# Movie Recommendation System – Instructions to Run

## Project Overview

The **Movie Recommendation System** recommends similar movies based on a selected movie using **content-based filtering**, **NLP**, and **cosine similarity**. The web interface is built with **Streamlit**, and movie posters are fetched via the **OMDb API**.

## Project Files

Ensure all these files are in the same folder: 1. 25k IMDb movie Dataset.csv – Movie dataset 2. app.py – Streamlit web application 3. model.pkl or movie\_dict.pkl – Pickled movie metadata 4. similarity.npy – Precomputed similarity matrix

## Required Libraries

Install Python 3.8+ and run:

pip install pandas numpy scikit-learn nltk streamlit requests

## Step-by-Step Instructions

### Step 1: Prepare the Dataset and Models

* Keep 25k IMDb movie Dataset.csv in the same folder as app.py.
* Ensure model.pkl / movie\_dict.pkl and similarity.npy are in the same folder.
* These files contain preprocessed movie data and similarity matrix needed for recommendations.

### Step 2: Run the Streamlit Application

1. Open a terminal or command prompt in the folder containing app.py.
2. Execute the command:

streamlit run app.py

1. A browser window/tab will open with the **Movie Recommendation System** interface.

### Step 3: Use the Application

1. Select a movie from the dropdown menu.
2. Click the **“Recommend”** button.
3. The top 5 similar movies will appear along with their **poster images**.
4. If a poster is not found, a placeholder image will be displayed.

### Step 4: Notes

* Ensure an **active internet connection** to fetch movie posters from the OMDb API.
* You can update model.pkl or similarity.npy if the dataset changes.
* Ideal for learning **content-based movie recommendations**, **cosine similarity**, and **Streamlit app development**.

## Optional Enhancements

* Combine with collaborative filtering to create a **hybrid recommendation system**.
* Include additional features like ratings, reviews, or release year for improved recommendations.
* Customize the web interface with advanced Streamlit components for a better user experience.

## Educational Purpose

This project is created **solely for educational purposes** to demonstrate the implementation of a content-based movie recommendation system using machine learning and NLP techniques.