Assignment: Django Mini Movie Gallery

Objective

Build a mini Django project to understand how templates and static files work. You'll create a basic Movie Gallery to display a list of movies with posters using HTML, CSS, and Django's view rendering.

Project Title: Django Movie Gallery

Instructions

Follow the steps below to build your project. No database is required. Use hardcoded data for movie details.

Step-by-Step Guide

Step	Task	Description
1	Create Django Project	django-admin startproject movie_project
2	Create App	python manage.py startapp movies
3	Register App	Add 'movies' to INSTALLED_APPS in settings.py
4	Setup URLs	Create movies/urls.py and include in project urls.py
5	Create Template Folder	Inside app: templates/movies/movie_list.html, etc.
6	Configure Templates	Set 'DIRS': [BASE_DIR / 'templates'] in TEMPLATES
7	Create Static Folder	static/movies/ with CSS, images
8	Configure Static	Add STATICFILES_DIRS = [BASE_DIR / 'static'] in settings.py
9	Add Dummy Movie Data	Use a list of dictionaries in views to simulate movies
10	Build Views	Render HTML pages using templates
11	Link CSS & Images	Use {% load static %} and reference CSS/images
12	Test the App	Run server and test each endpoint

Required Endpoints

URL Path	View Name	Description	Response Type
/	home_view	Welcome Page	HTML
/movies/	movie_list_view	Shows list of all movies	HTML Template

Assignment: Django Mini Movie Gallery

/movies/<int:id>/	movie_detail_view	Shows single movie details	HTML Template
/api/movies/	movie_json_view	Returns JSON of all movies	JSON Response

Deliverables

- Template files (HTML)

- Static files (CSS, images)

- Python files (views.py, urls.py)

- Screenshots of working output

Bonus (Optional)

- Add Bootstrap to improve the design

- Make the detail page dynamic

- Add a navigation bar in base.html

Submission Deadline

_____ (Fill in the date assigned by your instructor)