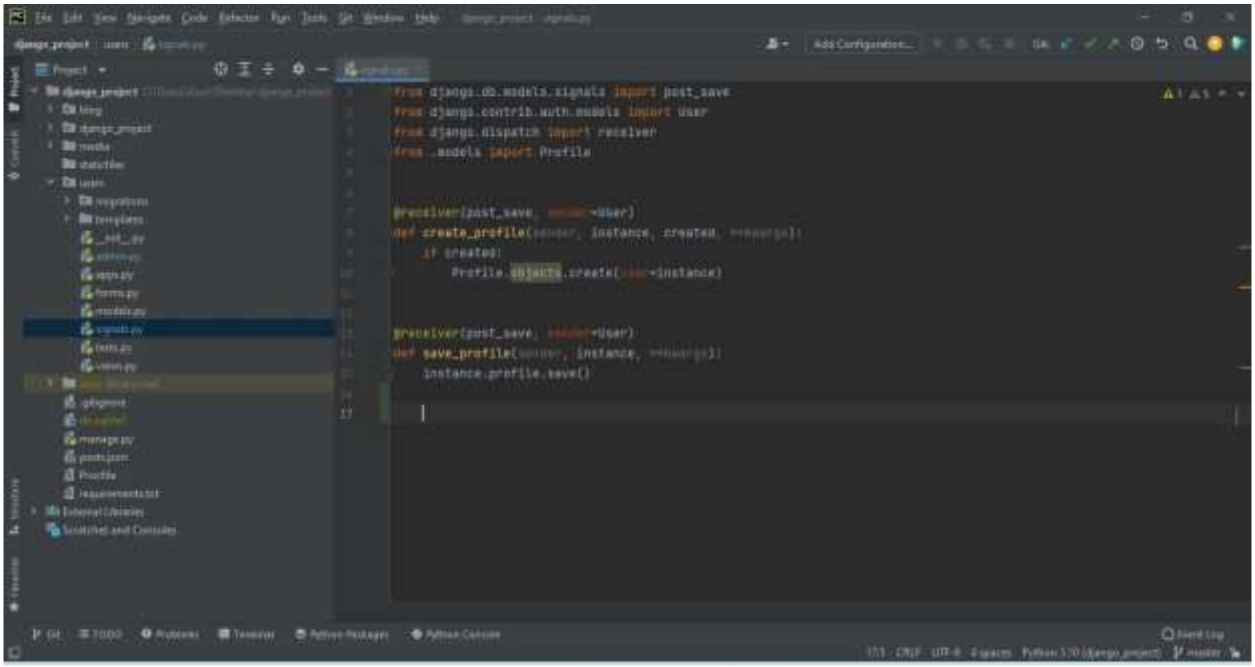
```python
from django.contrib.auth.models import User
from django.contrib.auth.forms import UserCreationForm
from .models import Profile


class UserRegisterForm(UserCreationForm):
    email = forms.EmailField()

    class Meta:
        model = User
        fields = ['username', 'email', 'password1', 'password2']


class UserUpdateForm(forms.ModelForm):
    email = forms.EmailField()

    class Meta:
        model = User
        fields = ['username', 'email']


class ProfileUpdateForm(forms.ModelForm):
    class Meta:
        model = Profile
        fields = ['image']
```

- django_project > users > **models.py**



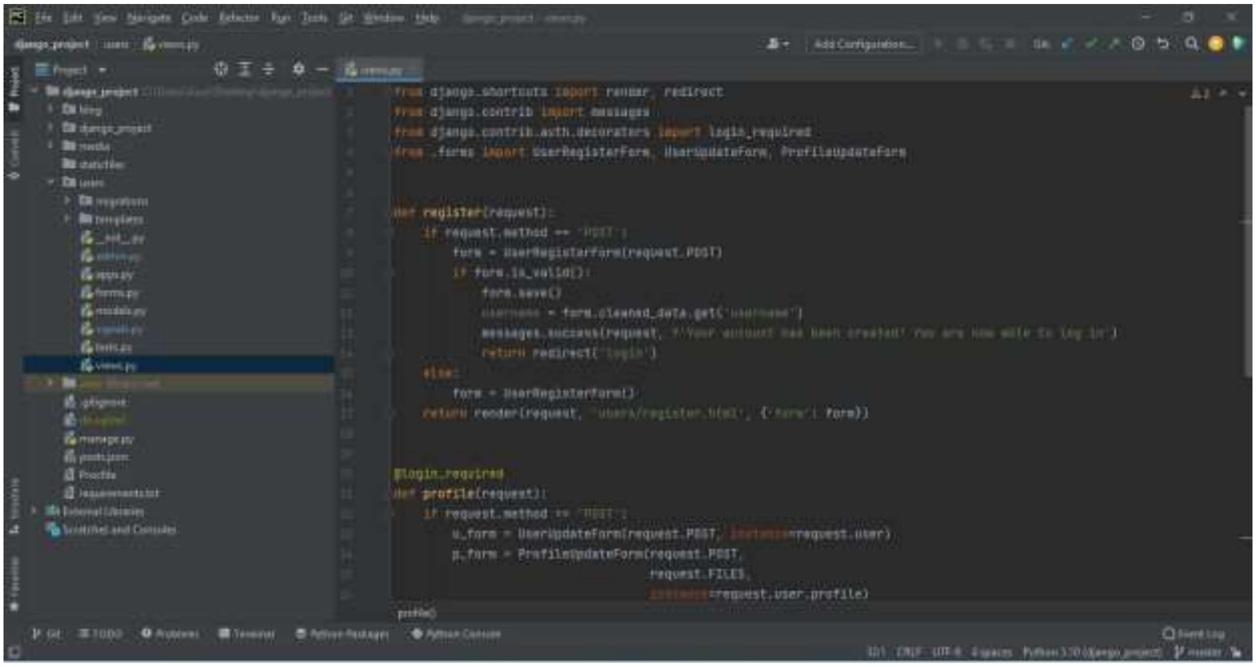
```python
from django.db import models
from django.contrib.auth.models import User
from PIL import Image


class Profile(models.Model):
    user = models.OneToOneField(User, on_delete=models.CASCADE)
    image = models.ImageField(upload_to='profile_pics')

    def __str__(self):
        return f'{self.user.username} profile'

    def save(self):
        super().save()

        img = Image.open(self.image.path)

        if img.height > 300 or img.width > 300:
            output_size = (300, 300)
            img.thumbnail(output_size)
            img.save(self.image.path)
```

- django_project > users > **signals.py**

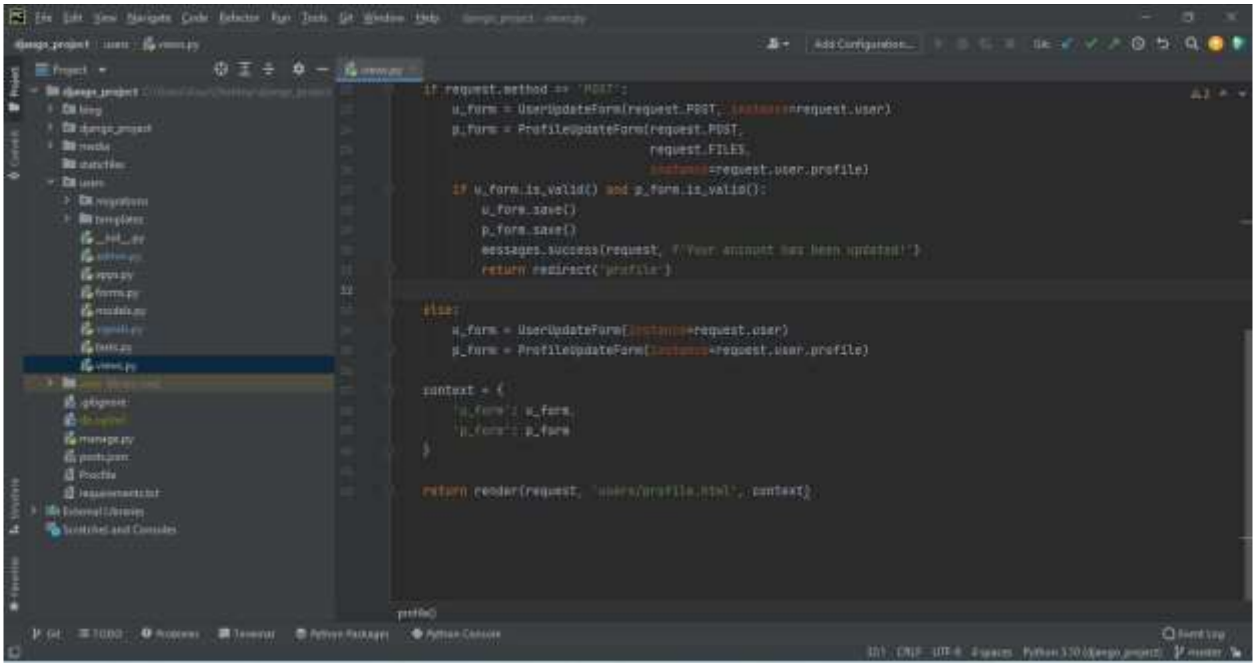


```python
from django.db.models.signals import post_save
from django.contrib.auth.models import User
from django.dispatch import receiver
from .models import Profile


@receiver(post_save, sender=User)
def create_profile(sender, instance, created, **kwargs):
    if created:
        Profile.objects.create(user=instance)


@receiver(post_save, sender=User)
def save_profile(sender, instance, **kwargs):
    instance.profile.save()
```

