STS backup code

Project name – dataset [STS]

App name – app

Dataset [STS]:

app files

admin.py

from django.contrib import admin  
from .models import DatasetDescription  
  
  
@admin.register(DatasetDescription)  
class DatasetDescriptionAdmin(admin.ModelAdmin):  
 list\_display = ('name', 'country', 'year')  
 list\_filter = ('country', 'year')  
 search\_fields = ('name', 'description')

apps.py

from django.apps import AppConfig  
  
  
class AppConfig(AppConfig):  
 default\_auto\_field = 'django.db.models.BigAutoField'  
 name = 'app'

forms.py

from django import forms  
from app.models import DatasetDescription  
  
  
class DatasetDescriptionForm(forms.ModelForm):  
 class Meta:  
 model = DatasetDescription  
 fields = '\_\_all\_\_'  
  
 widgets = {  
 'name': forms.TextInput(attrs={'class': 'form-control', 'placeholder': 'Enter Name Of The Dataset'}),  
 'description': forms.Textarea(attrs={'class': 'form-control', 'placeholder': 'Enter Description'}),  
 'country': forms.TextInput(attrs={'class': 'form-control', 'placeholder': 'Enter Country'}),  
 'number\_of\_classes': forms.NumberInput(attrs={'class': 'form-control', 'placeholder': 'Enter total number of classes present in the dataset'}),  
 'number\_of\_scenes': forms.NumberInput(attrs={'class': 'form-control', 'placeholder': 'enter the total number of scenes'}),  
 'total': forms.NumberInput(attrs={'class': 'form-control', 'placeholder': 'enter the total number of images'}),  
 'image\_size': forms.TextInput(attrs={'class': 'form-control', 'placeholder': 'Enter Image Size'}),  
 'sign\_size': forms.TextInput(attrs={'class': 'form-control', 'placeholder': 'Enter Sign Size'}),  
 'is\_include\_video': forms.CheckboxInput(attrs={'class': 'form-check-input'}),  
 'link': forms.URLInput(attrs={'class': 'form-control', 'placeholder': 'Enter Link', 'required': False}),  
 'year': forms.NumberInput(attrs={'class': 'form-control'}),  
 }  
  
 labels = {  
 'name': 'Name',  
 'description': 'Description',  
 'country': 'Country',  
 'number\_of\_classes': 'Classes',  
 'number\_of\_scenes': 'Scenes',  
 'total': 'Total',  
 'image\_size': 'Image Size',  
 'sign\_size': 'Sign Size',  
 'is\_include\_video': 'Includes Video',  
 'link': 'Link',  
 'year': 'Year',  
  
 }

models.py

from django.db import models  
from django.shortcuts import redirect, render  
  
  
class DatasetDescription(models.Model):  
 name = models.CharField(max\_length=255)  
 description = models.CharField(max\_length=500)  
 country = models.CharField(max\_length=255)  
 number\_of\_classes = models.PositiveIntegerField()  
 number\_of\_scenes = models.PositiveIntegerField()  
 total = models.PositiveIntegerField()  
 image\_size = models.CharField(max\_length=255)  
 sign\_size = models.CharField(max\_length=255)  
 is\_include\_video = models.BooleanField()  
 link = models.URLField(null=True)  
 year = models.PositiveIntegerField()  
  
 def \_\_str\_\_(self):  
 return self.name

app.urls.py

from django.urls import path  
from . import views  
from .views import home, edit\_dataset, datasets, create  
  
urlpatterns = [  
 path('', home, name='home'),  
 path('datasets/', datasets, name='datasets'),  
 path('datasets/create/', create, name='create'),  
 path('datasets/edit/<int:dataset\_id>/', edit\_dataset, name='edit\_dataset'),  
 path('datasets/delete/<int:dataset\_id>/', views.delete\_dataset, name='delete\_dataset'),  
  
]

views.py

from django.shortcuts import render, redirect, get\_object\_or\_404  
from django.urls import reverse  
  
from .forms import DatasetDescriptionForm  
from .models import DatasetDescription  
  
  
def home(request):  
 if request.method == 'POST':  
 form = DatasetDescriptionForm(request.POST)  
 if form.is\_valid():  
 form.save()  
 return redirect('datasets') # Redirect to the home page after successful form submission  
 else:  
 form = DatasetDescriptionForm()  
  
 dataset\_descriptions = DatasetDescription.objects.all()  
 context = {  
 'form': form,  
 'dataset\_descriptions': dataset\_descriptions  
 }  
 return render(request, 'home.html', context)  
  
  
def create(request):  
 if request.method == 'POST':  
 form = DatasetDescriptionForm(request.POST)  
 if form.is\_valid():  
 form.save()  
 return redirect('datasets') # Redirect to the home page after successful form submission  
 else:  
 form = DatasetDescriptionForm()  
  
