



LOADING AND ERROR HANDLING





```
import React, { useState, useEffect } from "react";

const KEY = '44397289';
const BASE = 'https://www.omdbapi.com'

const App = ()=>
{
  const [movies, setMovies] = useState([]);
  const [query, setQuery] = useState("interstellar");
  const [isLoading, setIsLoading] = useState(false);
  const [error, setError] = useState("");

  useEffect(()=>
  {
    const fetchMovies = async () =>
    {
      try
      {
        setError("");
        setIsLoading(true);
        if(query.length < 3)
        {
          setMovies([]);
          throw new Error("Please type to search a movie");
        }

        const res = await fetch(`${BASE}?apikey=${KEY}&s=${query}`);

        console.log("Response is ", res);

        if(!res.ok && res.status !== 200)
        {
          throw new Error("Something went wrong with fetching movies");
        }
        const data = await res.json();
        if(data.Response === 'False')
        {
          throw new Error(data.Error)
        }
        setMovies(data.Search);
        setError("");
      }
      catch (error)
      {
        setError(error.message);
      }
      finally
      {
        setIsLoading(false);
      }
    }

    fetchMovies();
  }, [query]);

  return (
    <div>
      { isLoading && <Loader /> }
      { !isLoading && !error && (<MoviesList movies={movies}/>) }
      { error && <ErrorMessage message={error}/> }
    </div>
  );
}

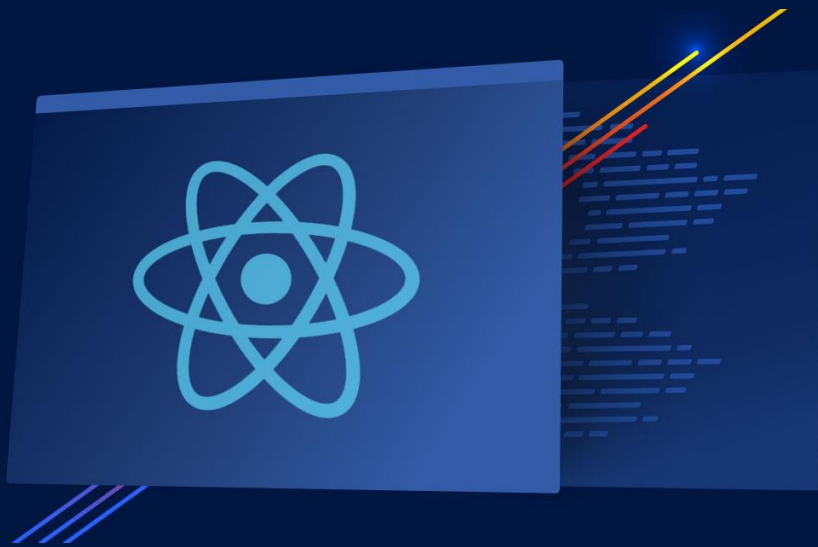
export default App;
```





Do you find it helpful?

Let me know down in the comments.



Click To Follow For More On LinkedIn