django_project > users > **views.py**



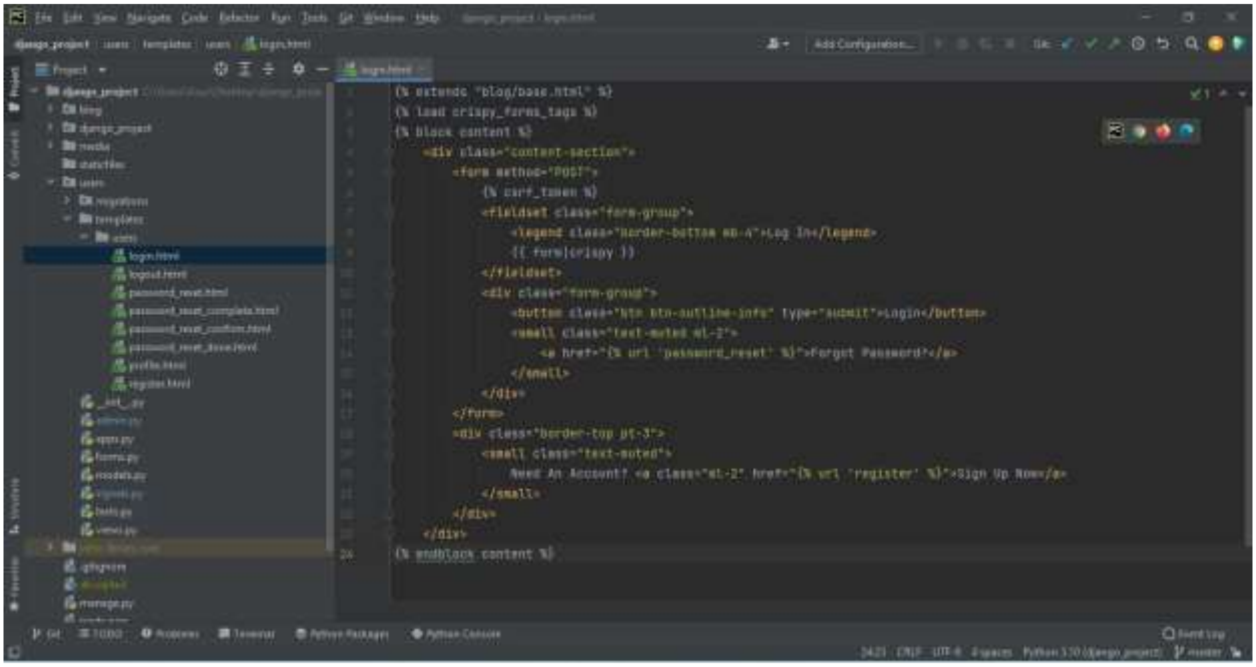
```python
from django.shortcuts import render, redirect
from django.contrib import messages
from django.contrib.auth.decorators import login_required
from .forms import UserRegisterForm, UserUpdateForm, ProfileUpdateForm


def register(request):
    if request.method == 'POST':
        form = UserRegisterForm(request.POST)
        if form.is_valid():
            form.save()
            username = form.cleaned_data.get('username')
            messages.success(request, f'Your account has been created! You are now able to log in')
            return redirect('login')
    else:
        form = UserRegisterForm()
    return render(request, 'users/register.html', {'form': form})


@login_required
def profile(request):
    if request.method == 'POST':
        u_form = UserUpdateForm(request.POST, instance=request.user)
        p_form = ProfileUpdateForm(request.POST,
                                   request.FILES,
                                   instance=request.user.profile)
```



```python
    if request.method == 'POST':
        u_form = UserUpdateForm(request.POST, instance=request.user)
        p_form = ProfileUpdateForm(request.POST,
                                   request.FILES,
                                   instance=request.user.profile)
        if u_form.is_valid() and p_form.is_valid():
            u_form.save()
            p_form.save()
            messages.success(request, f'Your account has been updated!')
            return redirect('profile')

    else:
        u_form = UserUpdateForm(instance=request.user)
        p_form = ProfileUpdateForm(instance=request.user.profile)

    context = {
        'u_form': u_form,
        'p_form': p_form
    }

    return render(request, 'users/profile.html', context)
```

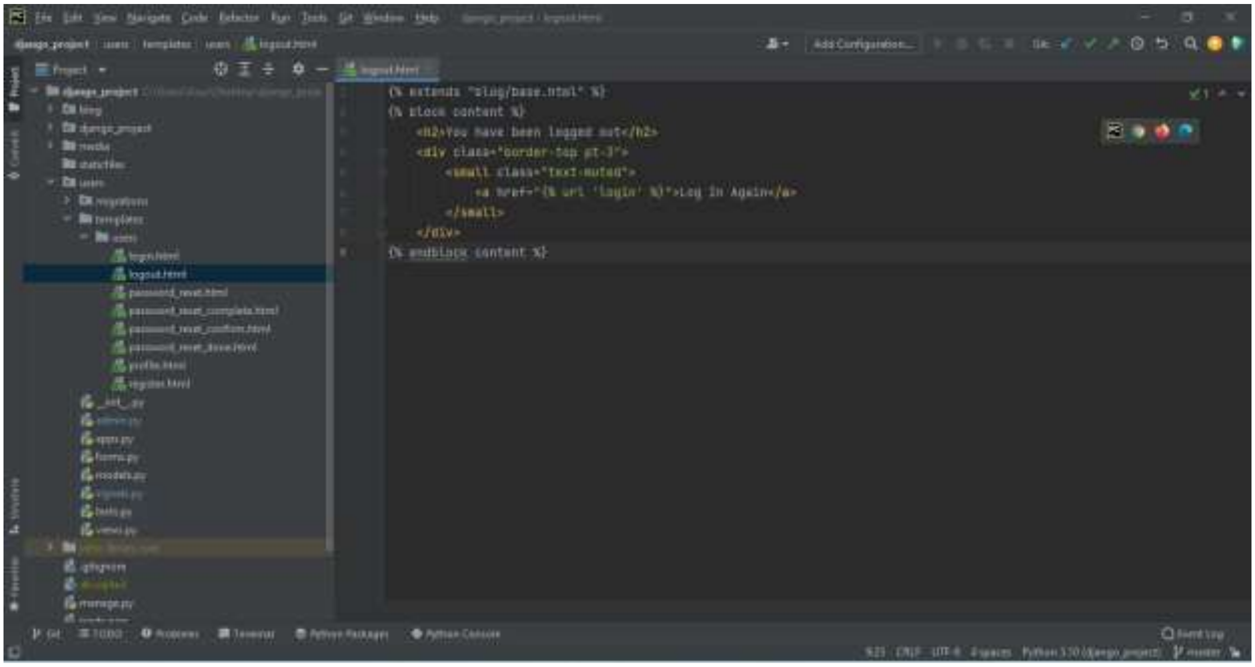
Users App Templates

- django_project > users > templates > users > **login.html**



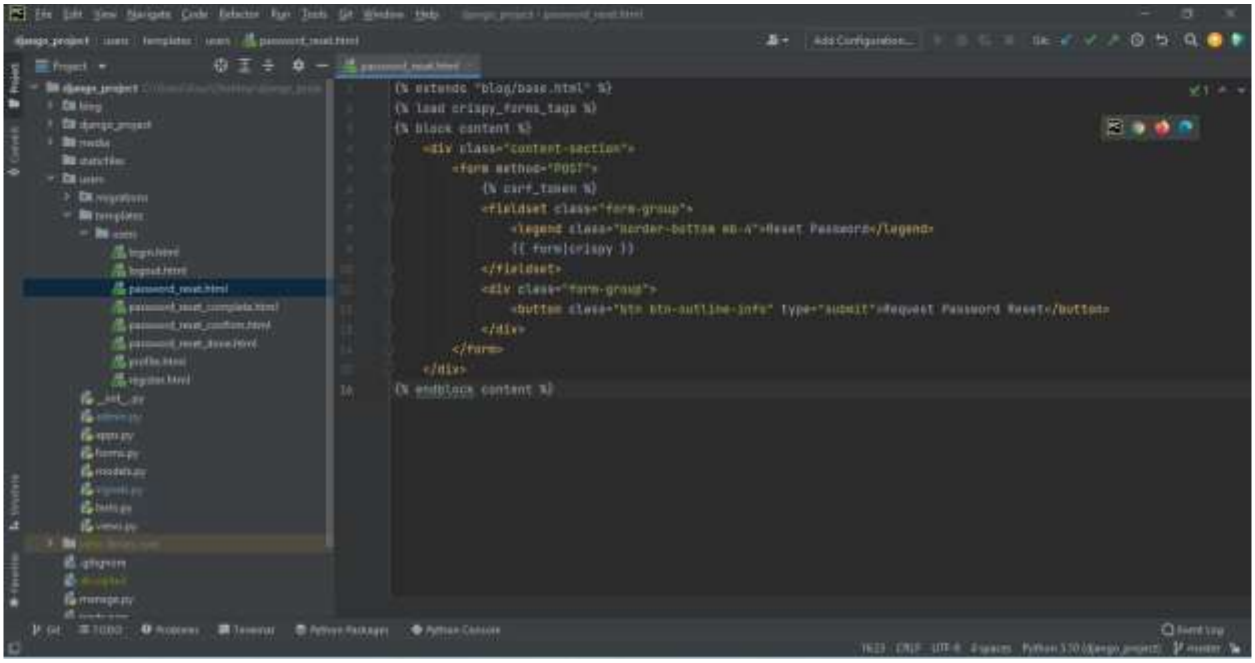
```
{% extends "blog/base.html" %}
{% load crispy_forms_tags %}
{% block content %}
    <div class="content-section">
        <form method="POST">
            {% csrf_token %}
            <fieldset class="form-group">
                <legend class="border-bottom mb-4">Log In</legend>
                {{ form|crispy }}
            </fieldset>
            <div class="form-group">
                <button class="btn btn-outline-info" type="submit">Login</button>
                <small class="text-muted ml-2">
                    <a href="{% url 'password_reset' %}">Forgot Password?</a>
                </small>
            </div>
        </form>
        <div class="border-top pt-3">
            <small class="text-muted">
                Need An Account? <a class="ml-2" href="{% url 'register' %}">Sign Up Now</a>
            </small>
        </div>
    </div>
{% endblock content %}
```

- django_project > users > templates > users > **logout.html**



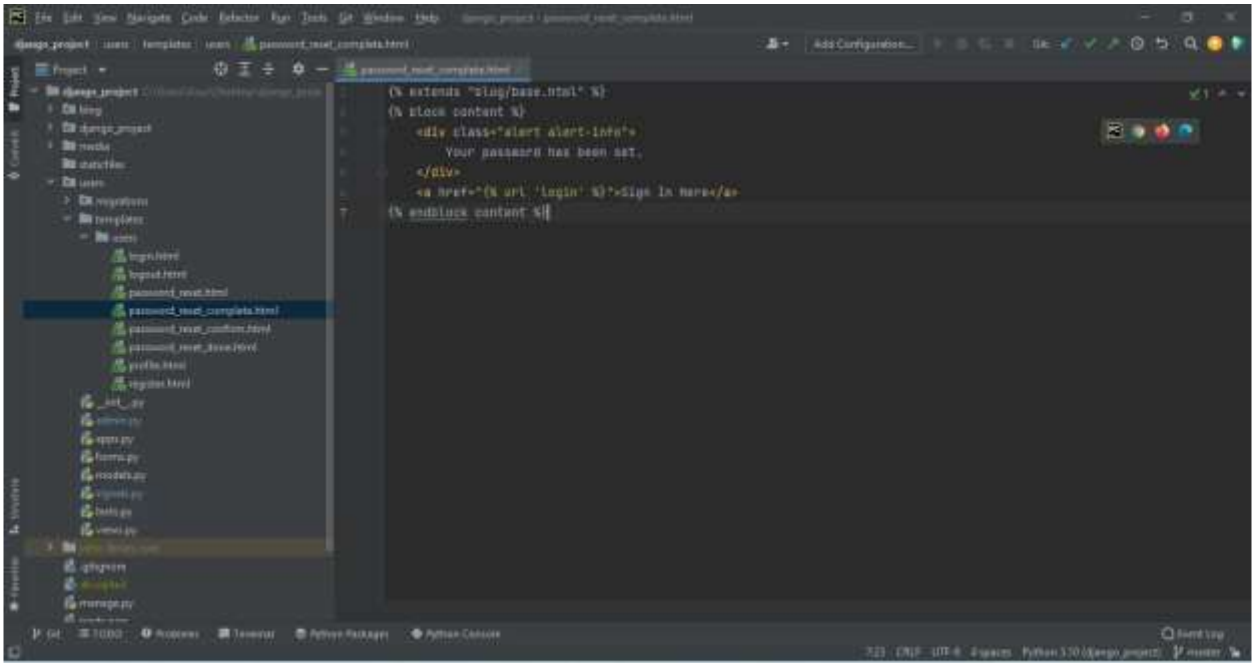
```
{% extends "blog/base.html" %}
{% block content %}
    <h2>You have been logged out</h2>
    <div class="border-top pt-3">
        <small class="text-muted">
            <a href="{% url 'login' %}">Log In Again</a>
        </small>
    </div>
{% endblock content %}
```