 dataset\_descriptions = DatasetDescription.objects.all()  
 context = {  
 'form': form,  
 'dataset\_descriptions': dataset\_descriptions  
 }  
 return render(request, 'create\_dataset\_description.html', context)  
  
  
def create\_dataset\_description(request):  
 if request.method == 'POST':  
 form = DatasetDescriptionForm(request.POST)  
 if form.is\_valid():  
 form.save()  
 return redirect('dataset\_description\_list')  
 else:  
 form = DatasetDescriptionForm()  
 return render(request, 'create\_dataset\_description.html', {'form': form})  
  
  
def datasets(request):  
 dataset\_description = DatasetDescription.objects.all()  
 return render(request, 'datasets\_list.html', {'dataset\_descriptions': dataset\_description})  
  
  
def edit\_dataset(request, dataset\_id):  
 dataset = get\_object\_or\_404(DatasetDescription, id=dataset\_id)  
  
 if request.method == 'POST':  
 form = DatasetDescriptionForm(request.POST, instance=dataset)  
 if form.is\_valid():  
 form.save()  
 return redirect(  
 'datasets') # Replace 'datasets' with the URL name of the page where the dataset list is displayed  
 else:  
 form = DatasetDescriptionForm(instance=dataset)  
  
 return render(request, 'edit\_dataset.html', {'form': form})  
  
  
def delete\_dataset(request, dataset\_id):  
 dataset = DatasetDescription.objects.get(id=dataset\_id)  
 dataset.delete()  
 return redirect('datasets')

app. template files

base.html

{% load static %}  
  
<html lang="en">  
 <head>  
  
 <meta charset="utf-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>Traffic Sign</title>  
 <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.7.3/font/bootstrap-icons.css">  
 <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css" integrity="sha512-Z/9eErsj7doTs+a1zDxfjvZKUVTG/HlZ4JSAAm47iFMIu8/sgsJFxd4YegVEtSZZtGX5zR+5g4awSNQY7Y8O3w==" crossorigin="anonymous" referrerpolicy="no-referrer" />  
 <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" integrity="sha384-PnpVVBsUW3+rU/itv5MCn7P5A99BvF47SVpI5tRvXfOTQfaTpqMzhP0mtRG70m4r" crossorigin="anonymous">  
 <link rel="stylesheet" type="text/css" href="{% static 'styles/style.css' %}">  
  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-9ndCyUaIbzAi2FUVXJi0CjmCapSmO7SnpJef0486qhLnuZ2cdeRhO02iuK6FUUVM" crossorigin="anonymous">  
 </head>  
 <body>  
 {% include 'navbar.html' %}  
 <div class="container">  
 <br>  
 {% if messages %}  
 {% for message in messages %}  
 <div class="alert alert-warning alert-dismissible fade show" role="alert">  
 {{ message }}  
 <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>  
 </div>  
 {% endfor %}  
  
 {% endif %}  
  
 {% block content %}  
 {% endblock%}  
 </div>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.min.js" integrity="sha384-7EdtqrqxTrpoBSgBzbl4VX8mQc+ztTl+3mePz/AfJ2D7CwtXQc7wQzgmDz7HJd83" crossorigin="anonymous"></script>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js" integrity="sha384-geWF76RCwLtnZ8qwWowPQNguL3RmwHVBC9FhGdlKrxdiJJigb/j/68SIy3Te4Bkz" crossorigin="anonymous"></script>  
 </body>  
</html>

Create\_dataset\_description.html

{% extends 'base.html' %}  
  
{% block content %}  
 <h1>Dataset Descriptions</h1>  
  
 <form method="POST" action="{% url 'home' %}">  
 {% csrf\_token %}  
 {{ form.as\_p }}  
 <button type="submit" class="btn btn-primary">Submit</button>  
 <a href="{% url 'datasets' %}" class="btn btn-secondary">Back</a>  
 </form>  
  
 <table class="table">  
 <!-- Display existing dataset descriptions -->  
 </table>  
{% endblock %}

Datasets\_list.html

{% extends 'base.html' %}  
  
{% block content %}  
<h1>Dataset Descriptions</h1>  
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css"/>  
<div class="text-right">  
 <button align="center" type="button" class="btn btn-primary" target="\_blank" onclick="location.href='{% url 'create' %}'">Add Dataset</button>  
</div>  
<style>  
 .text-right{  
 display: flex;  
 justify-content: right;  
 }  
  
</style>  
<style>  
 .navy-table-head th{  
 background-color: navy;  
 color: white;  
 }  
</style>  
  
