



ABORT CONTROLLER : CLEAN UP FUNCTION





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

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

  useEffect(() =>
  {
    const controller = new AbortController();
    const fetchMovies = async () =>
    {
      try
      {
        setError("");
        setIsLoading(true);

        if(query.length < 3)
        {
          throw new Error("Please type to search a movie");
        }

        const res = await fetch(`https://www.omdbapi.com/?apikey=${KEY}&s=${query}`, { signal: controller.signal });

        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)
      {
        if(error.name !== "AbortError")
        {
          setError(error.message);
        }
      }
      finally
      {
        setIsLoading(false);
      }
    }

    fetchMovies();

    return () =>
    {
      controller.abort();
    }
  }, [query]);

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

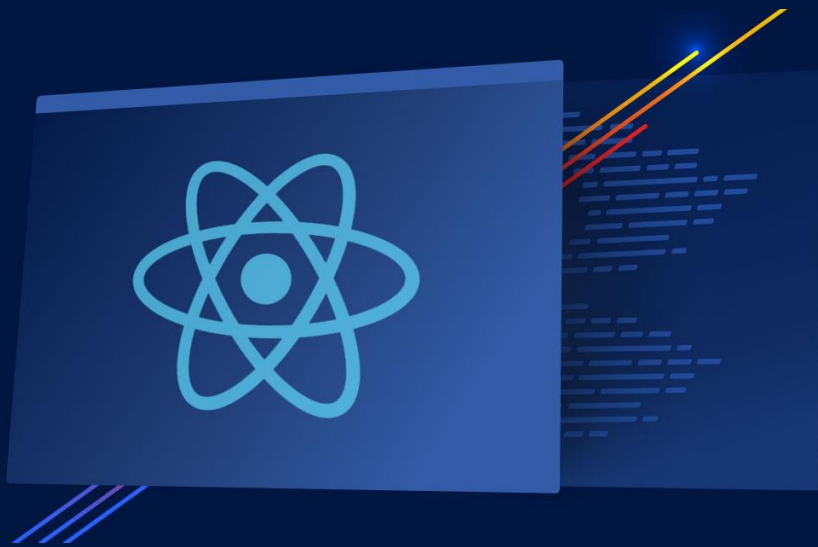
export default App;
```





Do you find it helpful?

Let me know down in the comments.



Click To Follow For More On LinkedIn