password_reset.html



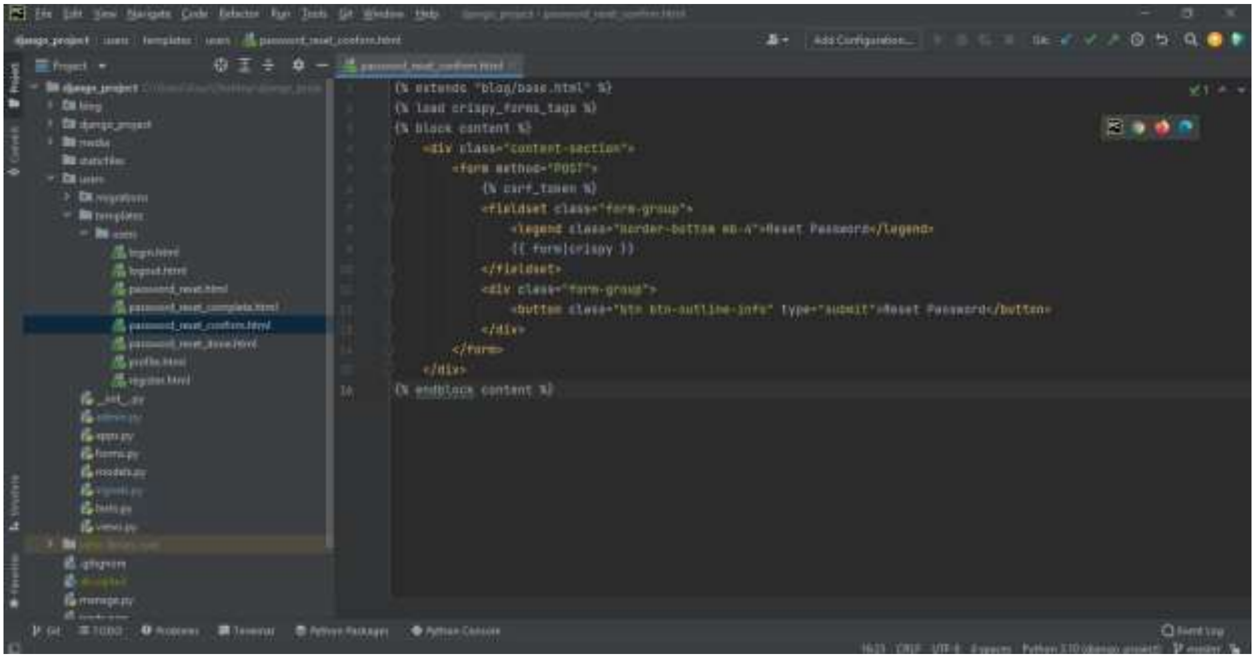
```
{% extends "blog/base.html" %}
{% load crispy_forms_tags %}
{% block content %}
    <div class="content-section">
        <form method="POST">
            {% csrf_token %}
            <fieldset class="form-group">
                <legend class="border-bottom mb-4">Reset Password</legend>
                {{ form|crispy }}
            </fieldset>
            <div class="form-group">
                <button class="btn btn-outline-info" type="submit">Request Password Reset</button>
            </div>
        </form>
    </div>
{% endblock content %}
```

- django_project > users > templates > users > **password_reset_complete.html**



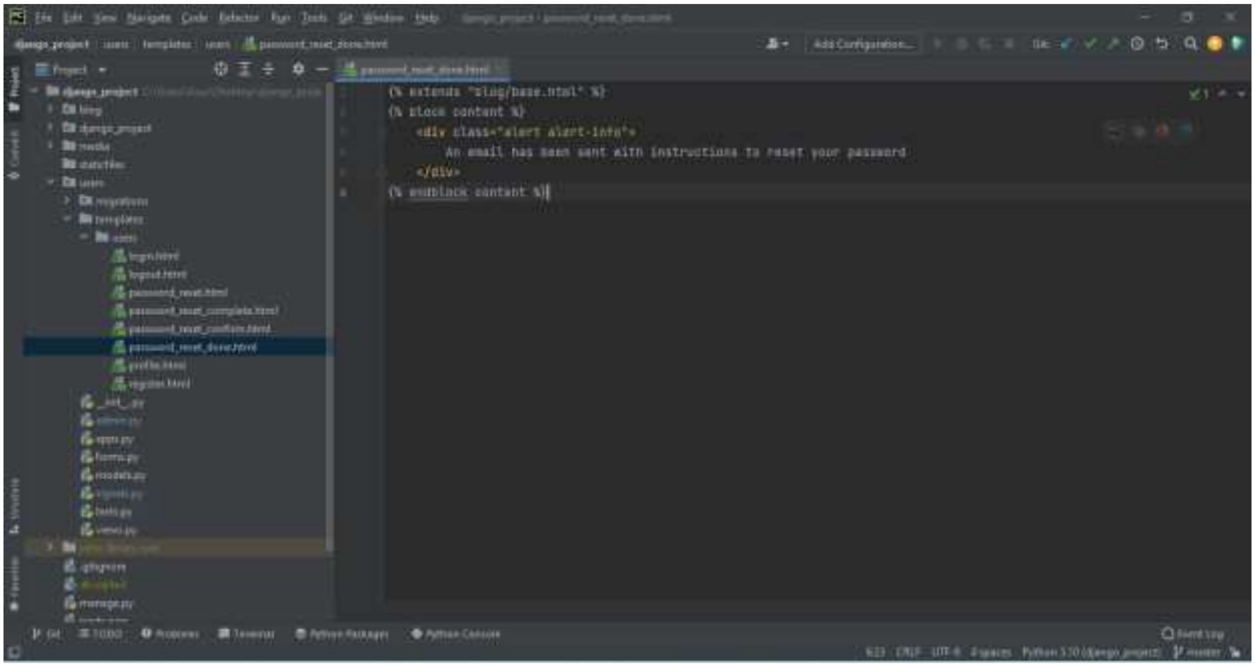
```
{% extends "blog/base.html" %}
{% block content %}
    <div class="alert alert-info">
        Your password has been set.
    </div>
    <a href="{% url 'login' %}">Sign In Here</a>
{% endblock content %}
```

password_reset_confirm.html



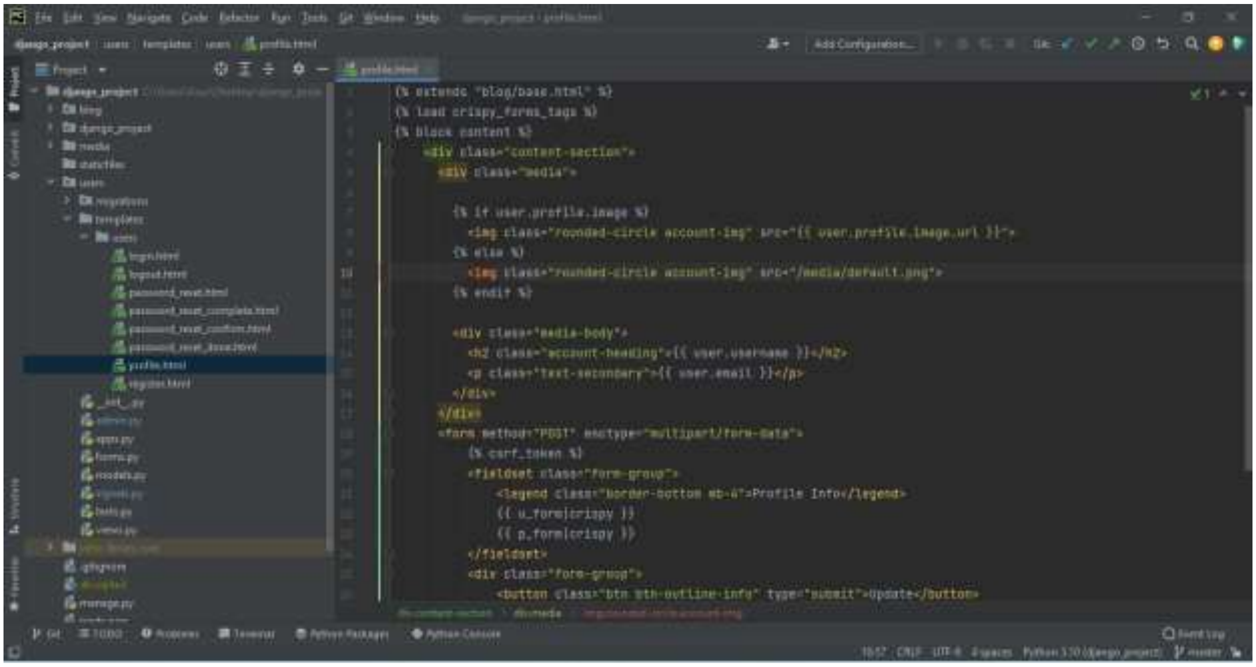
```
{% extends "blog/base.html" %}
{% load crispy_forms_tags %}
{% block content %}
    <div class="content-section">
        <form method="POST">
            {% csrf_token %}
            <fieldset class="form-group">
                <legend class="border-bottom mb-4">Reset Password</legend>
                {{ form|crispy }}
            </fieldset>
            <div class="form-group">
                <button class="btn btn-outline-info" type="submit">Reset Password</button>
            </div>
        </form>
    </div>
{% endblock content %}
```