  
<div class="table-responsive">  
 <table class="table table-striped table-class">  
 <thead class="navy-table-head" >  
 <tr>  
 <th scope="col" style="width: 10%;">Sl.No.</th>  
 <th scope="col" style="width: 10%;">Name</th>  
 <th scope="col" style="width: 30%;">Description</th>  
 <th scope="col" style="width: 10%;">Country</th>  
 <th scope="col" style="width: 10%;">Classes</th>  
 <th scope="col" style="width: 10%;">Scenes</th>  
 <th scope="col" style="width: 10%;">Total</th>  
 <th scope="col" style="width: 10%;">Image Size</th>  
 <th scope="col" style="width: 10%;">Sign Size</th>  
 <th scope="col" style="width: 10%;">Includes Video</th>  
 <th scope="col" style="width: 5%;">Link</th>  
 <th scope="col" style="width: 5%;">Year</th>  
 <th scope="col" style="width: 5%;">Actions</th>  
 </tr>  
 </thead>  
 <tbody>  
 {% for dataset\_description in dataset\_descriptions %}  
 <tr>  
 <td>{{ forloop.counter }}</td>  
 <td>{{ dataset\_description.name }}</td>  
 <td>{{ dataset\_description.description }}</td>  
 <td>{{ dataset\_description.country }}</td>  
 <td>{{ dataset\_description.number\_of\_classes }}</td>  
 <td>{{ dataset\_description.number\_of\_scenes }}</td>  
 <td>{{ dataset\_description.total }}</td>  
 <td>{{ dataset\_description.image\_size }}</td>  
 <td>{{ dataset\_description.sign\_size }}</td>  
 <td>  
 {% if dataset\_description.is\_include\_video %}  
 Yes  
 {% else %}  
 No  
 {% endif %}  
 </td>  
 <td style="max-width: 200px; overflow: hidden; text-overflow: ellipsis;">{{ dataset\_description.link }}</td>  
 <td>{{ dataset\_description.year }}</td>  
 <td>  
 <a href="{% url 'edit\_dataset' dataset\_description.id %}" target="\_blank" class="text-primary me-2">  
 <i class="fa-solid fa-pen-to-square"></i>  
 </a>  
 <a href="#" class="text-danger delete-link" data-url="{% url 'delete\_dataset' dataset\_description.id %}">  
 <i class="fa-solid fa-trash"></i>  
 </a>  
 </td>  
 </tr>  
 {% endfor %}  
 </tbody>  
 </table>  
</div>  
  
  
<div class="pagination">  
 {% if dataset\_descriptions.has\_previous %}  
 <a href="?page={{ dataset\_descriptions.previous\_page\_number }}" class="text-primary">  
 Previous  
 </a>  
 {% endif %}  
  
 {% if dataset\_descriptions.has\_next %}  
 <a href="?page={{ dataset\_descriptions.next\_page\_number }}" class="text-primary">  
 Next  
 </a>  
 {% endif %}  
</div>  
<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>  
<script>  
 $(document).ready(function() {  
 $('.delete-link').click(function(e) {  
 e.preventDefault(); // Prevent the default behavior of the link  
 var deleteUrl = $(this).data('url');  
 if (confirm("Are you sure you want to delete this dataset?")) {  
 $.ajax({  
 url: deleteUrl,  
 type: 'POST',  
 data: {  
 'csrfmiddlewaretoken': '{{ csrf\_token }}'  
 },  
 success: function(response) {  
 window.location.href = "{% url 'datasets' %}"; // Redirect to the datasets list page  
 },  
 error: function(xhr, errmsg, err) {  
 console.log(xhr.status + ": " + xhr.responseText);  
 }  
 });  
 }  
 });  
 });  
</script>  
{% endblock %}

Edit\_dataset.html

{% extends 'base.html' %}  
  
{% block content %}  
 <h1>Edit Dataset</h1>  
  
 <form method="POST">  
 {% csrf\_token %}  
 {{ form.as\_p }}  
 <button type="submit">Save</button>  
 </form>  
{% endblock %}

Home.html

{% extends 'base.html' %}  
{% load static %}  
{% block content %}  
  
<div id="imageCarousel" class="carousel slide" data-bs-ride="carousel">  
 <div class="carousel-inner">  
 <div class="carousel-item active">  
 <img src="{% static 'images/wp7190999.jpg' %}" class="d-block w-100" alt="Image 1">  
 </div>  
 <div class="carousel-item">  
 <img src="{% static 'images/wp7191038.jpg' %}" class="d-block w-100" alt="Image 2">  
 </div>  
 <div class="carousel-item">  
 <img src="{% static 'images/wp7191064.jpg' %}" class="d-block w-100" alt="Image 3">  
 </div>  
 </div>  
 <button class="carousel-control-prev" type="button" data-bs-target="#imageCarousel" data-bs-slide="prev">  
 <span class="carousel-control-prev-icon" aria-hidden="true"></span>  
 <span class="visually-hidden">Previous</span>  
 </button>  
 <button class="carousel-control-next" type="button" data-bs-target="#imageCarousel" data-bs-slide="next">  
 <span class="carousel-control-next-icon" aria-hidden="true"></span>  
 <span class="visually-hidden">Next</span>  
 </button>  
</div>  
  
