

# RE-FETCHING QUERIES

in this GraphQL tutorial I'll show you how we can re-fetch queries to automatically update a component without have to refresh the page



## AddBook.js

```
import React from "react";
import { useQuery, useMutation } from "@apollo/client";
import { getAuthorsQuery, addBookMutation, getBooksQuery } from '../queries/queries';

function DisplayAuthors(){
  const { loading, data } = useQuery(getAuthorsQuery);
  if(loading){
    return( <option disabled>Loading authors</option> );
  } else {
    return data.authors.map(author => {
      return( <option key={ author.id } value={author.id}>{ author.name }</option> );
    });
  }
}

function AddBook() {

  const [book, setBook] = React.useState({ name: "", genre: "", authorId: "" });

  const [addBookMut] = useMutation(addBookMutation)

  const handleSubmit = e => {
    e.preventDefault();
    addBookMut({
      variables: {
        name: book.name,
        genre: book.genre,
        authorId: book.authorId
      },
      refetchQueries: [{ query: getBooksQuery }]
    });
  }

  ...
}
```

Insert  
code

# My Reading List

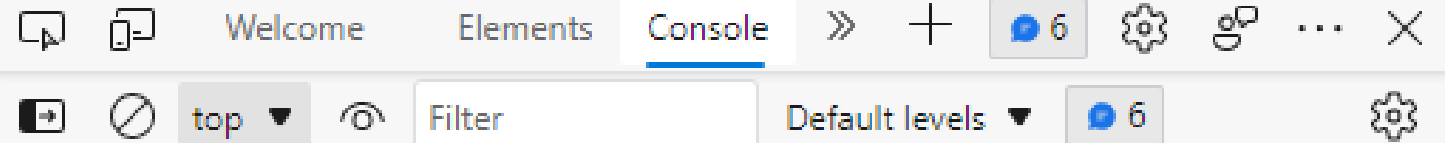
- Name of the Wind
- The Final Empire
- The Long Earth
- The Hero of Ages
- The Colour of Magic
- The Light Fantastic
- Hogfather
- The Long Universe

Book name:

Genre:

Author:

+



[HMR] Waiting for update signal from WDS...

[log.js:24](#)

Download the Apollo DevTools for a better development experience: [invariant.ts:42](https://chrome.google.com/webstore/detail/apollo-client-dev-eloper-t/jdkknkkbebbapilgoeccciglkfbmbnfm)  
<https://chrome.google.com/webstore/detail/apollo-client-dev-eloper-t/jdkknkkbebbapilgoeccciglkfbmbnfm>