- django_project > users > templates > users > **password_reset_done.html**

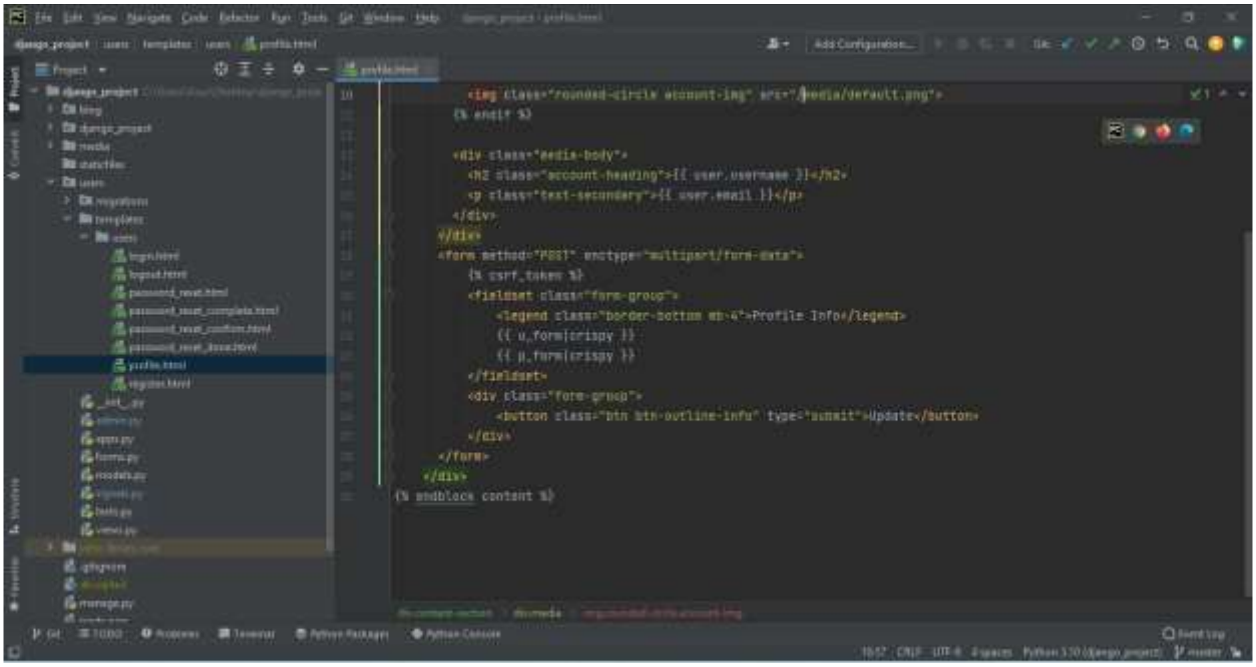


```
{% extends "blog/base.html" %}
{% block content %}
    <div class="alert alert-info">
        An email has been sent with instructions to reset your password
    </div>
{% endblock content %}
```

profile.html

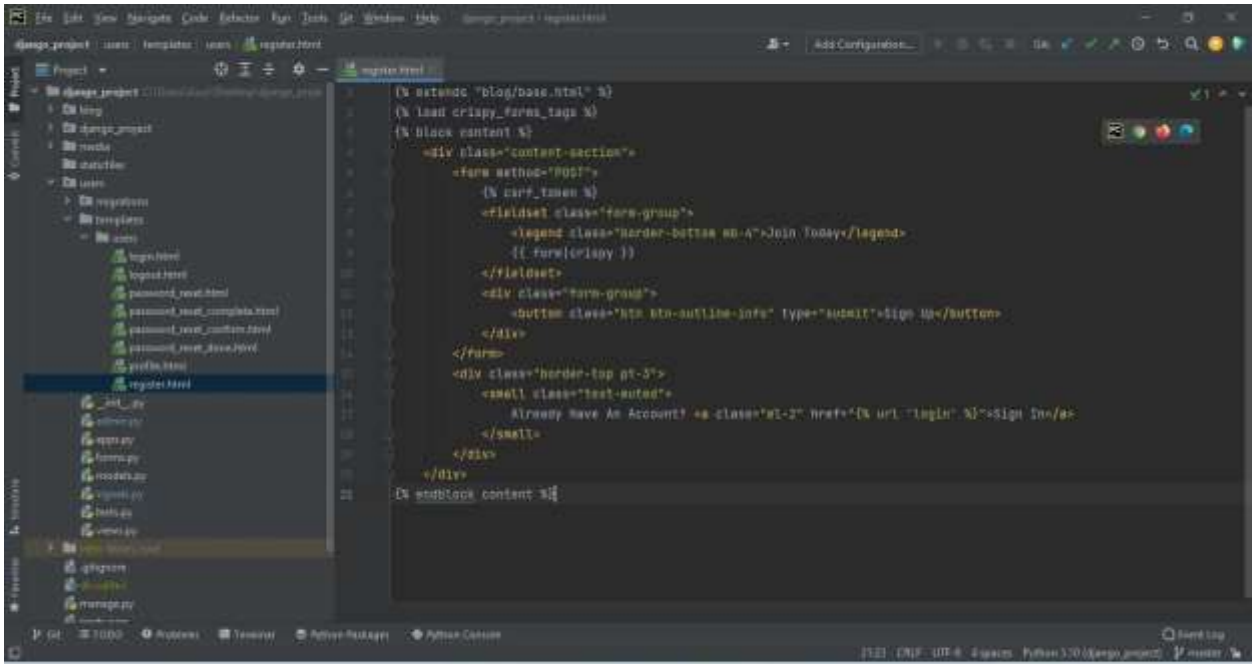


```
{% extends "blog/base.html" %}
{% load crispy_forms_tags %}
{% block content %}
    <div class="content-section">
      <div class="media">

        {% if user.profile.image %}
          <img class="rounded-circle account-img" src="{{ user.profile.image.url }}">
        {% else %}
          <img class="rounded-circle account-img" src="/media/default.png">
        {% endif %}

        <div class="media-body">
          <h2 class="account-heading">{{ user.username }}</h2>
          <p class="text-secondary">{{ user.email }}</p>
        </div>
      </div>
      <form method="POST" enctype="multipart/form-data">
          {% csrf_token %}
          <fieldset class="form-group">
              <legend class="border-bottom mb-4">Profile Info</legend>
              {{ u_form|crispy }}
              {{ p_form|crispy }}
          </fieldset>
          <div class="form-group">
              <button class="btn btn-outline-info" type="submit">Update</button>
```


File Edit View Navigate Code Refactor Run Tools Git Window Help
django_project › users › templates › users › profile.html
Add Configuration...
Project
Git
TODO
Problems
Terminal
Python Packages
Python Console
Event Log

```
          <img class="rounded-circle account-img" src="/media/default.png">
        {% endif %}

        <div class="media-body">
          <h2 class="account-heading">{{ user.username }}</h2>
          <p class="text-secondary">{{ user.email }}</p>
        </div>
      </div>
      <form method="POST" enctype="multipart/form-data">
          {% csrf_token %}
          <fieldset class="form-group">
              <legend class="border-bottom mb-4">Profile Info</legend>
              {{ u_form|crispy }}
              {{ p_form|crispy }}
          </fieldset>
          <div class="form-group">
              <button class="btn btn-outline-info" type="submit">Update</button>
          </div>
      </form>
    </div>
{% endblock content %}
```

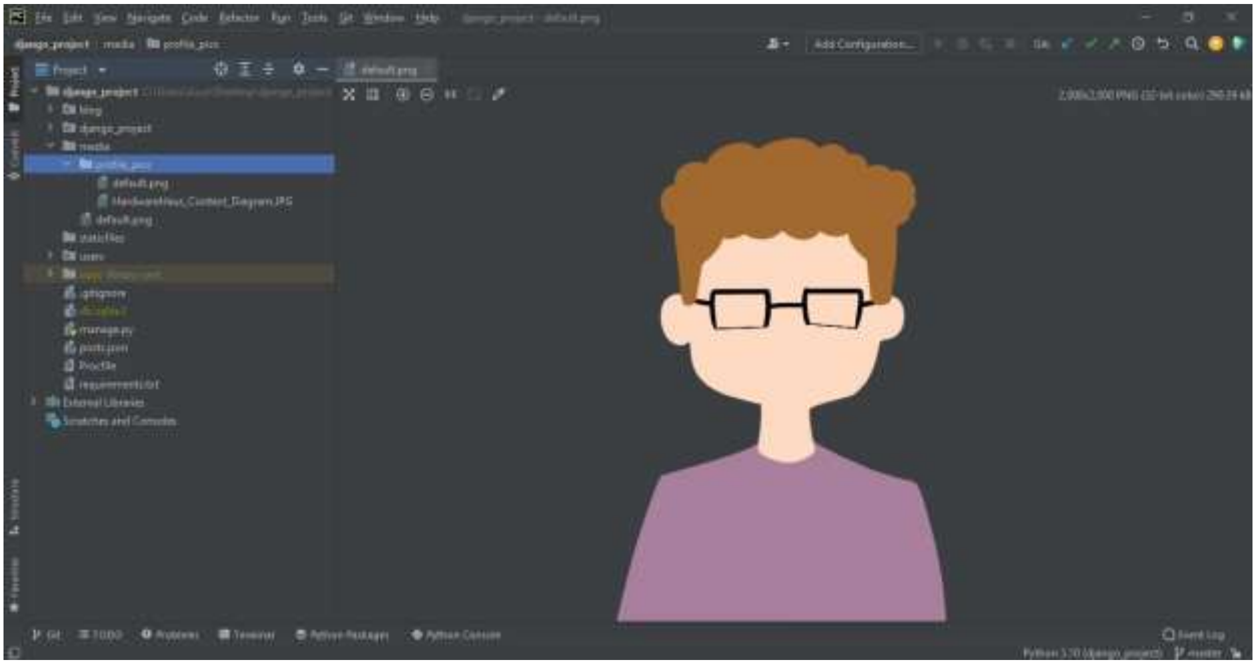
- django_project > users > templates > users > **register.html**



```
{% extends "blog/base.html" %}
{% load crispy_forms_tags %}
{% block content %}
    <div class="content-section">
        <form method="POST">
            {% csrf_token %}
            <fieldset class="form-group">
                <legend class="border-bottom mb-4">Join Today</legend>
                {{ form|crispy }}
            </fieldset>
            <div class="form-group">
                <button class="btn btn-outline-info" type="submit">Sign Up</button>
            </div>
        </form>
        <div class="border-top pt-3">
            <small class="text-muted">
                Already Have An Account? <a class="ml-2" href="{% url 'login' %}">Sign In</a>
            </small>
        </div>
    </div>
{% endblock content %}
```