<script>  
  
 $('.carousel').carousel({  
 interval: 1000  
});  
</script>  
  
{% endblock %}

Navbar.html

<nav class="navbar navbar-expand-lg navbar-dark" style="background-color: navy;">  
 <div class="container-fluid">  
 <a class="navbar-brand" href="#" style="color: white;">Traffic Sign</a>  
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">  
 <span class="navbar-toggler-icon"></span>  
 </button>  
 <div class="collapse navbar-collapse" id="navbarSupportedContent">  
 <ul class="navbar-nav me-auto mb-2 mb-lg-0">  
 <li class="nav-item">  
 <a class="nav-link active" aria-current="page" href="{% url 'home' %}" style="color: white;">Home</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="#" style="color: white;">Detect</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="#" style="color: white;">Results</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="{% url 'datasets' %}" style="color: white;">Datasets</a>  
 </li>  
 <li class="nav-item dropdown">  
 <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false" style="color: white;">  
 Dropdown  
 </a>  
 <ul class="dropdown-menu">  
 <li><a class="dropdown-item" href="#">Action</a></li>  
 <li><a class="dropdown-item" href="#">Another action</a></li>  
 <li><hr class="dropdown-divider"></li>  
 <li><a class="dropdown-item" href="#">Something else here</a></li>  
 </ul>  
 </li>  
 </ul>  
 </div>  
 </div>  
</nav>

App->Static->images



Image 1



Image 2



Image 3

App->Static->styles

Style.css

.custom-image {  
 width: 100%;  
 height: 100vh;  
 object-fit: cover;  
 border: 2px solid red;  
}  
  
  
.navy-table-head th{  
 background-color: navy;  
 color: white;  
 }  
  
.text-right{  
 display: flex;  
 justify-content: right;  
 }

Dataset project files

Settings.py

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/3.2/howto/deployment/checklist/  
  
# SECURITY WARNING: keep the secret key used in production secret!  
SECRET\_KEY = 'django-insecure-o@uhr=n0v=m7yhlwz29\_svzo90\*nod81@@6=(73\_%%^czb\_s08'  
  
# 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',  
 '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 = 'dataset.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 = 'dataset.wsgi.application'  
  
  
# Database  
# https://docs.djangoproject.com/en/3.2/ref/settings/#databases  
  
DATABASES = {  
 'default': {  
 'ENGINE': 'django.db.backends.mysql',  
 'NAME': 'ts\_dataset',  
 'USER': 'root',  
 'PASSWORD': 'sqll',  
 'HOST': 'localhost',  
 'PORT': '3306',  
 }  
}  
  
  
# Password validation  
# https://docs.djangoproject.com/en/3.2/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/3.2/topics/i18n/  
  
LANGUAGE\_CODE = 'en-us'  
  
TIME\_ZONE = 'UTC'  
  
USE\_I18N = True  
  
USE\_L10N = True  
  
USE\_TZ = True  
  
  
# Static files (CSS, JavaScript, Images)  
# https://docs.djangoproject.com/en/3.2/howto/static-files/  
  
STATIC\_URL = '/static/'  
  
# Default primary key field type  
# https://docs.djangoproject.com/en/3.2/ref/settings/#default-auto-field  
  
DEFAULT\_AUTO\_FIELD = 'django.db.models.BigAutoField'

project->urls.py

from django.contrib import admin  
from django.urls import path, include  
  
urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('', include('app.urls'))  
]

manage.py

#!/usr/bin/env python  
*"""Django's command-line utility for administrative tasks."""*import os  
import sys  
  
  
def main():  
 *"""Run administrative tasks."""* os.environ.setdefault('DJANGO\_SETTINGS\_MODULE', 'dataset.settings')  
 try:  
 from django.core.management import execute\_from\_command\_line  
 except ImportError as exc:  
 raise ImportError(  
 "Couldn't import Django. Are you sure it's installed and "  
 "available on your PYTHONPATH environment variable? Did you "  
 "forget to activate a virtual environment?"  
 ) from exc  
 execute\_from\_command\_line(sys.argv)  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()

Requirement.txt

asgiref==3.4.1  
Django==3.2.19  
mysqlclient==1.4.6  
Pillow==8.4.0  
PyMySQL==1.0.2  
pytz==2023.3  
sqlparse==0.4.4  
typing\_extensions==4.1.1

note : don’t forget to makemigrations and migrate in order to get migrations files