

```

import pickle

import streamlit as st

import requests


def fetch_poster(movie_id):

    url =
    "https://api.themoviedb.org/3/movie/{}?api_key=8265bd1679663a7ea12ac168da84d2e8&l
    anguage=en-US".format(movie_id)

    data = requests.get(url)

    data = data.json()

    poster_path = data['poster_path']

    full_path = "https://image.tmdb.org/t/p/w500/" + poster_path

    return full_path


def recommend(movie):

    index = movies[movies['title'] == movie].index[0]

    distances = sorted(list(enumerate(similarity[index])), reverse=True, key=lambda x: x[1])

    recommended_movie_names = []

    recommended_movie_posters = []

    for i in distances[1:6]:

        # fetch the movie poster

        movie_id = movies.iloc[i[0]].movie_id

        recommended_movie_posters.append(fetch_poster(movie_id))

        recommended_movie_names.append(movies.iloc[i[0]].title)


    return recommended_movie_names,recommended_movie_posters


st.header('watch your favorite movie')

```

```
movies = pickle.load(open('movie_list.pkl','rb'))
similarity = pickle.load(open('similarity.pkl','rb'))
```

```
movie_list = movies['title'].values
```

```
selected_movie = st.selectbox(
    "Type or select a movie",
    movie_list
)
```

```
if st.button('Show Recommendation'):
```

```
    recommended_movie_names,recommended_movie_posters =
    recommend(selected_movie)
```

```
    col1, col2, col3, col4, col5 = st.columns(5)
```

```
    with col1:
```

```
        st.text(recommended_movie_names[0])
```

```
        st.image(recommended_movie_posters[0])
```

```
    with col2:
```

```
        st.text(recommended_movie_names[1])
```

```
        st.image(recommended_movie_posters[1])
```

```
    with col3:
```

```
        st.text(recommended_movie_names[2])
```

```
        st.image(recommended_movie_posters[2])
```

```
    with col4:
```

```
        st.text(recommended_movie_names[3])
```

```
        st.image(recommended_movie_posters[3])
```

```
    with col5:
```

```
        st.text(recommended_movie_names[4])
```

```
st.image(recommended_movie_posters[4])
```