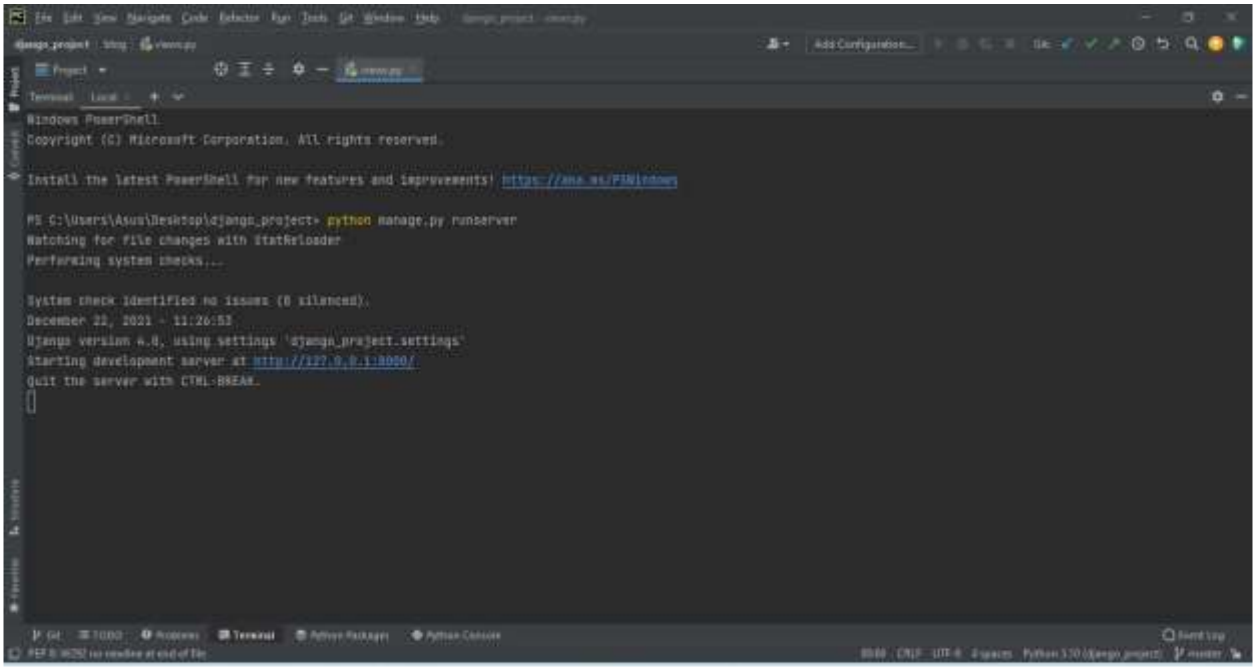
Media Files

- django_project > media > **default.png**



RUNSERVER Command

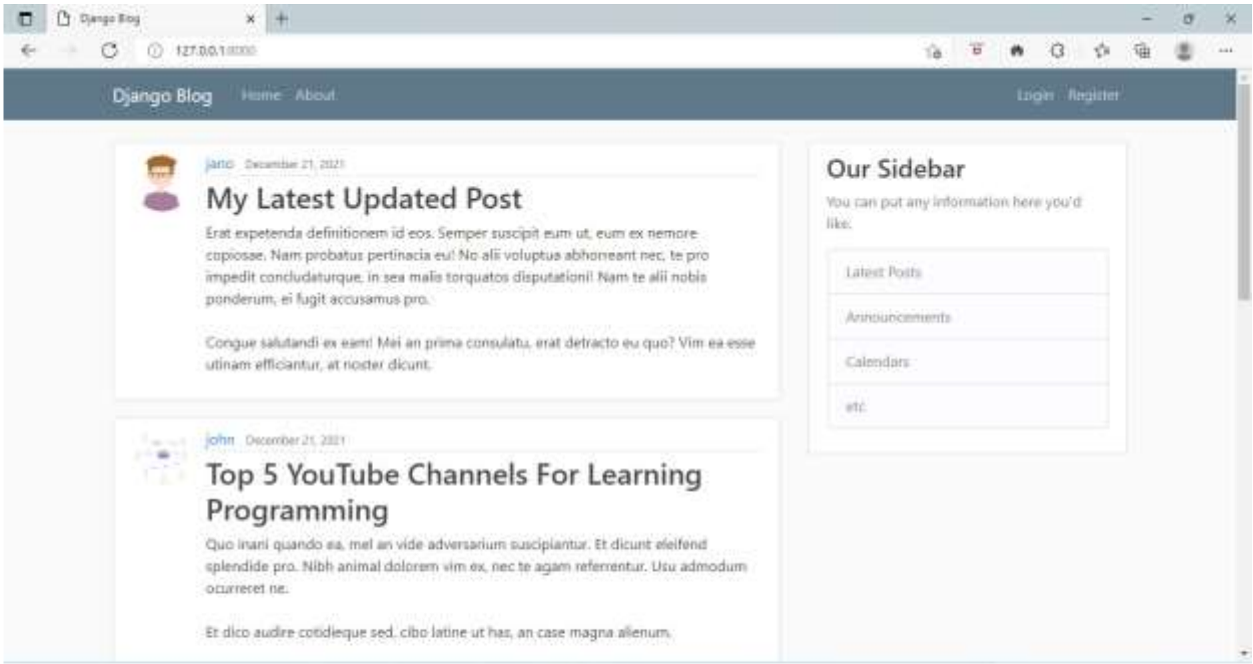
- **python manage.py runserver**



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Asus\Desktop\django_project> python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
December 22, 2021 - 11:26:53
Django version 4.0, using settings 'django_project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

DJANGO BLOG FINISHED RESULT/OUTPUT

- **Home Page**



jano December 21, 2021

My Latest Updated Post

Erat expetenda definitionem id eos. Semper suscipit eum ut, eum ex nemore copiosae. Nam probatus pertinacia eu! No alii voluptua abhorreant nec, te pro impedit concludaturque, in sea malis torquatos disputationi! Nam te alii nobis ponderum, ei fugit accusamus pro.

Congue salutandi ex eam! Mei an prima consulatu, erat detracto eu quo? Vim ea esse utinam efficiantur, at noster dicunt.

john December 21, 2021

Top 5 YouTube Channels For Learning Programming

Quo inani quando ea, mel an vide adversarium suscipiantur. Et dicunt eleifend splendide pro. Nibh animal dolorem vim ex, nec te agam referrentur. Usu admodum ocurreret ne.

Et dico audire cotidieque sed, cibo latine ut has, an case magna alienum.

Our Sidebar

You can put any information here you'd like.

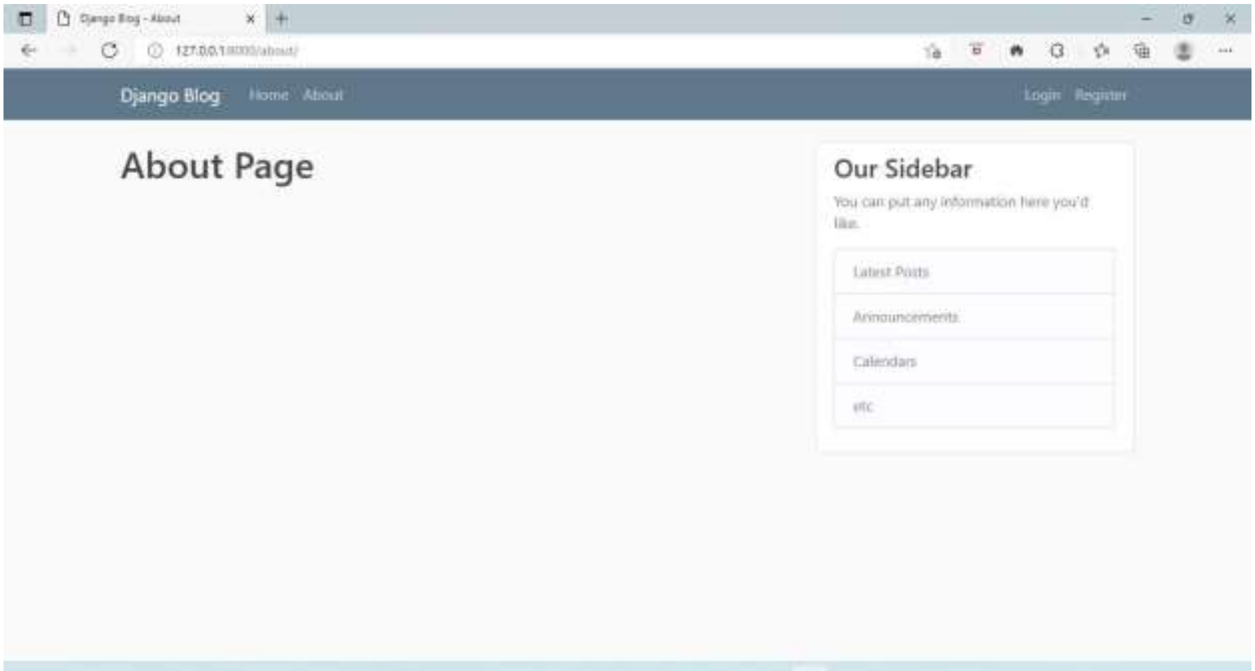
Latest Posts

Announcements

Calendars

etc

- **About Page**



About Page

Our Sidebar

You can put any information here you'd like.

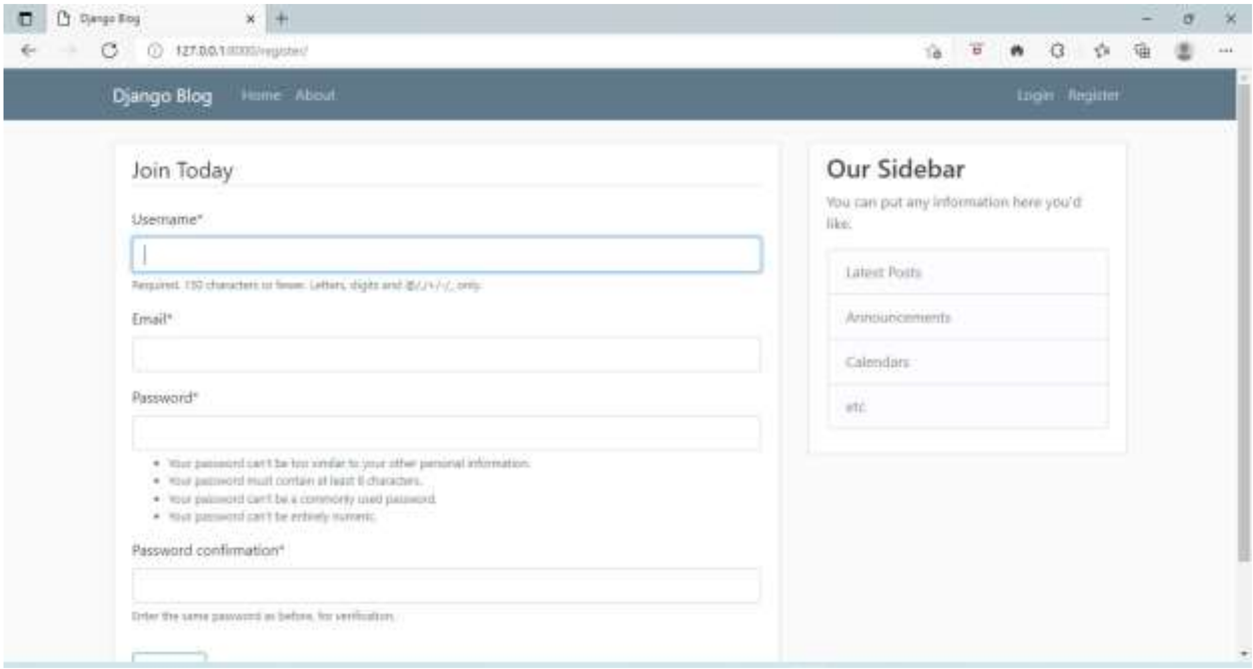
Latest Posts

Announcements

Calendars

etc

Register Page



Join Today

Username*

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email*

Password*

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

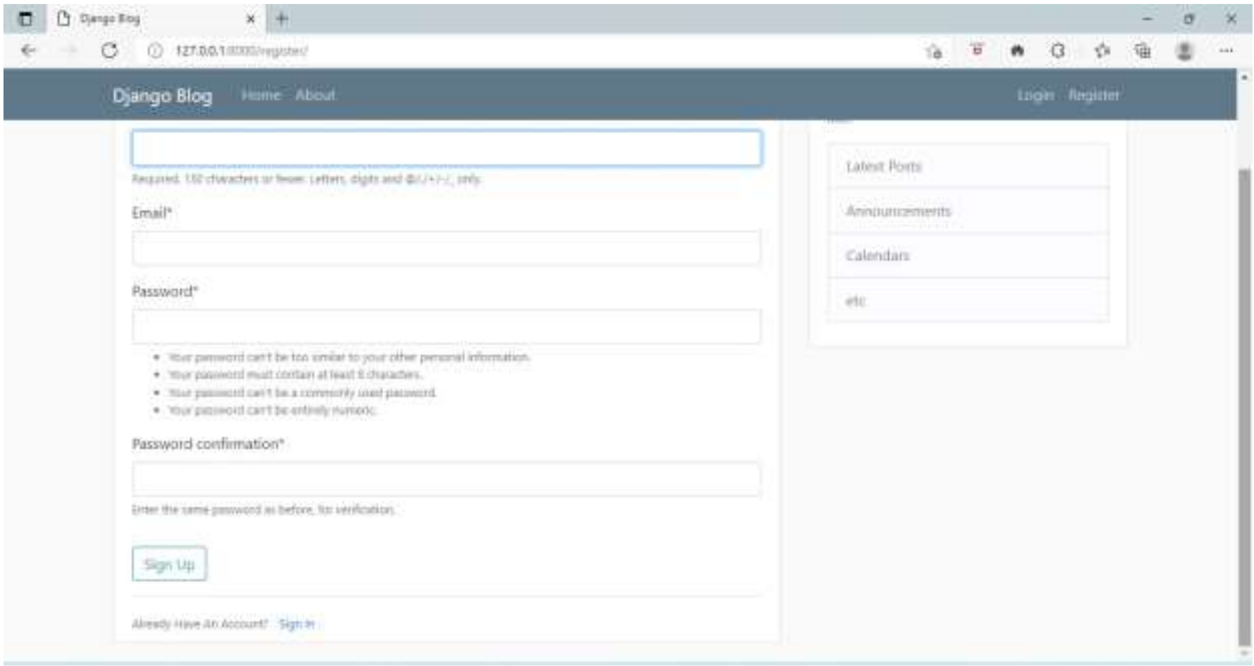
Password confirmation*

Enter the same password as before, for verification.

Our Sidebar

You can put any information here you'd like.

- Latest Posts
- Announcements
- Calendars
- etc



Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email*

Password*

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

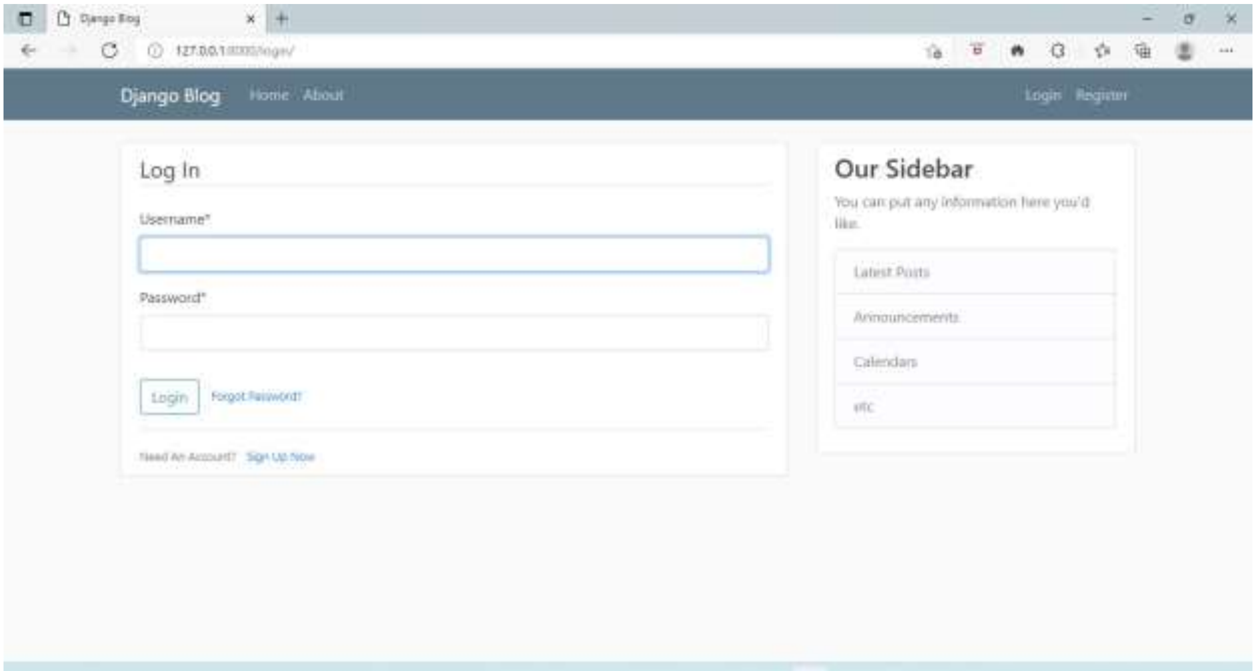
Password confirmation*

Enter the same password as before, for verification.

Already Have An Account? Sign In

- Latest Posts
- Announcements
- Calendars
- etc

- **Login Page**



Log In

Username*

Password*

Login Forgot Password?

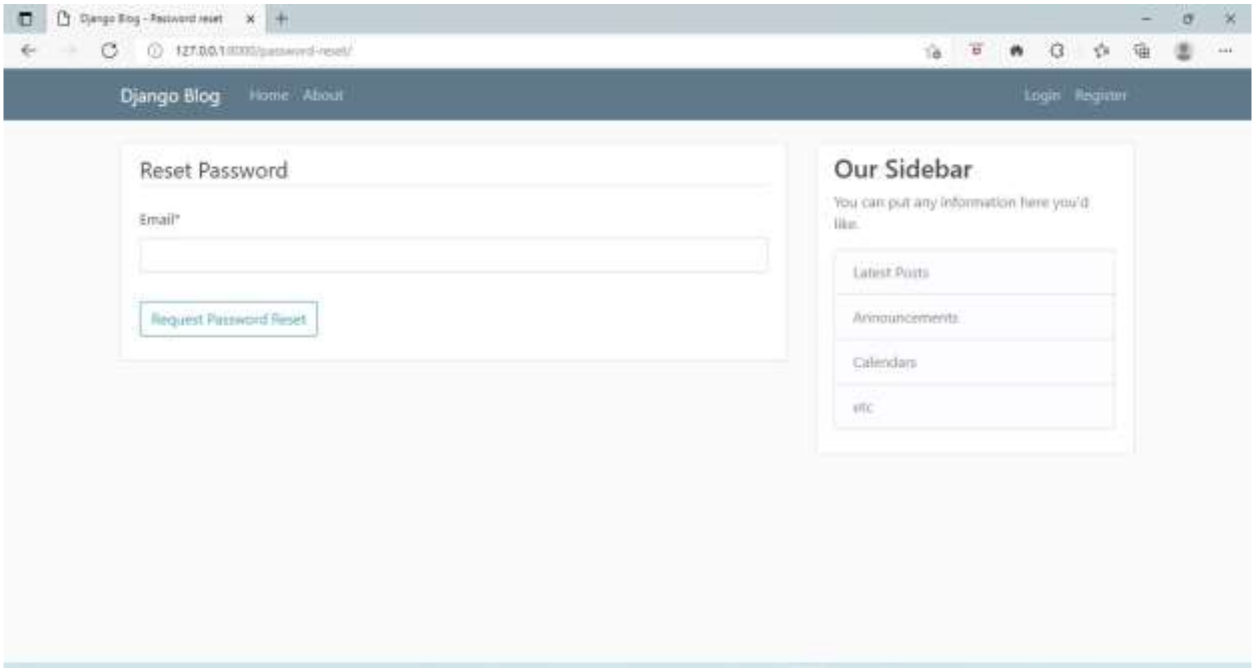
Need An Account? Sign Up Now

Our Sidebar

You can put any information here you'd like.

- Latest Posts
- Announcements
- Calendars
- etc

- **Forgot Password Page**



Reset Password

Email*

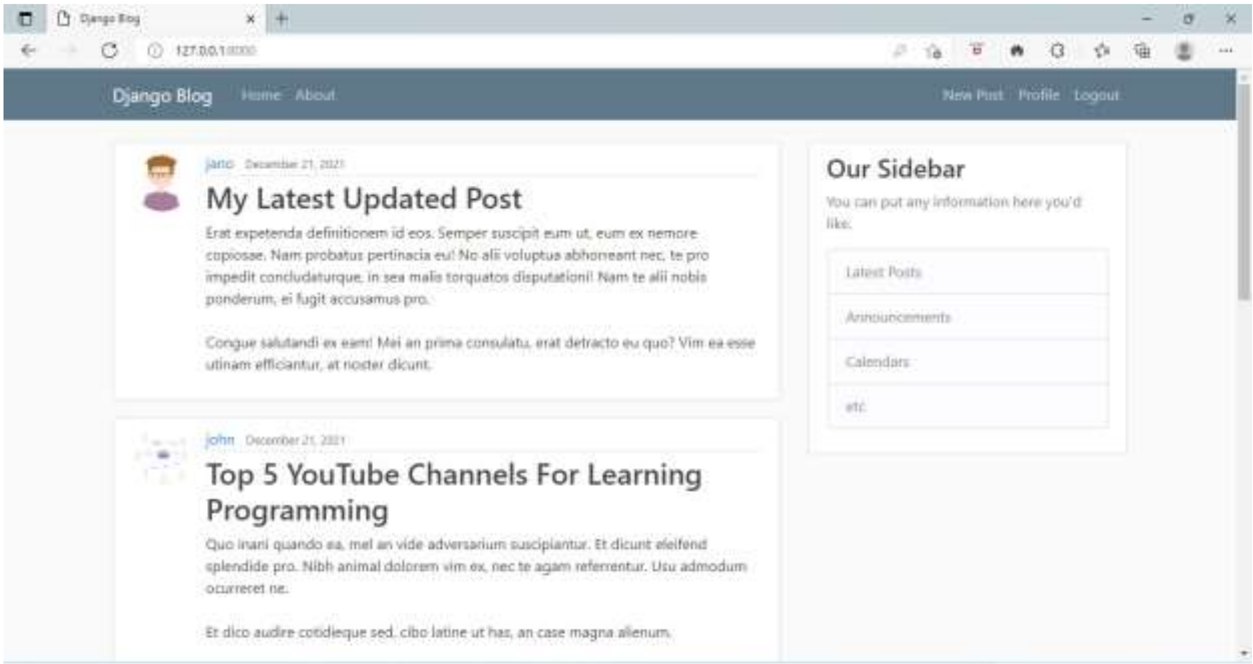
Request Password Reset

Our Sidebar

You can put any information here you'd like.

- Latest Posts
- Announcements
- Calendars
- etc

Logged In User Home Page



janto December 21, 2021

My Latest Updated Post

Erat expetenda definitionem id eos. Semper suscipit eum ut, eum ex nemore copiosae. Nam probatus pertinacia eu! No alii voluptua abhorreant nec, te pro impedit concludaturque, in sea malis torquatos disputationi! Nam te alii nobis ponderum, ei fugit accusamus pro.

Congue salutandi ex eam! Mei an prima consulatu, erat detracto eu quo? Vim ea esse utinam efficiantur, at noster dicunt.

john December 21, 2021

Top 5 YouTube Channels For Learning Programming

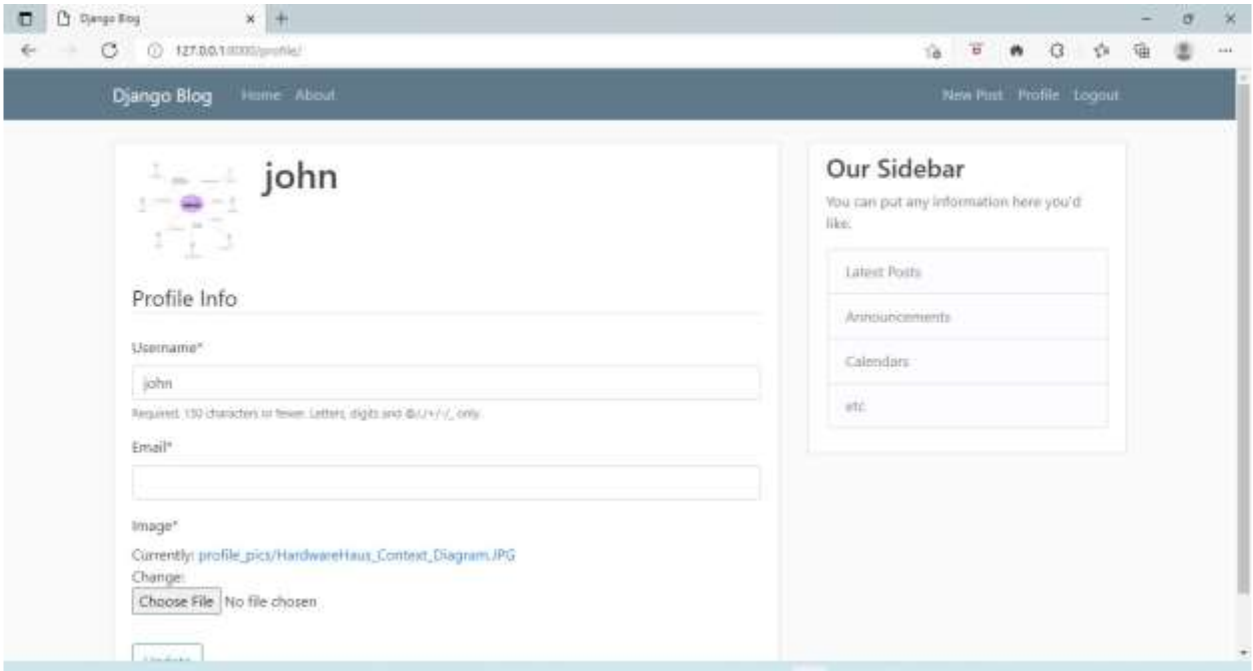
Quo inani quando ea, mel an vide adversarium suscipiantur. Et dicunt eleifend splendide pro. Nibh animal dolorem vim ex, nec te agam referrentur. Usu admodum ocurreret ne.

Et dico audire cotidieque sed, cibo latine ut has, an case magna alienum.

Our Sidebar

You can put any information here you'd like.

- **Profile Page**



john

Profile Info

Username*

john

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

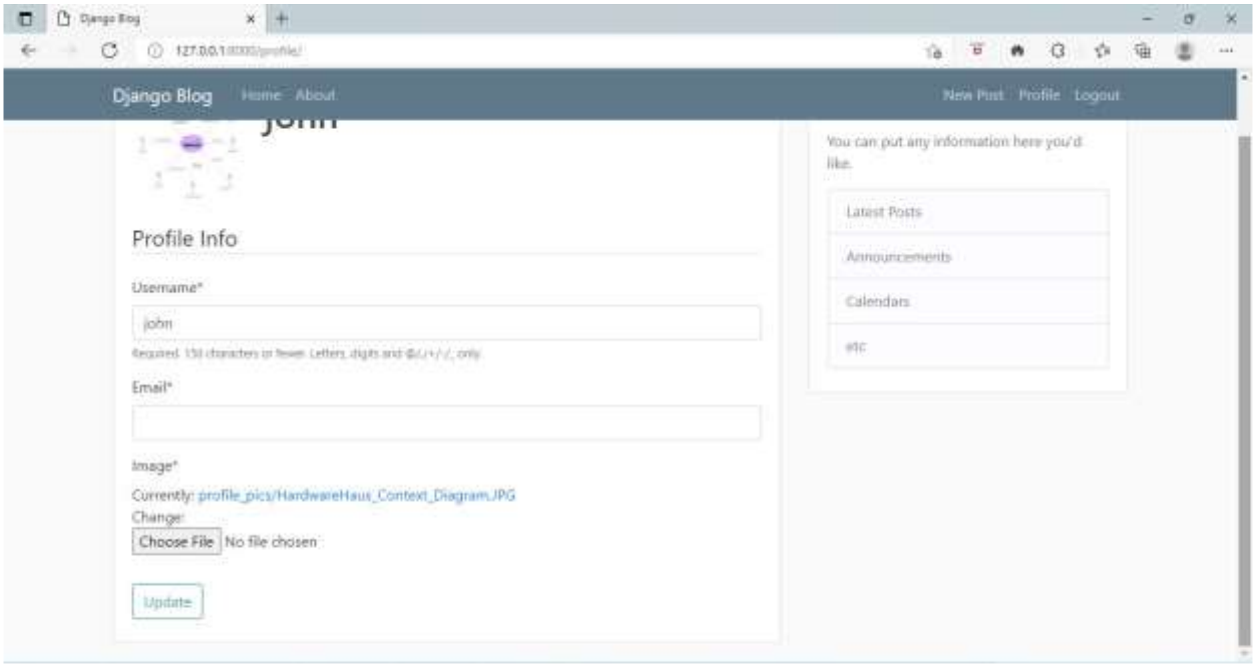
Email*

Image*

Currently: profile_pics/HardwareHaus_Context_Diagram.JPG
Change:
Choose File No file chosen

Our Sidebar

You can put any information here you'd like.



john

Profile Info

Username*

john

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email*

Image*

Currently: profile_pics/HardwareHaus_Context_Diagram.JPG

Change:

Choose File No file chosen

Update

You can put any information here you'd like.

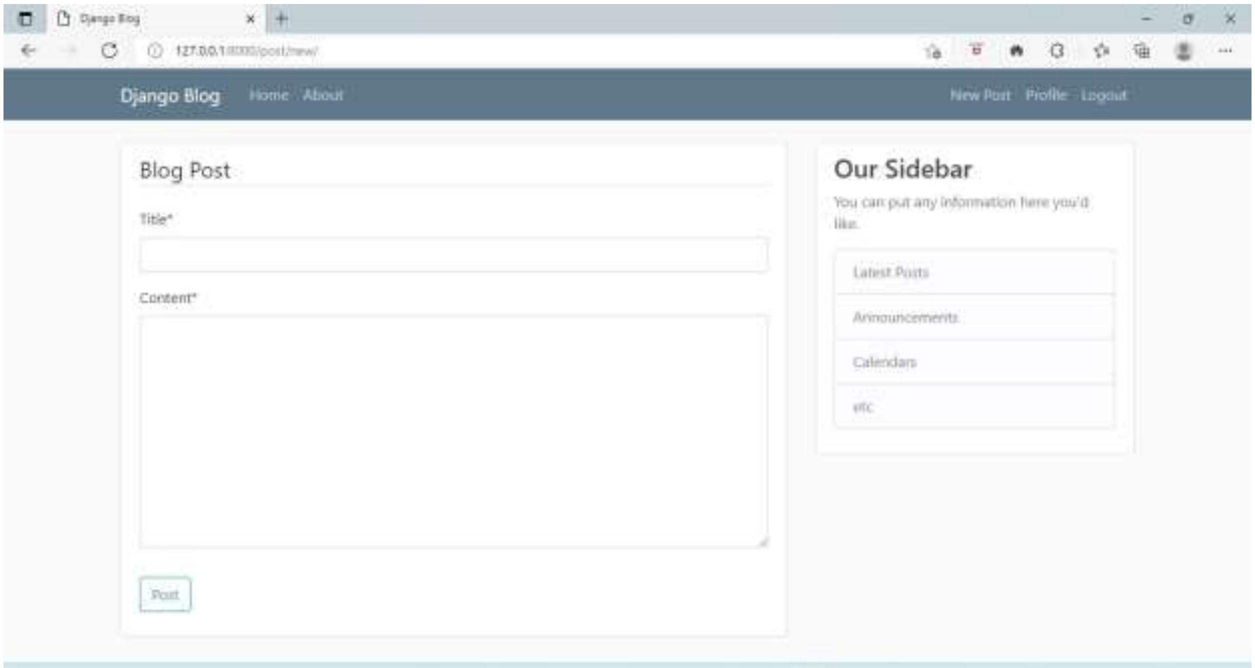
Latest Posts

Announcements

Calendars

etc

- **New Post Page**



Blog Post

Title*

Content*

Post

Our Sidebar

You can put any information here you'd like.

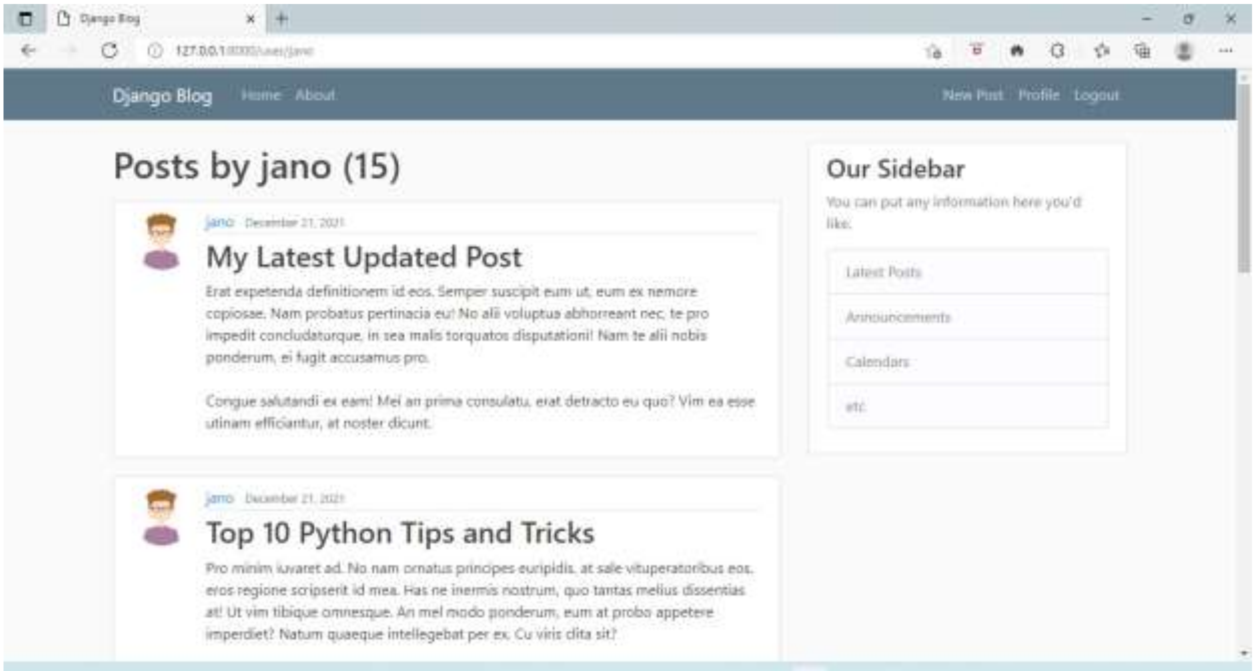
Latest Posts

Announcements

Calendars

etc

User Posts Page



Posts by jano (15)

jano December 21, 2021

My Latest Updated Post

Erat expetenda definitionem id eos. Semper suscipit eum ut, eum ex nemore copiosae. Nam probatus pertinacia eu! No alii voluptua abhorreant nec, te pro impedit concludaturque, in sea malis torquatos disputationi! Nam te alii nobis ponderum, ei fugit accusamus pro.

Congue salutandi ex eam! Mei an prima consulatu, erat detracto eu quo? Vim ea esse utinam efficiantur, at noster dicunt.

jano December 21, 2021

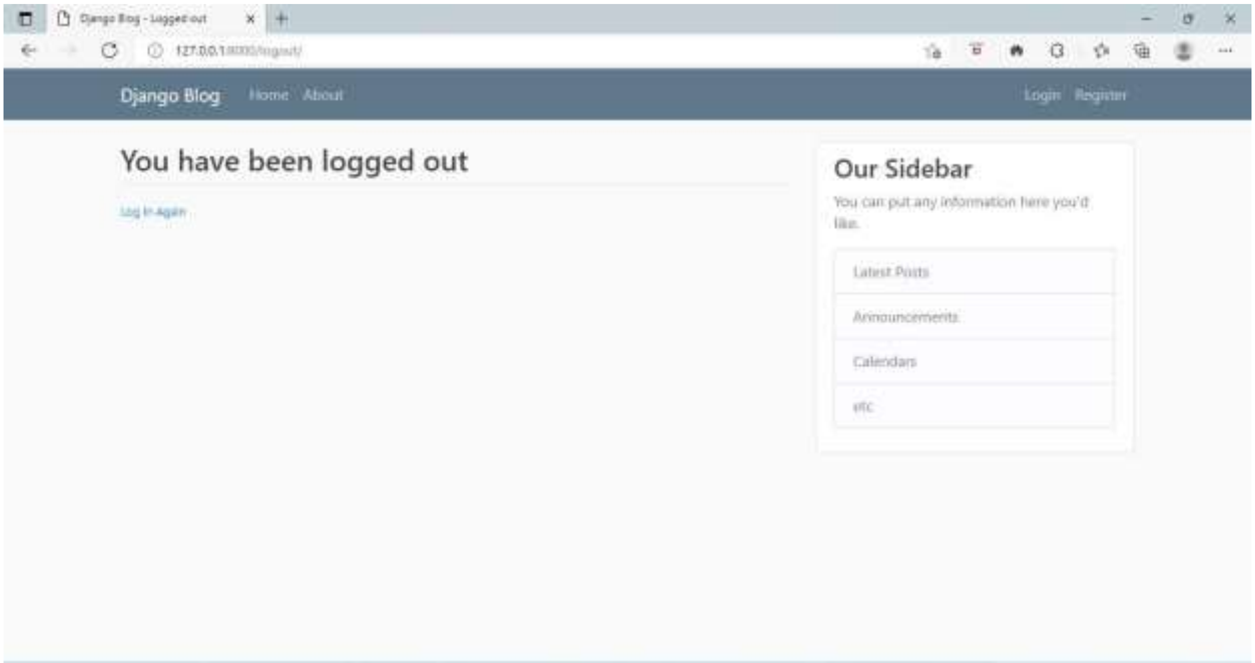
Top 10 Python Tips and Tricks

Pro minim iuvaret ad. No nam ornatus principes euripidis, at sale vituperatoribus eos, eros regione scripserit id mea. Has ne inermis nostrum, quo tantas melius dissentias at! Ut vim tibique omnesque. An mel modo ponderum, eum at probo appetere imperdiet? Natum quaeque intellegebat per ex. Cu viris clita sit?

Our Sidebar

You can put any information here you'd like.

- **Logout Page**

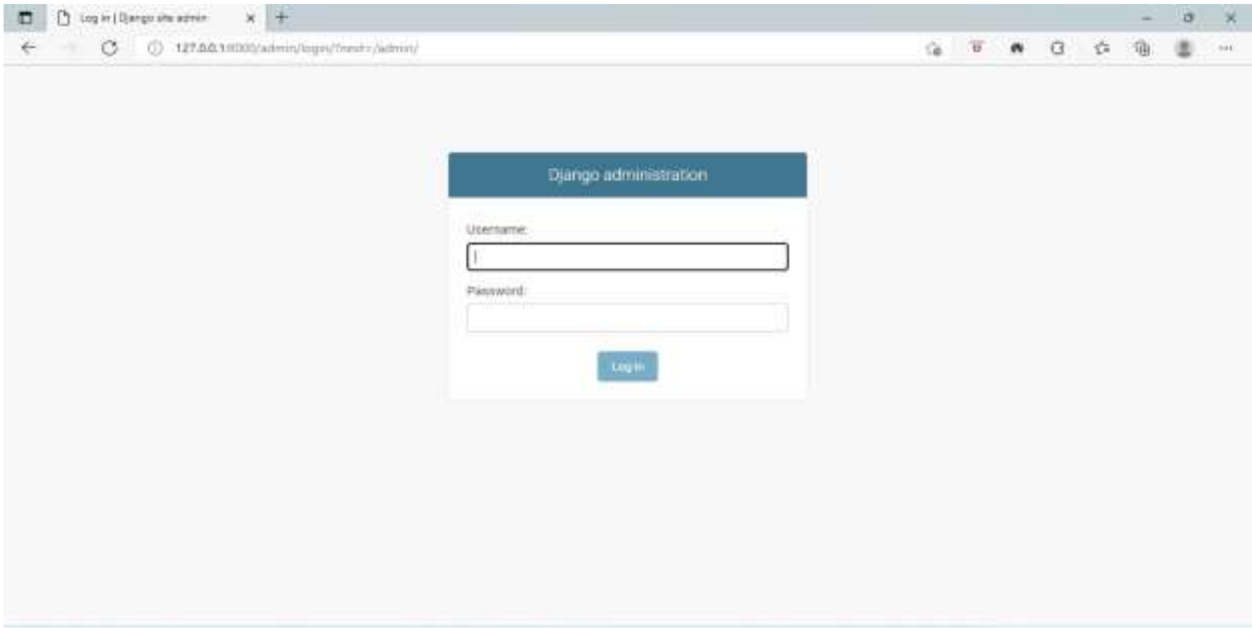


You have been logged out

Our Sidebar

You can put any information here you'd like.

- **Admin Login Page**


Log in | Django site admin

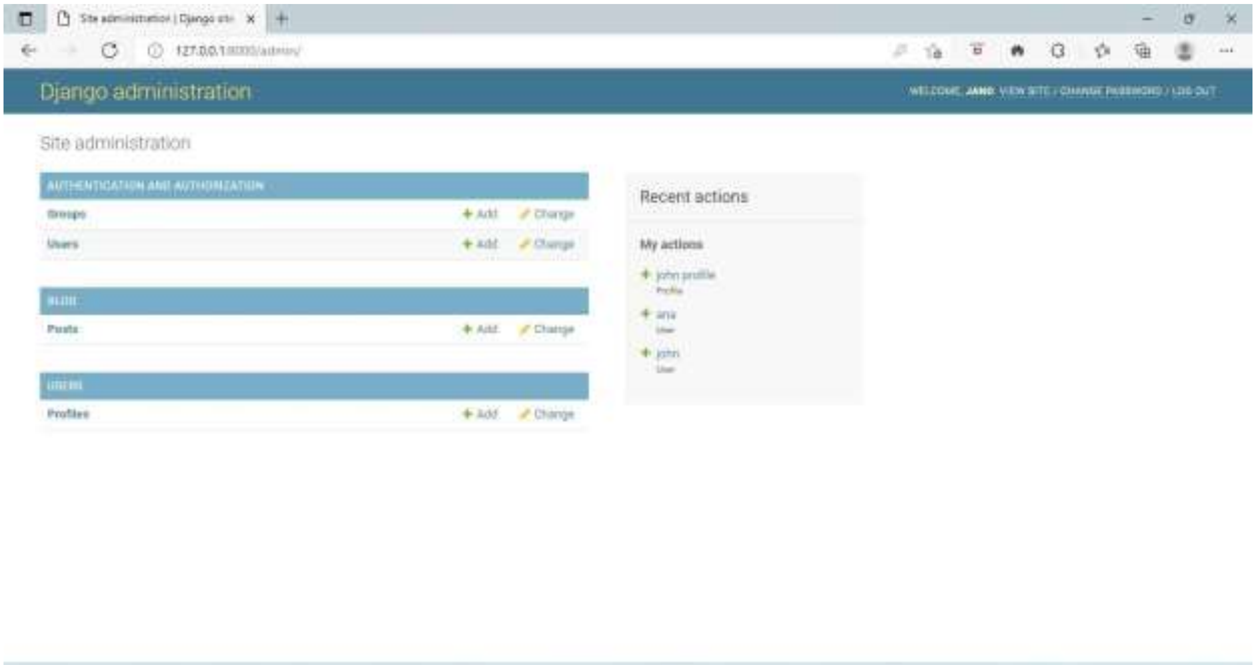
127.0.0.1:8000/admin/login/?next=/admin/

Django administration

Username:

Password:

- **Admin Home Page (Logged In)**


Site administration | Django site admin

127.0.0.1:8000/admin/

Django administration

WELCOME, **JANO**. VIEW SITE / CHANGE PASSWORD / LOG OUT

Site administration

AUTHENTICATION AND AUTHORIZATION

Groups + Add ✎ Change

Users + Add ✎ Change

BLOG

Posts + Add ✎ Change

USERS

Profiles + Add ✎ Change

Recent actions

My actions

+ john profile
Profile

+ ana
User

+ john
User